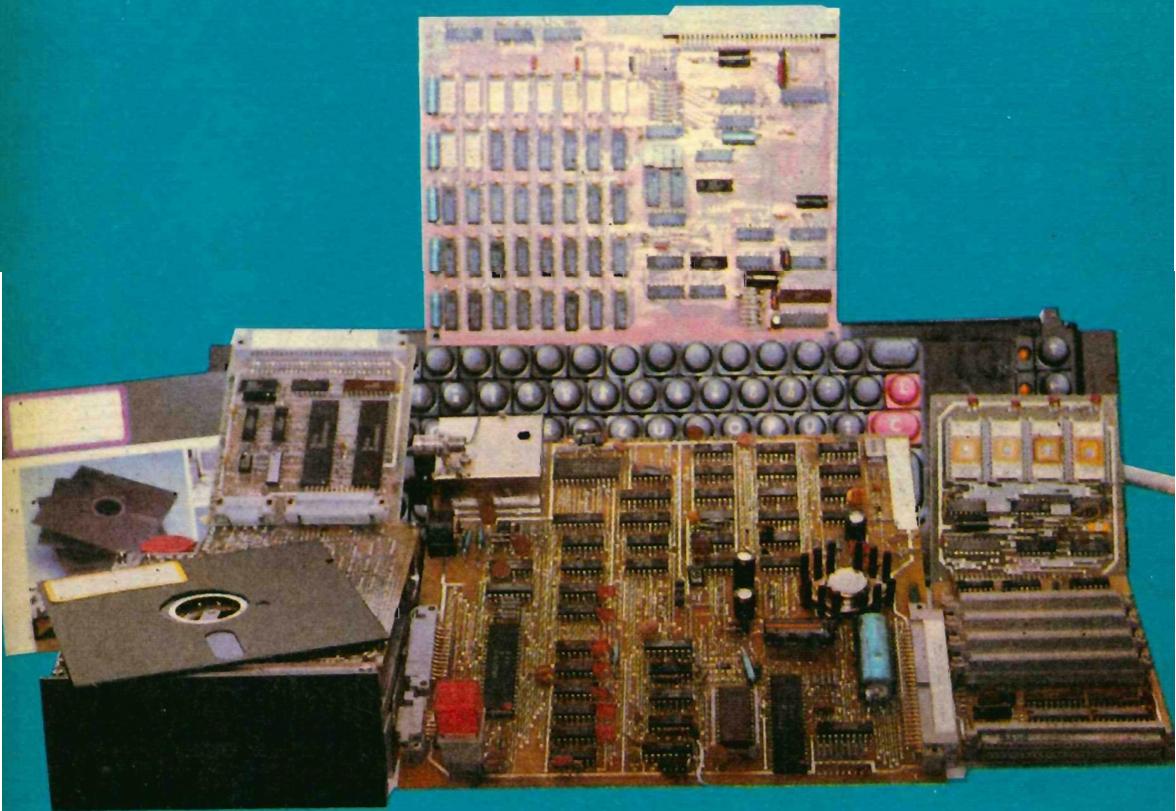


BETRIEBSSYSTEM CP/M AUFBAU UND ANWENDUNG



Manfred Kramer



6. Literatur

- [1] -; CP/M-Manual, Firmenschrift Digital Research. USA 1979
- [2] -; CP/M-Alteration Guide, Firmenschrift Digital Research. USA 1979
- [3] R. Zaks, CP/M-Handbuch mit MP/M. Sybex-Verlag 1981
- [4] R. Paul/H. Riedel, CP/M und Wordstar. Te-Wi-Verlag München 1981
- [5] B. Pohl, Vom Umgang mit CP/M. IWT-Verlag, Vaterstetten 1982
- [6] -; Utility software manual, Firmenschrift Microsoft Corp. USA 1978
- [7] M. Kramer, Praktische Mikrocomputertechnik. Berlin 1987
- [8] -; Anleitung für den Programmierer Teil I unter dem Betriebssystem SCPX 1526. Firmenschrift VEB robotron Karl-Marx-Stadt 1984
- [9] W. Dames, Beschreibung des Betriebssystems CP/A. Akademie der Wissenschaften der DDR, Institut für Informatik und Rechentechnik, Berlin 1985
- [10] W. Kammer/W. Spindler, RAM-Disk für K1520-Systeme. In: Mikroprozessortechnik, Berlin 2 (1988) Heft 3
- [11] H.-J. Bachmann, RAM-Speichererweiterung für Z 1013. In: Mikroprozessortechnik, Berlin 2 (1988) Heft 4
- [12] R. Brosig, Z-1013-Tastatur mit Raffinessen. In: Mikroprozessortechnik, Berlin 2 (1988) Heft 7
- [13] K. Schmidt, Betriebssystem SCP für Personalcomputer. Berlin 1988
- [14] D. Petschke, Textverarbeitung mit Personalcomputern. Berlin 1988
- [15] U. Hempel, Datenbanken mit Personalcomputern. Berlin 1988

Manfred Kramer

Betriebssystem CP/M Aufbau und Anwendung

**Brandenburgisches
Verlagshaus**

Kramer, M.;
Betriebssystem CP/M – Aufbau und Anwendung. –
Berlin: Brandenburg. Verl.-Haus 1990. –
128 S.; 6 Bilder, 2 Programme

ISBN 3-327-01024-2

1. Auflage
© Brandenburgisches Verlagshaus – Berlin, 1990
Lizenz-Nr. 5
Computersatz: System SCLX
Druck und buchbindereische Weiterverarbeitung:
Druckhaus Schöneweide – Berlin – 3 4375-0
Lektor: Steffen Würtenberger
Redaktionsschluß: 1. Dezember 1989
LSV 3539
Bestellnummer: 747 378 8
DDR 7,- M

7.80

Inhaltsverzeichnis

Vorwort	4
1. Einleitung	6
1. 1. Begriffe und Definitionen	6
1. 1. 1. Datenaufzeichnung auf Disketten	9
1. 1. 2. Namenskonventionen und Dateiarbeit	14
1. 2. Speicheraufteilung	16
1. 3. Programmablauf	18
1. 4. Fehlermeldungen	20
2. Programm-Module	21
2. 1. BIOS - die Anpassung an den Computer	21
2. 1. 1. Konsole, Druckeranschluß und serielle Schnittstellen	21
2. 1. 2. Diskettenlaufwerke und deren Anpassung	26
2. 1. 3. Virtuelle Diskettenlaufwerke im Speicher	34
2. 2. CCP - die Benutzeroberfläche	35
2. 3. BDOS - Kern des Betriebssystems CP/M	38
3. Kurzbeschreibung ausgewählter Programme	65
3. 1. Systemprogramme MOVCPM, STAT und PIP	65
3. 2. Power und ZSID	68
3. 3. Public-domain-Software	75
3. 4. WordStar	76
3. 5. M80, L80 und C80	80
4. Praxis-Tips	88
4. 1. Erstellen einer Systemdiskette	88
4. 2. Anpassung von Programmen (Installation)	89
4. 3. Rechnerkopplung	91
5. Programme	93
5. 1. Urlader	93
5. 2. CBIOS für Z 1013	94
6. Literatur	2. US

Vorwort

Fast 20 Jahre nach der Entwicklung des Mikroprozessors im Jahre 1971 ist auf dem Gebiet der 8-bit-Mikrorechner eine Stabilisierung eingetreten. Das bezieht sich vor allem auf die Struktur des technischen Aufbaus (Hardware) und das Betriebssystem CP/M. Es wurde zum Standard bei den 8-bit-Mikrorechnern. Selbst weit verbreitete Heimcomputer wie der C 64; für die eine Fülle von Software vorhanden ist, wurden mit einer zweiten CPU erweitert, um das Betriebssystem CP/M mit seinen leistungsfähigen Programmen nutzen zu können.

Stabilisierend wirkte bei der internationalen Entwicklung auch die Verschiebung der Interessen auf die 16-bit-Technik, wo Industrie und Amateure mit in etwa gleichem technischen Aufwand noch mehr Leistung angeboten bekommen. Jedoch kann der einzelne schon die Leistungsmöglichkeiten der 8-bit-CP/M-Rechner keinesfalls voll ausnutzen. Diese Technik wird deshalb trotz der rasanten Weiterentwicklung der Bauelemente noch für längere Zeit Bestand haben. Das zeigt sich z.B. auch in der Neuentwicklung und Umstellung der Fertigung des Mikroprozessors U 880 von der NMOS- auf die CMOS-Technologie. Die neuen, höher integrierten Speicherschaltkreise erleichtern zudem den Aufbau des Rechners, indem sich die Anzahl der nötigen Bauelemente drastisch vermindert. So liefern zwei Schaltkreise des Typs U 61464 ebenso einen Arbeitsspeicher von 64 KByte wie die 32 Stück U 256 in dem bekannten PC 1715 von robotron.

Das Disketten-Betriebssystem CP/M wurde Mitte der 70er Jahre von einer amerikanischen Firma für den Mikroprozessor 8080 entwickelt. In der DDR werden unterschiedliche zu CP/M kompatible Betriebssysteme verwendet, die auf den Prozessor U 880 umgestellt wurden und dadurch keine Urheberrechte Dritter verletzen, z.B. BCU vom Werkzeugmaschinenbau, mikro-DOS von der Wilhelm-Pieck-Universität Rostock, CPA von der Akademie der Wissenschaften usw. Am weitesten verbreitet ist natürlich SCP

in den Betrieben, mit denen die Büro- und Personalcomputer von robotron seit etwa 1984 ausgeliefert werden.

Für den Amateur rückt dieses Betriebssystem aus zwei Gründen zunehmend in den Mittelpunkt des Interesses:

- Durch Ablösung älterer Floppy-Disk-Laufwerke in den Betrieben gelangen diese an den Amateur.
- Das Angebot leistungsfähiger, billiger Speicherschaltkreise im Amateurbedarfshandel erlaubt es, Diskettenlaufwerke im RAM- oder ROM-Speicher des Computers zu simulieren und so CP/M mit Kassetten als Speichermedium zu nutzen.

Der wichtigste Grund für das anhaltende Interesse ist die große Fülle von auch in der DDR vorhandener Software für dieses Betriebssystem. Die bekannten Programme "Wordstar", "Turbo-Pascal", "Dbase" usw. interessieren auch den Amateur, weil er sie aus der internationalen Computerliteratur oder die vergleichbaren Erzeugnisse von robotron von der Arbeit her kennt. Ein wichtiger Aspekt ist dabei auch die Zusammenarbeit mit anderen Amateuren und die Austauschbarkeit der Programme unabhängig vom jeweiligen Rechner.

Das Betriebssystem CP/M ist nicht auf einen bestimmten Mikrorechner festgelegt, sondern kann auf jedem Computer mit dem U 880 und mehr als 20 kByte RAM-Speicher laufen. Die Broschüre soll allen Interessenten, auch Arbeitsgemeinschaften und Besitzern von Mikrorechnern der KC-Typen, am konkreten Beispiel des Z 1013 die notwendigen Informationen und Kenntnisse über die interne Struktur des Betriebssystems zur Anpassung eines Eigenbau- oder industriellen Heimcomputers vermitteln. Kurze Erläuterungen bekannter Programme runden die Ausführungen ab. Hinweise zu Druckfehlern oder sonstige Mitteilungen werden gern unter folgender Adresse entgegengenommen: M. Kramer, PSF 156, Berlin 1136.

Berlin, im August 1989

Manfred Kramer, Y23VO

1. Einleitung

Das Betriebssystem eines Computers ist das Programm, das dem Bediener als Schnittstelle für die Nutzung aller Möglichkeiten des Rechners zur Verfügung steht. Es organisiert die Initialisierung aller System- und Gerätefunktionen, steuert die Zusammenarbeit aller angeschlossenen Geräte, die Auswertung von Kommandos und Parametereingaben sowie die Eingabe, Ausgabe und Speicherung von Daten und Programmen.

1.1. Begriffe und Definitionen

Bild 1 zeigt die materiellen Bestandteile ("Hardware") eines CP/M-Computers in der Systematik ihrer Aufgabenstellung. Bekanntlich wird die Leistungsfähigkeit eines Computers von den

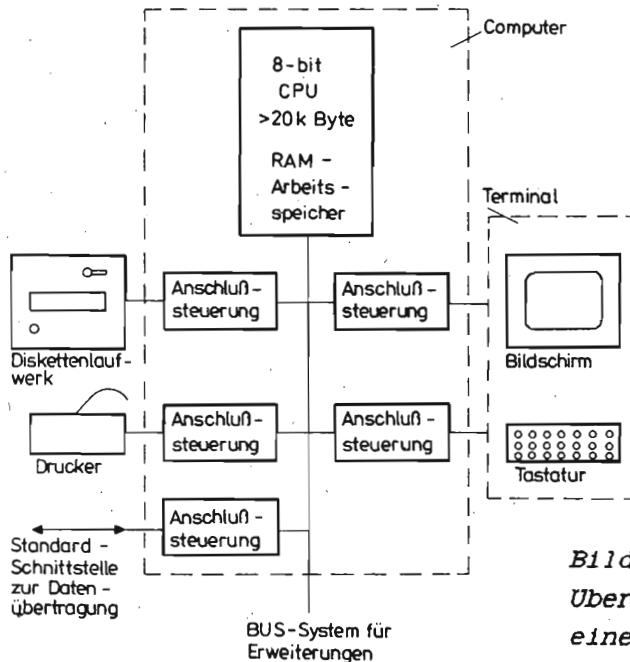


Bild 1
Übersichtsschaltbild
eines CP/M-Computers

verfügbarer Programmen (Software) im gleichen Maße bestimmt wie von den technischen Daten (Arbeitsgeschwindigkeit, Speichergröße usw.) der Hardware. Die konstruktive Gestaltung ist bei den einzelnen Fabrikaten unterschiedlich, die grundsätzliche Funktion jedoch gleich. Der Computer im engeren Sinne besteht aus Mikroprozessor, Speicher und Anschlußmöglichkeiten für Geräte zur Bedienung des Computers, sowie zur Anzeige, Ein- und Ausgabe von Daten. Als Mikroprozessoren wurden anfangs der 8080, später der Z 80 und der 8085 eingesetzt. Diese beiden Schaltkreise sind bezüglich der Software abwärtskompatibel, d. h., der U 880 kann die Programme für den 8080 ohne Änderung abarbeiten. Um neue oder modernisierte Programme auch auf älteren Geräten betreiben zu können, werden sie oft mit dem Befehlssatz des 8080 geschrieben. Andere 8-bit-Prozessoren wie der bekannte 6502 oder der U 882 verwenden intern einen anderen Befehlssatz und können deshalb das Betriebssystem CP/M und die darunter lauffähigen Programme nicht nutzen. Für die 16-bit-Prozessoren 8086 und 68000 gibt es Simulationsprogramme, mit denen CP/M-Programme und -Disketten betrieben werden können.

Der Arbeitsspeicher besteht meist aus einem Schreib-/Lese-Speicher (RAM) von 64 kByte. Größere oder kleinere Speicher sind möglich, jedoch nicht sehr verbreitet. Da der RAM seinen Dateninhalt mit der Abschaltung der Betriebsspannung verliert, ist noch ein ROM (Festwertspeicher) vorhanden, der das Laden des Betriebssystems unterstützt. Die in Bild 1 als Anschlußsteuerung bezeichneten Baugruppen können je nach Rechnertyp komplett Leiterkarten oder einzelne Schaltkreise sein, so kann eine CENTRONICS-Schnittstelle mit 2 TTL-Schaltkreisen aufgebaut werden. Mit modernen Bauelementen kann daher der in Bild 1 als Computer bezeichnete Teil auf einer einzigen Leiterplatte mit den Maßen 95 mm * 170 mm untergebracht werden.

Die Bezeichnung der Baugruppen hat oft historische Wurzeln. Die wichtigste Baugruppe außerhalb des eigentlichen Computers ist in Bild 1 mit dem Begriff Terminal bezeichnet, auch der Ausdruck Console ist üblich. Neben der gezeigten Konfiguration sind auch einige andere Möglichkeiten üblich, z.B. können auch Tastatur und Diskettenlaufwerk konstruktiv mit dem Computer vereinigt und nur der Bildschirm ein getrenntes Gerät sein.

Die Tastatur erlaubt dem Bediener die Eingabe von Befehlen und Daten in den Rechner. Diese bestehen aus einzelnen Zeichen, die zusammengesetzt Wörter einer Programmiersprache oder andere "Datenblöcke" sein können. Es sind sowohl Buchstaben und Ziffern als auch Sonderzeichen wie Punkt und Fragezeichen usw. zulässig. Man spricht deshalb von einer alphanumerischen Tastatur. Zur Rückmeldung und Datenausgabe an den Bediener ist der Bildschirm des Computers das wichtigste Gerät. Moderne Rechner verfügen über eine Bildschirmansteuerung, die nicht nur die alphanumerischen Zeichen der Tastatur, sondern beliebige grafische Darstellungen anzeigen kann.

Die angedeutete Zusammenfassung von Bildschirm und Tastatur ist vor allem bei Grafikarbeitsplätzen üblich. Die Bezeichnung Terminal bedeutet soviel wie Abschluß oder Endgerät. Das Terminal enthält dann meist einen eigenen Computer. Dies ist zweckmäßig, weil Grafikschaltungen viel Speicherplatz benötigen, der sonst beim Arbeitsspeicher des Computers fehlen würde. Schon bei einfacher schwarz/weiß-Darstellung mit 640 Bildpunkten waagerecht und 200 senkrechten Bildpunkten ergeben sich z. B. etwa 16 KByte. Ein für den Amateur interessanter Anwendungsfall ist die Nutzung eines Heimcomputers oder einer Schreibmaschine als Terminal. Über eine standardisierte Schnittstelle V.24 ist Ankoppln an den CP/M-Computer problemlos möglich.

Erläuterungen sind oft verständlicher, wenn sie an einem konkreten Beispiel erfolgen. Es sind deshalb einige Programme beigelegt, an denen die Einzelheiten des Betriebssystems näher gezeigt werden. Die Programme im Abschnitt 5. wurden zunächst für einen Eigenbaucomputer nach [7] erarbeitet, ließen sich jedoch auf den Mikrorechnerbausatz Z 1013 sehr einfach umstellen, weil sich die Schaltungskonzepte stark ähneln. Die zugrundeliegende Computerkonfiguration besteht aus der auf 64 kByte DRAM erweiterten Grundplatine [11], die um folgende Leiterplatten des Computers nach [7] erweitert wurde:

- Ansteuerkarte für 2 Diskettenlaufwerke mit U 8272
- Ansteuerkarte für Grafikbildschirm mit U 82720
- Ansteuerkarte für serielle Schnittstellen V.24 und IFSS.

Anstelle der letztgenannten kann man nach geringen Programmänderungen sicher auch eine zum Bausatz gehörende Leiterplatte

einsetzen. Als Tastatur wird die Lösung nach [12] verwendet, außerdem ist eine RAM-Floppy-Karte nach [10] vorhanden. Die Schaltung der Grundplatine wurde so geändert, daß ein aus Gattern aufgebautes RS-Flipflop nach dem Reset den RAM erst ab Adresse 8000 einschaltet und dafür ein EPROM 2716 ab Adresse 0 arbeitet. Dieser kopiert seinen Inhalt auf eine Adresse oberhalb von 8000 und startet das Ladeprogramm für das erste Diskettenlaufwerk. Beim Programmlauf wird mit dem ersten IORQ-Signal das Flipflop umgeschaltet, sperrt den EPROM mit OE und gibt den RAM ab Adresse 0 frei. Die Taktfrequenz wurde auf 4 MHz umgestellt. Der Bildschirm-RAM ist auf Adresse FC00 verlegt und die Ausgabe auf 64 Spalten mit 16 Zeilen umgestellt, weil dafür Programme von anderen Computern (BC, PC, [7]) vorliegen und die Nutzung dann auch ohne Grafikkarte möglich ist.

1. 1. 1. Datenaufzeichnung auf Disketten

Das Diskettenlaufwerk in Bild 1 bewirkt den qualitativen Unterschied eines CP/M-Computers zum Heimcomputer mit Kassetten als Speichermedium. Disketten sind runde Plastescheiben, die ähnlich wie beim Magnetband mit einem magnetisierbaren Stoff beschichtet sind. Ubliche Bezeichnungen sind auch Folienspeicher und Floppy Disk. Die Scheiben rotieren in Schutzhüllen und können mit Hilfe der Mechanik des Laufwerks und der zugehörigen Elektronik vom Rechner mit Informationen beschrieben und später beliebig oft gelesen und beschrieben werden. (Die Lebensdauer einer Diskette ist natürlich durch den Verschleiß begrenzt.) In noch höherem Maße als bei Kassetten sind die Diskettenlaufwerke hochempfindliche Präzisionsgeräte, die mit Vorsicht zu behandeln und vor allem vor Verschmutzung zu schützen sind. Die Bilder 2 und 3 zeigen schematisch den Anschluß und den Aufbau eines Diskettenlaufwerkes. Die technischen Daten der Laufwerke haben sich im Laufe ihrer nun etwa 15jährigen breiten Anwendung ständig verbessert. Zur Zeit (1989) werden meist 5,25"-Laufwerke (" - Zoll = 25,4 mm) mit 2 Schreib-/Leseköpfen verwendet, die bei 300 Umdrehungen je min in 80 Spuren die Daten mit 250 KHz aufzeichnen. In den Spuren werden die Daten in Blöcken, den

sogenannten Sektoren, gespeichert. Den Anfang der Spur kennzeichnet das Indexloch, das fotoelektrisch abgetastet wird (siehe Bild 3). Bei den "hardsektorierten" Disketten wird jeder Sektor durch ein Loch angekündigt. Ist nur ein Loch vorhanden, so sind die Disketten softsektorisiert, d.h. die Sektoren werden durch Programm bezüglich Anfang und Länge festgelegt. Die Sektoren können dann unterschiedlich groß sein. Ursprünglich wurden auf 8"-Disketten 77 Spuren mit 26 Sektoren zu je 128 Byte verwendet. Jeder Sektor ist durch einen unmittelbar davor aufgezeichneten Identifikationsblock, der Diskettenseite, Spur und Sektornummer enthält, gekennzeichnet. Durch Vergrößerung der Sektoren fallen Identifikationsblöcke weg, so daß bei konstanter Spurlänge die für Aufzeichnungen nutzbare Kapazität steigt. Es sind deshalb auch Sektorformate mit 256, 512 und 1024 Byte üblich. Anfangs, und auch heute noch bei älteren Computern mit 8"-Laufwerken anzutreffen, erfolgte die Aufzeichnung der Datenimpulse mittels Frequenzmodulation (FM) entsprechend Bild 4. Im Abstand von 4 µs liegen Taktimpulse. Wenn eine

BUS - System	Anschluß - steuerung		Elektronik (Kopf- und Motor - steuerung)	Mechanik (Antriebe, Verriegelungen usw.)
	-----	reserve	1	
	-----	Kopf laden / head load	2	
	-----	Auswahl / device select 3	3	
	-----	Index	4	
	-----	Auswahl / device select Ø	5	
	-----	Auswahl / device select 1	6	
	-----	Auswahl / device select 2	7	
	-----	Motor an / motor on	8	
	-----	Schrittrichtung / direction	9	
	-----	Schritt / step	10	
	-----	Schreibdaten / write data	11	
	-----	Schreibfreigabe / write gate	12	
	-----	Spur / track Ø	13	
	-----	Schreibschutz / write protected	14	
	-----	Lesedaten / read data	15	
	-----	Seite 1 / side 1 select	16	
	-----	Bereitschaft / ready	17	
			18	
			19	
			20	
			21	
			22	
			23	
			24	
			25	
			26	
			27	
			28	
			29	
			30	
			31	
			32	
			33	
			34	
Computer		Kabel (alle ungeraden an ØV) (alle Signale neg. Logik)		
		Laufwerk (Stift-Nr. für 5,25")		

Bild 2 Anschlußbild eines Diskettenlaufwerks

logische "1" geschrieben werden soll, kommt ein Datenimpuls dazwischen, so daß in diesem Falle die doppelte Frequenz entsteht. Zur Rückgewinnung der Daten werden die von der Diskette gelesenen Taktimpulse zur Synchronisation verwendet, damit wirken sich Schwankungen der Drehzahl nicht so stark aus. Bekanntlich ist die maximale Aufzeichnungsfrequenz bei magnetischen Medien begrenzt. Man kann jedoch die Aufzeichnungsdichte verdoppeln, ohne daß an das magnetische Material höhere Anforderungen gestellt werden, wenn man die bei FM immer vorhandenen Taktimpulse wegläßt und an deren Stelle auch Datenimpulse (sofern im Bitmuster vorhanden) schreibt. Die maximale Frequenz mit 2 µs je Bitzelle bleibt dabei gleich. Nur bei mehreren O-Impulsen muß ein Taktimpuls dazwischen geschrieben werden, um die Daten lesen zu können.

Dieses Aufzeichnungsverfahren heißt "modifizierte Frequenzmodulation" (MFM). Bezuglich der Daten spricht man von doppelter Dichte (englisch: double density) gegenüber der "einfachen Dichte" (englisch: single density) bei FM. Die in letzter Zeit

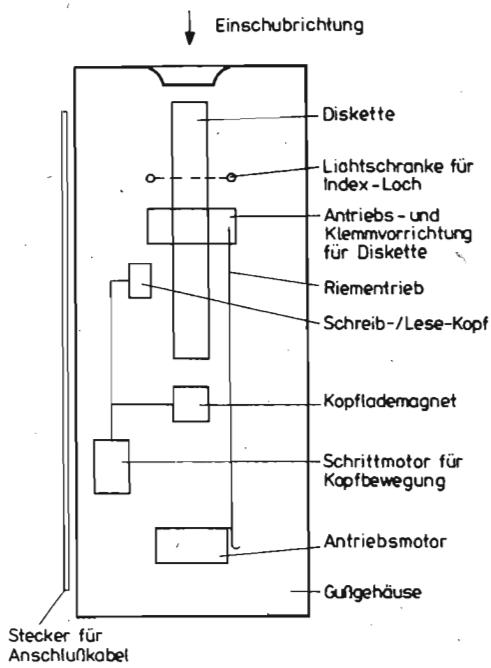


Bild 3

Draufsicht auf ein Diskettenlaufwerk (Prinzip)

aufgekommenen "hochdichten" Laufwerke arbeiten mit höherer Drehzahl und anderem Diskettenmaterial.

Auch bei der Mechanik der Laufwerke vollzog sich eine Weiterentwicklung zu höherer Leistung. Bei 5,25"-Disketten wurde zwar schon fast von Anfang an die Aufzeichnung mit doppelter Dichte benutzt, jedoch beherrschte man zunächst nur 40 Spuren. Später ging man auf zweiseitige Aufzeichnung und dann durch größere Präzision auch auf 80 Spuren über. Die unterschiedliche Dichte der Spuren auf gleich großen Disketten (5,25") wird auf einen Zoll bezogen, d.h., 48 TPI (tracks per inch = Spuren pro Zoll) entsprechen 40 Spuren insgesamt, 96 TPI entsprechen 80 Spuren auf der Diskette. Diese Angabe ist also von der Modulationsart unabhängig, erst beide Angaben zusammen ergeben das exakte Maß für die maximale Kapazität der Diskette. Bei der Drehzahl von 300 min^{-1} und $4 \mu\text{s}$ je bit ergibt sich bei einer 5,25"-Diskette ein Maximum von 6250 Bytes je Spur. Die Einteilung der Spuren in Sektoren überlässt der Hersteller dem Anwender. Dieser Vorgang heißt "Formatieren" und erlaubt es, die gleichen Disketten mit unterschiedlichen Betriebssystemen und auf unterschiedlichen Laufwerken zu betreiben. Durch "Lücken" zwischen den Sektoren erfolgt die Synchronisation der Aufzeichnung, d.h., bei Verwendung eines großen Sektors anstelle mehrerer kleiner werden höhere Anforderungen an Mechanik und Elektronik gestellt.

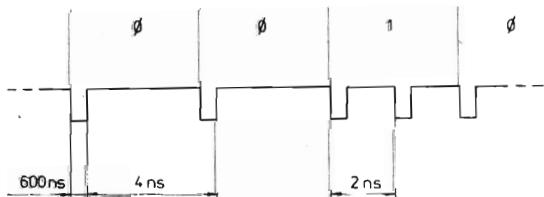


Bild 4
Aufzeichnungsverfahren
FM (Frequenzmodulation)

Die oben genannten, international üblichen Bezeichnungen stehen in dieser Form auch meist auf den Diskettenpackungen, so daß der Amateur ihnen vor firmeninternen, abweichenden, teilweise sogar irreführenden Bezeichnungen (z.B. "1.6-er") den Vorzug geben sollte. Ist eine Diskette vom Hersteller als einseitig gekennzeichnet, so heißt das nur, daß die 2. Seite nicht geprüft wurde und der Hersteller keine Garantie für die Fehlerfreiheit dieser Seite gibt. Fehler sind jedoch trotzdem äußerst selten.

Beispiele für Diskettenbezeichnungen:

8" SS SD	- 8 Zoll, einseitig, einfache Dichte
5,25" DS DD 96 TPI	- 5,25 Zoll, doppelseitig, doppelte Aufzeichnungsdichte, 80 Spuren

Im Gegensatz zur üblichen Praxis bei Heimcomputern setzt man bei CP/M-Computern einen möglichst kleinen ROM-Speicher ein. Das Betriebssystem wird meist auch von Diskette geladen, um flexibel auf unterschiedliche Computer oder Anforderungen reagieren zu können; ein Programm auf Diskette lässt sich einfacher ändern als im EPROM. Der dafür nötige Speicherplatz auf der Diskette ist für Anwenderprogramme nicht mehr nutzbar. In der Praxis werden meist 2 bis 3 Spuren am Anfang der Diskette dafür genutzt, so daß zur exakten Angabe des Diskettenformats eines CP/M-Computers auch die Zahl der Systemspuren gehört. Durch die unterschiedlichen Laufwerke mit 8", 5,25", 3,5" und 3", ein- oder zweiseitig mit unterschiedlich großen Sektoren und verschiedenen Modulationsverfahren, entstand eine Vielzahl von Diskettenformaten, die das Betriebssystem CP/M unterstützt.

Beispiele für Diskettenformate:

Bürocomputer A 5120	PC 1715
3 Laufwerke 5,25" SS DD 48 TPI	2 Laufwerke 5,25" DS DD 96 TPI
16 Sektoren zu 256 Byte/Spur	2*5 Sektoren zu 1024 Byte/Spur
3 Systemspuren	2 Systemspuren
143 KByte je Diskette nutzbar	780 KByte je Diskette nutzbar

Manchmal ist es notwendig, die gespeicherten Daten gegen versehentliches Löschen durch den Bediener zu schützen. Dazu enthält die Diskettenhülle eine optoelektronisch abgetastete Kerbe. Bei 8"-Disketten muß sie offen, bei 5,25"-Disketten zugeklebt sein, um das Schreiben zu sperren. Die Elektronik des Laufwerks gibt ein Fehlersignal an die Ansteuerung des Computers, wenn trotzdem versucht wird zu schreiben.

Diese Erläuterungen sollten eine Übersicht zu Disketten und Laufwerken geben. Als Amateur sollte man ein von der Industrie ausgesondertes Laufwerk nur erwerben, wenn Unterlagen mit technischen Daten verfügbar sind.

1. 1. 2. Namenskonventionen und Dateiarbeit

Sowohl Hardware als auch Software erhalten bei Diskettenbetriebssystemen Namen, mit denen sie von Programmen bezeichnet werden können.

Man unterscheidet beim Computer logische und physikalische Geräte. "Logisch" ist z.B. der Drucker als Ausgabegerät allgemein, "physikalisch" sind dann der Typ K 6313 mit CENTRONICS-Schnittstelle oder ein Fernschreiber. Die Namen röhren oft von der englischen Bezeichnung der zugehörigen Geräte her. Die nachfolgende Tabelle führt die Namen der logischen Geräte auf. Es ist zu beachten, daß der Doppelpunkt mit zum Namen gehört!

Logische Geräte:

CON: Konsole; Bedienergerät, besteht logisch aus einem Eingabe- und einem Ausgabekanal, im allgemeinen Bildschirm und Tastatur.

RDR: Leser für Zeichen; (engl. reader) früher Lochstreifenleser, heute serielle Schnittstelle, z.B. V.24 oder 20-mA-Stromschleife.

PUN: Zeichenausgabe; früher Lochstreifenstanzer, (engl. punch) heute serielle Schnittstelle, z.B. V.24.

LST: Drucker (engl. list-device).

Den logischen Geräten können vom Betriebssystem bis zu je 4 physikalische Geräte zugeordnet werden, zwischen denen man dann per Software umschalten kann (siehe Abschnitt 2).

Bei der Software ist die Bezeichnungssystematik vielseitiger. Alle sinngemäß zusammengehörigen Daten bilden unabhängig von ihrer physischen Aufteilung in große oder kleine Sektoren eine Datei (auch englisch: file = Ablage). Sowohl die Anweisungen eines Programmes als auch Daten, z.B. eine Gruppe von Meßwerten, können logisch zusammengehören und zu verschiedenen Dateien gehören. Um als Bediener die Dateien unterscheiden zu können, erhalten alle Dateien einen Namen, mit dem sie im Inhaltsverzeichnis (englisch: directory) der Diskette dokumentiert werden. Der Name von Dateien besteht aus zwei Teilen, gewissermaßen einem Vor- und einem Nachnamen, die durch einen

Punkt getrennt sind und die der Programmierer bei der Erstellung festlegt. Die Zeichen <[(,;:=;*?%/\)]> sind im Namen nicht zulässig, weil ihnen an anderer Stelle eine feste Bedeutung zu kommt, ansonsten muß nur das erste Zeichen des Namens ein Buchstabe sein. Der Teil vor dem Punkt ist der eigentliche Dateiname, er kann maximal 8 Stellen lang sein. Der Punkt und die danach stehende Namenserweiterung (maximal 3 Stellen, auch als Extend bezeichnet) können auch fehlen. Die Namenserweiterung gibt jedoch bei den meisten Dateien Auskunft über die Art. Oft legen Programme schon die Namenserweiterung fest, z. B.:

name.COM Programmdateien

Diese Dateien enthalten Programme, die durch Eingabe des Namens über die Tastatur (ohne .COM) von der Diskette in den Speicher geladen und dann gestartet werden. Bei der Erstellung solcher lauffähiger Programme mittels Assembler und Linker oder Compiler wird die Endung .COM automatisch erzeugt.

name.MAC

Es gibt einen Macro-Assembler (M80), der auf Quellprogramme in Assemblersprache mit dieser Endung voreingestellt ist.

name.REL

Mit diesem Extend sind verschiebbliche Programme gekennzeichnet, die vom Assembler M80 erzeugt wurden.

name.PRN (englisch: printer)

Der Assembler erzeugt auf Anforderung auch Dateien für den Drucker.

name.BAK

Wenn man mit einem Editor, z.B. WORDSTAR, einen Text bearbeitet, behält dieser die vorherige Bearbeitungsversion zur Sicherheit als sogenannte back-up-Datei mit dem Extend .BAK. Der festgelegte Name geht immer auf die neueste Version des Textes über.

name.BAS

Diese Namenserweiterung erhalten BASIC-Programme vom Interpreter MBASIC von Microsoft. Die Dateien können in ASCII oder im internen Format des Interpreters ausgegeben werden.

name.PAS

Der Compiler TURBO-PASCAL vergibt für Quellprogramme diese Endung, wenn man nicht eine andere festlegt.

Bei der Auswahl eigener Namenserweiterungen sollte daraus so weit wie möglich der Typ hervorgehen, z.B.: name.TXT für allgemeine Texte, name.DOK für Dokumentationen usw.

Wenn man ein Programm in Assemblersprache erarbeitet, entsteht eine ganze Gruppe von Dateien, die den gleichen Namen haben und sich nur im Extend unterscheiden. Solche Dateigruppen können vorteilhaft mit einem "Joker" im Namen bezeichnet werden. Durch das Zeichen * werden alle bis zum Trennzeichen folgenden Zeichen im Namen ersetzt.

Beispiele:

*.BAS erfaßt alle BASIC-Programme

W*.* erfaßt alle Dateien, deren erster Buchstabe ein W ist, z.B. die zu WORDSTAR gehörigen WS.COM, WSMGS.OVR und WSOVLY1.OVR

Mit dem ? wird ein einzelnes Zeichen an der entsprechenden Stelle im Namen ersetzt. Beispiel:

TEST.?AS erfaßt BASIC- und PASCAL-Programme

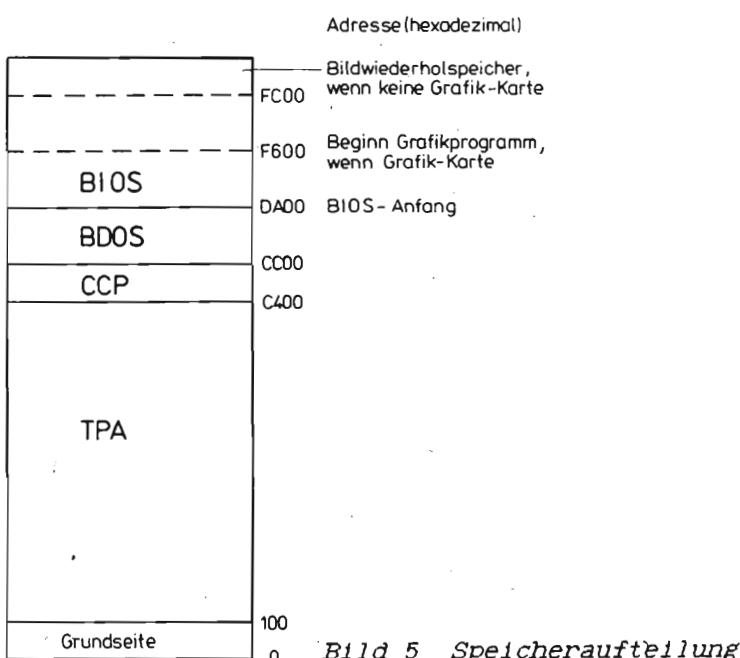
Die vereinfachte Bezeichnung wird von älteren Programmen nicht immer unterstützt.

Dateien können mit den sogenannten Attributen "Schreibschutz" und "System" versehen werden. Im Gegensatz zur Sperre der Diskette durch die Kerbe in der Hülle wirkt dieser Schreibschutz nur auf die betroffene Datei, nicht auf die ganze Diskette. Systemdateien werden durch ihr Attribut "versteckt", d.h. sie sind im Inhaltsverzeichnis nicht sichtbar.

1.2. Speicheraufteilung

Ein Vorzug des Betriebssystems CP/M ist die saubere Gliederung in Module mit exakten Schnittstellen. Bild 5 zeigt die Aufteilung des Speichers in funktionell zusammengehörige Bereiche, wobei die angegebenen Adressen sich auf das Beispiel-CBIOS in Abschnitt 5 beziehen. Der Adressbereich von 0 bis OFFH heißt Grundseite und enthält die zwei wichtigsten Einsprünge ins Betriebssystem, Flagbytes, mit denen Gerätezuordnungen gespeichert werden sowie Pufferbereiche:

Adresse (HEX)	Funktion
0000-0002	Warmstart
0003	IOBYTE, enthält die Zuordnung zeichenorientierter Geräte (LST:, PUN:, RDR:)
0004 (low-Teil)	aktueller Laufwerk
0004 (high-Teil)	aktuelle Benutzernummer
0005	Sprung zum BDOS
0038	Restartadresse für Debugger ZSID u. ä.
005C-007C	Dateisteuerblock (engl. filecontrol block, fcb)
007D-007F	wahlweise aktueller Block
0080-00FF	Zwischenspeicher, Puffer (engl. direct memory access, dma)



Erläuterungen zur Funktion folgen im Zusammenhang mit der Darstellung der Systembestandteile.

Ab Adresse 100H liegt freier Speicher, den der Anwender mit seinen Programmen nutzen kann. Dieser Bereich heißt TPA.

Diese Abkürzung wurde aus der englischen Bezeichnung transient program area gebildet, was man mit "Feld für ladbare Programme" übersetzen kann. Beim Aufruf eines Programms mit seinem Namen wird es ab der Adresse 100H geladen und danach auf diese Adresse gesprungen, d.h. beim Betriebssystem CP/M beginnen alle Anwenderprogramme immer auf Adresse 100H. Die Größe des TPA ist von der konkreten Rechnerkonfiguration abhängig, beträgt aber meist mehr als 48 KByte, weil verschiedene Programme mindestens diese Speichergröße erfordern.

Die Abkürzung CCP steht für console command processor. Das ist ein Bestandteil des Betriebssystems, mit dem die notwendigsten Bedienerfunktionen ausgeführt werden können. Er hat unabhängig von der konkreten Adresse im Speicher eine Länge von 2 KByte. Der Modul BDOS ist der Kern des Betriebssystems, die Bezeichnung steht als Abkürzung von basic disk operating system (= grundlegendes Disketten-Betriebssystem). Er hat eine Länge 3,5 KByte und verfügt über eine Vielzahl von Funktionen, die man alle über die Adresse 5 erreichen kann und die im Abschnitt 2.3. ausführlicher behandelt werden.

Für den Amateur ist das BIOS der wichtigste Teil des Betriebssystems, weil hier die Anpassung an die unterschiedlichen Computer-Konfigurationen erfolgt. Die Abkürzung kommt von basic input/output system, frei übersetzt mit "Steuerprogramme für die Peripheriegeräte", worunter Tastatur, Bildschirm, Diskettenlaufwerke und serielle Schnittstellen zu verstehen sind.

Die Länge des BIOS ist rechnerabhängig. Wenn z.B. mehrere Drucker mit unterschiedlichen Schnittstellen vorhanden sind und viele verschiedene Diskettenformate benutzt werden sollen, benötigt dieser Modul natürlich mehr Speicher, CCP und BDOS verschieben sich nach unten, und der TPA verkleinert sich entsprechend.

1.3. Programmaufruf

Man unterscheidet residente und transiente Kommandos. Die residenten sind im Systembestandteil CCP vorhanden, die transienten sind Programme, die erst noch von Diskette in den Speicher

geladen werden müssen. Wie schon erwähnt, können nur Programme mit dem extend .COM als transiente Befehle arbeiten. Der Aufruf erfolgt nur mit dem Programmnamen, also ohne Eingabe des Extend. Den Abschluß der Eingabe bildet die Taste "Wagenrücklauf" (CR von engl. carriage return), auch oft mit "Enter" oder "Return" bezeichnet. Erst diese Taste macht die Eingabe gültig, vorher ist noch durch die Taste "Rückschritt" und Überschreiben eine Korrektur der Eingabe möglich.

Bei den eingegebenen Namen werden große und kleine Buchstaben nicht unterschieden, d.h., es ist gleichgültig, welche man wählt. Das Betriebssystem prüft zunächst, ob es sich um ein residentes Kommando handelt und verbleibt in diesem Falle im CCP zur Ausführung. Ist das nicht der Fall, wird eine entsprechende COM-Datei auf Diskette gesucht. Unabhängig von der vorhandenen Anzahl ist immer nur ein Diskettenlaufwerk aktiv. Beim Systemstart ist dies das Laufwerk A:.

Durch Eingabe des Laufwerksbezeichners kann das aktuelle Laufwerk gewechselt werden, z.B. C: (CR). Befindet sich ein Aufruf nicht als .COM-Datei auf dem aktuellen Laufwerk, so muß ihm der entsprechende Laufwerksbezeichner vorangestellt werden. Ist ein entsprechendes Programm nicht vorhanden, erscheint nach erfolgloser Suche als Fehlermeldung die Wiederholung des Kommandos mit nachgestelltem Fragezeichen.

Verschiedene Programme arbeiten mit anderen Dateien zusammen, z. B. ein Editor mit verschiedenen, in Bearbeitung befindlichen Textdateien. Beim Aufruf von Kommandos können deshalb noch Parameter erforderlich sein, um den Befehl genauer festzulegen. Die Parameter sind vom Programmnamen durch ein Leerzeichen zu trennen.

Beispiel:

Das aktuelle Laufwerk sei b:, Eingabe:

a:ws test.mac (CR)

Dieser Befehl lädt das Programm WS.COM (WORDSTAR) von Laufwerk a: in den Speicher auf Adresse 100H, startet es und lädt anschließend von Laufwerk b: die Datei TEST.MAC zur Bearbeitung. Manche Programme verwenden auch mehrere Parameter.

1. 4. Fehlermeldungen

Bei einem so komplizierten System, wie es ein Computer ist, sind technische Störungen und Fehler des Bedieners unvermeidlich. Durch die Anwenderprogramme und das Betriebssystem erfolgt eine ständige Überwachung der einzelnen Funktionen und im Falle eines Fehlers dessen Meldung an den Bediener.

Im Interesse eines geringen Speicherplatzbedarfs sind die Fehlermeldungen des Betriebssystems sehr knapp formuliert. Bei den Anwenderprogrammen kann man wegen der Möglichkeit des Nachladens der Meldungen von Diskette großzügiger vorgehen. Nachfolgend sind die Fehlermeldungen des Betriebssystems erläutert.

Bad Load	Fehlermeldung des CCP beim Laden von Programmen.
Bad Sector	Fehler beim Lesen oder Schreiben eines Sektors auf Diskette. Ursachen können z. B. sein: Diskette nicht formatiert, Toleranzen der Laufwerke usw.
Bdos Err on d:	Allgemeiner Fehler beim Diskettenzugriff, meist folgt noch eine weitere Meldung, siehe unten. d: steht für das betroffene Laufwerk, z. B. A:. Erfolgt die Meldung aus einem laufenden Programm heraus, so wird dieses mit ^C abgebrochen, bei anderen Tasten wird der Fehler ignoriert.
R/O	<u>Diskette</u> ist schreibgeschützt, weil sie z. B. ohne nachfolgenden Warmstart gewechselt wurde.
Select	Ein nicht vorhandenes Laufwerk wurde angewählt.
File R/O	<u>Datei</u> ist schreibgeschützt, lässt sich nicht ohne Änderung dieses Attributs löschen.
no File	erscheint, wenn die bei einem Kommando als Parameter angegebene Datei nicht gefunden wurde.
filename?	Ein aufgerufenes Programm oder eine als Parameter eingegebene Datei wurde nicht gefunden.
File exists	Wenn mit dem Kommando REN eine Datei erzeugt werden soll, die bereits vorhanden ist, erscheint diese Fehlermeldung. Abhilfe: Löschen oder anderer Name.

2. Programm-Module

2.1. BIOS - die Anpassung an den Computer

Das BIOS enthält die Programme zum Betrieb der Peripheriegeräte und kann entsprechend der Rechnerkonfiguration unterschiedlich groß sein. Um die Geräte unabhängig von der Länge des BIOS betreiben zu können, enthält es am Anfang eine Tabelle mit Sprungbefehlen zu den zugehörigen Unterprogrammen, deren Reihenfolge vorgegeben ist. Auch die Parameterübergabe ist festgelegt, d. h., es liegt fest, welche Register mit Adressen oder Zeichen geladen werden. Dadurch entsteht eine exakte Schnittstelle zum restlichen Teil des Betriebssystems und zu den Anwenderprogrammen.

Das BIOS ist im Speicher unmittelbar hinter dem Modul BDOS angeordnet, dadurch ist diesem der Anfang bekannt. Die Adressen der Einsprungtabelle werden vom BDOS und in Ausnahmefällen auch von Anwenderprogrammen als Unterprogramme aufgerufen. Die konkrete Anordnung der Einsprungtabelle ist dem Programmbeispiel CBIOS zu entnehmen.

Man unterscheidet Peripheriegeräte, welche die Daten zeichenweise behandeln, z. B. eine Tastatur, und solche, die nur mit Datenblöcken arbeiten, wie Disketten- oder Festplattenlaufwerke.

Auf die Parameterübergabe und andere Einzelheiten der Unterprogramme des CBIOS wird nachfolgend eingegangen.

2.1.1. Konsole, Druckeranschluß und serielle Schnittstellen

Bei den zeichenorientierten Geräten wird bei jedem Aufruf nur ein Byte, meist in Form von ASCII-Code verwendet und für die Parameterübergabe gilt die folgende Festlegung:

Ausgabe: Register C; Eingabe: Register A.

Den 4 logischen Geräten Konsole CON:, Lesekanal RDR:, Schreibkanal PUN: und Drucker LST: können maximal je 4 physikalische zugeordnet werden, von denen nur je eines aktiv sein kann. Die Auswahl ist im IOBYTE auf Adresse 3 der Grundseite gespeichert. Die Belegung des IOBYTE zeigt Bild 6. 4 Gruppen zu 2 bit steuern die Zuordnung für die jeweiligen Kanäle. Sie wirken wie Schalter in der Elektrotechnik. Die Zuweisung der Ein-/Ausgabe-geräte arbeitet für jeden Kanal unabhängig (siehe Programmbeschreibungen BDOS und STAT).

Die Konsole besteht logisch aus 2 Teilen, einem Eingabe- und einem Ausgabekanal, meist Tastatur und Bildschirm. Die Möglichkeit der Auswahl unterschiedlicher Konsolen hat für einen CP/M-Computer geringe Bedeutung, weil im Gegensatz zu größeren Rechnern daran nicht mehrere Bediener arbeiten. Im Beispielprogramm CBIOS werden deshalb nur zwei Bildschirmansteuerungen, alphanumerisch und Grafik, unterschieden.

Bei der Initialisierung der Peripherieschaltkreise testet das Programm, ob der Grafik-Controller vorhanden ist. In diesem Falle wird das bit 0 des IOBYTE auf H-Pegel gesetzt. Bei jeder Zeichenausgabe auf den Bildschirm testet das Unterprogramm CONOUT, wie das bit 0 gesetzt ist und ruft das entsprechende Ausgabeprogramm auf. Hat man z. B. außer dem Grafikbildschirm auch noch den alphanumerischen in Betrieb, so kann man durch Löschen des bit 0 im IOBYTE die Ausgabe auf diesen per Software umschalten.

Die Zuweisung der logischen Geräte RDR:, PUN: und LST: arbeitet sinngemäß wie bei der Konsole beschrieben, so daß weitere Erläuterungen zur Auswertung des IOBYTE überflüssig sind. Im Beispiel-CBIOS sind die Möglichkeiten der Gerätezuweisung außer beim Bildschirm nicht genutzt, um Speicherplatz zu sparen. Da die Einsprungtabelle am BIOS-Anfang im RAM steht, kann man bei Bedarf jederzeit eigene Gerätetreiberprogramme eintragen und testen.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit Ø
Drucker - Ausgabe	Stanz - Ausgabe	Leser - Eingabe		Konsole - Ein-/Ausgabe			

Bild 6

Zuordnung der Geräte im IOBYTE

Nachfolgend sollen die Unterprogramme des BIOS kurz erläutert werden. Die Namen dafür sind in dieser Form üblich.

Unterprogramm Konsolenstatus feststellen: JP CONST

Dieses Unterprogramm meldet, ob an der Konsole ein Zeichen zur Eingabe in den Computer bereitsteht oder nicht. Es übergibt Register A = 0, wenn Kein Zeichen anliegt, sonst A = FFH.

Dabei ist die Art des Konsolgerätes unwichtig. Beispiel:

Eine Tastatur kann an einem 8-bit-Port den ASCII-Code liefern oder über eine V.24-Schnittstelle angeschlossen sein. Im erstgenannten Fall wird z.B. bit 7 high, wenn eine Taste gedrückt ist, beim zweiten liefert der Schaltkreis SIO ein Statussignal. In beiden Fällen muß das Unterprogramm CONST des BIOS das Statussignal des Geräts abfragen und im Register A die entsprechende Information zurückgeben. Diese Verfahrensweise gilt auch für die anderen Unterprogramme der zeichenorientierten Geräte, d.h. die Parameterübergabe ist von der verwendeten Hardware unabhängig und muß entsprechend programmiert werden.

Unterprogramm Konsoleneingabe JP CONIN (console input)

Im Gegensatz zu CONST wartet dieses Unterprogramm, bis an der Konsole ein Zeichen bereitsteht und liefert es dann in Register A. Die technische Lösung des Konsolgeräts ist dabei wie oben ohne Bedeutung.

Unterprogramm Konsolausgabe JP CONOUT (console output)

Das Unterprogramm erhält das Byte im Register C. Die technische Lösung (interner Bildschirm, Schnittstelle usw.) ist ohne Belang. Während bei der Konsoleneingabe fast nur ASCII-Code üblich ist, kommen bei der Ausgabe auch andere Zeichen in Frage. Früher, als vorwiegend Fernschreiber als Konsole eingesetzt wurden, reichten die Steuerzeichen "Wagenrücklauf" und "Zeilenvorschub" aus. Mit dem Einzug elektronischer Bildschirmterminals entstanden neue Möglichkeiten für die Programmgestaltung, die unter dem Begriff "Menütechnik" bekannt wurden. Bei Fernschreibern ist das einmal Ausgedruckte nicht leicht zu beseitigen, ein Bildschirm kann jedoch schnell teilweise gelöscht und neu beschrieben werden. Deshalb sind heute einige weitere

Steuerzeichen speziell für den Bildschirm üblich, die mit dem Betriebssystem CP/M nichts direkt zu tun haben, sondern rechnerabhängig sind. Aus diesem Grund gibt es hier keine Einheitlichkeit, vielmehr lassen sich alle größeren Programme für bestimmte Steuerzeichen "installieren". In der DDR sind durch das Betriebssystem SCP von *robotron* die nachfolgenden weit verbreitet:

Steuerzeichen für Bildschirm

dez	hex	Wirkung
1	1	Cursor an Anfang des Bildschirms
7	7	Piep
8	8	Cursor ein Zeichen nach links
10	0a	Cursor eine Zeile nach unten (Zeilenvorschub, linefeed)
12	0c	Bildschirm löschen und Cursor an den Anfang (formfeed)
13	0d	Cursor an Zeilenanfang (Wagenrücklauf, return, enter)
20	14	löschen ab Cursor bis Ende Bildschirm
21	15	Cursor ein Zeichen nach rechts
22	16	löschen ab Cursor bis Ende der Zeile
24	18	löschen gesamte Zeile und Cursor an Zeilenanfang setzen
26	1a	Cursor eine Zeile nach oben
27	1b	Einleitung Cursorpositionierung Die anschließenden Bytes geben an: Zeilennummer +80H Spaltennummer +80H
127	7f	Cursor ein Zeichen nach links und dort 00 ausgeben
130	82	Cursor einschalten
131	83	Cursor ausschalten
132	84	normale Darstellung
133	85	inverse Darstellung
134	86	intensive Darstellung
135	87	intensiv inverse Darstellung

Im Beispiel-CBIOS sind fast alle dieser Steuerzeichen implementiert.

Die Unterprogramme für Drucker und Datenkanäle sind zwar vom Betriebssystem vorgesehen, werden jedoch nur von anderen Programmen benutzt, nicht von CCP und BDOS. Zur Vereinfachung oder um Speicherplatz zu sparen kann man sie auch weglassen und die Einsprungtabelle mit je zwei NOP- und einem RET-Befehl versehen. Auch die Datenkanäle RDR: und PUN: kann man als eine in Eingabe und Ausgabe geteilte logische Einheit ansehen, zumal bei neueren Geräten der Lesekanal meist physikalisch mit dem Schreibkanal vereinigt ist, z.B. in Form einer V.24-Schnittstelle mit einem SIO-Schaltkreis. Für beide Kanäle ist deshalb in neueren Programmen auch die gemeinsame Bezeichnung AUX: üblich. Vom Betriebssystem CP/M wird dies jedoch nicht unterstützt, es gibt kein Kommando, mit dem man beide Kanäle gleichzeitig ansprechen könnte. Die beiden Datenkanäle waren zur Zeit der Lochstreifentechnik vorwiegend für die Sicherung von Daten und Programmen gedacht, sie haben heute für die Kopplung von Computern untereinander und die Datenübertragung Bedeutung. In diesem Sinne sind sie auch für den Amateur von Interesse.

Unterprogramm Lesekanal RDR: JP READER

Das Unterprogramm liefert das Zeichen in Register A. Soll dieser Kanal in einem eigenen Computer nicht benutzt werden, so sollte der Programmaufruf das Dateiende-Kennzeichen 1AH zurückgeben, weil einige Programme bei Aufruf des Eingabekanals dies irgendwann erwarten. Die 2 Befehle "ld a,1ah" und "ret" können auch direkt in die Sprungtabelle eingetragen werden.

Unterprogramm Schreibkanal PUN: JP PUNCH

Das Unterprogramm erhält das auszugebende Zeichen in Register C.

Unterprogramm für Drucker LST: JP LIST

Die Druckerausgabe erhält das Zeichen ebenfalls in Register C.

Unterprogramm Druckerstatus JP LISTST

Dieses Unterprogramm wird von Anwenderprogrammen kaum genutzt.

Um unnötigen Aufwand bei der Programmierung des BIOS auf der einen Seite und Sicherheit gegen Programmabsturz auf der anderen zu erreichen, ist im Beispiel-CBIOS eine Routine eingefügt, die ungeprüft das Statussignal A = FFH (Drucker bereit) zurückliefert.

2.1.2. Diskettenlaufwerke und deren Anpassung

Im Gegensatz zu den Geräten des vorigen Abschnitts arbeiten Diskettenlaufwerke einschließlich entsprechender Simulationsprogramme mit Blöcken von Daten, nicht nur mit einem einzelnen Byte beim Aufruf. Die zugehörigen Programme sind daher etwas komplexer.

Man kann die Unterprogramme für den Betrieb der Diskettenlaufwerke in solche zur Steuerung des Laufwerks und solche für den Datenverkehr einteilen. Die Steuerung kann unterschiedliche technische Lösungen verwenden, z.B. verschiedene Floppy-Disk-Controller oder unterschiedliche Zeitverzögerungen nach dem Laden des Schreib-/Lesekopfes usw. Für den Datenverkehr des Betriebssystems ist das ohne Bedeutung. Der technische Aspekt ist zwar im Beispiel-BIOS vollständig dokumentiert, im folgenden soll jedoch vorwiegend auf solche Programmteile des BIOS näher eingegangen werden, die sich auf den Datenverkehr beziehen.

Ein Diskettenzugriff läuft in mehreren Schritten ab. Zunächst muß das Laufwerk angewählt, danach der Schreiblesekopf auf die richtige Spur bewegt werden. Dazu ist mindestens bei der ersten Benutzung des Laufwerkes nach dem Einschalten eine Einstellhilfe für Spur 0 nötig, die durch ein optoelektronisch erzeugtes Signal entsteht. Nach dem Einstellen der Spur kann mit Hilfe der Identifikationsfelder der richtige Sektor gelesen werden. Dabei wird ein Pufferbereich im Speicher benötigt, in den die Daten zunächst gelangen. Dieser Pufferspeicher des Betriebssystems CP/M ist auf 128 Byte festgelegt, seine Adresse einstellbar. Falls das Diskettenformat mit größeren Sektoren arbeitet, und das ist heute die Regel, muß im BIOS ein weiterer Datenpuffer für den Diskettenzugriff bereitstehen. Das BIOS muß

dann das physikalisch vorhandene Diskettenformat in das Standardformat 128 Byte je Sektor umwandeln. Weitere Elemente der Schnittstelle zum Betriebssystem sind Tabellen, in denen die wichtigsten technischen Daten der Laufwerke, wie maximale Kapazität, Sektoren je Spur usw. gespeichert sind.

Unterprogramm Kaltstart JP BOOT

Der erste Sprung der Tabelle am BIOS-Anfang wird nur nach dem Reset des Computers, z. B. nach dem Einschalten, benutzt. Der Ausdruck kommt aus dem Englischen und bedeutet soviel wie sich an den eigenen Stiefeln hochziehen. Gemeint ist, daß sich das Betriebssystem mit Hilfe eines Urladers, der sich im EPROM befindet und nur das Lesen des bzw. der ersten Sektoren auf der Diskette erlaubt, selbst lädt und startet. Dazu müssen die Peripherieschaltkreise und einige RAM-Bereiche, z. B. die Grundseite initialisiert werden. Im Beispiel-CBIOS werden die Systembestandteile BIOS, CCP und BDOS von den Systemspuren der Diskette geladen. Das geschieht auch bei den professionellen Computern (z. B. PC 1715). Weil auch das BIOS geladen wird, kann man die Systemdiskette nur für den entsprechenden Computertyp verwenden. Natürlich ist es mit modernen Schaltkreisen auch möglich, das komplette Betriebssystem inclusive BIOS vom EPROM zu laden. Das spart Zeit und Speicherplatz auf der Diskette (Die Systemspuren fallen weg!), jedoch sind Änderungen umständlicher. Der Kaltstart ist genaugenommen kein Unterprogramm, sondern geht in das Unterproram "Warmstart" über.

Unterprogramm Warmstart JP WBOOT

Der Warmstart erfüllt 2 wichtige Funktionen. Wenn das Betriebssystem anläuft, liest es beim ersten Zugriff die Inhaltsverzeichnisse der angesprochenen Diskettenlaufwerke und speichert dazu eine Prüfsumme. Vor jedem Schreibvorgang wird diese auf Veränderung überprüft. Hat man die Diskette inzwischen gewechselt, gilt sie als schreibgeschützt, und es erscheint eine Fehlermeldung. Deshalb muß nach jedem Diskettenwechsel ein Warmstart ausgeführt werden, wenn man nicht nur lesen will. Dazu wird entweder ein entsprechendes Programm aufgerufen oder auf der Ebene des Betriebssystems die Taste ^C gedrückt.

Die 2. Funktion ist das Aktualisieren des CCP im Speicher. Vor einigen Jahren arbeiteten viele Computer noch mit weniger als 64 KByte Speicher, und um für größere Programme Platz zu gewinnen, durften diese den CCP überschreiben. Den Abschluß dieser Programme bildet dann ein Warmstart, bei dem vor der Rückkehr auf die Ebene des Betriebssystems der CCP von der Diskette nachgeladen wurde. Heute ist der verfügbare Speicher größer, so daß ein anderer Weg möglich ist: Beim Kaltstart wird der CCP in einen Speicherbereich im BIOS kopiert und beim Warmstart nur von dort zurück übertragen. Das geht wesentlich schneller als das Lesen von Diskette. Natürlich ist das Einlesen des Inhaltsverzeichnisses davon nicht betroffen, so daß ein Diskettenzugriff beim Warmstart trotzdem nötig ist. Den Abschluß des Warmstarts bildet der Sprung in den CCP.

Unterprogramm Spur 0 einstellen JP HOME

Das Unterprogramm soll den Schreiblesekopf des aktuellen Laufwerks auf Spur 0 einstellen. Ein Zugriff auf das Laufwerk ist nicht unbedingt nötig. Es genügt, die Adresse der Spur für einen späteren Zugriff abzuspeichern.

Unterprogramm Laufwerk auswählen JP SELDSK

Auch dieses Unterprogramm dient der Vorbereitung eines nachfolgenden Diskettenzugriffs und besteht nur in der Abspeicherung des ausgewählten Laufwerks. Der Eingabeparameter "Laufwerksnummer" ist beim Aufruf in Register C enthalten (Laufwerk A: als 0, B: als 1 usw.). Das Unterprogramm liefert im Registerpaar HL die Adresse einer dem angewählten Laufwerk zugeordneten Tabelle im Speicher. Sie heißt Disk-Parameter-Header DPH (head = engl. Kopf) und enthält einen Zwischenspeicher für das BDOS und die Adressen weiterer Speicherplätze, auf denen Kennwerte des Laufwerks stehen. Das Unterprogramm enthält auch eine einfache Kontrolle auf Eingabefehler des Bedieners. Die Laufwerke werden beginnend mit A: bis maximal P: bezeichnet. Die tatsächlich vorhandene Zahl kann an dieser Stelle im BIOS mit der vom BDOS angeforderten verglichen und ein Fehler mit HL = 0 dem BDOS gemeldet werden. Dieses löst eine Fehlermeldung an der Konsole aus. Die DPH der einzelnen Laufwerke sind

hintereinander angeordnet, dadurch kann deren Adresse auf einfache Weise direkt aus der Laufwerksnummer, ausgehend von einer Anfangsadresse als Basis, berechnet werden.

Ein DPH hat folgenden Aufbau:

DPBAS	EQU	\$; Bezugsbasis für erstes Laufwerk
DPHA:	DW	0	; Adresse Konvertierungstabelle
	DW	0	; Notizspeicher für BDOS
	DW	0	
	DW	0	
	DW	DIRBUF	; Adresse directory-buffer
	DW	DPB	; Adresse Laufwerksbeschreibung
	DW	CHKO	; Adresse Prüffeld Laufwerk A:
	DW	ALLO	; Adresse Belegungsfeld Laufw. A:

Die Marke DPHA: dient nur der Kennzeichnung für den Programmierer. Eine Konvertierungstabelle wird heute bei 5,25"-Laufwerken nicht mehr verwendet, deshalb steht auf dieser Adresse eine 0; sie war jedoch bei 8" üblich, um den Diskettenzugriff zu beschleunigen.

Wenn ein Programm auf eine Datei zugreift, wird normalerweise mehr als ein Sektor gelesen. Ordnet man die Sektoren nacheinander auf der Spur an, hat sich die Diskette schon weitergedreht, ehe der nächste erreicht ist, und es vergeht die Zeit für eine weitere Umdrehung bis zum nächstmöglichen Zugriff. Bei 100 Sektoren entsteht dadurch schon merklicher Zeitaufwand. Deshalb wurden früher die Sektoren nicht in ihrer natürlichen Reihenfolge, sondern im Raster 1, 7, usw. genutzt. Der Abstand zwischen den zusammengehörigen Sektoren heißt auch Sektorversatz. Näheres dazu folgt beim Unterprogramm SECTRAN, das diese Transformation ausführt. Wenn nicht durch den Wert 0 auf diesem Speicherplatz auf die Sektortransformation verzichtet wird, können dort für jedes Laufwerk unterschiedliche Tabellenadresse stehen. Bedeutung hat dies auch noch heute, wenn in einem Gerät 8"- und 5,25"-Laufwerke vorhanden sind. Die Adresse DIRBUF zeigt auf einen Speicherbereich der für alle Laufwerke einen gemeinsamen Puffer für das Inhaltsverzeichnis enthält. Die Adresse DPBA enthält den Beginn einer Tabelle mit technischen

Daten des zugehörigen Laufwerks, die Disk-Parameter-Block heißt und für jeden Laufwerktyp mit unterschiedlichem Format getrennt vorhanden sein muß. Der Disk-Parameter-Block bewirkt die Anpassung der Kennwerte eines Laufwerks an das Betriebssystem CP/M und wird deshalb später noch ausführlicher beschrieben. Das Prüffeld ist ebenfalls für jedes Laufwerk getrennt vorhanden. Es enthält die Kontrollsumme des Directory und erlaubt so dem BDOS die Prüfung auf Diskettenwechsel. Die letzte Adresse des DPH zeigt auf einen Speicherbereich, in dem die Belegung (engl. allocation) der Diskette gespeichert ist. Im Beispiel-BIOS enthält das Unterprogramm noch einen Test auf erstmalige Benutzung, der durch die verwendete Hardware bedingt ist und nicht durch das Betriebssystem.

Unterprogramm Spur einstellen JP SETTRK

Die Nummer der vom Betriebssystem geforderten Spur steht beim Aufruf im Registerpaar BC. Ein Doppelregister erlaubt auch die Ansteuerung von Festplatten, bei denen auch mehr als 256 Spuren möglich sind. Bei Diskettenlaufwerken genügt die Auswertung des Registers C. Wie schon beim Unterprogramm HOME gelangt der Wert für die Spur nur in den Speicher, ein Diskettenzugriff ist mit dem Aufruf des Unterprogramms nicht verbunden.

Unterprogramm Sektor auswählen JP SETSEC

Die Auswahl eines Sektors geschieht in völlig gleicher Weise wie die einer Spur, der Inhalt des Registers C enthält die Sektornummer und wird abgespeichert. Zu beachten ist, daß das BDOS nur mit 128-Byte-Sektoren arbeitet, d.h., die vom Betriebssystem in diesem Unterprogramm angeforderte Sektornummer stimmt bei größeren Sektoren auf der Diskette nicht mit der im Identifikationsfeld eingetragenen überein.

Unterprogramm Pufferspeicher festlegen JP SETDMA

Die Abkürzung DMA steht für "direct memory access" (direkter Speicherzugriff) und bezeichnet einen Speicherbereich, der beim nächsten Diskettenzugriff 128 Byte aufnehmen oder liefern soll. Eingangsparameter ist das Registerpaar BC, die Routine speichert den Wert nur ab.

Unterprogramm Sektorlesen JP READ

Nach dem Aufruf dieses Unterprogramms sollen die Daten im Puffer stehen. Wenn beim Diskettenzugriff keine Fehler auftraten, steht im Register A = 0, sonst A = 01.

Unterprogramm Sektorschreiben JP WRITE

Das Unterprogramm arbeitet äquivalent dem Lesen. Es benutzt die im Datenpuffer enthaltenen 128 Byte als Eingabe und liefert in Register A den Wert 0, wenn kein Fehler auftrat, sonst A = 01. Wie schon mehrfach betont, arbeiten heute fast alle Diskettenformate mit größeren Sektoren als 128 Byte. Nicht immer ist deshalb beim Lesen oder Schreiben wirklich ein Diskettenzugriff nötig. Bei Sektoren auf der Diskette von 1024 Byte muß das BIOS die Umwandlung der vom BDOS berechneten Sektornummer in die auf der Diskette vorhandene besorgen. Außerdem kann auf den Zugriff verzichtet werden, wenn sich der geforderte Sektor bereits im Puffer des Diskettenleseprogramms befindet. Er muß dann nur von dort in den Puffer DMA des Betriebssystems übertragen werden.

Unterprogramm Sektortransformation JP SECTRAN

Dieses Unterprogramm ist im Beispiel-BIOS vereinfacht worden, weil keine 8"-Diskettenlaufwerke vorgesehen waren. Bei Bedarf findet man den Algorithmus für die Umrechnung der Sektornummer in [2]. Natürlich kann auch bei 5,25"-Disketten mit Sektorgrößen von 1024 Byte die Geschwindigkeit des Zugriffs durch einen Sektorversatz erhöht werden, jedoch läßt sich das vorteilhafter beim Formatieren der Diskette erledigen. Die Zeit für die Sektortransformation tritt dann später nicht bei jedem Zugriff auf einen Sektor auf.

Weitere Unterprogramme des Beispiel-BIOS

Viel Platz nimmt im BIOS der sogenannte physische Treiber für die Diskettenlaufwerke ein. Er ist auf die konkrete Schaltung der Anschlußsteuerung bezogen und muß bei einer anderen, z. B. mit dem bekannten Schaltkreis 1793, ersetzt werden. Der Algorithmus für die Umwandlung der großen in die 128-Byte-Sektoren entstand in Anlehnung an [2]; bei Schaltungsänderungen oder

anderen Laufwerken (z. B. SS DD 48 TPI) muß man ihn ebenfalls geringfügig ändern. Unabhängig von diesen technischen Fragen ist die Anpassung unterschiedlicher Diskettenlaufwerke an das Betriebssystem. Die Schnittstelle dazu ist der Disk-Parameter-Block DPB. In dieser Tabelle sind die technischen Daten des Diskettenlaufwerks für das BDOS in kodierter Form gespeichert.

Ein DPB hat folgenden Aufbau:

DPBA:	DW	SPT ; SECTORS PER TRACK
	DB	BSH ; BLOCK SHIFT
	DB	BLM ; BLOCK MASK
	DB	EXM ; EXTEND MASK
	DW	DSM
	DW	DRM ; DIRECTORY MARK
	DW	ALL ; ALLOCATION
	DW	CKS ; CHECK SUM
	DW	OFF ; OFFSET

Die Anfangsadresse der Marke DPBA (A steht für Laufwerk A;) findet das BDOS im DPH. Der erste Tabellenplatz gibt die Anzahl der 128-Bytesektoren auf einer Spur an. Sind sie physisch größer, muß man die umgerechnete Zahl eintragen. Außer mit Sektoren rechnet das BDOS auch mit Blöcken aus mehreren Sektoren, deren Anzahl in bestimmten Grenzen wählbar ist. Im Blockverschiebungsfaktor BSH und in der Blockmaske BLM ist die Blockgröße BLS wie folgt enthalten:

BLS (Bytes)	BSH	BLM
1024	3	7
2048	4	15
4096	5	31
8192	6	64
16384	7	127

Diese Blöcke bestimmen die Eintragung im Belegungsfeld

Die Extendmaske EXM gibt an, wieviele 16-kByte-Eintragungen von einem Eintrag im Inhaltsverzeichnis erfaßt sind. Sie hängt von der Blockgröße und der Gesamtkapazität der Diskette DSM ab:

BLS	EXM (DSM<255)	EXM(DSM>255)
1024	0	-
2048	1	0
4096	3	1
8192	7	3
16384	15	7

Die Kapazität der Diskette einschließlich möglicher Systemspuren steht im Wert DSM in der Form "Anzahl der Blöcke - 1". Der Platz für das Inhaltsverzeichnis ist in der Form der "Anzahl der Eintragungen - 1" in den DRM-Bytes enthalten. Das Inhaltsverzeichnis beginnt unmittelbar hinter den Systemspuren. Jede Eintragung umfaßt 32 Byte, so daß in Abhängigkeit von der Zahl der Eintragungen DRM und der Blockgröße BLS einige Datenblöcke schon von Anfang an durch das Inhaltsverzeichnis belegt sind. Das Belegungsfeld ALL enthält die vom Inhaltsverzeichnis belegten Datenblöcke auf H-Pegel gesetzt.

Der Wert im Datenwort CKS bestimmt die Zahl der bei Diskettenwechsel zu überprüfenden Bytes der Eintragungen im Verzeichnis. Bei nicht austauschbaren Datenträgern, z.B. Festplatten oder RAM-Disk steht hier der Wert 0, sonst $(DRM + 1)/4$.

Unter dem Offset (engl. = Versatz) ist die Anzahl der Systemspuren zu verstehen.

Die im DPB festgelegten Werte bestimmen auch die Größe der Felder für Belegung und Checksumme im DPH. Das Belegungsfeld ALV berechnet sich zu $(DSM/8) + 1$, CKS benötigt $(DRM + 1)/4$ Bytes. Sind 2 DPB identisch (gleiche Laufwerke), so können sie auch von verschiedenen DPH adressiert werden und brauchen damit nur einmal im BIOS zu stehen. Im Beispiel-BIOS sind 2 unterschiedliche DPB enthalten, um Disketten mit und ohne Systemspuren (780- und 800-K-Format) gleichzeitig lesen zu

Können. Die Adresse des 2. DPB wird dann mittels Programm bei einem Laufwerk in den DPH eingetragen.

Die nachstehende Tabelle enthält als zusätzliches Beispiel übliche DPB für einseitige Laufwerke mit 40 Spuren.

			SCP	CPA
DPB40:	DW	SPT	20H	28H
	DB	BSH	04	03
	DB	BLM	0FH	07
	DB	EXM	01	00
	DW	DSM	49H	0C7H
	DW	DRM	3FH	3FH
	DW	ALL	80H	OC0H
	DW	CKS	10H	10H
	DW	OFF	03	00

2.1.3. Virtuelle Diskettenlaufwerke im Speicher

Die Arbeitsweise eines Diskettenlaufwerks lässt sich auch durch ein Programm im Speicher simulieren. Für den Anwender entsteht dadurch kein Unterschied bei der Bedienung. Das hat den Vorteil, daß alle Zugriffe mit vergleichsweise sehr hoher Geschwindigkeit ablaufen. Es sind Schaltungen mit ROM- und RAM-Schaltkreisen möglich, wobei natürlich ROM-Schaltungen nur gelesen werden können und RAM-Schaltungen den Dateninhalt bei Abschaltung der Versorgungsspannung verlieren.

In [7] ist eine Lösung angegeben, die einen Teil des TPA dafür verwendet. Diese Lösung ist zwar besonders einfach, hat aber den Nachteil, daß dann der für Programme verfügbare Speicher nur noch recht klein ist. Mit den heute zu Verfügung stehenden hochintegrierten Speicherschaltkreisen lassen sich jedoch mit relativ geringem Aufwand Schaltungen aufbauen, die außerhalb des TPA liegen und somit die Möglichkeiten eines konventionellen CP/M-Computers wesentlich erweitern. Im Beispiel-BIOS ist

eine RAM-Floppy-Karte nach [11] implementiert. Der Speicher dieser Karte liegt nicht im Adressbereich des Prozessors, sondern wird über Ein-/Ausgabeadressen angesteuert. Das Betriebssprogramm dafür unterscheidet sich von dem für die Diskettenlaufwerke, so daß im BIOS in Abhängigkeit vom aufgerufenen "Laufwerk" in den Diskettentreiber oder in das RAM-Floppy-Programm verzweigt werden muß.

Bei einem Computer nach [7] wurde noch eine RAM-Floppy mit einer normalen 64-K-Speicherplatine aufgebaut, die sich in Blöcken zu 32 KByte aus- und einschalten läßt. Zum Betrieb dieser Karte läßt sich auch der TPA bis zur Adresse 8000 abschalten. Bei dieser Technik spricht man von Bank-Betrieb der Speicher. Im Betriebssprogramm dazu darf man beim Aufruf nicht vergessen, den Stack des aufrufenden Programms zu retten.

2.2. CCP - die Benutzeroberfläche

Der Modul CCPwickelt die Kommunikation zwischen Bediener und Betriebssystem ab. Bei Betriebsbereitschaft des Systems zeigt er die Eingabeaufforderung, den sogenannten Prompt, an. Er besteht aus dem Buchstaben des aktuellen Laufwerkes und dem Zeichen >. Wenn dieses Bild erscheint, können Kommandos und Parameter eingegeben werden. Der CCP hat dazu unmittelbar am Programmanfang einen internen Pufferspeicher von 127 Byte Größe. Bei der Eingabe gelangen die Daten von der Konsole zunächst in diesen Speicher, wo sie bei Eingabefehlern noch in bescheidenem Umfang editiert werden können. Die Tabelle auf Seite 36 gibt eine Übersicht der dazu nötigen Steuerzeichen. Den Anfang des CCP bilden 2 Einsprünge, danach folgen 2 Byte. Das 1. gibt die Puffergröße, das 2. die Belegung des Pufferspeichers an und wird bei Eingabe heraufgezählt. Anschließend folgt der Puffer selbst. Der 1. Einsprung unterscheidet sich in der Wirkung vom 2. dadurch, daß er den Eingabepuffer nicht löscht. Er kann benutzt werden, um schon beim Anlauf des Betriebssystems ein Programm zu laden und zu starten. Der Programmname muß dazu (ohne .COM) im Eingabepuffer des CCP einge tragen sein, wenn dieser von Diskette geladen wird.

Steuerzeichen	Bezeichnung	Wirkung
DEL	Löschen	gelösches Zeichen wird ausgegeben, Cursor eine Position nach rechts
^H	Rückschritt	Cursor geht ein Zeichen nach links und schreibt dorthin ein Leerzeichen
^U		Löschen der ganzen Zeile im Puffer
^X		Löschen der ganzen Zeile im Puffer und auf dem Bildschirm
^R		Neuanzeige des Pufferinhalts nach Korrekturen
^E		Fortsetzung der Eingabe in der nächsten Zeile
^I	Tabulator	Cursor geht an Anfang der nächsten 8-Spalten-Position
^P		Druckerausgabe gleichzeitig zur Konsole ein/aus
^Z	EOF	Dateiendeckenzeichen für ASCII
^S		Konsolausgabe ein/aus
^M		Eingabezeile ausführen (Wagenrücklauf, Enter, Return)

Beispiel:

Anfang des CCP ohne Programmstart:

C800 C3 5C CB C3 58 CB 7F 00 20 20 20...

Anfang des CCP mit Programmstart:

C800 C3 5C CB C3 58 CB 7F 03 50 57 52 00 20...

Ist im Zähler des Puffers eine 0 eingetragen, so gibt der CCP den Prompt aus und wartet auf eine Eingabe. Der CCP enthält einige Funktionen, die nach dem Aufruf sofort ausgeführt werden. Das sind die sogenannten residenten Kommandos des Betriebssystems:

DIR

Dieses Kommando listet das Inhaltsverzeichnis einer Diskette auf der Konsole. Ist kein Laufwerk angegeben, wird das aktuelle verwendet. Als Parameter sind außer den Laufwerksbezeichnern auch Dateibezeichnungen einschließlich den Gruppenbezeichnern möglich. Beispiel: Aktuelles Laufwerk sei A:, die Kommandoeingabe:

```
dir b:*.?as (cr)
```

liefert eine Anzeige von BASIC- und PASCAL-Programmen auf Laufwerk B:.

TYPE

Das Kommando listet eine Datei auf der Konsole aus. Es ist nur für ASCII-Dateien sinnvoll und erfordert die Namenseingabe vollständig ohne die Gruppenbezeichner * und ?.

Bei diesem Befehl kann die Ausgabe vorteilhaft mit der Tastenkombination ^S angehalten und fortgesetzt werden.

REN

Mit diesem Kommando kann man einer Datei einen neuen Namen geben, Beispiel:

```
ren name.neu=eman.alt
```

ERA

Dieses Kommando löscht die vollständig bezeichnete Datei. Dabei erhält nur das Inhaltsverzeichnis ein Kennzeichen, die Datei selbst ist noch vorhanden und kann mit Hilfsprogrammen wieder "hervorgeholt" werden, wenn nicht durch nachfolgendes Schreiben der nun als frei gemeldete Speicherplatz auf der Diskette schon erneut belegt wurde.

SAVE

Dieses Kommando erlaubt es, den Inhalt des TPA (ab der Adresse 100H!) als Datei auf Diskette zu bringen. Als Parameter müssen die Dateilänge und der vollständige Name angegeben werden. Die Längenangabe erfolgt dabei in "Speicherseiten", das sind Blöcke von 256 Bytes. Wesentlich leistungsfähiger sind die Programme ZSID und POWER (siehe Abschnitt 3), die wählbare Adressbereiche abspeichern können.

USER

Das Betriebssystem erlaubt es, 15 verschiedene Benutzerbereiche auf einer Diskette voneinander abzugrenzen. Als Parameter für

die Kommandoeingabe dient die Benutzernummer, z. B. :

user 2 (cr)

Für den Amateur hat das wenig Bedeutung. Beim Warmstart wird immer Bereich 0 eingestellt.

Wird ein anderes als die oben genannten Kommandos über die Konsole eingegeben, so veranlaßt der CCP die Suche nach einer Datei mit entsprechendem Namen und dem Extend .COM, das Laden und den Start wie bereits beschrieben.

2. 3. BDOS - Kern des Betriebssystems CP/M

Die wichtigste Aufgabe des BDOS ist es, dem Anwender die Verwaltung der Dateien auf den Disketten zu vereinfachen. Zwar enthält das BIOS Routinen zum Lesen beliebiger Sektoren, jedoch wäre eine manuelle Steuerung des Zugriffs umständlich und fehlerträchtig. Auch sind verschiedene Lösungen möglich, so daß der Programmaustausch schwieriger wäre, wenn jeder Anwender eine eigene nehmen könnte,

Das BDOS erlaubt aber auch den Zugriff auf andere Peripheriegeräte. Der Einsprung in das BDOS erfolgt über die Adresse 5, die Standardschnittstelle des Betriebssystems CP/M. Dadurch ist es möglich, Programme rechnerunabhängig zu gestalten. Unterschiede in der Größe des TPA wirken sich nicht aus, wenn man anstelle der Unterprogramme des BIOS die BDOS-Aufrufe mit Adresse 5 benutzt. Es gibt 40 Funktionsaufrufe des BDOS. Jeder Aufruf erfolgt nach folgender Systematik:

- Laden der Funktionsnummer in Register C;
- Laden weiterer möglicher Parameter im Registerpaar DE;
- CALL 5.

Bei der Rückkehr von diesem Aufruf übergibt das BDOS 8 bit im Register A, 16 bit im Registerpaar HL. Liegt die Funktionsnummer außerhalb des zulässigen, so wird eine 0 übergeben.

Ahnlich wie beim BIOS sollen zunächst die zeichenorientierten Funktionen besprochen werden. Sie sind vorwiegend für Programme in Assembler nützlich, weil die höheren Sprachen meist vergleichbare oder komfortablere Befehle für die Ein- und Ausgabe enthalten (z. B. PRINT, WriteLn).

Funktion 0: System rücksetzen

Eingangsparameter:

Register C: 00H

Diese Funktion gibt die Steuerung an das CP/M auf dem CCP-Niveau zurück. Der CCP initialisiert das Diskettensystem und wählt Laufwerk A: an. Die Funktion hat den gleichen Effekt wie ein Sprung auf die Adresse WBOOT.

Funktion 1: Konsoleneingabe

Eingangsparameter:

Register C: 01H

Ausgangsparameter:

Register A: ASCII-Zeichen

Die Funktion "Konsoleneingabe" liest das nächste Zeichen von der Konsole in Register A ein. Alphanumerische Zeichen und die Steuerzeichen Wagenrücklauf, Zeilenvorschub und Rückwärts-schritt (ctrl-h) werden zur Konsole zurückgegeben. Tabulator-zeichen (ctrl-i) werden auf die nächste Tabulatorposition (alle acht Zeichen) mit Leerzeichen aufgefüllt. Es wird über-prüft, ob Start/Stop für Bildschirmrollen (ctrl-s) und Start/-Stop der Druckerausgabe (ctrl-p) auftritt. Das BDOS kehrt nicht eher zum aufrufenden Programm zurück, bevor nicht ein Zeichen auf der Konsole eingegeben wurde.

Funktion 2: Konsolenausgabe

Eingangsparameter:

Register C: 02H

Register E: ASCII-Zeichen

Das in Register E enthaltene Zeichen wird zur Konsole gesendet.
Test auf Start/Stop von Bildschirmrollen und Druckerausgabe.

Funktion 3: Lesereingabe

Eingangsparameter:

Register C: 03H

Ausgangsparameter:

Register A: ASCII-Zeichen

Die Lesereingabe liest ein Zeichen vom logischen Leser in das Register A (siehe I/O-Byte-Definition im Abschnitt 2.1.1). Die Steuerung kehrt nicht zurück, bevor das Zeichen gelesen ist.

Funktion 4: Stanzerausgabe

Eingangsparameter:

Register C: 04H

Register E: ASCII-Zeichen

Die Funktion "Stanzerausgabe" sendet das im Register E enthaltene Zeichen zum logischen Stanzer.

Funktion 5: Druckerausgabe

Eingangsparameter:

Register C: 05H

Register E: ASCII-Zeichen

Die Funktion "Druckerausgabe" sendet das im Register E enthaltene Zeichen zum logischen Listen-Ausgabegerät.

Funktion 6: direkte Konsolenein-/ausgabe

Eingangsparameter:

Register C: 06H

Register E: OFFH (Eingabe) oder
 ASCII-Zeichen (Ausgabe)

Ausgangsparameter:

Register A: Zeichen oder Status
 (kein Wert)

Die Benutzung dieser Funktion sollte vermieden werden, da die normalen Steuerzeichen des CP/M umgangen werden.

Bei Eintritt in die Funktion 6 enthält Register E entweder den hexadezimalen Wert FF, was eine Konsoleneingabe kennzeichnet, oder ein ASCII-Zeichen. Wenn der Eingangsparameter FFH ist, so kehrt die Funktion mit a = 0 zurück, falls kein Zeichen von der Konsole bereitsteht. Ansonsten enthält Register A das nächste Eingabezeichen von der Konsole.

Ist der Wert des Registers E nicht gleich FFH, behandelt ihn Funktion 6 als gültiges ASCII-Zeichen und sendet dieses zur Konsole.

Funktion 7: I/O-Byte holen

Eingangsparameter:

Register C: 07H

Ausgangsparameter:

Register A: I/O-Byte

Die Funktion "I/O-Byte holen" gibt den momentanen Wert des I/O-Bytes im Register A zurück. (siehe Abschnitt 2.1.1 für die Definition des I/O-Byte)

Funktion 8: I/O-Byte setzen

Eingangsparameter:

Register C: 08H

Register E: I/O-Byte

Funktion 9: Zeichenkette ausgeben

Eingangsparameter:

Register C: 09H

Registerpaar DE: Adresse der Zeichen

Die Funktion "I/O-Byte setzen" ändert den Wert des I/O-Bytes in den im Register E übergebenen Wert.

Die Funktion zur Ausgabe einer Zeichenkette sendet Zeichen aus dem Speicher beginnend ab der in DE übergebenen Adresse zur Konsole bis in der Zeichenkette ein "\$" erkannt wird. Tabulatoren werden expandiert wie in Funktion 2 und Start/Stop-Zeichen für Bildschirmrollen und Druckerausgabe ausgewertet.

Funktion 10: Eingabe in Konsolenpuffer

Eingangsparameter:

Register C: OAH

Registerpaar DE: Pufferadresse

Ausgangsparameter:

Zeichen von der Konsole im Puffer

Diese Funktion liest eine Zeile von eingegebenen Zeichen in einen Puffer, der durch DE adressiert ist. Die Eingabe wird abgeschlossen, wenn der Eingabepuffer überläuft oder wenn (CR) eingegeben wird. Der Puffer erhält folgende Form:

DE	+0	+1	+2	+3	+4	+5	+6	+7	+n
mx	nc	c1	c2	c3	c4	c5	c6	??	

Darin ist mx (1 bis 255) die maximale Anzahl von einzulesenden Zeichen und nc die tatsächlich eingegebene Zahl (vom BDOS bei Rückkehr gesetzt). Es folgen die eingelesenen Zeichen. Ist nc kleiner als mx, folgt dem letzten Zeichen unbestimmter Speicherinhalt, der im obigen Beispiel mit "???" gekennzeichnet ist. Die im Abschnitt 2.2 (CCP) beschriebenen Steuerzeichen werden während der Eingabe erkannt.

Die folgende Funktion überprüft, ob ein Zeichen auf der Konsole eingegeben wurde. Falls ein Zeichen eingegeben wurde, enthält Register A den Wert FFH, anderenfalls den Wert OOH.

Funktion 11: Konsolenstatus holen

Eingangsparameter:

Register C: 0BH

Ausgangsparameter:

Register A: Konsolenstatus

Funktion 12: Versionsnummer holen

Eingangsparameter:

Register C: 0CH

Ausgangsparameter:

Registerpaar HL: Versionsnummer

Funktion 12 erlaubt es, Programme zu erstellen, die von der Version unabhängig sind. Es wird ein 2-Byte-Wert übergeben, wobei H = 00H angeibt, daß es sich um CP/M handelt (H = 01H bei mp/m und L = 00 bei allen Versionen vor 2.0). Alle folgenden liefern Werte im Bereich von 21H bis 2FH. Für die Praxis des Amateurs hat diese Funktion keinerlei Bedeutung.

Das nachfolgende Programmbeispiel soll den Aufruf einiger BDOS-Funktionen demonstrieren. Es zeigt, wie man ähnliche Probleme in eigenen Assemblerprogrammen lösen kann. Einige allgemeine Gesichtspunkte der Programmierung, wie Absicherung gegen falsche Eingabe, Nutzung von Bildschirmsteuerzeichen usw., wurden nicht berücksichtigt, weil es hier nur auf die BDOS-Aufrufe ankam.

aseg

org 100h

	TITLE	'DIALOG'
bdos	equ	5 ;Standardeinsprung
start:	ld de, text1	;Adresse Text
	ld c, 9	;Ausgabe Zeichenkette
	call bdos	
	ld c, 1	;Konsoleingabe
	call bdos	
	bit 6, a	;Test auf kleine Buchstaben
	jr z, dia1	
	res 5, a	;Umwandlung in grosse
dia1:	cp 'J'	;Ja
	jr nz, exit	;zum Programmende
	ld de, text2	;Adresse Text fuer Drucker
	ld c, 9	;Textausgabe
	call bdos	
	ld de, puffer	;Zwischenspeicher Text ;(Druckpuffer)
	ld c, 10	;Texteingabe in Puffer
	call bdos	
	ld h1, puffer+1	;Druckerausgabe vorbereiten
	ld b, (h1)	;Zeichenanzahl
dia2:	inc h1	;Textanfang
	ld a, (h1)	
	push bc	
	push h1	
	ld c, 5	;Druckerausgabe
	ld e, a	;Zeichen umladen
	call bdos	
	pop h1	
	pop bc	
	djnz dia2	
exit:	ld c, 0	
	call bdos	
text1:	defm	'Wollen Sie den Drucker testen? (j/n)\$'
text2:	defm	0dh, 0ah, 'Geben Sie einen Text ein!', 0dh, 0ah, '\$'
puffer:	db	0ffh, 0
end		

Die folgenden BDOS-Funktionen befassen sich mit der Arbeit des Diskettenteils des Computers. Sie werden nach der Behandlung der einzelnen Aufrufe ebenfalls mit einem kleinen Beispiel näher erläutert. Vor der Besprechung soll noch auf eine Voraussetzung für den Diskettenzugriff eingegangen werden, die das Betriebssystem erfordert. Bei den meisten Diskettenzugriffen werden Laufwerksbezeichner, Dateiname und u. U. weitere Daten benötigt, die in normierter Form im Speicher stehen müssen und Dateisteuerblock oder FCB (von engl. file control block) heißen. Man unterscheidet 2 Arten des Dateizugriffs. Beim sequentiellen werden die Blöcke einer Datei nacheinander bearbeitet, z. B. indem eine ganze Datei in den Speicher geladen wird. Dazu ist ein FCB mit 33 Byte erforderlich. Beim direkten Zugriff muß der FCB 36 Byte lang sein. Ein FCB hat folgenden Aufbau:

Byte	Symbol	Bedeutung
0	dr	Laufwerkskode: 0 aktuelles LW 1 LW A: 2 LW B: usw.
		Benutzernummer im Inhaltsverzeichnis
1...8	f1...f8	Dateiname in ASCII-Zeichen (bit 7 = 0)
9...11	t1...t3	Dateityp bit 7 = 1 von t1 bedeutet Datei schreibgeschützt, bit 7 = 1 von t2 bedeutet Systemdatei
12	ex	aktuelle Extendnummer (0...31)
13, 14	s1, s2	reserviert
15	rc	Anzahl 128-Byte-Sektoren (0...FF)
16...31	dmi...16	Blocknummern im Directory-Eintrag, sonst 0
32	cr	aktuelle Blocknummer für nächsten sequentiellen Zugriff.
Für direkten Zugriff sind zusätzlich notwendig:		
33	r0	Blocknummer L-Teil
34	r1	Blocknummer H-Teil
35	r2	für Überlauf

Der FCB ist vor jeder Dateioperation vom Anwenderprogramm bereitzustellen, voreingestellt ist in der Grundseite die Adresse 5CH. Soll eine Datei sequentiell bearbeitet werden, müssen die Positionen 12 (ex) bis 32 (cr) zu Anfang auf 0 gestellt werden. In Übereinstimmung mit dem FCB ist der Eintrag der Datei im Inhaltsverzeichnis der Diskette aufgebaut. Wie schon im Abschnitt 2.1.2. bei der Beschreibung des Disk-Parameter-Blocks erläutert, vergibt das Betriebssystem CP/M den Speicherplatz auf der Diskette in Form von Blöcken, die in bestimmten Stufen als Vielfache von 128 Byte organisiert sind und durchnummieriert werden. Das Feld dm enthält die zur Datei gehörigen Blocknummern. Ist die Datei größer als 16 Blöcke, reicht der Platz nicht aus und es wird ein weiterer Eintrag mit dem gleichen Namen und den weiteren Blocknummern vorgenommen. Im Feld ex erscheint dann die nächste Extendnummer. Mit dem Begriff "Sektor" sind unabhängig vom Diskettenformat im nachfolgenden Text 128-Byte-Sektoren gemeint. Die Benutzernummer ist im Inhaltsverzeichnis im Byte 0 gespeichert, wo beim FCB der Laufwerkskode steht.

Funktion 13: Diskettensystem rücksetzen

Eingangsparameter:

Register C: ODH

Diese Funktion wird benutzt, um vom Programm aus das Dateisystem in den Anfangszustand zu versetzen. Dabei sind alle Laufwerke für Schreiben und Lesen zugelassen, Laufwerk A: ist angewählt und die DMA-Adresse ist auf den Standardwert 0080H festgelegt. Diese Funktion ermöglicht z.B. in einem Anwenderprogramm Diskettenwechsel ohne Neuladen des Betriebssystems.

Funktion 14 kennzeichnet das in E bezeichnete Laufwerk als das gültige Laufwerk für die folgenden Dateioperationen. Dabei gilt

Funktion 14: Laufwerk anwählen

Eingangsparameter:

Register C: OEH

Register E: Laufwerkskode

E = 0 für Laufwerk A:, 1 für Laufwerk B: usw. Das Laufwerk ist dadurch aktiviert und verbleibt bis zum nächsten Warmstart oder Rücksetzen des Diskettensystems im on-line-Status. Wenn im on-line-Zustand die Diskette gewechselt wird, geht das Laufwerk automatisch in den Schreibschutz über (siehe Funktion 28). Dateisteuerblöcke, die den Laufwerkskode 0 aufweisen, beziehen sich immer auf das momentan angewählte Laufwerk. Laufwerkskodes von 1 bis 16 ignorieren die Standardanwahl und beziehen sich direkt auf ein Laufwerk von A: bis P:.

Funktion 15: Datei eröffnen

Eingangsparameter:

Register C: OFH

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Directory-Kode

Die Dateieröffnung wird benutzt, um eine Datei zu aktivieren, die bereits auf der angewählten Diskette existieren muß. Das EDOS sucht im Inhaltsverzeichnis nach Übereinstimmung mit dem Dateinamen (Positionen 1 bis 12 des durch DE adressierten Dateisteuerblocks (Bytes ex und s1 auf Null gesetzt), wobei auch

ein Fragezeichen in jeder Positionen als Übereinstimmung gilt. Wenn ein Dateiname im Inhaltsverzeichnis mit dem des FCB übereinstimmt, wird die zugehörige Information in die Bytes dmi bis dm16 des Dateisteuerblocks kopiert, wodurch der Zugriff auf diese Datei bei nachfolgenden Lese- und Schreiboperationen möglich ist. Auf eine Datei kann nicht zugegriffen werden, bevor nicht die Eröffnung erfolgreich durchgeführt wurde. Die Funktion 15 übergibt in Register A einen "Directory-Kode" mit dem Wert von 0 bis 3, falls die Eröffnung erfolgreich war oder OFFH falls die Datei nicht gefunden wurde. Wenn im FCB Fragezeichen auftreten, wird die erste übereinstimmende Datei ausgewählt. Der Datenzähler cr muß vom Programm auf Null gesetzt werden, wenn die Datei sequentiell vom ersten Sektor angelesen werden soll.

Funktion 16: Datei schließen

Eingangsparameter:

Register C: 10H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Directory-Kode

Die Funktion führt die Umkehrung der Dateieröffnung durch. Wurde der durch DE adressierte Dateisteuerblock durch eine vorhergehende Eröffnung oder Kreation (Anlegen einer Datei, siehe Funktionen 15 und 22) aktiviert, so speichert die Dateischließung den FCB dauerhaft im Inhaltsverzeichnis. Der FCB-Suchvorgang ist derselbe wie bei der Eröffnung. Der Directory-Kode, der bei einem erfolgreichen Abschluß übergeben wird, ist 0 bis 3. FFH kommt zurück, falls der Dateiname im Inhaltsverzeichnis nicht zu finden war.

Nach Leseoperationen braucht eine Datei nicht geschlossen zu werden. In jedem Fall ist die Datei nach Schreiboperationen zu

schließen, um den neuen Eintrag im Inhaltsverzeichnis abzuspeichern.

Funktion 17: erste Eintragung suchen

Eingangsparameter:

Register C: 11H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Directory-Kode

Die Funktion "erste Eintragung suchen" gewährt dem Programmierer Zugriff auf die Daten des Inhaltsverzeichnisses der Diskette. Sie durchsucht dieses auf Übereinstimmung mit dem durch DE adressierten FCB. Der Wert FFH kommt zurück, wenn die Datei nicht gefunden wurde, anderenfalls die Werte 0, 1, 2, oder 3. Diese zeigen an, daß die Datei vorhanden ist. In dem Falle wird der gültige DMA-Bereich mit der Verzeichniseintragung gefüllt.

Je nach Parameter in Register A bei der Rückkehr stehen die 32 Byte des Verzeichniseintrags im DMA-Puffer ab dem 1. Byte (wenn A = 0 war) oder um je 32 Byte verschoben, d.h. A = 1 ergibt die Werte ab 33. Byte usw. Ein Fragezeichen in irgendeiner Position von f1 bis ex gilt als Übereinstimmung mit dem entsprechenden Feld im Inhaltsverzeichnis des angewählten Laufwerks. Wenn das dr-Feld ein Fragezeichen enthält, wird auf allen Benutzerbereichen geprüft. Wenn das dr-Feld kein Fragezeichen enthält, wird das s2-Byte automatisch auf Null gesetzt.

Die folgende Funktion entspricht der Funktion 17 nur mit dem Unterschied, daß die Suche im Inhaltsverzeichnis ausgehend vom letzten übereinstimmenden Eintrag fortgesetzt wird. Funktion 18 über gibt ebenfalls den Wert FFH, wenn kein Eintrag mehr gefunden wurde.

Funktion 18: nächste Eintragung suchen

Eingangsparameter:

Register C: 12H

Ausgangsparameter:

Register A: Directory-Kode

Funktion 19: Datei löschen

Eingangsparameter:

Register C: 13H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Directory-Kode

Diese Funktion entfernt Dateien, die mit dem durch DE adressierten FCB übereinstimmen. Dateiname und Dateityp können mehrdeutig sein (d.h. Fragezeichen in verschiedenen Positionen), aber der Laufwerkskode darf nicht mehrdeutig sein wie in den Funktionen 17 und 18.

Die Funktion 19 übergibt den Wert FFH, wenn die Datei nicht gefunden wurde, ansonsten Werte von 0 bis 3.

Wurde der durch DE adressierte Dateisteuerblock mit einer der Funktionen 15 oder 22 aktiviert, liest die Funktion 20 den nächsten Sektor in den Puffer mit der aktuellen DMA-Adresse. Der Sektor wird von der Position cr gelesen und das cr-Feld automatisch auf die nächste Position erhöht. Wenn das cr-Feld

überläuft, wird automatisch auf den folgenden Extend umgeschaltet und das cr-Feld für die nächste Leseoperation auf Null gesetzt. Wenn die Operation erfolgreich war, wird der Wert OOH im Register A übergeben. Ein Wert ungleich OOH entsteht, wenn auf der nächsten Position keine Daten existieren, weil das Ende der Datei erreicht ist.

Funktion 20: sequentiell lesen

Eingangsparameter:

Register C: 14H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Directory-Kode

Funktion 21: sequentiell schreiben

Eingangsparameter:

Register C: 15H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Directory-Kode

Wenn der durch DE adressierte FCB mit einer der Funktionen 15 oder 22 aktiviert wurde, schreibt Funktion 21 den Sektor auf der aktuellen DMA-Adresse in die durch diesen FCB gekennzeichnete Datei. Der Sektor wird an der durch cr vorgegebenen Position abgelegt und das cr-Feld wird auf die nächste Position

erhöht. Läuft das cr-Feld über, wird automatisch das nächste Extend eröffnet und das cr-Feld in Vorbereitung der nächsten Schreiboperation auf Null gesetzt. Schreiboperationen können in existierenden Dateien ausgeführt werden, wobei veränderte Datenblöcke die schon existierenden überschreiben. Vom Register A wird nach einer erfolgreichen Schreiboperation der Wert OOH übergeben, anderenfalls (Diskette voll) A = i.

Funktion 22: Datei anlegen

Eingangsparameter:

Register C: 16H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Directory-Kode

Diese Funktion gleicht der Eröffnungsfunktion mit dem Unterschied, daß der FCB eine Datei bezeichnen muß, die noch nicht im Inhaltsverzeichnis existiert. Das BDOS legt die Datei an und initialisiert das Inhaltsverzeichnis der Diskette mit einer leeren Datei. Bei Rückkehr ist Register A = 0, 1, 2 oder 3, wenn die Operation erfolgreich war oder FFH, wenn kein weiterer Platz im Inhaltsverzeichnis vorhanden ist. Die Funktion aktiviert den FCB, so daß keine Eröffnung notwendig ist.

Funktion 23 benutzt einen durch DE adressierten FCB, der anders aufgebaut ist als bei den bisher beschriebenen Funktionen. Es werden alle Dateien mit dem in den ersten 16 Bytes angegebenen Namen in den mit den zweiten 16 Bytes angegebenen Namen umbenannt. Der Laufwerkskode dr auf Position 0 wird benutzt, während der Kode auf Position 16 mit Null angenommen wird. Bei Rückkehr ist das Register A auf einen Wert von 0 bis 3 gesetzt, wenn die Umbenennung erfolgreich war oder FFH, falls keine Datei mit dem ersten angegebenen Namen vorhanden ist.

Funktion 23: Datei umbenennen

Eingangsparameter:

Register C: 17H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Fehlerkode

Die Anwahldaten (Funktion 24) zeigen an, auf welche Laufwerke das BDOS bereits zugegriffen hat.

Das niedwertigste Bit von Register L entspricht dem Laufwerk A: und das höchstwertigste Bit des Registers H dem sechzehnten Laufwerk P:. Eine "0" zeigt an, daß das Laufwerk noch nicht angewählt wurde, eine "1" markiert ein benutztes Laufwerk.

Funktion 24: Anwahldaten holen

Eingangsparameter:

Register C: 18H

Ausgangsparameter:

Registerpaar hl: Anwahldaten

Register A: Register L

Funktion 25 holt die Nummer des momentan ausgewählten Standardlaufwerks in Register A. Die Laufwerksnummer liegt im Bereich von 0 bis 15 entsprechend den Laufwerken A: bis P:.

Nach einem Kaltstart, einem Warmstart oder dem Rücksetzen des Diskettensystems ist die DMA-Adresse auf 80H gesetzt. Man kann

die Funktion 26 dazu benutzen, um diesen Standardwert zur Adressierung des Speicherbereiches, in dem die Daten liegen, auf den in DE vorgegebenen Wert zu ändern.

Funktion 25: Abfrage aktuelles Laufwerk

Eingangsparameter:

Register C: 19H

Ausgangsparameter:

Register A: aktuelles Laufwerk

Funktion 26: DMA-Adresse setzen

Eingangsparameter:

Register C: 1AH

Registerpaar DE: DMA-Adresse

Funktion 27: Adresse Belegungsfeld holen

Eingangsparameter:

Register C: 1BH

Ausgangsparameter:

Registerpaar HL: Adresse Belegungsfeld

Funktion 27 übergibt die Adresse der Belegungstabelle des DPB für das ausgewählte Laufwerk. Die Information in der Belegungstabelle kann jedoch ungültig sein, wenn dieses Laufwerk nur für Leseoperationen freigegeben ist. (Siehe auch DPB-Beschreibung im Abschnitt 2.1.2.)

Funktion 28: Schreibschutz

Eingangsparameter:

Register C: 1CH

Diese Funktion gewährleistet Schreibschutz für das angewählte Laufwerk. Jeder Schreibversuch auf dieser Diskette führt zur Fehlermitteilung (mit d: als Laufwerksbezeichner):

bdos err on d: r/o

Funktion 29: Schreibschutzvektor holen

Eingangsparameter:

Register C: 1DH

Ausgangsparameter:

Registerpaar HL: r/o vektor

Funktion 29 übergibt in Registerpaar HL einen bit-Vektor, der diejenigen Laufwerke angibt, bei denen Schreibschutz gesetzt ist. Wie bei Funktion 24 entspricht das niedwertigste bit dem Laufwerk A:, das höchswertigste bit dem Laufwerk P:. Das r/o-bit wird entweder durch die Funktion 28 oder durch Diskettenwechsel 13 gesetzt.

Funktion 30: Dateiattribute setzen

Eingangsparameter:

Register C: 1EH

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Fehlerkode

Diese Funktion erlaubt die programmtechnische Manipulation der Dateiattribute Schreibschutz und System (siehe Abschnitt 1.1.2.). Das Registerpaar DE adressiert einen FCB mit den gewünschten Attributen. Funktion 30 sucht im Inhaltsverzeichnis nach Übereinstimmung und ändert den entsprechenden Eintrag.

Funktion 31: DPB-Adresse holen

Eingangsparameter:

Register C: 1FH

Ausgangsparameter:

Registerpaar HL: DPB-Adresse

Die Funktion übergibt die Adresse des Disk-Parameter-Blockes des aktuellen Laufwerks (im BIOS) im Registerpaar HL.

Durch Aufruf der Funktion 32 kann ein Anwenderprogramm die aktuelle Benutzernummer ändern oder abfragen. Ist Register E = FFH, wird der aktuelle Wert der Benutzernummer im Register A übergeben, wobei der Wert im Bereich von 0 bis 31 liegt. Andernfalls wird die Benutzernummer auf den Wert der unteren

5 bit von Register E gesetzt. Man beachte, daß mit dem Kommando USER des CCP nur ein Benutzerbereich bis 15 möglich ist!

Funktion 32: Benutzernummer holen/setzen

Eingangsparameter:

Register C: 20h

Register E: OFFH (holen) oder
Benutzernummer

Ausgangsparameter:

Register A: Benutzernummer oder
kein Wert

Funktion 33: wahlfrei lesen

Eingangsparameter:

Register C: 21H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Rückkehrkode

Die Funktion 33 arbeitet mit einer Blocknummer, die aus dem 24-bit-Wert der 3 dem FCB folgenden Bytes gebildet wird (Bytes r0, r1 und r2). Byte r0 enthält den L-Teil, r1 den H-Teil der Blocknummer, r2 ist auf 0 zu setzen. Um eine Datei im wahlfreien Zugriff zu verarbeiten, muß sie zuvor im Extend 0 eröffnet werden. Bei der Rückkehr enthält Register A einen Fehlerkode oder den Wert 0, wenn die Operation erfolgreich war. In diesem

Fall enthält der aktuelle DMA-Bereich den gelesenen Sektor. Im Gegensatz zum sequentiellen Lesen wird die Blocknummer nicht erhöht. Bei jeder wahlfreien Leseoperation werden die Werte für Extend ex und den aktuellen Block cr gesetzt. Deshalb kann die Datei ausgehend von der durch wahlfreien Zugriff bestimmten Position sequentiell weiter bearbeitet werden. In diesem Fall wird der letzte Datenblock beim Übergang vom wahlfreien zum sequentiellen Lesen noch einmal gelesen bzw. der letzte Sektor wird überschrieben.

Im Register A können folgende Fehlerkodes übergeben werden:

01 Leseversuch von nicht vorhandenen Daten;

03 das aktuelle Extend kann nicht geschlossen werden;

04 Suche nach nicht vorhandenem Extend;

06 Suche über das physikalische Ende der Diskette hinaus.

Funktion 34: wahlfrei schreiben

Eingangsparameter:

Register C: 22H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Rückkehrkode

Die Funktion 34 arbeitet äquivalent dem Lesen. Die übergebenen Fehlerkodes sind die gleichen wie beim wahlfreien Lesen, wobei zusätzlich der Fehlercode 5 auftreten kann. Er zeigt an, daß ein neues Extend wegen Überlaufs des Inhaltsverzeichnisses nicht angelegt werden kann. Sind in einem Datenblock 128 Bytes geschrieben, so ist damit der ganze Block vorhanden.

Bei der Ermittlung der Dateigröße nach Funktion 35 adressiert das Registerpaar DE einen Dateisteuerblock im Format für wahlfreien Zugriff, d.h. die Bytes r0 bis r2 sind vorhanden. Der Dateisteuerblock enthält einen eindeutigen Dateinamen, der im

Inhaltsverzeichnis gesucht wird. Bei Rückkehr enthalten die Bytes für wahlfreien Zugriff die Adresse des Sektors, der dem Dateiende folgt. Die virtuelle Größe der Datei stimmt mit der realen überein, wenn die Datei sequentiell geschrieben wurde. Ist die Datei jedoch im wahlfreien Modus geschrieben worden, dann können "Löcher" in der Plazierung der Sektoren existieren und die Datei kann weniger Sektoren enthalten, als angezeigt wird.

Funktion 35: Dateigröße berechnen

Eingangsparameter:

Register C: 23H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Dateigröße im Feld für

wahlfreien Zugriff

Funktion 36: Feld für wahlfreien Zugriff
setzen

Eingangsparameter:

Register C: 24H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Feld für wahlfreien Zugriff gesetzt

Funktion 36 setzt das Feld für wahlfreien Zugriff einer Datei, die bis zu einem bestimmten Punkt sequentiell gelesen oder geschrieben wurde. Die Funktion kann auf 2 Arten nützlich sein:

Erstens ist es oft notwendig, eine Datei zuerst sequentiell zu lesen und nach Informationen ("Schlüssel") zu durchsuchen. Ist die Information gefunden, wird Funktion 36 aufgerufen, um die Position des zugehörigen Sektors zu bestimmen. Die so gewonnene Sektorposition kann in einer Tabelle für späteren Zugriff abgelegt werden. Nachdem die Datei durchsucht wurde und alle Schlüsseldatensätze in einer Tabelle abgelegt wurden, kann man auf einen bestimmten Sektor mit einem wahlfreien Zugriff unter Nutzen der vorher gespeicherten Sektornummer zurückgreifen.

Die 2. Verwendungsmöglichkeit tritt beim Umschalten vom sequentiellen Lesen oder Schreiben auf den wahlfreien Zugriff auf. Wenn eine Datei bis zu einem bestimmten Punkt sequentiell bearbeitet wurde, kann nach Aufruf der Funktion 36 die Datei ab dem bis dahin erreichten Punkt wahlfrei bearbeitet werden.

Funktion 37: Laufwerke zurücksetzen

Eingangsparameter:

Register C: 25H

Registerpaar DE: RESET-Maske

Ausgangsparameter:

Register A: 0

Im Gegensatz zur Funktion 13, die alle Diskettenlaufwerke anspricht, können hier beliebige Laufwerke aus den 16 möglichen mit Hilfe der Maske zum Rücksetzen ausgewählt werden. Die Zuordnung der Laufwerke beginnt mit bit 0 = A:; alle auf 1 gesetzten bits kennzeichnen die zurückzusetzenden Laufwerke. Die Funktionen 38 und 39 sind reserviert. Sie waren für Laufwerkszugriff und -freigabe vorgesehen und sind bei der derzeit letzten Version 2.2 des Betriebssystems nicht implementiert. Funktion 40 entspricht Funktion 34, wobei zusätzlich ein zuvor angelegter Block mit 0 aufgefüllt wird, bevor die Daten geschrieben werden.

Funktion 40: Schreiben mit wahlfreiem Zugriff und Initialisierung

Eingangsparameter:

Register C: 28H

Registerpaar DE: FCB-Adresse

Ausgangsparameter:

Register A: Rückkehrkode

Das nachfolgende Programmbeispiel zeigt die Anwendung einiger BDOS-Funktionen für den Diskettenzugriff. Es ist Teil eines Programms für einen EPROM-Programmierzusatz.

```
.z80
aseg
org 100h

;Programmteil zum Abspeichern eines Speicherbereichs
;ab Adresse 4000h auf Diskette, Dateiname waehlbar
;Extend: .ROM, Test ob Datei schon existiert

bdos equ 5

init: ld de,txt1 ;Dateiname ?
      ld c,9 ;Textausgabe
      call bdos
      ld de,fcb1 ;Eingabepuffer
      ld c,10 ;Textkette eingeben
      call bdos ;GROSSE BUCHSTABEN !
      ld hl,fcb1+2 ;Voreinstellung des fcb
      ld de,fcb2+1 ;später wirksamer fcb
      ld bc,33 ;Anzahl Bytes
      ldir ;uebertragen
      ld de,fcb2 ;test ob schon vorhanden
      ld c,17 ;erste Eintragung suchen
      call bdos
      inc a ;test directorycode
      jr z,make
      ld de,txt2 ;Datei vorhanden
```

```

ld      c, 9
call   bdos
ld      hl, fcb1+1      ;Puffer loeschen
xor    (hl)
ld      a, 20h
ld      b, 8
loei: inc   hl
ld      (hl), a
djnz  loei
jr     init      ;neuen Namen eingeben
make: ld    de, fcb2
ld    c, 22      ;Datei anlegen
call   bdos
inc   a
jr     nz, make1
ld    de, txt4      ;Fehlermeldung
ld    c, 9
call   bdos
jr     exit
make1: ld   hl, 4000h-128  ;EPROM-Puffer
ld   de, 128      ;Sektorgroesse
ld   a, (typ)
ld   b, a        ;Anzahl Sektoren
loop: push bc      ;Schleife Sektoren schreiben
push de
add   hl, de
push hl
ld   d, h
ld   e, 1
ld   c, 26      ;dma einstellen
call   bdos
ld   de, fcb2
ld   c, 21      ;sequentiell schreiben
call   bdos
or    a
jr   z, loop1      ;kein Fehler
ld   de, txt3
ld   c, 9      ;Fehlermeldung
call   bdos
jr   exit
loop1: pop  hl
pop  de
pop  bc
djnz loop
ld   de, fcb2
ld   c, 16      ;Datei schliessen

```

```
call    bdos
ld      de, txt5          ; Ausgabe OK
ld      c, 9
call    bdos
exit:  ld      c, 0
       call    bdos

; Eingabepuffer fuer Dateinamen
fcb1: db      8           ; max. Anzahl
      db      0           ; zaehler
      ds      8, 20h        ; Platz fuer Dateinamen
      db      'ROM'         ; vorgegebener Extend
      ds      21, 0         ; Voreinstellung

; fcb fuer Diskettenzugriff
fcb2: db 0
      ds 33

typ:   db 64              ; EPROM-Typ

txt1: defm  Odh, Oah, 'Dateiname :$ '
txt2: defm  Odh, Oah, 'Datei existiert schon !$'
txt3: defm  Odh, Oah, 'Diskette voll !$'
txt4: defm  Odh, Oah, 'Directory voll !$'
txt5: defm  Odh, Oah, 'OK$'

end
```

3. Kurzbeschreibung ausgewählter Programme

Die starke Verbreitung des Betriebssystems CP/M beruht zu einem Großteil auf der Fülle leistungsfähiger Programme. Einige werden als "transiente Kommandos" mit dem Betriebssystem geliefert. Neben den vielen handelsüblichen Programmen ist für den Amateur auch die "public domain"-Software sehr interessant, die von Amateuren geschaffen wurde und im allgemeinen kostenlos zur Verfügung steht. Obwohl es hier noch Unklarheiten gibt, kann doch davon ausgegangen werden, daß die Urheberrechte an Programmen wie bei anderen schöpferischen Leistungen zu beachten sind, und nicht dadurch wegfallen, daß sich Programme leicht kopieren lassen oder kostenlos vertrieben werden. Nachfolgend sollen einige mit dem Betriebssystem CP/M gelieferte oder anderweitig verfügbare Programme vorgestellt werden, wobei wegen des Platzbedarfs nur die für den Amateur interessantesten zur Sprache kommen können. Von robotron werden viele dieser Programme mit gleicher Leistung jedoch teilweise anderer Bezeichnung angeboten, die Beschreibung findet man bei Bedarf in [13].

3.1. Systemprogramme MOVCPM, STAT und PIP

Das Programm **MOVCPM** erlaubt es, die Systembestandteile CCP und BDOS auf eine gewünschte Adresse im Speicher zu bringen. Natürlich muß dann das BIOS auch entsprechend übersetzt werden. Die Syntax lautet:

MOVCPM n * (CR)

Damit wird ein Betriebssystem für einen n kByte großen TPA erzeugt. Das Programm erlaubt beim Abschluß zwei Möglichkeiten der Ergebnisauswertung. Entweder kann mit dem Systemprogramm SYSGEN sofort eine Systemdiskette angelegt werden oder man speichert mit dem residenten Befehl SAVE die beiden Teile CCP und BDOS als eine Datei. Dies ist günstiger, weil hier mehr

Möglichkeiten der Anpassung mittels anderer Hilfsprogramme bestehen (siehe Abschnitt 4.).

Das Programm STAT hat heute geringe Bedeutung. Es dient der Zuweisung von Peripheriegeräten mit Hilfe des IOBYTE und kann auch Daten des Diskettensystems liefern und verändern.

Beispiele:

A> STAT DAT.COM (CR)	Speicherplatzbedarf der Datei DAT.COM auf aktuellem Laufwerk A
A> STAT C:*.* (CR)	Übersicht des Speicherplatzbedarfs aller Dateien auf dem Laufwerk C
STAT DEV: (CR)	Anzeige der Zuordnungen logischer zu physischen Geräten
STAT log. gerätename:=physischer gerätename (CR)	Ändern der aktuellen Gerätezuordnung
STAT LW:=R/O (CR)	Ändern des Schreib/Lese-Zustandes in Schreibschutz, wird durch Warmstart aufgehoben.
STAT LW:datei \$R/O (CR)	Schreibschutz für eine Datei setzen
STAT LW:datei \$R/W (CR)	Schreibschutz für eine Datei aufheben
STAT [LW:]DSK: (CR)	Ausgabe der LW-Charakteristik

Das Programm PIP dient dem Dateiaustausch. Neben vielfältigen Varianten des Kopierens von Dateien auf den Diskettenlaufwerken ist damit auch sehr gut eine einfache Rechnerkopplung über die seriellen Schnittstellen möglich. Natürlich müssen die Anschlüsse nicht nur logisch, sondern auch technisch zueinander passen, d.h. nach dem gleichen Wirkprinzip arbeiten. Der logische Stanzerausgang des einen Computers, z.B. eine 20-mA-Stromschleife (IFSS), kann mit dem logischen Leserkanal (ebenfalls 20-mA-Stromschleife) eines anderen Computers problemlos zusammenarbeiten. Beim Aufruf des Kommandos PIP können noch Parameter zur Präzisierung des Befehls oder der Datei in eckigen Klammern angefügt werden, die in der auf Seite 67 folgenden Tabelle aufgeführt sind.

Beispiele:

a) Kopieren von Dateien

Achtung: Befindet sich auf der Diskette schon eine Datei mit dem Namen der zu erzeugenden Kopie, so wird diese über-

schrieben. Die Angabe des Laufwerkes ist nur nötig, wenn es sich nicht um das aktuelle handelt.

PIP [LW:]dateiname kopie=[LW:]dateiname original (CR)

Vereinfachungen:

- Dateiname von Original und Kopie ist gleich
PIP LW:=[LW:]dateiname original (CR)
- Original befindet sich auf dem aktuellen LW
PIP LW:dateiname kopie=dateiname original (CR)
PIP LW:=[LW:]dateiname original (CR) (gleicher Name)
- Kopie soll auf aktuelles LW geschrieben werden
PIP dateiname kopie=LW:dateiname original (CR)

Parameter	Erläuterung
V	anschließendes Vergleichen zwischen Original und Kopie
N	Einfügen von Zeilennummern in die Kopie
QString (CTRL-Z)	Abbruch des Kopievorgangs, wenn die Zeichenfolge "string" im Original erkannt wird (Zeichenfolge selbst wird noch übertragen)
Sstring (CTRL-Z)	Kopieroperation beginnt ab der angegebenen Zeichenfolge
E	Anzeige der Übertragenen Datei auf Bildschirm
O	Übertragen von Objektdateien (nicht ASCII), Ignorieren des Dateiendezeichens (CTRL-Z)
U	Umsetzung aller Klein- in Großbuchstaben
L	Umsetzung aller Groß- in Kleinbuchstaben
Gn	Auswahl der Datei vom Nutzerbereich n

b) Verketten von Dateien

PIP LW:verkettete datei=LW:datei1,...,datein (CR)

- wird die Befehlszeile länger als eine Bildschirmzeile, so muß (CTRL-E) betätigt werden.
- die Dateien werden in angegebener Reihenfolge aneinandergefügt.

c) Drucken von Dateien

A> PIP LST:=DOKU.TXT[E] (CR)

Drucken der Datei DOKU.TXT und gleichzeitige
Anzeige auf dem Bildschirm

d) Einlesen von Lochstreifen/serieller Schnittstelle

PIP LW:filename=RDR: (CR)

e) Ausgabe auf Stanzer/serielle Schnittstelle

PIP PUN:=LW:filename (CR)

Mit Befehl a) Kann mittels PIP auf einfache Weise eine Datei von der Konsole auf Diskette gebracht werden. Die Eingabe muß man dann mit (CTRL-Z) abschließen:

PIP TEST.TXT =CON: (CR)

Dies ist ein Test zum....^Z

Das Programm **SUBMIT** erlaubt die Ausführung einer Befehlsfolge mit Hilfe einer ASCII-Datei, in der Befehle so stehen, wie sie sonst an der Konsole eingegeben werden. Die ASCII-Datei kann mit einem beliebigen Editor oder mit PIP erzeugt werden und muß das Extend .SUB aufweisen. Es sind variable Parameter zulässig, die mit \$1 ... \$n bezeichnet werden müssen und durch Leerzeichen voneinander zu trennen sind.

3.2. Universelle Dienstprogramme POWER und ZSID

Die oben vorgestellten Systemprogramme bieten oft nur das Notwendigste und repräsentieren einen Entwicklungsstand, bei dem die Programme noch für geringsten Speicherplatzbedarf ausgelegt waren. Einige modernere Programme bieten mehr Komfort oder erlauben zusätzliche Funktionen.

Für den Amateur sind vor allem Programme von Bedeutung, die Diskettenarbeit und Inbetriebnahme eigener Programme unterstützen. Solche Programme werden oft als Monitor oder Debugger bezeichnet. Herausragende Beispiele dafür sind die beiden nachfolgend beschriebenen Programme.

POWER ist ein universelles Manipulationsprogramm für Diskettenarbeit. Ein vergleichbares Programm wird von *robotron* unter der Bezeichnung **DIENST** angeboten [13]. Nach dem Aufruf meldet sich das Programm mit Namen, Versionsnummer und einem Copyright-Vermerk. Gibt man ein Fragezeichen ein, so erscheint eine Übersicht der Funktionsaufrufe, die zumeist den Zweck bereits angeben. Durch nachfolgende Parameter können viele häufig vorkommende Aufgabenstellungen der Diskettenarbeit sehr komfortabel erledigt werden. Die Eingabe dieser Parameter weicht teilweise vom sonst bei CP/M üblichen ab, sie erfolgt teilweise im Dialog, d.h. auf Anfrage, und man kann die Dateien auch mit Ziffern bezeichnen, aufeinanderfolgende auch mit Bindestrich, beispielsweise 3-14. Folgende Abkürzungen werden in der Beschreibung der Kommandos verwendet:

com	Kommando
ufn	Filename
adr	Adresse, hexadezimal angegeben

Kommandos:

CHECK Errechnet eine Kontrollsumme für die ausgewählten Dateien.

CM *aadr eadr adr2* Vergleicht einen Speicherbereich byteweise von Anfangsadresse *aadr* bis Endadresse *eadr* mit dem Bereich ab Anfangsadresse *adr2*.

COPY Die ausgewählten Dateien werden auf ein anderes Laufwerk kopiert. Die Optionen sind:

- [A] automatisches Kopieren auch der Überlagerungen;
- [B] automatisches Ändern der Namenserweiterung in .BAK;
- [C] fragt, wenn File bereits existiert;
- [D] löscht automatisch vorhandene Files gleichen Namens;
- [T] stellt die Reihenfolge des Kopierens nach der Größe;
- [M] markiert originale und kopierte Files;
- [Q] ändert den Namen während des Kopierens.

DIR [*UX1*] Anzeige der Directory.

U - alle USER-Felder;
X - alle gewünschten Laufwerke;
1 - Anzahl der Spalten.
Kennzeichnung der Files:

* - File ist R/O (schreibgeschützt)
(..) - File ist SYS (systemgeschützt)
Durch die Option [S] Kann die Anzeige von Systemfiles unterdrückt werden.

DISK Anzeige des DISK-Status.

DUMP adr Anzeige eines Speicherbereichs in ASCII.

DUMPX HEX- und ASCII-Anzeige auf Bildschirm

DUMPH HEX-Anzeige auf Bildschirm

DUMPA ASCII-Anzeige ohne Tabulatoren.

Die Anzeige erfolgt als Text. Nicht druckbare Zeichen werden nicht angezeigt. Die Adresse kann man wie folgt eingeben:

- 128 Bytes ab aktueller Position;
- , i - i Bytes ab aktueller Position;
- , , - weiter ab aktueller Position;
- adr - 1 Byte von adr;
- adr, i - i Byte ab adr;
- adr, , - 128 Bytes von adr;
- adr,, - alle Bytes.

DS adr, len Ab Adresse adr werden die Bytes einzeln hexadezimal, binär, als ASCII-Zeichen und dezimal angezeigt. Es kann dann eventuell ein neuer Wert eingegeben werden. Ein in spitze Klammern gesetzter Buchstabe zeigt an, in welcher Form die Eingabe erfolgt (H - hexadezimal, D - dezimal, B - binär, A - Zeichen, wobei ein vorangestelltes ^ aus dem Zeichen die 5 bis 7 ausblendet /CTRL/). Folgende Eingaben schalten zwischen den einzelnen Eingabeformaten um:

- . h es folgen hexadezimale Eingaben
- . d es folgen dezimale Eingaben
- . b es folgen binäre Eingaben
- . a es folgen Zeicheneigaben
- .. das Kommando DS wird beendet

ERA Löschen der ausgewählten Dateien.

ERA [R] Löschen mit vorheriger Frage.

EXIT Programm POWER beenden.

FILL aadr eadr byte Füllt Speicher von aadr bis eadr mit byte.

GO ufn adr Lädt File ufn nach adr und startet es.

GROUP Zeigt alle Filegruppen auf Display an.

Als Filegruppen gelten die Dateiblöcke, die in der Directory

die gestreute Speicherung auf Diskette angeben. Es wird die logische Nummer jedes verwendeten Dateiblockes der Datei angegeben, vor jeder Angabe steht noch die Extend-Nummer.

JP adr arg Es wird Programmstart ausgeführt, arg ist die Kommandozeile wie im Betriebssystem.

EX adr arg Startet Programm ab Adresse, dann zurück nach POWER

LOAD ufn adr Mit diesem Kommando kann die Datei ufn auf Adresse adr geladen werden. Der Bereich von POWER (bis etwa 4000H) darf nicht überschrieben werden (siehe LOG).

SAVE ufn adr [sect] Das Kommando speichert die angegebene Zahl von Sektoren auf Diskette. Diese Angabe kann entfallen, wenn vorher das File ufn mit LOAD geladen wurde.

LOG zeigt die gestellten Schalter auf Display an. Gleichzeitig wird der freie Anwenderspeicher angegeben. Die Schalter können durch die Angabe in eckige Klammern gestellt werden. Jede Angabe schaltet die Wirkung um:

- A - automat overlaying files when COPY
- B - automat rename old files to .BAK when COPY
- C - asks if file exists when COPY
- D - automat skipping existing files when COPY
- P - paging (ON - paging, OFF - scrolling)
- Q - renaming files when copying
- R - request Y/N on currend files
- S - display or hide SYSTEM files
- T-serie file copying (ON - abort, OFF -next smaller file)
- M - mark original and copied files

MOVE aadr eadr aadr2 Speichertransport. Es wird der Speicherbereich aadr bis eadr nach aadr2 verschoben. Der Transport erfolgt aufsteigend, Überdeckungen sind zu vermeiden.

PASS Eingabe Paßwort. Es kann nach PASS ein Paßwort eingegeben werden, wobei nur das Wort "POWER" (Großbuchstaben!) als gültig akzeptiert wird.

READ trk sec Lesen von Diskette. Von Spur trk, Sektor sec wird i Sektor vom aktuellen Laufwerk gelesen und auf Speicherbereich 80H bis OFFH transportiert.

WRITE trk sec adr nsec Ab Spur trk, Sektor sec werden nsec

Sektoren vom Speicherbereich adr (oder, falls adr fehlt, dann von 80H) geschrieben.

READ trk sec kind nsec Ab Spur trk, Sektor sec werden nsec Sektoren gelesen und auf Display dargestellt oder in den Speicher geladen. kind bedeutet dabei:

XX HEX- und ASCII-Darstellung;

XH HEX-Darstellung;

XA ASCII-Darstellung.

READGR ngroup mem nsec Es werden nsec Sektoren einer Speichergruppe (logische Nummer wie DM in Directory) von Diskette gelesen und nach dem Speicher ab Adresse mem transportiert.

WRITEGR ngroup mem nsec Schreiben auf Diskette, jedoch vom Speicher ab Adresse mem, logische Gruppennummer ngroup mit nsec Sektoren.

READER ngroup kind nsec Wie vorher Lesen der Dateigruppe und Anzeige auf Display. kind darf XX, XH oder XA sein.

RECLAIM Wiederherstellen gelöschter Files, bereits gelöschte Files werden wieder gültig, wenn sie im Inhaltsverzeichnis noch den alten Dateninhalt haben.

REN Die ausgewählten Files können umbenannt werden. Dazu wird bei jedem File nach dem neuen Namen gefragt. Die Angabe eines Sterns "*" im Namen oder der Namenserweiterung lässt den alten Namensteil unverändert:

*.ext --> setzt eine andere Namenserweiterung ext;

name.* --> setzt einen anderen Namen name.

RESET d: Dieses Kommando gestattet das Rücksetzen einzelner Laufwerke und damit den Diskettenwechsel auf diesen Laufwerken. (Mit ^C können alle Laufwerke zurückgesetzt werden.)

RUN [ufn param] Lädt und startet das .COM-File ufn. Es dürfen Parameter angeben werden.

SEARCH aadr eadr byte... Sucht im Speicher nach Bytes ab aadr bis eadr. Die gefundenen Bytes werden mit ihrer Adresse angezeigt. Für Byte kann außerdem angegeben werden:

? zeigt alle Bytes des Bereiches an;

"" schließt ein ASCII-Zeichen ein, das gesucht wird.

SET [-{1..8,S,R,X}] Setzt Files-Kennzeichen. Es wird bei Angabe einer Ziffer n das n-te Zeichen des Filenamens mit bit 7 = 1 gesetzt. S, R und X setzen bit 7 = 1 in der Namenserweiterung.

SET [+{....}] Setzt Files-Kennzeichen zurück. Parameter wie vorher, es wird im angegebenen Zeichen bit 7 auf 0 gesetzt.

SET Zeigt die gesetzten 7. bits an.

SETHOOK Setzt DIR/SYS-Attribute.

SETSYS

SETRO Setzt R/O- oder R/W-Attribute

SETWR

SIZE Die Größe der Datei wird in Sektoren, freien Sektoren, kbytes und Gesamtbelegung angezeigt.

SORT i Sortiert Files zur Anzeige:

i = 0 unsortiert;

i = 1 sortiert nach den Filenamen;

i = 2 wie 1, SYS-File zuletzt;

i = 3 sortiert nach Namenserweiterungen;

i = 4 wie 3, SYS-Files zuletzt.

SPEED i Setzt Ausgabegeschwindigkeit

i = 0 schnell;

...

...

i = 9 sehr langsam.

STAT d: Zeigt freien Speicherplatz auf Diskette d und allen anderen aktiven Laufwerken. Aktiv sind alle die Laufwerke, die seit dem Rücksetzen schon einmal angesprochen wurden.

TEST Liest alle Sektoren und zeigt dies an. Fehlerhafte Sektoren werden mit ===..., == angezeigt. TEST repariert alle Filegruppen und defekten Files und errechnet eine Kontrollsumme. Nicht reparierbare Spuren werden mit einem besonderen Filenamen belegt und mit in die Directory aufgenommen (R/O und SYS-Attribute gesetzt).

TEST S Test wie vorher, es werden zusätzlich die Systemspuren überprüft.

TYPE [A,H,X] Zeigt Files auf Display mit Spur und Sektor an. ^K beendet die Ausgabe, es wird das nächste File angezeigt. Die Ausgabe kann mit einer Option gesteuert werden:

typea Ausgabe in ASCII;

typeh Ausgabe in HEX;

typex Ausgabe in ASCII und HEX.

USER i Schaltet zum USER i um

XUSER 1 Setzt die Quelle für das Kopieren auf Anwender 1.
UR1 bis 4 Man kann Anwenderkommandos in POWER installieren und aufrufen. Der Ansprung muß in folgenden Speicherplätzen stehen:
UR1 140 bis 147H; UR3 150 bis 157H;
UR2 148 bis 14FH; UR4 158 bis 15FH.

Das Programm ZSID ist ein komfortabler Debugger zum Test von Programmen. Nach dem Aufruf verschiebt er sich an das obere Ende des Speichers, und man kann die zu testenden Programme wie gewohnt auf Adresse 100H laden. Bei der Ein- und Ausgabe gilt folgender Zeichensatz:

Hex-Zahlen (voreingestellt) : 0 bis 9, A bis F
Dez-Zahlen : 0 bis 9 (durch vorangestelltes # gekennzeichnet, z. B. #59)
Textkette (engl. :string) : in Apostrophe eingeschlossen.
Symbole können verwendet werden, falls eine Symboldatei (name.SYM) geladen wurde. In eckige Klammern eingeschlossene Parameter sind wahlweise zu benutzen, können auch entfallen.

Kommandos:

A[adr] Befehlswörter übersetzen einer eingegebenen Z80-Mnemonik, Verlassen von A durch Eingabe eines Punktes.

Cadr Abarbeiten eines Unterprogramms.

D[adr][,eadr] Speicherbereich (adr - eadr) auf Bildschirm anzeigen, Standard: 12 Zeilen

Fadr,eadr,byte Füllen des Speicherbereichs (adr - eadr) mit dem Byte.

G[adr][,breakadr1][,breakadr2] Start ab adr mit den Unterbrechungspunkten breakadr1 und breakadr2

H[hhhhi][,hhhh2] Rechnen mit hexadezimalen Zahlenwerten, Ausgabe von hhhh1 + hhhh2 und hhhh1 - hhhh2 oder hhhh1 ssssi ddddi oder Liste der Adressen aus der Symboleitabelle.

Iname1.com, name2.com,..., nami.sym,... Initialisierung eines Dateisteuerblocks

L[-]L[adr][,eadr] Auflisten eines Speicherbereichs in Z80-Mnemonik, Datenbytes werden als ?? = adr ausgegeben.

Madr,eadr,vadr Verschieben von zusammenhängenden Speicherbereichen (adr - eadr) nach vadr...

P[-]P[adr][, awert] Durchlaufzähler setzen auf adr und initialisieren mit awert (Standard: 1)

R[adr] Laden der mit dem I-Kommando festgelegten Dateien ab adr + 100H, ohne Angabe auf Adresse 100H.

S[W]adr Speichermanipulation ab adr W (byteweise), beenden mit der Eingabe eines Punktes.

[‐]T[anz][, zusatz] Abarbeitung und Protokollierung von anz Programmschritten. [, zusatz] aktiviert die Zusatzfunktionen TRACE. UTL oder HIST. UTL.

[‐]U[W]... Das U-Kommando arbeitet entsprechend dem Kommando T, protokolliert jedoch nur Start- und Passpunkte.

X[buchstabe] CPU-Status anzeigen und durch Eingabe eines Buchstabens die Möglichkeit zur Änderung der

Flags : C, Z, M, E, I,

und Register : A, B, D, H, S, A', B', D', H', X, Y, P

3. 3. Public-Domain-Software

Von Amateuren aus Freude an der Sache geschaffene Programme können durchaus professionelle Leistung bringen und sind auf Grund der kostenlosen Weitergabe besonders interessant. Es soll deshalb hier auf 2 Programme hingewiesen werden, die von DDR-Amateuren geschaffen wurden.

Das Programm **RAZ80** von *H. D. Frey*, Erfurt ist ein komfortabler Reassembler für Z80-Kode. Die erzeugten Quellprogramme mit dem Extend .RAZ kann man mit WordStar und dem Assembler M80 (siehe Abschnitt 3.4.) weiterverarbeiten. Als Parameter ist beim Aufruf die zu reassemblierende Datei anzugeben, anderenfalls erscheint ein Menü mit Bedienungshinweisen. Mit verschiedenen wahlweise einzugebenden zusätzlichen Parametern kann die Ausgabe gesteuert werden. Da das Programm mit Dokumentation bei den Amateuren im Umlauf ist, soll eine ausführlichere Beschreibung hier nicht erfolgen.

Von *H. Höning*, Berlin stammt der leistungsfähige Tracer **TR80**. Er verfügt über etwas mehr Möglichkeiten als ZSID auf dieser Strecke. So kann wahlweise bei jedem Programmschritt ein vorinstellbarer Speicherbereich angezeigt werden. Auch dieses

Programm ist mit Dokumentation im Umlauf, so daß weitere Erläuterungen nicht nötig sind.

Das Programm **NSWEET** von D. Rand aus Canada ist auch in der DDR verbreitet. Es erlaubt neben anderen als wichtigste Funktion das Komprimieren von Dateien bei der Aufzeichnung auf Disketten. Bei Sicherheitskopien von Quellprogrammen und Dokumentation kann man bis etwa 30% des Platzes einsparen. Beim Aufruf meldet es sich mit einem kurzen Menü, danach werden die auf der Diskette vorhandenen Dateien nacheinander (nach Betätigung der CR-Taste) angezeigt. Sollen Dateien komprimiert oder dekomprimiert werden, so müssen sie zunächst mit dem Buchstaben t markiert werden. Mit dem Buchstaben q erfolgt dann die Auswahl komprimieren/dekomprimieren. Das Ansehen von Dateien ist auch in komprimierter Form mit dem v-Kommando möglich. Das ist manchmal notwendig, weil die Dekompression bei langen Dateien doch etwas Zeit benötigt und außerdem der Platz auf der Diskette u.U. knapp ist.

3.4. WordStar

Das Programm **WS** ist besonders bekannt, weil Schreibarbeiten heutzutage zu den häufigsten Tätigkeiten gehören und es als eines der ersten Programme die Form des Bildschirm-Menüs einsetzte. Während der Bedienung wird außer dem in Bearbeitung befindlichen Text auf Wunsch im oberen Teil des Bildschirms eine kurze Erklärung der in der augenblicklichen Betriebsart möglichen Kommandoeingaben abgebildet. Dadurch kann auch der Ungeübte sofort arbeiten. Wer die Kommandos beherrscht, kann die Hilfe stufenweise abschalten. Das Programm arbeitet in 2 Betriebsarten, den Modi "Textverarbeitung" und "Programmbearbeitung". Ersterer ist in [14] ausführlich beschrieben, da von robotron ein vergleichbares Programm unter der Bezeichnung TP angeboten wird. Die Bearbeitung von Programmdateien ist wesentlich einfacher, es lohnt sich jedoch, dafür das Komfortable WS einzusetzen. Nach dem Aufruf meldet sich das Programm mit einem Anfangsmenü, in dem man u.a. die Betriebsarten auswählen kann. Die nachfolgende Übersicht stellt die einzelnen Kommandos vor.

Kursorpositionierung, Rollen, Blättern

^S	i Zeichen nach links	^D	i Zeichen nach rechts
^A	i Wort nach links	^F	i Wort nach rechts
^E	i Zeile nach oben	^X	i Zeile nach unten
^W	Rollen 1 Zeile rückwärts	^Z	Rollen 1 Zeile vorwärts
^R	3/4 Bild rückwärts	^C	3/4 Bild vorwärts
^QS	Zeilenanfang	^QD	Zeilenende
^QE	in 1. Bildzeile	^QX	in vorletzte Bildzeile
^QW	Rollen rückwärts	^QZ	Rollen vorwärts
^QR	an Dateianfang	^QC	an Dateiende
^QB	an Blockanfang	^QK	an Blockende
^QV	an Position vor letztem Such- oder BlockKommando		
^QP	an Position vor letztem Kommando		
^Qn	an Marke n (n : 0 bis 9)		

Tabulator-Kommandos

^I	Tabulatorsprung
^OI	Setzen Tabulator
^ON	Löschen Tabulator
^OF	Tabulatorzeile = Musterzeile
^OV	variable Tab. ein/aus

Kommandos zur Texteingabe

^V	Einfügen ein/aus
(CR)	Ende Absatz, Zeilenschaltung
^N	Zeile einfügen
^PO	Leerstelle im Wort
^P-	Minuszeichen (unabhängig von ^OE)

Lösckommandos

	Zeichen links vom Kursor	^QY	ab Kursor bis Zeilende
^G	Zeichen auf Kursorposition		
^T	ab Kursor bis Wortende	^Y	gesamte Zeile
^Q	ab Zeilenanfang bis Kursor	^KY	markierter Block

Formatierungskommandos

^OC	Zeile zentrieren	^OT	Tab.-Zeile ein/aus
^OL	Einstellen linker Rand	^OJ	Blocksatz ein/aus
^OR	Einstellen rechter Rand	^OH	Trennhilfe ein/aus
^OX	Randbegrenzung aufheben	^OE	Weicher Trennstrich ein/aus
^OG	Absatz einrücken 1 Tab.	^OD	Steuerzeichenanzeige ein/aus
^OS	Setzen Zeilenabstand	^OP	Seitenanzeige ein/aus
^OW	Wortumschlag ein/aus	^B	Formieren

Suchen und Ersetzen

^QF Suchen
^QA Suchen und Ersetzen
^L Wiederholen ^QF bzw. ^QA
^QV Kursorposition vor letztem ^QF-/^QA-Kommando

Parameter bei Suchen und Ersetzen

N Suchen n mal (ab Kurs) **B rückwärts suchen**
U Ignoriere Groß/Klein **G gesamte Datei (nur bei ^QA)**
W nur ganze Wörter **N ohne Abfrage (nur bei ^QA)**

Spezialzeichen im Suchbegriff:

^P^A Jedes Zeichen befriedigt diese Stelle
^P^S Jedes Sonderzeichen befriedigt diese Stelle
^Ox Jedes Zeichen befriedigt diese Stelle außer "x"
^N Jedes Zeilenende befriedigt diese Stelle

Marken- und Blockkommandos

^Kn Setzen/Löschen einer Markierung n (n : 0 bis 9)
^KN Spaltenmodus ein/aus
^KB Blockanfangsmarke setzen/löschen
^KK Blockendemarke setzen/löschen
^KV Blocktransport (einfügen mit löschen)
^KC Blockkopie (einfügen ohne löschen)
^KY Block löschen
^KH Blockmarkenanzeige ein/aus

Dateikommandos

^KW Block als Datei auf Disk schreiben
^KR Datei von Disk lesen und einfügen
**^R letzten Dateinamen aktivieren
(in ^KW, ^KR und Anfangsmenü)**
^KO Kopieren einer Datei
^KE Umbenennen einer Datei
^KJ Löschen einer Datei
^KL Ändern Arbeitslaufwerk
^KF Inhaltsverzeichnisanzeige ein/aus
^KP Drucken einer Datei

Kommandos für Speicherung und Abbruch

- ^KS Speichern und Weiterarbeit mit der Datei
- ^KX Speichern Datei, zum Betriebssystem
- ^KD Speichern Datei, zum Anfangsmenü
- ^KQ Abbrechen ohne Speichern der Datei, zum Anfangsmenü

Sonstige Kommandos

- ^QQ Kontinuierliches Wiederholen
- ^U Unterbrechen einer Funktion
- ^J Hilfsmenü

DRUCKSTEUERUNG

Kommandos im Text

^PC Druckpause	^PA Zeichenabstand
^PH Rückschritt	alternativ
^PS Unterstreichen ein/aus	^PN Zeichenabstand
^PX Durchstreichen ein/aus	Standard
^PY Farbband- umschaltung ein/aus	^P(CR) Rücklauf ohne Zeilen- schaltung
^PD Doppeldruck ein/aus	^PQ Anwendersteuerzeichen 1
^PB Fettdruck (Schatten) ein/aus	^PW Anwendersteuerzeichen 2
^PV Tiefschrift ein/aus	^PE Anwendersteuerzeichen 3
^PT Hochschrift ein/aus	^PR Anwendersteuerzeichen 4

Punktkommandos

(n = Zeilen, m = Zeichen bzw. Spalte)

.PLn Setzen Zeilen pro Blatt	.FO..	Fußzeile
.MTn Setzen oberer Rand	.OP	Seitennummerdruck aus
.MBn Setzen unterer Rand	.PN	Seitennummerdruck ein
.HMn Setzen Kopfabstand	.PNr	Seitennummer setzen
.FMn Setzen Fußabstand	.POm	Verschiebung linker
.PCm Setzen Seiten-Nr.-Spalte	.Druckrand	Druckrand
.PA Vorschub neue Seite	.IGtext	Kommentarzeile
.CPn eventuell neue Seite	..text	Kommentarzeile
.HE.. Kopfzeile	^PK	kein Punktkommando

KOMBO-Druck-Kommandos

. DF...	Variablen-Datei aufrufen	. SV...	Variablenwert setzen
. RV...	Variablen-Werte lesen	. FI...	Datei einfügen
. AV...	Variableneingabe von Tastatur	. DM...	Meldung auf Bildschirm
		. CS...	Bildschirm löschen
		&name&	Variablenbezeichner

Kommandos für Typenraddrucker

- . LHn Zeilenabstand, n in 1/48"
- . SRn Hoch-/Tiefschrift Vorschub, n in 1/48"
- . CWr Zeichenabstand, m in 1/120"
- . UJr Mikrojustifikation ein(r=1)/aus(r=0)
- . BPr Vor-Rückw.-Druck ein(r=1)/aus(r=0)
- ^PB Schattenschrift ein/aus
- ^PF 95. Druckzeichen
- ^PG 96. Druckzeichen
- ^PP Proportionalschrift ein/aus

Das Kennzeichen (CR) bedeutet Abschlußtaste, RETURN, ENTER.

3. 5. M80, L80 und C80

In vielen Fällen ist die Programmierung in Assemblersprache unumgänglich, insbesondere bei der Anpassung von Peripheriegeräten oder anderen Problemen, wo es auf höchstmögliche Arbeitsgeschwindigkeit und geringen Speicherplatzbedarf ankommt. Standardwerkzeuge beim Betriebssystem CP/M sind dazu die Programme

M80 Assembler

L80 Linker (Programmverbinder)

C80 Crossreferenz

Man kann den Assembler **M80** mit und ohne Parameter aufrufen. Im letzten genannten Fall gibt er einen Stern (*) aus und zeigt damit an, daß er bereit ist, eine Kommandozeile entgegenzunehmen. Für die Quelldateien ist als Extend .MAC voreingestellt, man kann jedoch auch beliebige andere verwenden. Die erste Anweisung des Quelltextes sollte .Z80 lauten, um mit Sicherheit den Z80-Modus

einzustellen. Der Assembler arbeitet auch im 8080-Modus, wodurch bei Programmen in Z80-Mnemonik Fehler entstehen. Der Anfänger findet in den Programmen im Abschnitt 4. viele Anregungen zur Nutzung des Assemblers.

Die Kommandozeile des Assemblers

- * ,=PROG Syntaxtest der Datei PROG
- * =PROG Erstellen der Objektdatei PROG.REL
(Objektdateiname = Quelldateiname mit Ausnahme der Namenserweiterung).
- * NAME=PROG Übersetzen der Quelldatei PROG.MAC und Erzeugen der Objektdatei PROG.REL.
- * ,PROG=PROG Erzeugt die Druckdatei PROG.PRN aus der Quelldatei PROG.MAC.
- * ,LST:=PROG Assemblerliste der Datei PROG.MAC wird am Drucker ausgegeben.
- * ,TTY:=PROG Ausgabe der Assemblerliste am Bildschirm
- * PROG,PROG=PROG Übersetzen der Quelldatei PROG.MAC, Erzeugen der Objektdatei PROG.REL und der Druckdatei PROG.PRN
- * ,TTY:=TTY: Lesen einer Zeile von der Tastatur, Übersetzen und Ausgabe am Bildschirm
- * ,LST:=TTY: Wie vorhergehender, jedoch Ausgabe am Drucker.

Es ist möglich, das Kommando gleich hinter den Aufruf zu schreiben, der Assembler kehrt dann nach der Kommandoausführung wieder in das Betriebssystem zurück. Befinden sich die Dateien nicht im aktuellen Laufwerk, muß der Laufwerksbezeichner vor dem Dateinamen stehen.

Beispiel:

M80 B:PROG=C:PROG

Schalter

- * PROG=PROG/S1/S2 Übersetzen der Quelldatei PROG.MAC und Erzeugen der Objektdatei PROG.REL unter Berücksichtigung der Schalter S1 und S2.

Folgende Schalter des Assemblers sind möglich:

Schalter	Bedeutung
/O	Der Assembler druckt die Adressen in der Druckdatei oktal (zur Basis 8).
/P	Jedes /P legt einen extra Stack von 256 Bytes zur Verwendung während der Übersetzung an.
/R	Erzwingt das Erzeugen einer Objektdatei mit dem gleichen Namen wie die Quelldatei.
/X	Der /X-Schalter setzt den Standard und die aktuelle Zuordnung auf Unterdrücken der Liste falscher Bedingungen.
/L	Erzwingt das Erzeugen einer Druckdatei mit dem gleichen Namen wie die Quelldatei.
/C	Veranlaßt den Assembler, eine spezielle Druckdatei mit dem gleichen Namen wie die Quelldatei anzulegen, für die Verwendung mit C80. Wenn das Erzeugen einer Cross-Referenz gewünscht wird, muß das Programm mit diesem Schalter übersetzt werden.
/Z	Veranlaßt den Assembler, Z80-Operationskodes zu übersetzen.
/I	Veranlaßt den Assembler, 8080-Operationskodes zu übersetzen.
/P	Das ist der Standard. Der Assembler druckt die Adressen in der Druckdatei hexadezimal.
/M	Der /M-Schalter initialisiert einen Blockdatenbereich, der mit DS definiert wurde, auf OOH.

Fehlerkodes

Fehlerkode	Bedeutung
A	Argumentfehler. Das zu der Pseudooperation gehörende Argument hat nicht das korrekte Format oder liegt außerhalb des zugelassenen Bereichs.

Fehlerkode	Bedeutung
C	Fehler der Bedingungsschachtelung. ELSE ohne IF, ENDIF ohne IF, zweimal ELSE für ein IF, ENDC ohne COND.
D	Doppelt definiertes Symbol. Bezugnahme auf ein Symbol, das mehrfach definiert wurde
E	Externer Fehler. Verwendung eines externen Symbols, das in diesem Zusammenhang nicht erlaubt ist.
M	Mehrfach definiertes Symbol.
N	Fehler einer Zahl. Fehler in einer Zahl; meist ein falsches Zeichen (z. B.: 8Q).
O	Falscher Operationskode oder nicht einwandfreie Syntax. SET, EQU oder MACRO ohne einen Namen; falsche Syntax in einem Operationskode oder falsche Syntax in einem Ausdruck (nichtpaarige runde Klammern, Anführungszeichen, aufeinanderfolgende Operatoren usw.)
P	Phasenfehler. Der Wert einer Marke oder eines EQU-Namens ist im Pass 2 anders als im Pass 1.
Q	Fragwürdig. Meistens ist eine Zeile nicht ordentlich abgeschlossen. Das ist eine Warnung.
R	Verschiebung. Nichterlaubte Verwendung einer Verschiebung in einem Ausdruck (z. B.: abs-rel). Daten, Code und Common-Bereiche sind verschieblich.
U	Undefiniertes Symbol. Ein Symbol, auf das in einem anderen Ausdruck Bezug genommen wird, ist nicht definiert.

Fehlercode	Bedeutung
V	<p>Wertfehler.</p> <p>Eine Pseudooperation (z. B.: .RADIX, .PAGE, DS, IF, IFE), hat im Pass 1 einen undefinierten Wert, obwohl er im Pass 1 bekannt sein müßte. Wenn das Symbol später im Programm definiert wird, erzeugt der Assembler im Pass 2 keinen U-Fehler.</p>

Heldungen

XNO END statement

Keine END-Anweisung.

Unterminated conditional

Wenigstens eine Bedingung ist am Ende der Datei nicht abgeschlossen worden.

Unterminated REPT/IRP/IRPC/MACRO

Es ist wenigstens ein Block nicht abgeschlossen worden.

Symbol table full

Bei der Bildung der Symbolebene durch den Assembler ist der verfügbare Speicher erschöpft.

Die Makroblöcke werden vollständig in der Symbolebene abgespeichert, einschließlich der Kommentare.

Um die Kommentare innerhalb der Makroblöcke aus der Symbolebene auszuschließen, sollten vor die Kommentare zwei Semikolons (;;) geschrieben werden.

{xx, No} Fatal Errors [, xx warnings]

Anzahl der schweren Fehler und Warnungen, die im Programm aufgetreten sind.

Diese Meldung wird am Ende jeder Übersetzung auf den Bildschirm und in die Druckdatei ausgegeben.

Der Programmverbinder L80 von der Firma *MICROSOFT* kann alle mit den Programmiersprachen *FORTRAN*, *COBOL*, *BASIC* und mit dem Assembler M80 hergestellten Programme problemlos verbinden. Diese Programmsysteme erzeugen .REL-Dateien, die alle notwendigen Informationen für den Linker enthalten.

Das Programm wird mit seinem Namen L80 aufgerufen. Danach erscheint das Systemzeichen "*", das auf die Bereitschaft zur Kommandoeingabe hinweist. Als Parameter können aber auch die zu linkenden Programme direkt im Aufruf stehen.

Das Kommandoformat lautet:

file1, file2, ..., filen

Die Namenserweiterung für alle File ist .REL; das ist nicht Bestandteil des Filenamens im Kommando, kann jedoch zusätzlich anders angegeben werden. Vor jedem Filenamen kann außerdem noch ein Laufwerkbezeichner angegeben werden, wenn sich das File nicht auf der aktuellen Diskette befindet.

Nach einer Kommandoeingabe, die mit (CR) abgeschlossen wird, beginnt das Linken. Alle Symbole, die undefiniert geblieben sind, werden auf dem Bildschirm ausgegeben, gefolgt von einem Sternchen *. Ist ein externes Symbol damit bezeichnet, kann zuerst die Bibliothek danach abgesucht werden. Dies kann durch eine Spezifikation mit einem sogenannten Schalter erfolgen. Diese Schalter beginnen durch einen Schrägstrich und bestehen aus einem Großbuchstaben. Schalter können beliebig aufeinander folgen, wenn damit eine vernünftige Wirkung erreicht wird. Die Schalter können in verschiedenen Stufen zur Anwendung kommen:

- auf der Kommandostufe

Es ist möglich, nur einen Schalter nach dem Systemzeichen * einzugeben bzw. diesen Schalter als erstes Zeichen einer Kommandokette anzugeben:

- /G Teilt L80 mit, daß die Ausführung des Programms, das bereits geladen sein muß, begonnen werden kann. Dieses Programm wird gestartet und abgearbeitet.
- /G:name Legt die Startadresse nicht auf 100H, sondern auf die Adresse, die der Name hat (AnsprungAdresse).
- /M Alle Globalbezüge der Programme, die bereits geladen worden sind, werden aufgelistet.

- /P:adr Das nachfolgend bezeichnete Programm wird ab der Adresse adr geladen.
- /D:adr Es wird der Datenbereich für das nachfolgend bezeichnete Programm ab Adresse adr geladen.
- unmittelbar nach dem Dateinamen
- Die folgenden Schalter beziehen sich ausschließlich auf eine Datei und sind sofort nach dem Dateinamen anzugeben:
- /S Durchsuchen der Datei nach unbefriedigten Externalen und Laden der erforderlichen Bibliotheksmodule.
- /N Der Name, der vor diesem Schalter steht, wird künftiger Programmname. Er steht dann als .COM-File auf der im Namen angegebenen Diskette. Ein Sprung zum Programmeintrittspunkt wird bei Bedarf eingefügt. Diese Datei wird jedoch nicht als zu linkende Datei aufgefaßt.
- /E Ende des Linkens und Rückkehr in das Betriebssystem. Die Systembibliothek wird daraufhin untersucht, ob ein nicht befriedigtes Globalsymbol befriedigt werden kann. L80 druckt 3 Zahlen aus: Die Startadresse, die Adresse des nächsten freien Speicherraumes und die Anzahl der benutzten 256-Byte-Seiten.
- Die wahlweise mögliche Form
- /E:name bestimmt mit name die Startadresse des Programms. name muß deshalb ein Globalsymbol sein.
- am Ende der Kommandokette werden die Schalter auf die gesamte Kommandokette wirksam

Weitere Schalter oder Schalterspezifikationen:

- /R Rücksetzen. Der Linker wird auf seinen Anfangszustand gesetzt. Es kann eine neue Kommandozeile folgen.
- /O legt alle Adreßeingaben auf eine oktale Basis.
- /H legt alle Adreßeingaben auf eine hexadezimale Basis (Standard).
- /U Zeigt den Anfangs- und Endpunkt eines Programms sowie des Datenbereichs dieses Programms an,

wenn ein Schalter /D vorhanden war. Ist /D nicht vorhanden, wird der Datenbereich innerhalb des Programms abgespeichert.

/X Für den Fall, daß ein Dateiname mit /N angegeben wurde, wird das gelinkte Programm im Intel-Hex-Code mit der Namenserweiterung .HEX auf Diskette geladen.

/Y Für den Fall, daß ein Dateiname mit /N angegeben wurde, wird eine Symboldatei für den symbolischen Debugger SDT80 erzeugt.

Beispiel für einen normalen Aufruf des Linkers:

L80 gdcbios, gdcbios/n/e (CR)

Es wird die vom Assembler M80 erzeugte Datei gdcbios.rel als Eingabedatei auf dem aktuellen Laufwerk erwartet, eine Datei mit dem Extend .COM erzeugt und auf Diskette gespeichert.

Mit dem Programm C80 kann man eine Liste erzeugen, die alle Marken des Programms mit den Zeilennummern enthält. Das ist für die Fehlersuche sehr nützlich. Es wird eine vom Assembler erzeugte Datei mit dem Extend .CRF erwartet.

Beispiel eines Aufrufs:

C80 gdcbios:gdcbios (CR)

4. Praxis-Tips

4.1. Erstellen einer Systemdiskette

Beim Anlauf des Computers muß das Betriebssystem in den Speicher gelangen. Wie bei den meisten Geräten wird es auch im vorliegenden BIOS von Diskette geladen. Dazu sind die 20 kByte in den Systemspuren reserviert.

Wenn ein CP/M-kompatibler Computer zur Verfügung steht, kann damit eine Systemdiskette relativ einfach hergestellt werden. Es ist jedoch nicht ohne weiteres möglich, die Programmmodulen BIOS, CCP und BDOS in die Systemspuren zu schreiben, dafür gibt es kein Kommando.

Im Betriebssystem ist zwar das Programm SYSGEN dafür vorhanden, jedoch muß das an die Hardware angepaßt werden und läuft daher nicht unverändert auf jedem Computer. Eine einfache Möglichkeit bietet das Programm POWER mit den Befehlen READ und WRITE, wenn eine Systemdiskette schon vorhanden ist und nur kopiert werden soll:

- POWER laden, Systemdiskette einlegen;
- READ 0 1 4000 160 (CR);
- neue, formatierte Diskette einlegen;
- WRITE 0 1 4000 160 (CR).

In der 2. Zeile werden 160 Sektoren ab Spur 0 Sektor 1 auf die Speicheradresse 4000 geladen. Das ist der ganze reservierte Platz in den Systemspuren, mehr als eigentlich gebraucht wird. Mit der letzten Zeile gelangt der Speicherinhalt auf der neuen Diskette an die gleiche Stelle.

Liegen die Teillprogramme nur auf Diskette als COM-File vor (BIOS.COM und z.B. CPM56.COM), so ist die Abspeicherung etwas umständlicher:

- POWER laden, Diskette mit den Programmen einlegen;
- LOAD BIOS.COM 4000 (CR), angezeigt wird Sektorzahl,
z. B. 37;

- LOAD CPM56.COM 6000 (CR), angezeigt wird Sektorzahl, CCP und BDOS stehen ab Adresse 6880 (!!);
- neue, formatierte Diskette einlegen;
- WRITE 0 1 4000 37 (CR);
- WRITE 2 1 6800 48 (CR).

In der letzten Zeile wird ein Sektor mehr als nötig geschrieben, weil der erste Sektor im Original einen Lader enthält. Nach diesen Kommandos enthält die neue Diskette die Programm-Module des Betriebssystems in der Form, wie das vorgestellte BIOS sie benötigt. Die getrennte Form für die Systembestandteile wurde gewählt, um bei Änderungen nur eines bearbeiten zu müssen.

Man kann eine Systemdiskette auch ohne CP/M-Computer einrichten. Mit einem Computer nach [7] kann das BIOS von Kassette in den Speicher geladen werden. Mit dem Debugger wird im Kaltstart nur der Lese- in einen Schreibbefehl geändert und dann mit Go-Kommando und Haltepunkt auf dem Warmstart CCP und BDOS auf die Diskette geschrieben! In ähnlicher Weise kann man den Lader benutzen, um das BIOS zu schreiben.

4.2. Anpassung von Programmen (Installation)

Wie schon im Abschnitt 2. kurz angedeutet wurde, lassen sich die kommerziellen Programmpakete an die unterschiedliche Hardware der einzelnen Computer fast immer mit einem Installationsprogramm anpassen. Die Unterschiede betreffen oft nur den Drucker, manchmal aber auch die Bildschirmausgabe. In beiden Fällen sind unterschiedliche Steuerzeichen gebräuchlich, auch die Anzahl der Zeilen und Spalten kann sich stark unterscheiden. Das Installationsprogramm führt den Bediener im Dialog und fragt dabei die nötigen Daten ab. Wegen der vielen Texte ist es oft sehr umfangreich. Soll nur wenig verändert werden, ist das sogenannte Patchen einfacher (von engl. to patch = flicken). Dabei wird das Programm mit einem Debugger in den Speicher geladen, z. B. werden die beiden Bytes verändert, in denen die Zeilen- und die Spaltenzahlen stehen, und auf die Diskette zurückgeschrieben. Bei den meisten Programmen befinden sich

diese Stellen am Anfang, und man kann sie leicht finden. Natürlich stehen die Angaben dort im Hex-Code, d.h. 10H und 40H bei 16 Zeilen zu 64 Spalten und 19H 50H bei 25 Zeilen zu 80 Spalten. Das ganze soll nachstehend am Beispiel des Programms WordStar näher erläutert werden. Dieses Programm erlaubt es, eine Fülle von Kennwerten zu installieren. Hier soll nur auf den Bildschirm näher eingegangen werden. Zunächst lädt man WS.COM mit POWER auf Adresse 4100:

load ws.com 4100 (CR)

Nach der Ausführung gibt das Programm die Anzahl der geladenen Sektoren aus, die man sich für das spätere Zurückschreiben merken muß. Nun kann man sich den Patch-Bereich mit dem Kommando

dumph 4200 42ff (CR)

ansehen. Das nachfolgende Ergebnis entsteht, wenn das WS.COM für den Grafikbildschirm installiert ist:

4200:	63	6B	65	72	20	53	44	20	31	31	35	32	20	53	49	4F	
4210:	20	20	20	20	20	20	20	20	20	20	20	20	20	20	0F	00	1A
4220:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4230:	00	00	00	00	00	00	22	00	05	8E	06	54	04	8E	06	00	00
4240:	00	00	00	00	00	00	00	7E	00	19	50	01	1B	00	00	00	00
4250:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4260:	00	00	00	00	C3	48	03	06	01	00	00	00	00	01	16	00	00
4270:	00	00	00	00	00	1B	52	00	00	00	00	00	00	1B	45	00	00
4280:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4290:	00	00	01	0C	C6	80	0E	06	C3	F2	02	01	0C	00	00	00	00
42A0:	00	00	00	00	00	00	C9	00	00	00	C9	00	00	00	00	00	00
42B0:	00	00	FC	00	00	00	00	00	00	00	00	00	C9	00	00	C9	00
42C0:	00	00	C9	00	00	00	00	00	C9	00	00	00	00	00	C9	03	00
42D0:	09	19	01	09	00	00	00	00	01	01	00	00	01	14	00	00	00
42E0:	2A	01	00	11	2A	00	19	22	EB	02	CD	00	00	3C	1F	C9	00
42F0:	0E	05	5F	CD	05	00	B7	C9	16	00	21	25	03	4E	0C	0D	00

Auf den Adressen 4248 und 4249 erkennt man die Anzahl der Zeilen und Spalten dieses Bildschirms. Das Programm WS erledigt die Bildschirmansteuerung entweder über die schon erwähnten Steuerzeichen, oder es greift direkt auf den Bildwiederholspei-

cher zu, wenn das technisch möglich ist. Bei den alphanumerischen Bildschirmen des Z 1013 sowie der Bürocomputer, des PC 1715 und des Computers nach [7] ist das möglich, weil sie direkt im Arbeitsspeicher des Mikroprozessors liegen. Als Vorteil erhöht sich die Geschwindigkeit der Bildschirmzugriffe, was beim Aufbau der Mantis deutlich zu merken ist. Die Art des Zugriffs ist auf Adresse 42B0 gespeichert. Steht dort 00, wirken die eingetragenen Steuerzeichen, bei FF erfolgt der Zugriff ab der in den nachfolgenden Zellen stehenden Anfangsadresse des Bildschirms. Soll das für den Grafikbildschirm installierte WS.COM mit dem alphanumerischen 1k-Bildschirm arbeiten, so sind nur wenige Speicherplätze zu patchen (im Beispiel mit dem Programm POWER):

ds 4248 (CR)

von 19H auf 10H (CR), 4249 von 50H auf 40H, 42B0 von 00 auf FF, 42B1 auf 00, 42B2 auf FC. Danach kann das Programm mit dem SAVE-Kommando auf die Diskette zurück gespeichert werden. Die Steuerzeichen sind mit vorangestellter Länge eingetragen, z.B. auf Adresse 4292 steht das Zeichen HOME. Bei manchen Computern sind die Steuerzeichen auch mehrere Zeichen lang.

Das Patchen ist natürlich nicht als Ersatz für eine komplette Installation gedacht, aber doch eine wichtige Möglichkeit für den Amateur.

4. 3. Rechnerkopplung

Mit dem Betriebssystem CP/M ist eine leichte Austauschbarkeit von Programmen gesichert, wenn man sich bei der Programmierung an die Systemschnittstellen hält. Trotzdem kann die Kopplung von CP/M-Computern erwünscht sein, um z.B. Dateien von großen auf kleine Disketten zu übertragen, wenn die Diskettenformate nicht kompatibel sind oder wenn nur mit Kassetten als Speichermedium gearbeitet werden soll. Hierzu sind spezielle Programme zwar verfügbar, jedoch bietet auch das Programm PIP schon eine sehr einfache Möglichkeit. Die im BIOS vorhandenen seriellen Schnittstellen (20-mA-Stromschleife) sind auch bei anderen Computern weit verbreitet. Die nachfolgend beschriebene Rech-

nerkopplung verwendet die logischen Schnittstellen RDR: und PUN:. Wird die Schnittstelle für den Drucker benutzt, so muß sie für die Datentübertragung anders programmiert werden (siehe BIOS). Die Leitungen der Schnittstellen sind entsprechend ihren Polaritäten und Sender mit Empfänger zu verbinden. Dann ist mit folgendem Kommando zunächst der Empfänger "scharf" zu machen:

PIP C:NAME.EXT = RDR: (CR)

Danach löst man den Sender aus:

PIP PUN: = NAME.EXT (CR)

Die obige Folge gilt für ASCII-Dateien, an deren Ende das Dateiende kennzeichen 1AH steht. PIP prüft auf dieses Zeichen und beendet die Übertragung, wenn es erkannt wird. Danach werden die übertragenen Zeichen aus dem Speicher des Empfängercomputers auf die Diskette gespeichert und die belegten Blöcke auf der Diskette in deren Directory eingetragen. Wie oben erwähnt, können lange Dateien mehrere Eintragungen erfordern. In diesem Falle ist die einfache Routine des BIOS im Abschnitt 5. etwas unvollkommen. Sie kann die Übertragung durch den sendenden Computer nicht stoppen, so daß während des Diskettenzugriffs die Datenübertragung beim Sender weiterläuft und die Daten beim Empfänger verloren gehen. Deshalb muß man dann die RAM-Floppy benutzen, wo der Zugriff ausreichend schnell geht. Sollen Programmdateien übertragen werden, ist der Parameter [O] an die Dateibezeichnung anzuhängen (siehe Abschnitt 3.1.).

5.

Programme**5.1. Urlader**

```

1          .z80
2      0000    aseg
3          org     100h
4          TITLE   'Lader'
5
6
7
8
9          ;Lader fuer erste 5 Sektoren und Sprung dorthin
10         ;(c) Y23VO 1989
11         ;public domain
12
13         ram     equ     8100h  ;Arbeitsbereich Ladeprogramm
14         stack   equ     ram-20h
15         dskbuf  equ     ram-10h ;Speicherplatz Ladedresse
16         hstsiz  equ     400h  ;Sektorgroesse
17
18         ;Floppy-Disk-Controller 8272/765
19         cfdc    equ     40h   ;Steuerung
20         dfdc    equ     41h   ;Daten
21         tc      equ     50h   ;Ende-Impuls
22
23         ;System: Bios ab Spur 0, Sektor 1
24         ;Intellader, CCP und BIOS ab Spur 2, Sektor 1
25         ;Umkopieren
26         0100    21 010E
27         0103    11 8100
28         0106    01 0400
29         0109    ED 80
30         0108    C3 8100
31         010E    C5
32
33         8100    31 80E0
34         8103    CD 8149
35         8106    06 05
36         8108    11 0400
37         8108    21 8500
38         810E    22 80F0
39         8111    C5
40         8112    05
41         8113    E5
42         8114    CD 8195
43         8117    B7
44         8118    20 10
45         811A    21 8130
46         811D    34
47         811E    E1
48         811F    D1
49         8120    C1
50         8121    19
51         8122    22 80F0
52         8125    10 EA
53         8127    C3 8500
54
55         812A    76
56         812B
57         812B    C9
58
59         ;Steuertabelle fuer Floppycontroller
60         CTAB:  DB      0c6h  ;MFM LESEN
61         UNIT:  DB      0      ;LAUFWERK A VOREINGESTELLY
62         TRCK:  DB      0      ;SPUR
63         HED:   DB      0      ;DISK-SEITE
64         HSTSEG: DB      1      ;SECTOR
65         N:     DB      3      ;N=3=1024 BYTES
66         EOT:   DB      5      ;5 SECTOREN JE SPUR
67         GPL:   DB      1BH   ;LUECKE
68         DTL:   DB      0FFH
69         RESLT: DB      0      ;STO, RESULT TAB F.FDC
70         8136    00
71         8137    00
72         8138    00
73         8139    00
74         813A    00
75         813B    00
76         813C    00
77         813D    00
78         813E    00
79         813F    00
80         8140    00
81         8141    00

```

```

82    8142    00      HSTTRK: DB      0
83    8143    02      HRTYPE: DB      2
84    8144    00      WRFLG: DB      0
85    8145    00      inhibit: db      0
86    8146    01      SEKSEC: DB      1      ;BOOS-SEKTOR
87    8147  0080      DMAADDR: DEFW  80H      ;BOOS-DMA
88    8149      erflng equ   $      ;nicht benutzt
89    8149      msg2   equ   $      ;nicht benutzt
90
91
91      :es folgt FDC780.MAC

```

5.2. CBIOS für Z 1013

```

1      .z80
2      0000`      asseg
3      org     100h
4      :(c) Y23V0
5      :Stand: 22.9.89
6      :public domain
7
8      TITLE  'CBIOS Y23V0'
9
10     :DEFINITION VON KONSTANTEN
11     :Schalter
12    0001      GC   EQU   1      ;Grafik mituebersetzen
13    0000      V24  EQU   0      ;LST: = V24-Drucker oder IFSS
14    0000      CNTR EQU   0      ;LST: = CENTRONICS-Schnittstelle
15    0001      Z1013 EQU   1      ;oder Y23VD
16
17     :Speicheradressen
18     if      z1013
19     DAA0      BIOS  EQU   0DA00H      ;56K-cpm
20     else
21     BIOS  EQU   0DE00H      ;57K-cpm
22     endif
23    CCO0      BDOS  EQU   BIOS-0E00H
24    C400      CCP   EQU   BIOS-1600H
25    F600      gdram equ   OF600H      ;Ramatabelle Grafik-Controller
26    FC00      BRAM  EQU   OFC00H      ;BILDSCHIRM-RAM
27    0003      IOBYT EQU   3      ;Zuweisung In-/Out-Geraete
28
29     :Adressen von Peripherieschaltkreisen
30     :PIO auf CPU-Karte
31     if      z1013
32    0002      PIOBD EQU   2      ;DATENEINGABE KANAL B
33    0003      PIOBC EQU   3      ;STEUERADRESSE KANAL B
34    0008      Latch  equ   8      ;Testaturlatch (A47)
35
36     else
37     PIOAD  EQU   OFCH      ;DATENAUSGABE KANAL A
38     PIOBD  EQU   OFDH      ;EINGABEDATEM tastatur
39     PIOAC  EQU   OFEH      ;STEUERADRESSE KANAL A
40     PIOBC  EQU   OFFH      ;STEUERADRESSE KANAL B
41     endif
42
43    008E      gdcc  equ   8Ah      ;lesen      schreiben
44    008F      gdcda equ   8Fh      ;status      daten
45
46     :CTC-SIO-Karte
47    006C      ctca  equ   6ch      ;taktgeber fuer sio a
48    006D      ctcb  equ   6dh      ;taktgeber fuer sio b
49    006F      ctcd  equ   6fh      ;zeitgeber uhr
50    0068      sioad equ   68h      ;sio kanal a: v24/kassette
51    0069      siobd equ   69h      ;sio kanal b: stromschleife
52    006A      siiac  equ   6ah
53    006B      siobc  equ   6bh
54
55     :Floppy-Disk-Controller 8272/765
56    0040      CFDC  EQU   40H      ;STEUERUNG FDC
57    0041      DFDC  EQU   41H      ;DATEN FDC
58    0050      TC    EQU   50H      ;TERMINAL COUNT (ENDE-IMPULS)
59
60     :2PIO-Karte, Centronics-Schnittstelle
61    00AC      p1ad  equ   0ACh      ;Daten
62    00AD      p1bd  equ   0ADh      ;Steuerung Drucker
63    00AE      p1ac  equ   0AEh
64    00AF      p1bc  equ   0AFh
65
66     :sonstiges
67    0003      NDISK EQU   3      ;ANZAHL LAUFWERKE
68
69     :CP/M UND SYSTEMKONSTANTEN
70    0400      BLKSIZ EQU   1024      ;PUFFERLAENGE
71    0400      HSTSIZ EQU   1024      ;DISKSEKTORLAENGE
72    0005      HSTSPY EQU   5      ;DISKSEKTOREN PRO SPUR
73    0008      HSTBLK EQU   HSTSIZ/128      ;CP/M-SEKT. je DISKSEKTOR
74    0028      CPNSPT EQU   HSTBLK*HSTSPT;CP/M-SEKTOREN je SPUR
75    0007      SECHSK EQU   HSTBLK-1      ;SEKTORMASKE
76     ;BOOS-KONSTANTEN BEIM SCHREIBEINTRITT
77    0001      WRDII EQU   1      ;VERZEICHNIS SCHREIBEN

```

```

o 0002          WRUAL EQU 2          ;UNDEFINIERT SCHREIBEN
79
80
81          ;hochkopieren
82 8500 21 8508          ,phase 8500h          ;Lader-Adresse !
83 8503 11 DA00          ld    h1,cbanf          ;Programmstart
84 8506 01 1FFF          ld    de,bios           ;Ziel im Ram
85 8509 ED 80          ld    bc,01FFFh          ;Laenge 8k
86 850B          ldir
87
88          cbanf equ $             ;.
89          .dephase
90          .phase bios
91  DA00 C3 DB60          BOT:   JP    BOOT      ;Kaltstart (nach RESET)
92  DA03 C3 DBE4          WBOOT: JP    WBOOT     ;"Warmstart"
93  DA06 C3 DFA3          CONST: JP    CONST      ;Consolstatus
94  DA09 C3 DFBF          cci:   JP    CONIN     ;Zeicheneingabe (A) von Tastatur
95  DA0C C3 DF38          cco:   JP    CONOUT    ;Zeichenausgabe (C) auf Bildschirm
96  DA12 C3 DF50          JP    LIST      ;Zeichenausgabe (C) auf Druckerkanal
97  DA15 C3 DF43          JP    PUNCH     ;Zeichenausgabe (C) auf Datenkanal PUN:
98  DA18 C3 DB52          JP    READER    ;Zeicheneingabe (A) von Datenkanal RDR:
99  DA1B C3 DB18          JP    HOME      ;Diskettenspur 0 einstellen
100 DA1E C3 DB55          JP    SELDSK    ;Diskettenlaufwerk auswählen
101 DA21 C3 DB4D          JP    SETTRK    ;Spur einstellen
102 DA24 C3 DB5A          JP    SETSEC    ;Sektor einstellen
103 DA27 C3 DC06          JP    SETDMA    ;Pufferspeicher festlegen
104 DA2A C3 DC1D          JP    DREAD     ;einen Sektor von Diskette lesen
105 DA2D C3 DB46          JP    WRT       ;einen Sektor auf Diskette schreiben
106 DA30 C3 DB49          JP    LISTST    ;Druckerstatus
107
108 DA33          DPBAS EQU $             ;-----
109 DA33 0000          ;1. Laufwerk
110 DPHA: DEFW 0          DEFW 0          ;Adresse Konvertierungstabelle
111 DA35 0000          DEFW 0          ;Notitzspeicher BDOS
112 DA37 0000          DEFW 0
113 DA39 0000          DEFW 0
114 DA3B E580          DEFW DIRBF    ;Adresse Directory-Puffer
115 DA3D DA63          DEFW OPBA     ;Laufwerkbeschreibung
116 DA3F E475          DEFW CHKO     ;Adresse Prueffeld
117 DA41 E425          DEFW ALL0     ;Adresse Belegungsfeld
118 DA43 0000          OPHB: DEFW 0          ;2. Laufwerk
119 DA45 0000          DEFW 0
120 DA47 0000          DEFW 0
121 DA49 0000          DEFW 0
122 DA4B E580          DEFW DIRBF
123 DA4D DA63          DEFW OPBA
124 DA4F E515          DEFW CHK1
125 DA51 E4C5          DEFW ALL1
126 DA53 0000          DPHC: DEFW 0          ;RAMDISK
127 DA55 0000          DEFW 0
128 DA57 0000          DEFW 0
129 DA59 0000          DEFW 0
130 DA5B E580          DEFW DIRBF
131 DA5D DA81          DEFW OPBC
132 DA5F E585          DEFW CHK2    ;kein check
133 DA61 E565          DEFW ALL2
134
135          ;DISK-PARAMETER-BLOCK FLOPPYLAUFWERKE 5,25 ZOLL
136          ;doppelte Dichte (DD), doppelseitig (DS), 80 Spuren (96TPI)
137          ;780 k
138 DA63 0028          DPBA: DEFW 40          ;40LOGISCHE=5 PHYSISCHE SEKTOREN/SPUR
139 DA65 04          DEFW 08        ;1K-BLOECKE
140 DA66 0F          DB 4
141 DA67 00          DB OFH
142 DA68 0185          DEFW 0185H
143 DA6A 007F          DEFW 07FH
144 DA6C 00C0          DEFW 0C0H
145 DA6E 0020          DEFW 20H
146 DA70 0004          DEFW 4          ;SYSTEMSPUREN
147          ;800 k
148 DA72 0028          DPBB: DEFW 40
149 DA74 04          DB 4
150 DA75 0F          DB OFH
151 DA76 00          DB 0
152 DA77 018F          DEFW 018FH
153 DA79 007F          DEFW 07FH
154 DA7B 00C0          DEFW 0C0H
155 DA7D 0020          DEFW 20H
156 DA7F 0000          DEFW 0          ;SYSTEMSPUREN
157 DA81 0010          DPBC: DEFW 16
158 DA83 03          DB 3
159 DA84 07          DB 07H
160 DA85 00          DB 0
161 DA86 00FF          DEFW 0255
162 DA88 003F          DEFW 03FH
163 DA8A 00C0          DEFW 0C0H
164 DA8C 0000          DEFW 0
165 DA8E 0000          DEFW 0          ;SYSTEMSPUREN
166          ;Steuertabelle fuer Floppycontroller
167 DA90 C6          CTAB: DB 0C6H          ;MFM LESEN
168 DA91 00          UNIT: DB 0          ;LAUFWERK A VOREINGESTELLT
169 DA92 00          TRCK: DB 0          ;SPUR
170 DA93 00          HED:  DB 0          ;DISK-SEITE
171 DA94 01          HSTSEC: DB 1          ;SEKTOR
172 DA95 03          N:   DB 3          ;N=3=1024 BYTES
173 DA96 05          EOT:  DB 5          ;5 SEKTOREN JE SPUR
174 DA97 18          GPL:  DB 1BH          ;LUECKE

```

```

175 DAA9C FF      DTL:   DB      0FFH
176 DA99 00       RESLT: DB      0          ;ST0, RESULT TAB F,FDC
177 DA9A 00       DB      0          ;ST1
178 DA9B 00       DB      0          ;ST2
179 DA9C 00       DB      0          ;CYLINDER
180 DA9D 00       DB      0          ;HEAD
181 DA9E 00       DB      0          ;RECORD
182 DA9F 00       DB      0          ;N
183 DAA0 00       SEKDSK: DB      0
184 DAA1 00       SEKTRK: DB      0
185 DAA2 00       HEAD:  DB      0
186 DAA3 00       SEKHST: DB      0
187 DAA4 00       HSTDOSK: DB     0
188 DAA5 00       HSTTRK: DB      0
189 DAA6 02       WRTYPE: DB     2
190 DAA7 00       WRFLG:  DB     0
191 DAA8 00       inhibt: db     0
192 DAA9 01       SEKSEC: DB     1          ;BDOS-SEKTOR
193 DAAA 0080     DMAADR: DW    80H          ;8005-DMA
194 DAAC E60E     DSKBUF: DW    HSTBUF
195
196
197 DAAE FC00     curs:  DW      BRAH      ;Cursor=Bildschirm-ram
198 DAB0 00       pos1:   db      0          ;Cursorpositionierung
199 DAB1 80       pos2:   db      80h      ;Cursor ein/aus
200 DAB2 18       ZK:    DB      18H      ;30 F.50 BAUD
201 DAB3 01       DB      01      ;2 F. 50 BAUD
202 DAB4 00       FLGIO: DB      0          ;steuerung in/out-geraeete
203
204
205 DAB5 0A 0D 59 32 MSG1:  DB      0AH,0DH,'Y23V0 CP/Y2 V.3 780K',0dh,0ah,0
206 DAB9 33 56 4F 20
207 DABD 43 50 2F 59
208 DAC1 32 20 56 2E
209 DAC5 33 20 37 38
210 DAC9 30 4B 00 0A
211 DADC 00
212 DACE 00 0A 4C 61 MSG2:  DB      0dh,0ah,'Laufwerk nicht bereit.'
213 DAD0 75 66 77 65
214 DAD6 72 6B 20 6E
215 DADA 69 63 68 74
216 DADE 20 62 65 72
217 DAE2 65,69 74 2E
218 DAE6 0A 00 61 62 MSG5:  DB      0AH,0DH,'abbrechen j/n ? ',0dh,0ah,0
219 DAEA 62 72 65 63
220 DAEF 68 65 6E 20
221 DAF2 20 6A 2F 6E
222 DAF6 20 3F 00 0A
223 DAF9 00
224 DAFB 00 0A 46 65 MSG3:  DB      0dh,0ah,'Fehler nicht zu beheben.',0dh,0ah,0
225 DAFF 68 6C 65 72
226 DB03 20 6E 69 63
227 DB07 68 74 20 7A
228 DB08 75 20 62 65
229 DB0F 68 65 62 65
230 DB13 6E 2E 0D 0A
231 DB17 00
232
233
234
235
236 DB18 21 0000 :LAUFWERK AUSWAEHLEN
237 DB18 79 :IN:C LAUFWERK (A ODER B)
238 DB1C FE 03 :OUT:HL DPH-VEKTOR
239 DB1E 00 SELDSK: LD      HL,0
240 DB1F 32 0AA0 LD      A,C
241 DB22 6F CP      ndisk
242 DB23 29 ret    NC
243 DB24 29 LD      (SEKDSK),A
244 DB25 29 ADD    HL,HL
245 DB26 29 ADD    HL,HL
246 DB27 11 0A33 ADD    HL,HL
247 DB2A 19 ADD    HL,DE      ;'16
248 DB2B E6 03 LD      DE,DPBAS      ;DPH-VEKTOR 1.LW
249 DB2D C8 and    3          ;Test lw b
250 DB2E FE 02 ret    z          ;nein
251 DB30 C8 cp      2          ;Test lw c
252 DB31 E5 ret    z          ;nein
253 DB32 21 0AA8 push   hl
254 DB35 CB 46 ld      hl,inhibit ;test ob 1.mal
255 DB37 20 0B bit    0,(hl)
256 DB39 3E FF jr      nz,sdsk1 ;nein, kein recal noetig
257 DB3B 77 ld      a,0ffh      ;markieren
258 DB3C 3E 01 ld      (hl).a
259 DB3E 32 DA91 ld      a,1
260 DB41 CD DDA9 call   recal
261 DB44 E1 sdsk1: pop   hl
262 DB45 C9 RET
263 DB46 3E FF LISTST: ld      a,0ffh      ;ohne Kontrolle!
264 DB48 C9 RET          ;LISTER STATUS
265
266
267
268 DB49 60 :SEKTOR TRANSFORMATION
269 DB4A 69 :IN: BC LOG.SEKTOR
270 DB4B 23 :DE = 0 = keine Transformation
271 DB4C C9 SECTRAN:LD    H,B
                    LD      L,C
                    INC   HL
                    ret

```

```

272 ;SEKTOR EINSTELLEN
273 ;IN: C SEKTOR
274 ;OUT: (SEKSEC) =C
275 DB4D 21 DAA9 SETSEC: LD      HL,SEKSEC
276 DB50 71          LD      (HL),C
277 DB51 C9          RET
278 ;KOPF AUF SPUR NULL STELLEN
279 DB52 01 0000 HOME: LD      BC,0
280 ;SPUR ANWAEHLEN
281 ;IN: C SPUR
282 ;OUT (SEKTRK)=C
283 DB55 79 SETTRK: LD      A,C
284 DB56 32 DAA1     LD      (SEKTRK),A
285 DB59 C9          RET
286 ;DMA SETZEN
287 ;IN: BC DMA
288 DB5A 60 SETDMA: LD      H,B
289 DB5B 69          LD      L,C
290 DB5C 22 DAAA     LD      (DMAADR),HL
291 DB5F C9          RET
292
293 ;-----[Kaltstart, nur nach RESET
294 ;CCP UND BOOS VON SPUR 2 DER DISKETTE LADEN
295 DB60 3E 71 BOOT: Id      a,71h :cpuram aus
296 DB62 03 F7        out    (017h),a
297 DB64 03 31        out    (31h),a :dram ein
298 DB66 3E C3        LD      A,OC3H
299 DB68 32 0000      LD      (0),A
300 DB6B 21 0A03      LD      HL,WBOTE
301 DB6E 22 0001      LD      (1),HL
302 DB71 32 0005      LD      (5),A
303 DB74 21 CC06      LD      HL,BOOS+6
304 DB77 22 0006      LD      (6),HL
305 DB7A 31 0080      LD      SP,B0H
306 DB7D CD DEE5      CALL   INIT
307 DB80 CD DD78      call   dskini
308 DB83 21 DAB5      LD      HL,MSG1
309 DB86 CD DF99      CALL   STR
310 DB89 AF          XOR    A
311 DB8A 32 E41C      LD      (HSTACT),A
312 DB8D 32 E41E      LD      (UNACNT),A
313 DB90 32 0004      LD      (4),A
314 DB93 CD D08F      CALL   INIFD :INITIALISIERUNG FLOPPY-CONTROLLER
315 DB96 00 00          LD      C,0 :LAUFWERK A VOREINGESTELLT
316 DB98 CD DB18      CALL   SELDSK
317 DB9B 06 2C          LD      B,2CH
318 DB9D 0E 02          LD      C,2 :SEKTORZAehler
319 DB9F CD DB55      CALL   SETTRK :SPUR-NR.
320 DBA2 16 02          LD      D,2 :SEKTOR
321 DBA4 21 C400      LD      HL,CCP :LADEZIEL
322 DBA7 C5          SLOAD1: PUSH  BC :NAECHSTEN SEKTOR
323 DBA8 D5          PUSH  DE
324 DBA9 E5          PUSH  HL
325 DBAA 4A          LD      C,D
326 DBAB CD DB4D      CALL   SETSEC :NEUEN SEKT_ANWAEHLEN
327 DBAE C1          POP   BC :BC=LADEADRESSE
328 DBAF C5          PUSH  BC :LADEADR RETTEN
329 DBB0 CD DB5A      CALL   SETDMA
330 DBB3 CD OC06      CALL   DREAD :SYSTEM RAUF SCHREIBEN: WFLOP EINSETZEN
331 DBB6 FE 00          CP      0 :FEHLER?
332 DBB8 C4 DF8F      CALL   NZ,RESTR :JA
333 DBB8 E1          POP   HL
334 DBBC 11 0080      LD      DE,128
335 DBBF 19          ADD   HL,DE :LADEADR.ERHOHEN
336 DBC0 01          POP   DE
337 DBC1 C1          POP   BC
338 DBC2 05          DEC   B :SEKTORZAHL - 1
339 DBC3 28 14          JR      Z,BOOT1 :ccp-kopie anlegen
340 DBC5 14          INC   D :NAECHSTEN SEKT.
341 DBC6 7A          LD      A,D
342 DBC7 FE 29          CP      29H :LETZTER SEKTOR PER TRACK
343 DBC9 38 DC          JR      C,SLOAD1 :1.SEKT.NAECHST.SPUR
344 DBCB 16 01          LD      D,1
345 DBCD 0C          INC   C
346 DBCE C5          PUSH  BC
347 DBCF D5          PUSH  DE
348 DBD0 E5          PUSH  HL
349 DBD1 CD DB55      CALL   SETTRK
350 DBD4 E1          POP   HL
351 DBD5 D1          POP   DE
352 DBD6 C1          POP   BC
353 DBD7 18 CE          JR      SLOAD1
354
355 DBD9 21 C400      ;-----[BOOT1: Id      hl,ccp :ccp-kopie anlegen
356 DBDC 11 EA0F      Id      de,kopie
357 DBDF 01 0880      Id      bc,128*17
358 DBE2 ED 80          ldir
359
360 DBE4 21 EA0F      XOR    A
361 DBE7 11 C400      LD      (HSTACT),A :Warmstart:ccp kopieren
362 DBEA 01 0880      LD      (UNACNT),A
363 DBED ED 80          ld      sp,B0H
364 DBEF AF          LD      BC,B0H
365 DBF0 32 E41C      LD      (SP,B0H)
366 DBF3 32 E41E      LD      BC,B0H
367 DBF6 31 0080      LD      SP,B0H
368 DBF9 01 0080      LD      BC,B0H

```

```

369 DBFC CD DB5A CALL SETDMA
370 DBFF 3A 0004 LD A,(4) ;AKTUELLES LAUFWERK HOLEN
371 DC02 4F LD C,A
372 DC03 C3 C400 JP CCP
373
374 :BLOCKBILDUNG FUER SEKTOREN FUER CP/M 2.0
375 DC06 3A DAA0 DREAD: ld a,(sekdisk) ;Test auf RAM-Disk
376 DC09 FE 02 cp 2
377 DC08 CA 0041 jp z,lese ;ram-disk
378 DC0E 3E 01 LD A,1 ;Sektor von Diskette lesen
379 DC10 32 E424 LD (READOP),A ;LESEFUNKTION
380 DC13 32 E423 LD (RSFLAG),A ;DATEN LESEN
381 DC16 3E 02 LD A,WRUAL
382 DC18 32 DAA6 LD (WRTYPE),A ;ALS NICHT DEF. BEHANDELN
383 DC1B 1A 60 JR RWOPER
384 DC1D 3A DAA0 WRIT: ld a,(sekdisk) ;Test auf RAM-Disk
385 DC20 FE 02 cp 2
386 DC22 CA D048 jp z,schr ;ja
387 DC25 AF XOR A
388 DC26 32 E424 LD (READOP),A ;SCHREIBEN DES ANGEWAELHTEN CP/M-SEKTORS
389 DC29 79 LD A,C ;SCHREIBTYP IN A
390 DC2A 32 DAA6 LD (WRTYPE),A
391 DC2D FE 02 CP WRUAL ;ORT DEFINIERT ?
392 DC2F 20 17 JR NZ,CHKUNA ;ORT TESTEN
393 :UNDEFINIERT SCHREIBEN
394 DC31 3E 08 LD A,BLKSI2/128 ;NAECHSTE UNBELEGTE SAETZE
395 DC33 32 E41E LD (UNACNT),A
396 DC36 3A DAA0 LD A,(SEKOSK)
397 DC39 32 E41F LD (UNADSK),A ;ZU SUCHENDES LAUFWERK
398 DC3C 3A DAA1 LD A,(SEKTRK) ;ZU SUCHENDE SPUR
399 DC3F 32 E420 LD (UNATRK),A
400 DC42 3A DAA9 LD A,(SEKSEC) ;ZU SUCHENDER SEKTOR
401 DC45 32 E421 LD (UNASEC),A
402 DC48 3A E41E CHKUNA: LD A,(UNACNT) ;TEST AUF SCHREIBEN IN EINEN UNBELEGTEN SEKTOR
403 DC4B B7 OR A
404 DC4C 28 34 JR Z,ALLOC ;SPRUNG WENN NICHT VORHANDEN
405 :ES SIND UNBELEGTE DATENAESATZE VORHANDEN
406 DC4E 3D DEC A ;HERUNTERZAEHLEN
407 DC4F 32 E41E LD (UNACNT),A
408 DC52 3A DAA0 LD A,(SEKOSK) ;GLEICHES LAUFWERK ?
409 DC55 21 E41F LD HL,UNADSK
410 DC58 BE CP (HL)
411 DC59 20 27 JR NZ,CHKUNA ;WENN NEIN, SPRUNG
412 :DIE LAUFWERKE SIND DIE GLEICHEN
413 DC5C 21 E420 LD HL,UNATRK
414 DC5E 3A DAA1 LD A,(SEKTRK)
415 DC61 BE CP (HL)
416 DC62 20 1E JR NZ,ALLOC ;WENN NEIN, SPRUNG
417 :DIE SPUREN SIND DIE GLEICHEN
418 DC64 3A DAA9 LD A,(SEKSEC) ;GLEICHER SEKTOR
419 DC67 21 E421 LD HL,UNASEC
420 DC6A BE CP (HL) ;SEKSEC=UNASEC ?
421 DC6B 20 15 JR NZ,ALLOC ;WENN NICHT, SPRUNG
422 :UEBEREINSTIMMUNG, UEBERGANG ZUM NAECHSTEN SEKTOR
423 DC6D 34 INC (HL) ;UNASEC=UNASEC+1
424 DC6E 7E LD A,(HL) ;SPURENDE ?
425 DC6F FE 28 CP CPMSPT ;CP/M-SEKTOREN/SPUR
426 DC71 38 09 JR C,NOOVS ;SPRUNG, WENN KEIN UEBERLAUF
427 :UEBERLAUF. NAECHSTE SPUR
428 DC73 36 00 LD (HL),0 ;UNASEC = 0
429 DC75 3A E420 LD A,(UNATRK)
430 DC78 3C INC A
431 DC79 32 E420 LD (UNATRK),A ;UNATRK=UNATRK+1
432 DC7C AF NOOVS: XOR A ;UEBEREINSTIMMUNG GEFUNDEN.
433 DC7D 32 E423 LD (RSFLAG),A ;UNBEDINGTE LESEOPERATION
434 DC80 18 08 JR RWOPER ;SCHREIBEN DURCHFUEHREN
435 DC82 AF ALLOC: XOR A ;KEIN UNBELEGTER DATENSATZ.
436 DC83 32 E41E LD (UNACNT),A ;VORHERIGES LESEN NOTWENDIG
437 DC86 3C INC A
438 DC87 32 E423 LD (RSFLAG),A ;RSFLAG = 1
439 :GEMEINSAMER KODE FUER SCHREIBEN UND LESEN
440 DC8A AF RWOPER: XOR A
441 DC8B 32 E422 LD (ERFLAG),A ;FEHLER RUECKSETZEN
442 DC8E 3A DAA9 LD A,(SEKSEC) ;DISKSEKTOR BERECHNEN
443 DC91 06 01 LD B,1
444 DC93 06 08 SC1: SUB B
445 DC95 28 06 JR Z,SC2
446 DC97 FA DC9D LD M,SC2
447 DC9A 04 INC B
448 DC9B 18 F6 JR SC1
449 DC9D 78 SC2: LD A,B
450 DC9E 32 DAA3 LD (SEKHST),A ;ZU SUCHENDER DISKSEKTOR
451 :DISKSEKTOR AKTIV ?
452 DCA1 21 E41C LD HL,HSTACT
453 DCA4 7E LD A,(HL)
454 DCAS 36 01 LD (HL),1
455 DCAT 87 OR A ;BEREITS GESETZT ?
456 DCAB 28 24 JR Z,FILHST ;SPEICHER FUENLEN
457 :AKTIVER PUFFER GLEICH GESUCHTEM ?
458 DCAA 3A DAA0 LD A,(SEKDISK)
459 DCAD 21 DAA4 LD HL,HSTDOSK ;GLEICHES LAUFWERKE
460 DCB0 BE CP (HL)
461 DCB1 20 12 JR NZ,NOMATC
462 :GLEICHES SPUR ?
463 DCB3 3A DAA1 LD A,(SEKTRK)
464 DCB6 21 DAAS CP (HL)
465 DCB9 BE

```

```

466 DBCA 20 09      DR   NZ,NOMATC
467
468 DC8C 3A DAA3    ;GLEICHER PUFFER
469 DC8F 21 DA94    LD   A,(SEKHST)
470 DCC2 BE          LD   HL,HSTSEC  ;SECHST=HSTSEC ?
471 DCC3 28 26      CP   (HL)
472 DCC5 3A E41D    JR   Z,MATCH  ;WENN JA, SPRUNG
473                   NOMATC: LO   A,(HSTWRIT) ;DISKSEKTOR GESENDRIBEN ?
474                   ;RICHTIGES LAUFWERK, ABER NICHT GLEICHER SEKTOR
475 DCC8 B7          OR   A
476 DCC9 28 03      JR   Z,FILMST
477 DCCE CD DDCD    CALL  WRHST  ;DISKPUFFER LEEREN
478 DCCE 3A DAO      FILHST: LD   A,(SEKOSK)
479 DCD1 32 DAA4    ;DISKPUFFER FUENNELN
480 DC04 3A DAA1    LD   (HSTDISK),A
481 DC07 32 DAA5    LD   A,(SEKTRK)
482 DC0A 3A DAA3    LD   (HSTTRK),A
483 DC0D 32 DA94    LD   A,(SEKHST)
484 DCE0 3A E423    LD   (HSTSEC),A
485 DCE3 B7          LD   A,(RSFLAG) ;LESEN NOTWENDIG
486 DCE4 C4 DDD8    OR   A
487 DCE7 AF          CALL NZ,REAHST
488 DCE8 32 E41D    XOR  A
489 DCEB 3A DAA9    LD   (HSTWRIT),A ;KEIN SCHREIBEN
490 DCEE E6 07      MATCH: LO   A,(SEKSEC)
491 DCF0 FE 00      AND  SECMSK
492 DCF2 20 02      CP   0
493 DCF4 3E 08      JR   NZ,MATCH2
494 DCF6 21 E58E    LD   A,B
495 DCF9 47          MATCH2: LD   HL,HSTBUF-128
496 DCFA 11 0080    ld   b,e
497 DCFD 19          ld   de,80h
498 DCFE 10 FD      match1: add hl,de
499 DD00 EB          djnz match1
500 DD01 2A DAAA    ex   de,hl
501 DD04 0E 80      ld   hl,(dmaadr)
502 DD06 3A E424    ld   c,128
503 DD09 B7          ld   a,(readop)
504 DDOA 20 06      or   a
505 DDOE 3E 01      jr   nz,rwmove
506 DDOE 32 E41D    ld   a,1
507 DD11 EB          ld   (hstwrft),a
508 DD12 1A          EX   DE,HL ;QUELLE/SENKE TAUSCHEN
509 DD13 13          RWMOVE: LD   A,(DE) ;QUELLZEICHEN
510 DD14 77          INC  DE
511 DD15 23          LD   (HL),A ;ZUR SENKE
512 DD16 0D          INC  HL
513 DD17 20 F9      DEC  C   :128*
514                   JR   NZ,RWMOVE
515 DD19 3A DAA0    ;DATENTRANSFER BEENDET
516 DD1C 32 E41F    LD   A,(SEKOSK)
517 DD1F 3A DAA1    LD   (UNADSK),A
518 DD22 32 E420    LD   A,(SEKTRK)
519 DD25 3A DAA9    LD   (UNATRK),A
520 DD28 32 E421    LD   A,(SEKSEC)
521 DD28 3A DAA6    LD   A,(WRTYPE) ;SCHREIBTYPE HOLLEN
522 DD2E FE 01      CP   WROIR ;IHV-EINTRAG ?
523 DD30 3A E422    LD   A,(ERFLAG) ;FEHLERFLAG HOLLEN
524 DD33 C0          RET  NZ   ;KEIN IHV, ZURUECK
525                   :FREMPUFFER LEEREN(Verzeichnis-EINTRAG)
526 DD34 B7          OR   A   ;FEHLER ?
527 DD35 C0          RET  NZ   ;ZURUCK FALLS JA
528 DD36 AF          XOR  A
529 DD37 32 E41D    LD   (HSTWRIT),A ;PUFFER GESENDRIBEN
530 DD3A CD DDCD    CALL  WRHST
531 DD3D 3A E422    LD   A,(ERFLAG)
532 DD40 C9          RET
533
534
535
536
537 0098          C   include MPRDSK.MAC
538
539 DD41 CD DD4F    C   ;ramdisk nach "Mikroprozessortechnik" 3/88
540 DD44 ED B2      C   .z80
541 DD46 AF          C   gadr equ 98h ;Basisadresse der Karte
542 DD47 C9          C   lese: call adre
543 DD48 CD DD4F    C   inir:
544 DD4B ED B3      C   xor  a
545 DD4D AF          C   ret
546 DD4E C9          C   schr: call adre
547 DD4F 3A DAA1    C   otir:
548 DD52 6F          C   xor  a
549 DD53 26 00      C   ret
550 DD55 29          C   adre: id   a,(sektrk)
551 DD56 29          C   id   l,a
552 DD57 29          C   id   h,0
553 DD58 29          C   add  h1,hl ;Umwandlung Spuren in Sektore
554 DD59 3A DAA9    C   add  h1,hl
555 DD5C 5F          C   add  h1,hl
556 DD5D 16 00      C   add  h1,hl
557 DD5F 19          C   id   a,(seksec) ;aktuelle Sektor dazu
558 DD60 AF          C   id   a
559 DD61 C8 1C      C   add  d,0
560 DD63 C8 1D      C   add  h1,de
561 DD65 C8 1F      C   xor  a
562 DD67 D3 9F      C   rr   h
                           rr   l
                           rr   a
                           out  (gadr+7),a :low-adr

```

```

563 DD69 7D C ld a,1
564 D06A D3 9E C out (gadr+6),a ;mittlere adr
565 D06C 7C C ld a,h ;Bank
566 D06D E6 03 C and 3
567 D06F F6 98 C or gadr
568 D071 4F C ld c,a
569 D072 06 80 C ld b,128
570 D074 2A DAAA C ld h1,(dmaadr)
571 D077 C9 C ret
572 ;Rendisk loeschen (mit E5)
573 D078 AF C dskini: xor a
574 D079 4F C ld c,a
575 D07A D3 9E C out (gadr+6),a
576 D07C AF C dskin1: xor a
577 D07D 47 C ld b,a
578 D07E D3 9F C out (gadr+7),a
579 D080 3E E5 C ld a,0e5h
580 D082 D3 98 C dskin2: out (gadr),a
581 D084 10 FC C djnz dskin2
582 D086 0C C inc c
583 D087 79 C ld a,c
584 D088 D3 9E C out (gadr+6),a
585 D08A FE 09 C cp 9
586 D08C 38 EE C jr c,dskin1
587 D08E C9 C ret
588 include FOC780.MAC
589 ;-----
590 ;TREIBER MIT U8272
591 ;-----
592 ;(c) Y23VO 1989
593 ;public domain
594 ;-----
595 D08F CD D0CS C INITD: CALL TCPLS
596 D092 06 00 C INITI: LD B,0 ;INITIALISIERUNG P8272
597 D094 10 FE C DJNZ INITI+2
598 D096 DB 40 C IN A,(DFOC)
599 D098 FE 80 C CP 80H
600 D09A 28 04 C JR Z,SPEZI
601 D09C DB 41 C IN A,(DFOC)
602 D09E 18 F2 C JR INITI
603 D0A0 21 DDC2 C SPEZI: LD HL,STAB-1 ;PARAMETER LADEN
604 D0A3 01 0303 C LD BC,0303H ;SPECIFY-COMM 3BYTES
605 D0A6 CD DE6B C CALL WCOM1 ;SCHREIBEN COMM
606 D0A9 01 0207 C RECAL: LD BC,0207H ;SPUR O EINSTELLEN
607 D0AC CD DE42 C CALL RDY ;DRIVE READY?
608 ;-----
609 D0AF 01 0108 C SENSE: LD BC,0108H ;PRUEFE INTERRUPT STATUS
610 D0B2 CD DE68 C CALL WCOM
611 D0B5 CD DE87 C CALL RBYTE ;RESULT REG 0:IC1,IC2,SE,EC,NR,HD,US1,US2
612 D0B8 47 C LD B,A
613 D0B9 FE 80 C CP 80H
614 D0BB C4 DE87 C CALL NZ,RBYTE ;PCN HOLLEN
615 D0BE CB 68 C BIT 5,B ;SEEK ENDE?
616 D0C0 28 ED C JR Z,SENSE
617 D0C2 C9 C RET
618 D0C3 DF C STAB: DB 0dfH ;XXXX=SRT,XXXX=HUT
619 D0C4 33 C DB 33H ;XXXXXXXX=HLT,X=ND
620 ;df,33 mit kopfladen,e1,3 ohne
621 ;-----
622 D0C5 D3 50 C TCPLS: OUT (TC),A ;ENDEIMPULS EIN
623 D0C7 CD DE80 C CALL DELAY
624 D0CA DB 50 C IN A,(TC)
625 D0CC C9 C RET
626 ;-----
627 ;SEKTOR SCHREIBEN/LESEN IM BETRIEBSART OHNE DMA-SCHALTKREIS
628 ;SECTOR IN (HSTSEC)
629 ;SPUR IN (HSTTRK)
630 ;AKT.LAUFWERK IN (HSTDISK)
631 ;ZIEL-/QUELLADRESSE = HSTBUF
632 ;KILL: A B DE HL
633 D0CD CD DE28 C WRHST: call sda
634 D0D0 CB 77 C bit 6,a ;Schreibschutz ?
635 D0D2 20 39 C jr nz,rwerr
636 D0D4 11 C wrhst1: DB 11H ;CODE LD DE...
637 D0D5 ED A3 C OUTI ;WRITE DATA
638 D0D7 3E C5 C LD A,0e5H ;SCHREIBKOMMANDO
639 D0D9 18 05 C JR RWIT
640 ;-----
641 D0DB 11 C REAHST: DB 11H ;CODE LD DE...
642 D0DC ED A2 C INI
643 D0DE 3E C6 C LD A,0e5H ;LESEKOMMANDO
644 D0E0 32 DA90 C RWIT: LD (CTAB).A
645 D0E3 ED 53 DE20 C LD (MODE).DE ;EINTRAGENINI/OUTI-OP-CODE
646 D0E7 CD DEBC C call heads
647 D0EA CD DE32 C CALL SEEK
648 D0ED 06 0A C ld b,10
649 D0EF C5 C RWOP: PUSH BC
650 D0F0 F3 C DI
651 D0F1 06 09 C LD B,9 ;9COMMANDOBYTES
652 D0F3 3A DA90 C LD A,(CTAB)
653 D0F6 4F C LD C,A
654 D0F7 CD DE42 C CALL RDY ;AUSGABEKOMMANDO
655 D0FA 2A DAAC C LD HL,(DSKBUF) ;Ladeadresse
656 D0FD 0E 41 C LD C,DFDC
657 D0FF CD DE13 C CALL RW
658 D0E2 CD D0C5 C CALL TCPLS ;ENDE-IMPULS
659 D0E5 CD DE90 C CALL RRSLT

```

```

660 DE08 FB C EI
661 DE09 C1 C POP BC
662 DE0A C8 C RET Z ;STO=0=KEINE FEHLER
663 DE0B 10 E2 C DJNZ RWOP
664 DE0D 3E 01 C rweerr: LD A,1
665 DE0F 32 E422 C 1d {erflag},a
666 DE12 C9 C RET ;ERROR
667 C
668 C
669 C
670 C
671 DE13 06 04 C :SCHREIBEN ODER LESEN 1024 BYTES
672 DE15 C5 C ;IN: HL QUELLE ODER ZIEL
673 DE16 06 00 C ;C: DATENPORT
674 DE18 DB 40 C RW: LD B,4
675 DE1A 07 C RW2: PUSH BC
676 DE1B 30 FB C LO B,0 ;256 BYTES
677 DE1D 07 C RW1: IN A,(CFDC)
678 DE1E 07 C RLCA ;RQM-TEST
679 DE1F D0 C JR NC,RW1
680 DE20 ED A2 C RLCA ;DIO
681 DE22 20 F4 C RLCA ;NON-DMA
682 DE24 C1 C RET NC ;FERTIG?
683 DE25 10 EE C MODE: INI ;INI BEI READ DATA SONST OUTI
684 DE27 C9 C JR NZ,RW1
685 C
686 DE28 01 0204 C SDS: LD BC,0204H ;PRUEFE LAUFWERK STATUS
687 DE2B CD 068 C CALL WC0M
688 DE2E CD DE07 C CALL RBYTE ;STATUS REG 3:FAULT,WP,RDY,TO,TS,HD,US1,U
689 DE31 C9 C RET
690 C
691 DE32 01 030F C SEEK: LD BC,030FH ;COMM SPUR EINSTELLEN
692 DE35 CD DE42 C CALL RDY
693 DE38 CD 0DAF C CALL SENSE
694 DE3B DB 40 C SKBSY: IN A,(CFDC)
695 DE3D E6 0F C AND OFH
696 DE3F 20 FA C JR NZ,SKBSY
697 DE41 C9 C RET
698 C
699 DE42 C5 C :-----ROY: PUSH BC ;LAUFWERK BETRIEBSFAEHIG ?
700 DE43 21 7FFF C Id h1,ffffh ;Wartezeit auf Bereitschaft
701 DE46 E5 C nrdy: push h1
702 DE47 C0 DE28 C CALL SDS ;nicht
703 DE4A E1 C pop h1
704 DE4B C8 6F C BIT 5,A ;READY-BIT IN STATUSREG.3
705 DE4D 20 18 C JR NZ,RDY1
706 DE4F 28 C dec h1
707 DE50 CD DE80 C call delay
708 DE53 7D C Id e,1
709 DE54 B4 C or h
710 DE55 20 EF C Jr nz,nrdy
711 DE57 21 DACE C LD HL,MSG2 ;FEHLERHELDUNG
712 DE5A CD DF99 C CALL STR
713 DE5D CD DA09 C CALL CCI
714 DE60 FE 6A C CP j
715 DE62 20 E2 C JR NZ,NRDY
716 DE64 C3 DF8F C JP restr ;Abbruch
717 DE67 C1 C RDY1: POP BC
718 C
719 DE68 21 DA90 C :-----WCOM: LD HL,CTAB ;COMM IN FDC SCHREIBEN
720 DE6B F3 C WCOM1: DI ;IN: B ANZAHL D. BYTES,C COMM
721 DE6C CD DE80 C CALL DELAY
722 DE6F D0 40 C IN A,(CFDC)
723 DE71 E6 C0 C AND OC0H
724 DE73 FE 80 C CP 80H ;RQM,DIO=OUT
725 DE75 20 F4 C JR NZ,WCOM1
726 DE77 79 C LD A,C
727 DE78 D3 41 C OUT (DFDC),A
728 DE7A 23 C INC HL
729 DE7B 4E C LD C,M
730 DE7C 10 E0 C DJNZ WCOM1
731 DE7E FB C EI
732 DE7F C9 C RET
733 C
734 DE80 C5 C :-----DELAY: PUSH BC ;VERZOEGERUNG F.STATUSFLAG 8272
735 DE81 06 0F C LD B,0FH
736 DE83 10 FE C DEL1: DJNZ DEL1
737 DE85 C1 C POP BC
738 DE86 C9 C RET
739 C
740 DE87 CD DE80 C :-----RBYTE: CALL DELAY ;1 BYTE LESEN
741 DE8A CD DEA6 C CALL IRDY
742 DE8D DB 41 C IN A,(DFDC)
743 DE8F C9 C RET
744 C
745 DE90 06 06 C :-----RRSLT: LD B,6 ;ILESE 7 RESULT BYTES
746 DE92 CD DE87 C CALL RBYTE
747 DE95 21 DA99 C LD HL,RESLT
748 DE98 77 C LD H,A
749 DE99 E6 C0 C AND OC0H ;ERROR?
750 DE9B 4F C LD C,A
751 DE9C CD DE87 C RESL1: CALL RBYTE
752 DE9F 23 C INC HL
753 DEAO 77 C LD H,A
754 DEA1 10 F9 C DJNZ RESL1
755 DEA3 79 C LD A,C ;FEHLERHELDUNG STATUS REG 0
756 DEA4 B7 C OR A

```

```

757 DEA5 C9 C RET
758 DEA6 DB 40 C IRDY: IN A,(CFDC) ;BEREIT F.DATENEINGABE ?
759 DEA8 07 C RLCA
760 DEA9 30 FB C JR NC,IRDY
761 DEAB E6 80 C AND 80H
762 DEAD 07 C RLCA
763 DEAE D8 C RET C
764
765
766 DEAF C3 DFBF C :FEHLERBEHANDLUNG
767 ERM: jp restr ;SPRUNG IN MONITOR
768
769 :KOMMANDO READ ID
770 DEB2 01 024A C RDID: LD BC,024AH
771 DEB5 CD DE42 C CALL RDY
772 DEB8 CD DE90 C CALL RRSLT
773 DEBB C9 C RET
774
775 DEBC B7 C HEADS: or a ;Carry-Flag loeschen
776 DEBD 3A DAA5 C LD A,(HSTTRK)
777 DECO 4F C LD C,A
778 DEC1 3A DAA4 C LD A,(HSTDISK)
779 DEC4 C8 19 C RR C
780 DEC6 38 0E C JR C,UNTEN
781 DEC8 E6 F3 C AND OFSH
782 DECA 32 DA91 C LD (UNIT),A
783 DECO 79 C LD A,C
784 DECE 32 DA92 C LD (TRCK),A
785 DED1 AF C XOR A
786 DED2 32 DA93 C LD (HED),A
787 DE05 C9 C RET
788 DED6 F6 04 C UNTEN: OR 4
789 DEDB 32 DA91 C LD (UNIT),A
790 DEDB 79 C LD A,C
791 DEDC 32 DA92 C LD (TRCK),A
792 DEDF 3E 01 C LD A,1
793 DEE1 32 DA93 C LD (HED),A
794 DEE4 C9 C ret
795 if z1013
796 include ZB10.MAC
797 .z80
798 ;Ein- /Ausgaberroutinen Drucker, serielle Kansae
799
800 :(c) Y23VO 15.6.1989
801 ;public domain
802
803
804 ;Initialisierung Peripherieschaltkreise
805 ;PROGRAMMIERUNG PIO FUER MATRIXTASTATUR
806 DEE5 C init:
807 if z1013
808 DEE5 21 DF27 C ld hl,tpiob
809 DEE8 0E 03 C ld c,pioac
810 DEEA 06 03 C ld b,lpiob
811 DEEC ED B3 C stir
812 else
813 ld b,LPIOA ;LAENGE
814 ld c,PIOAC ;ADRESSE PORT A COMMANDO
815 ld hl,TPIOA ;TAB.-ANFANG
816 OTIR
817 ld b,LPIOB
818 INC C
819 OTIR
820 endif
821 ;Initialisierung SIOA f.V24 AUSGABE
822 ;1200 Baud 8 Bit
823 DEEE 3E 05 C ld a,5 ;Bytes für CTC
824 DEF0 D3 6C C out (ctca).a
825 DEF2 3E 00 C ld a,13
826 DEF4 D3 6C C out (ctca).a
827 DEF6 06 07 C ld b,lsioa ;Anzahl Bytes für SIO
828 DEF8 0E 6A C ld c,sioac ;Zieladresse SIO
829 DEFA ED B3 C stir
830 ;Rechnerkopplung mit Stromschleife 20mA; 600 Baud
831 ;$I0B
832 DEFc 3E 05 C ld A,5 ;Initialisierung CTC
833 DEFe D3 60 C OUT (CTCb).A
834 DF00 3E 08 C LD A,8
835 DF02 D3 60 C OUT (CTCb).A
836 DF04 06 07 C LD B,lsiob ;Initialisierung SIO
837 DF06 0E 68 C LD C,SIOBc ;Statusport
838 DF08 ED B3 C OTIR
839 if cntr
840 ;drucker mit CENTRONICS-Schnittstelle
841 ;2-PIO-Karte
842 ld e,0cfh
843 out (piac).a
844 xor a
845 out (piac).a
846 ld e,0cfh
847 out (pibc).a
848 ld e,2
849 out (pibc).a
850 ld e,3
851 out (piac).a
852 out (pibc).a
853 ld e,0ffh

```

```

854
855
856
857 DF0A DB 8E
858 DF0C E6 02
859 DF0E 20 0E
860
861 DF10 21 E41C
862 DF13 11 F600
863 DF16 01 08A9
864 DF19 ED B0
865 DF1B CD F600
866
867 DF1E 0E OC
868 DF20 CD DF38
869 DF23 C9
870
871 DF24 CF C0 07
872 0003
873 DF27 CF FF 07
874 0003
875
876 DF2A 18 04 44 03
877 DF2E 41 05 28
878 0007
879 DF31 18 04 C4 05
880 DF35 28 D3 41
881 0007
882
883
884
885
886
887 DF38 3A 0003
888 DF3B E6 01
889 DF3D C2 F603
890 DF40 C3 E2E0
891
892
893
894
895 DF43
896
897 DF43 CD DF81
898 DB 68
899 DF46 0F
900 DF49 30 F8
901 DF4B DB 69
902 DF4D E6 7F
903 DF4F C9
904
905
906
907
908
909 DF50
910
911 DF50 CD DF81
912 DF53 DB 68
913 DF55 E6 04
914 DF57 28 F7
915 DF59 79
916 DF5A D3 69
917 DF5C C9
918
919
920
921
922
923 DF5D
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945 DF5D C5
946 DF5E DB 6A
947 DF60 CB 47
948 DF62 28 12
949 DF64 DB 68
950 DF66 FE 91
C      out    (pibd).a
C      endif
C      :test: ob grafik-kontroller vorhanden ist
C          in     s,(gdcc)
C          and   2
C          jr    nz,initi
C
C      if
C          gc
C          ld    hl,gdcnf
C          ld    de,gdcram
C          ld    bc,lgdc
C          ldir
C          call  gdciini      ;wenn vorhanden
C
C      endif
C      initi: ld    c,och      ;Bildschira loeschen
C          call  CONOUT
C          RET
C
C      :pio-tabellen
C      TPIOA: DB  OCFH,OC0H,7
C      LPIOA: EQU  $-TPIOA
C      TPIOB: DB  OCFH,OFFH,7
C      LPIOB: EQU  $-TPIOB
C
C      :sio-tabellen
C      TSIOA: DEFB  18h,4,044H,3,041H,5,28H ;Cx16, 1 Stop, 7bit, keine parität
C      TSIOB: DEFB  18h,4,0C4h,5,28h,3,41h ;Cx64, 1 Stop, 7bit, keine Parität
C
C      lsioa equ  $-tsioa
C      Tsioa: db   18h,4,0C4h,5,28h,3,41h ;Cx64, 1 Stop, 7bit, keine Parität
C
C      lslob equ  $-tsiob
C      Tslob: db   18h,4,044H,3,041H,5,28H ;Cx16, 1 Stop, 7bit, keine parität
C
C      :SUBR. ZUR AUSGABE UEBER BILDSCHIRM MIT TEST AUF CTRL/C
C      ;**IN: C...ZEICHEN
C      ;**OUT: -
C      ;*KILL: A,F
C      CONOUT: ld    a.(lobyt)
C          and   i
C          JP    nz,gdcram+3      ;Grafikbildschira
C          JP    bsico            ;alphanumerischer Bildschirm
C
C      :EINGABE EINES ZEICHENS UEBER DAS GERAET RDR:
C      ;**IN: -
C      ;**OUT: A..ZEICHEN
C      READER EQU  $
C      :EINGABE SIO KANAL B, STROMSCHLEIE
C      ssio: call  break
C          IN    A,(SIOBC)      ;statusabfrage
C          RRCA
C          JR    NC,ssio
C          IN    A,(SIOBD)      ;zeicheneingabe
C          AND   7FH
C          RET
C
C      :AUSGABE eines Zeichens ueber Geraset pun:
C      ;**IN: C..ZEICHEN
C      ;**OUT: -
C      ;*KILL: A,F
C      PUNCH EQU  $
C      :AUSGABE sio KANAL B, stromschleife, ZEICHEN IN C
C      ssio: call  break
C          IN    A,(SIOBC)
C          AND   4
C          JR    Z,ssio
C          LD    A,C
C          OUT   (SIOBD),A
C          RET
C
C      :AUSGABE UEBER GERAET "LST:" MIT TEST AUF CTRL/C
C      ;**IN: C..ZEICHEN
C      ;**OUT: -
C      ;*KILL: A,F
C      LIST EQU  $
C      if
C          V24
C          :Druckprogramme V24, 1200 Baud
C          :in: Zeichen in C
C          :out: A,C
C          :kill: A,F
C          latv24: push  bc
C          v241: in    a.(sioac)
C          and   4                  ;Sendepuffer leer ?
C          jr    z,v241
C          lst2: call  break
C          id    a,10h
C          out   (sioac).a      ;reset extern status
C          in    a,(sioac)        ;status lesen
C          and   20h              ;CTS-Test
C          jr    nz,lst2        ;warten auf Freigabe
C          ld    a,c
C          out   (sioad).a      ;Zeichenausgabe
C          pop   bc
C          rat
C
C      else
C          :Drucker mit IFSS-Schnittstelle 1200 Baud
C          SD1152: PUSH  BC
C              IN    A,(SIOAC)
C              BIT   0,A
C              JR    Z,SDR3
C              SDR1: IN    A,(SIOAD)
C              CP    91H             ;Freigabe Drucker

```

```

951 DF68 28 0C C JR Z,SDR3
952 DF6A FE 94 C CP 94H ;Status Drucker
953 DF6C 28 11 C JR Z,SDR4
954 DF6E DB 6A C SDR2: IM A,(SIOAC) :
955 DF70 CB 47 C BIT 0,A
956 DF72 28 FA C JR Z,SDR2
957 DF74 18 EE C JR SDR1
958 DF76 DB 6A C SDR3: IM A,(SIOAC)
959 DF78 CB 57 C BIT 2,A ;Sendepuffer leer ?
960 DF7A 28 FA C LD A,C
961 DF7C 79 C OUT (SIOAD),A ;Zeichenausgabe
962 DF7D 03 6B C POP BC
963 DF7F C1 C SDR4: RET
964 DF80 C9 C
965 C
966 C
967 C
968 C
969 C
970 C
971 C
972 C
973 C
974 C
975 C
976 C
977 C
978 C
979 C
980 C
981 C
982 C
983 C
984 C
985 C
986 C
987 C
988 C
989 C
990 C
991 C
992 DFB1 CD DFA3 C
993 DFB4 B7 C
994 DFB5 C8 C
995 DFB6 CD DFBF C
996 DFB9 FE 03 C
997 DFB8 C0 C
998 DFBC C3 0000 C
999 C
1000 C
1001 DFBF 31 0080 C
1002 DF92 21 DAFB C
1003 DF95 CD DF99 C
1004 DF98 76 C
1005 C
1006 C
1007 C
1008 C
1009 C
1010 DF99 7E C
1011 DF9A 4F C
1012 DF9B 23 C
1013 DF9C B7 C
1014 DF9D C8 C
1015 DF9E CD DF38 C
1016 DFA1 18 F6 C
1017 C
1018 C
1019 0004 C
1020 0004 C
1021 0004 C
1022 0001 C
1023 0000 C
1024 C
1025 C
1026 C
1027 C
1028 C
1029 C
1030 C
1031 C
1032 C
1033 C
1034 C
1035 C
1036 C
1037 C
1038 C
1039 C
1040 C
1041 DFA3 F3 C
1042 DFA4 E5 C
1043 DFA5 D5 C
1044 DFA6 C5 C
1045 DFA7 3E 0F C
1046 DFA9 D3 0B C
1047 DFAB 21 E2D4 C
JR Z,SDR3
CP 94H ;Status Drucker
JR Z,SDR4
IM A,(SIOAC) :
BIT 0,A
JR Z,SDR2
JR SDR1
IM A,(SIOAC)
BIT 2,A ;Sendepuffer leer ?
LD A,C
OUT (SIOAD),A ;Zeichenausgabe
POP BC
RET
endif
if cntr
;Drucker mit CENTRONICS-Schnittstelle
;2-PIO-Karte
;Drucker an pio1.Kanal a Daten, Kanal b Steuerung
cntr: call break
in a,(pibd) ;warten auf Fertigmeldung des Druckers
rra
rre
jr c,centri
ld a,c
out (pibd),a ;Ausgabedaten
ld a,02fH
centri: dec a
jr nz,centri
ld a,0 ;Abtastimpuls erzeugen
out (pibd),a
dec a
out (pibd),a
ret
endif
;-----*
;SUBR. ZUM TEST, OB UNTERBRECHUNG EINGEGEBEN WURDE
;**IN: -
;**OUT: A=0...KEIN ZEICHEN
; A SONST...ZEICHEN
;*KILL: A,F
BREAK: CALL CONST
OR A
RET Z
call conin
cp 3
ret nz
jp 0
;-----*
;Abbruch bei nicht beherrschbarem Fehler
restr: ld sp,80H
id hl,msg3 ;Fehlermeldung
call str ;Ausgabe
halt
;-----*
;SUBROUTINE ZUR AUSGABE EINER ZEICHENFOLGE, ABSCHLUSS DER FOLGE MIT 00
;**IN: ANFANGSADR. IN HL
;**OUT: -
;*KILL: AF,HL
STR: LD A,(HL)
LD C,A
INC HL
OR A
RET Z
CALL CONOUT
JR STR
include Z1SCI40A.MAC
HWMODI EQU 4 ;LATCHADR. HARDWAREMODIFIK.
ZKPOS EQU 4 ;ZEITK.POSITIV-ENTPRELLUNG
ZKNEG EQU 4 ;ZEITK.NEGATIV-ENTPRELLUNG
K7659 EQU 1 ;1 BEI TASTATUR K7659
K7652 EQU 0 ;1 BEI TASTATUR K7652
;*****
;-----*
;TASTATURABFRAGE FUER K 7652/59 MIT HARDWARE_NACH MP 7/88-
;*****



;MODEFLAGS
;BIT 0 : LANGE REPEATERVERZ.DUCHLAUFEN
; 1 : STRINGAUSGABE
; 2 : GRAFIKMODE ON
; 3 : LEER
; 4 : CAPS
; 5 : TASTATURPEEP ON
; 6 : LEER
; 7 : TASTE BETAETIGT

;TASTATURSTATUS
;CONST: DI
PUSH HL
PUSH DE
PUSH BC
LD A,0FH
OUT (LATCH),A
LD HL,MOFL

```

```

1048 DFAE CD DFE4 C CALL MINIT
1049 DF81 21 E2D3 C LD HL,MLZ
1050 DF84 36 00 C LD M,O
1051 DF86 B7 C OR A
1052 DF87 C1 C POP BC
1053 DF88 D1 C POP DE
1054 DF89 E1 C POP HL
1055 DF8A FB C EI
1056 DF8B C8 C RET Z
1057 DF8C 3E FF C LD A,OFFH ;TASTE GEDR.
1058 DFBF C9 C RET

1059
1060
1061 ;ZEICHEN VON TASTATUR 'A' MIT WARTEN AUF BETAETIGUNG
1062 ;
1063 DFBF F3 C CONIN: DI
1064 DFC0 E5 C PUSH HL
1065 DFC1 D5 C PUSH DE
1066 DFC2 C5 C PUSH BC
1067 DFC3 CD DFD8 C CALL TAST
1068 DFC6 B7 C OR A
1069 DFC7 CC DFD1 C CALL Z,SETO
1070 DFC8 C1 C PDP BC
1071 DFCB D1 C POP DE
1072 DFCC E1 C POP HL
1073 DFCD FB C EI
1074 DFCE 28 EF C JR Z,CONIN
1075 DFD0 C9 C RET

1076
1077 DFD1 F5 C SETO: PUSH AF
1078 DFD2 AF C XOR A
1079 DFD3 32 E2D3 C LD (MLZ),A
1080 DFD6 F1 C POP AF
1081 DFD7 C9 C RET

1082
1083 ;TASTATURPOLLING
1084 ;
1085 DFD8 3E OF C TAST: LD A,OFH
1086 DFDA 03 08 C OUT (LATCH),A
1087 DFD9 21 E2D4 C LD HL,M0FL
1088 DFD9 CB 4E C BIT 1,M ;STRINGMODE?
1089 DFE1 C2 E111 C JP NZ,AUSG$ ;AUSGABE
1090 DFE4 DB 02 C MINIT: IN A,(PIOBD)
1091 DFE6 2F C CPL
1092 DFE7 E6 OF C AND OFH ;MASKE
1093 DFE9 20 0A C JR NZ,MSTA1
1094 DFE8 CB 86 C RES 0,M ;RUECKSETZEN REPEATKENNZ.
1095 DFE9 CB BE C RES 7,M ;KEINE BETAETIGUNG
1096 DFFF 32 E2D3 C LD (MLZ),A
1097 DFF2 D3 08 C OUT (LATCH),A
1098 DFF4 C9 C RET
1099 DFF5 3A E2D3 C MSTA1: LD A,(MLZ)
1100 DFF8 B7 C OR A
1101 DFF9 28 33 C JR Z,MP1 ;=1 WENN VORHER REPEAT
1102 DFFB CB 46 C BIT 0,M
1103 DFFD 28 08 C JR Z,MRO

1104
1105 ;KLEINE REPEATVERZ.
1106 ;
1107 DFFF 06 1C C :
1108 E001 0E 00 C MRV1: LD B,28
1109 E003 0D C MRV2: LD C,0
1110 E004 20 FD C DEC C
1111 E008 10 F9 C JR NZ,NKV2
1112 E008 18 24 C DJNZ MRV1
1113 C
1114 ;GROSSE REPEATVERZ.+NEGATIV ENTPRELLUNG
1115 ;
1116 E00A 06 80 C MRO: LD B,B0H
1117 E00C 0E 00 C MRL: LD C,0
1118 E00E DB 02 C MR2: IN A,(PIOBD)
1119 E010 2F C CPL
1120 E011 E6 OF C AND OFH
1121 E013 28 09 C JR Z,MR3
1122 E015 0D C DEC C
1123 E016 20 F6 C JR NZ,MR2
1124 E018 10 F2 C DJNZ MRL
1125 E01A CB C6 C SET 0,M
1126 E01C 18 10 C JR MP1
1127 E01E 06 04 C MR3: LD B,ZKNEG ;NEG. ENTPRELLZ. B*5ms
1128 E020 0E 00 C MR4: LD C,0OH
1129 E022 DB 02 C MR5: IN A,(PIOBD)
1130 E024 2F C CPL
1131 E025 E6 OF C AND OFH
1132 E027 20 E1 C JR NZ,MRO
1133 E029 0D C DEC C
1134 E02A 20 F6 C JR NZ,MRS
1135 E02C 10 F2 C DJNZ MR4

1136
1137 ;SPALTENABTESTUNG
1138 ;
1139 E02E CB 7E C MP1: BIT 7,M ;ENTPRELLUNG?
1140 E030 20 0F C JR NZ,MPO ;POS-ENTPRELLZ. B*5ms
1141 E032 06 04 C LD B,ZKPOS
1142 E034 0E 80 C ENTPR: LD C,B0H
1143 E036 DB 02 C ZS11: IN A,(PIOBD)
1144 E038 2F C CPL

```

```

1145 E039 E6 0F C AND OFH
1146 E03B CB C RET Z
1147 E03C DD C DEC C
1148 E03D 20 F7 C JR NZ,ZS11
1149 E03F 10 F3 C DJNZ ENTPR
1150 E041 AF C MPO1: XOR A
1151 E042 FE 07 C MPS: CP 7
1152 E044 20 02 C JR NZ,MP2
1153 E046 C6 01 C ADD A,1
1154 E048 D3 08 C MP2: OUT (LATCH),A
1155 E04A 57 C LD D,A
1156 E04B DB 02 C MP3: IN A,(PIOBD)
1157 E04D 2F C CPL
1158 E04E E6 0F C AND OFH
1159 E050 20 DB C JR NZ,MP4
1160 E052 7A C LD A,D
1161 E053 C6 01 C ADD A,1H
1162 E055 FE 0C C CP 12 ;12 SPALTEN
1163 E057 20 E9 C JR NZ,MPS
1164 E059 C8 86 C RES 0,M
1165 E05B AF C FZUE: XOR A
1166 E05C C9 C RET
1167 C:
1168 C:;ERMITTLUNG TASTENPOSITION
1169 C:
1170 E05D 01 0708 C MP4: LD BC,708H
1171 E060 ED 41 C OUT (C),B ;SPALTE 7 AKT.
1172 E062 CB 22 C SLA D
1173 E064 CB 22 C SLA D ;MULT,SPALTE MIT B
1174 E066 CB 22 C SLA D
1175 E068 3D C DEC A
1176 E069 82 C ADD A,D
1177 E06A SF C LD E,A ;E=TASTENPOSITION 0-95
1178 C:
1179 C:;AUSWERTUNG SHIFT
1180 C:
1181 E068 32 E2D3 C LD (MLZ),A
1182 E06E DB 02 C IN A,(PIOBD)
1183 E070 E6 0F C AND OFH
1184 E072 EE 08 C XOR 8
1185 E074 7B C LD A,E
1186 E075 20 03 C JR NZ,FKTZU
1187 E077 C6 60 C ADD A,60H
1188 E079 SF C LD E,A
1189 C:
1190 C:;FUNKTIONSZUORDNUNG
1191 C:
1192 E07A 2A E2D8 C FKTZU: LD HL,(FTP0I) ;FUNKT.-TASTENTAB.-POINTER
1193 E07D ED 4B E2D9 C LD HL,(LFTB) ;LAENGE TABELLE
1194 E081 ED 81 C CPIR NZ,MP21 ;SUCHEN TAST.POS.
1195 E083 20 58 C DEC HL
1196 E085 2B C LD BC,(FTP0I)
1197 E086 ED 4B E2DB C SBC HL,BC ;HL:=GEF.POS.REL.
1198 E08A ED 42 C SLA L ;*2
1199 E08C CB 25 C LD BC,(FTAPO) ;F-TASTEN-ADRESS-TB
1200 E08E ED 4B E2DD C ADD HL,BC
1201 E092 09 C LD C,M
1202 E093 4E C INC HL
1203 E094 23 C LD H,M
1204 E095 66 C LD L,C
1205 E096 69 C LD BC,MOFL ;MODEFLAGS
1206 E097 01 E2D4 C LD A,(BC)
1207 E09A 0A C LD 0,A ;LOESCHEN LANGE REPEAT
1208 E09B CB 87 C RES 7,A
1209 E09D CB 7F C BIT 7,A
1210 E09F C2 E05B C JP NZ,FZUE
1211 E0A2 CB FF C SET 7,A
1212 E0A4 02 C LD (BC),A
1213 E0A5 E9 C JP (HL)
1214 C:
1215 E0A6 31 0080 C J100: LD SP,080H
1216 E0A9 C3 0100 C JP 100H
1217 E0AC 2E 04 C GRF: LD L,4
1218 E0AE 18 06 C JR AUSF
1219 E0B0 2E 20 C BEEP: LD L,20H
1220 E0B2 18 02 C JR AUSF
1221 E0B4 2E 10 C SLOCK: LD L,10H
1222 E0B6 0A C AUSF: LD A,(BC)
1223 E0B7 AD C XOR L
1224 E0B8 02 C LD (BC),A
1225 E0B9 AF C XOR A
1226 E0B8 SF C LD E,A
1227 E0B8 C3 E183 C JP SETFF
1228 E0B8 AF C MONRET: XOR A
1229 E0BF D3 D4 C OUT (HWMODI),A
1230 E0C1 C3 F000 C JP 0F000H ;ROM START
1231 E0C4 2E 80 C B3264: LD L,80H
1232 E0C6 18 02 C JR AUSF1
1233 E0C8 2E 20 C ZGUM: LD L,20H
1234 E0CA DB 04 C AUSF1: IM A,(HWMODI)
1235 E0CC AD C XOR L
1236 E0CD 18 0A C JR AUSF2
1237 E0CF DB 04 C F2MHZ: IN A,(HWMODI)
1238 E0D1 E6 BF C AND 0BFH
1239 E0D3 18 04 C JR AUSF2
1240 E0D5 DB 04 C F4MHZ: IN A,(HWMODI)
1241 E0D7 F6 40 C OR 40H

```

```

1242 E0D9 D3 04 C AUSF2: OUT (HWMODI),A
1243 E0DB AF C AUSF3: XOR A
1244 E0DC 32 E2D3 C LD (MLZ),A
1245 E0DF C9 C RET
1246
1247 ;ZEICHEN AUS TABELLE
1248
1249 E0E0 2A E2D5 C MP21: LD HL,(TAPOI)
1250 E0E3 16 00 C LD D,0
1251 E0E5 19 C ADD HL,DE ;BERECHN. TABELLENPOS.
1252 E0E6 7E C LD A,M ;HOLEN ZEICHEN TABELLE
1253 E0E7 5F C LD E,A
1254
1255 ;CONTROL FUNKTION
1256
1257 E0E8 DB 02 C IN A,(PIOBD)
1258 E0EA CB 5F C BIT 3,A ;ABFRAGE CTRL-TASTE
1259 E0EC 20 0E C JR NZ,F$
1260 E0EE 7B C LD A,E
1261 E0EF E6 9F C AND 9FH
1262 E0F1 5F C LD E,A
1263 E0F2 DB 02 C IN A,(PIOBD)
1264 E0F4 E6 0F C AND 0FH
1265 E0F6 20 04 C JR NZ,F$
1266 E0F8 3E 80 C LD A,B0H
1267 E0FA B3 C ADD A,E
1268 E0FB 5F C LD E,A
1269
1270 ;STRING FUNKTION
1271
1272 E0FC CB 7B C F$: BIT 7,E
1273 E0FE 28 28 C JR Z,MNULL
1274 E100 32 E2D3 C LD (MLZ),A
1275 E103 2A E2D7 C LD HL,(POSTA)
1276 E106 7E C F$1: LD A,M
1277 E107 87 C OR A
1278 E108 28 1E C JR Z,MNULL
1279 E10A 23 C INC HL
1280 E10B 88 C CP E
1281 E10C 20 F8 C JR NZ,F$1
1282 E10E 22 E2D1 C LO (NEXT$),HL
1283 E111 2A E2D1 C AUSG$: LD HL,(NEXT$) ;NAECHSTES STR.ZEICHEN
1284 E114 7E C LD A,M
1285 E115 23 C INC HL
1286 E116 22 E2D1 C LD (NEXT$),HL
1287 E119 21 E2D4 C LD HL,MOFL
1288 E11C CB CE C SET 1,M
1289 E11E CB 7F C BIT 7,A
1290 E120 20 02 C JR NZ,AUSGE
1291 E122 B7 C OR A
1292 E123 C0 C RET NZ
1293 E124 CB 8E C AUSGE: RES 1,M
1294 E126 AF C XOR A
1295 E127 C9 C RET
1296
1297 ;GRAFIK-MODE?
1298
1299 E128 21 E2D4 C MNULL: LD HL,MOFL
1300 E12B 7B C LD A,E
1301 E12C CB 56 C BIT 2,M ;GRAFIKMODE?
1302 E12E 28 0B C JR Z,MSM1
1303 E130 FE 08 C CP 8
1304 E132 28 07 C JR Z,MSM1
1305 E134 FE 09 C CP 9
1306 E136 28 03 C JR Z,MSM1 ;CL+CR KEIN GRFK.Z.
1307 E138 C6 80 C ADD A,B0H
1308 E13A 5F C LD E,A
1309
1310 ;CAPS-TASTATURUMSCHALTUNG
1311
1312 E13B CB 66 C MSM1: BIT 4,M
1313 E13D 28 19 C JR Z,MTE ;KEINE UMSCHAL. BIT 4=0
1314 E13F 7B C LD A,E
1315 E140 FE 41 C CP 41H ;C-FLAG GES. FUER A^41H
1316 E142 38 14 C JR C,MTE
1317
1318 ;KONVERTIERUNG GROSS "-> KLEIN
1319
1320 E144 FE 5E C MT5: CP 5EH ;C-FLAG GESETZT FUER A^5BH
1321 E146 30 05 C JR NC,MT61
1322 E148 C6 20 C ADD A,20H
1323 E14A SF C LD E,A
1324 E14B 18 0B C JR MTE
1325 E14D FE 61 C MT61: CP 61H ;C-FLAG GES. FUER A^61H (a)
1326 E14F 38 07 C JR C,MTE
1327 E151 FE 7E C CP 7EH ;C-FLAG GES. FUER A^7BH
1328 E153 30 03 C JR NC,MTE
1329 E155 06 20 C SUB 20H
1330 E157 5F C LD E,A
1331
1332 ;TASTATURBEEPIMPULS
1333
1334 E158 7B C MTE: LD A,E
1335 E159 32 E2D3 C LD (MLZ),A ;LETZTES ZEICHEN = (0004)
1336 E15C CB FE C SET 7,M
1337 E15E CB 6E C BIT 5,M ;BEEP ON?
1338 E160 28 3A C JR Z,MLED1

```

1339	E162	01 2039	C	PEEP1:	LD	8C,2039H	;ZK
1340	E165	C5	C	MPE3:	PUSH	BC	
1341	E166	C5	C		PUSH	BC	
1342	E167	79	C		LD	A,C	;TONH.=0->PAUSE
1343	E168	B7	C		OR	A	
1344	E169	3E 0F	C		LD	A,0FH	
1345	E16B	28 02	C		JR	Z,MPE4	
1346	E16D	3E 0E	C		LD	A,0EH	
1347	E16F	D3 08	C	MPE4:	OUT	(LATCH),A	
1348	E171	C1	C		POP	BC	;C=f, B=t
1349	E172	C5	C		PUSH	BC	
1350	E173	00	C	MPE1:	DEC	C	;NEGATIVE HALBWELLE
1351	E174	20 FD	C		JR	NZ,MPE1	
1352	E176	3E 0F	C		LD	A,0FH	
1353	E17B	D3 08	C		OUT	(LATCH),A	
1354	E17A	C1	C		POP	BC	
1355	E17B	C5	C		PUSH	BC	
1356	E17C	0D	C	MPE2:	DEC	C	;POSITIVE HALBWELLE
1357	E17D	20 FD	C		JR	NZ,MPE2	
1358	E17F	C1	C		POP	BC	
1359	E180	10 E4	C		DJNZ	MPE3	
1360	E182	C1	C		POP	BC	
1361			C				
1362			C				
1363			C				
1364	E183	E5	C	SETFF:	PUSH	HL	
1365	E184	21 E2D4	C		LO	HL,MOFL	
1366	E187	3E 0E	C		LD	A,0EH	;LED-FLIP-FLOP
1367	E189	D3 08	C		OUT	(LATCH),A	;NEU SETZEN
1368	E18B	CB 5E	C		BIT	3,M	
1369	E18D	28 04	C		JR	Z,HGR	
1370	E18F	3E 0C	C		LD	A,0CH	
1371	E191	03 08	C		OUT	(LATCH),A	
1372	E193	CB 66	C	MGR:	BIT	4,M	
1373	E195	E1	C		POP	HL	
1374	E196	28 04	C		JR	Z,MLED1	
1375	E198	3E 0D	C		LD	A,0DH	
1376	E19A	D3 08	C		OUT	(LATCH),A	
1377	E19C	3E 07	C	MLED1:	LO	A,7H	
1378	E19E	D3 08	C		OUT	(LATCH),A	
1379	E1A0	78	C		LO	A,E	
1380	E1A1	B7	C		OR	A	
1381	E1A2	C9	C		RET		
1382			C				
1383			C				
1384			C				
1385			C				
1386			C				
1387			C	FTAB:	DB	.2EH	;GRAFIK ON/OFF
1388			C		DB	35H	;SHIFT-LOCK
1389			C		DB	0A1H	;BEEP ON/OFF
1390			C		DB	43H	;MON-RET/RST 38H
1391			C		DB	48H	:32/64
1392			C		DB	4CH	;ZG-UMSCH.
1393			C		DB	0A8H	:2 MHZ
1394			C		DB	0A8H	:4 MHZ
1395			C		DB	0AOH	;JMP 100
1396			C				
1397			C				
1398			C				
1399			C				
1400			C				
1401			C	FATAB:	DW	GRF	
1402			C		DW	SLOCK	
1403			C		DW	BEEP	
1404			C		DW	MONRET	
1405			C		DW	B3264	
1406			C		DW	ZGUM	
1407			C		DW	F2MHZ	
1408			C		DW	F4MHZ	
1409			C		DW	J100	
1410			C				
1411			C				
1412			C				
1413			C		TAB:	DB	;A4/1
1414			C			31H	
1415			C			51H	;B2/Q
1416			C			41H	;C2/A
1417			C			59H	;D3/Y
1418			C			32H	;A5/2
1419			C			57H	;B3/W
1420			C			53H	;C3/S
1421			C			58H	;D4/X
1422			C			33H	;A6/3
1423			C			45H	;B4/E
1424			C			44H	;C4/D
1425			C			43H	;D5/C
1426			C			34H	;A7/4
1427			C			52H	;B5/R
1428			C			46H	;C5/F
1429			C			56H	;D6/V
1430			C			35H	;A8/5
1431			C			54H	;B6/T
1432			C			47H	;C6/G
1433			C			42H	;D7/B
1434			C			36H	;A9/6
1435			C			5AH	;B7/Z
			C			48H	;C7/H
			C			4EH	;D8/N

1436	DB	37H	:A10/7
1437	DB	55H	:B8/U
1438	DB	4AH	:C8/J
1439	DB	4DH	:D9/M
1440	DB	38H	:A11/8
1441	DB	49H	:B9/I
1442	DB	4BH	:C9/K
1443	DB	2CH	:D10/.
1444	DB	39H	:A12/9
1445	DB	4FH	:B10/0
1446	DB	4CH	:C10/L
1447	DB	2EH	:D11/.
1448	DB	30H	:A13/0
1449	DB	50H	:B11/P
1450	DB	5CH	:C11/DE
1451	DB	20H	:D12/-
1452	DB	7EH	:A14/SZ
1453	DB	5DH	:B12/UE
1454	DB	58H	:C12/AE
1455	DB	5CH	:A15/DE
1456	DB	28H	:B13/+
1457	DB	23H	:C13/DOPPELKREUZ
1458	DB	0	:A3/GRAPHIK
1459	DB	40H	:D2/KOMMERZIELLES A
1460	DB	3EH	:A16/!
1461	DB	0DH	:D14/ENTER/CTRL-M
1462	DB	5	:E2/CURSOR HOCH /CTRL-E
1463	DB	8	:E6/CURSOR LINKS /CTRL-H
1464	DB	20H	:E4/LEERTASTE
1465	DB	0	:B1/CAPS ON/OFF
1466	DB	18H	:E1/CURSOR RUNTER /CTRL-J
1467	DB	4	:E7/CURSOR RECHTS µD
1468	DB	0	
1469	DB	0	
1470	DB	0	
1471	DB	0	
1472	DB	0	
1473	DB	0	
1474	DB	0	
1475	DB	0	
1476	DB	0	
1477	DB	7	
1478	DB	3	
1479	DB	0	
1480	DB	82H	:D1,D13/SHIFT
1481	DB	81H	:C1/CTRL-TASTE
1482	DB	9	:A2/SLOW-FAST
1483	DB	18H	:B14/DEL=µG
1484	DB	0	:A17/JC
1485	DB	83H	:A1/MONITORRUECKKEHR
1486	DB	85H	:S4/'8:'
1487	DB	16H	:S3/'A:'
1488	DB	0	:ES/TAB
1489	DB	84H	:E3/ESC
1490	DB	91H	:S1/32/64-ZEICHEN
1491	DB	7FH	:S5/'C:'
1492	DB	0	:S7/'EXIT'
1493	DB	0	:S9/INS=µV
1494	DB	0	:S2/2.ZG
1495	DB	0	:S6/'DIR'
1496	DB	0	:S8/'QUIT'
1497	DB	0	:S10/DEL
1498	DB	0	
1499	DB	0	
1500	DB	0	
1501	DB	0	
1502	DB	0	
1503	DB	0	
1504	DB	0	
1505	DB	0	
1506	DB	0	
1507	DB	0	
1508	DB	0	
1509	DB	0	
1510	DB	0	
1511	TAB2:	21H	:A4/1
1512	DB	71H	:B2/q
1513	DB	61H	:C2/e
1514	DB	79H	:D3/y
1515	DB	22H	:A5/
1516	DB	77H	:B3/w
1517	DB	73H	:C3/a
1518	DB	78H	:D4/x
1519	DB	40H	:A6/\$
1520	DB	65H	:B4/e
1521	DB	64H	:C4/d
1522	DB	63H	:D5/c
1523	DB	24H	:A7/DOLLAR
1524	DB	72H	:B5/r
1525	DB	66H	:C5/l
1526	DB	76H	:D6/v
1527	DB	25H	:A8/%
1528	DB	74H	:B6/t
1529	DB	67H	:C6/g
1530	DB	62H	:D7/b
1531	DB	26H	:A9/&
1532	DB	7AH	:B7/z

1533			DB	68H	:C7/h
1534			DB	6EH	:D8/n
1535			DB	2FH	:A10//
1536			DB	75H	:B8/u
1537			DB	6AH	:C8/j
1538			DB	6DH	:D9/*
1539			DB	2BH	:A11/(
1540			DB	69H	:B9/i
1541			DB	6BH	:C9/
1542			DB	3BH	:D10/:
1543			DB	29H	:A12/)
1544			DB	6FH	:B10/o
1545			DB	6CH	:C10/1
1546			DB	3AH	:D11/:
1547			DB	3DH	:A13/=
1548			DB	70H	:B11/p
1549			DB	7CH	:C11/o*
1550			DB	5FH	:D12/
1551			DB	3FH	:A14/3
1552			DB	7DH	:B12/ü
1553			DB	7BH	:C12/B
1554			DB	SEH	:A15/CONTROL_ZEICHEN_H
1555			DB	2AH	:B13/*
1556			DB	27H	:C13/'
1557			DB	0	:A3
1558			DB	1EH	:D2/NL
1559			DB	3CH	:A16/'
1560			DB	ODH	:D14/ENTER/CR
1561			DB	OBH	:E2/C.RUNTER
1562			DB	8H	:E6/C_LINKS
1563			DB	20H	:E4/LEERTASTE
1564			DB	0	:B1/CAPS
1565			DB	0AH	:E1/C.HOCH
1566			DB	9H	:E7/C.RECHTS
1567			DB	0	
1568			DB	0	
1569			DB	0	
1570			DB	0	
1571			DB	0	
1572			DB	0	
1573			DB	0	:D1,D13/SHIFT
1574			DB	0	:C1/CTRL-TASTE
1575			DB	0	:A2/J 100
1576			DB	0	:B14/BEEP
1577			DB	0	:
1578			DB	060H	:A1/`
1579			DB	94H	:S4/'TYPE'
1580			DB	90H	:S3/'POWER'
1581			DB	18H	:E5
1582			DB	19H	:E3
1583			DB	0	:S1/2 MHz
1584			DB	8FH	:S5/'POWER COPY'
1585			DB	8AH	:S7/'POWER SIZE ***'
1586			DB	1CH	:S9
1587			DB	0	:S2/4 MHz
1588			DB	86H	:S6/'POWER STAT'
1589			DB	92H	:S8
1590			DB	1DH	:S10
1591			DB	0	:
1592			DB	0	
1593			DB	0	
1594			DB	0	
1595			DB	0	
1596			DB	0	
1597			DB	0	
1598			DB	0	
1599			DB	0	
1600			DB	0	
1601			DB	0	
1602			DB	0	
1603			DB	0	
1604			DB	0	
1605			DB	0	
1606			DB	0	
1607			DB	0	
1608					ENDIF
1609					:
1610					:
1611					IF K7659
1612					:
1613					FUNKTIONSTASTEN-POS.TABELLE
1614					:
1615	E1A3	52	FTAB:	DB 52H	:GRAFIK DN/OFF
1616	E1A4	35		DB 35H	:SHIFT-LOCK
1617	E1A5	55		DB 55H	:BEEP ON/OFF
1618	E1A6	47		DB 47H	:2 MHz
1619	E1A7	48		DB 48H	:4 MHz
1620	E1A8	49		DB 49H	:32*32/16*64
1621	E1A9	4A		DB 4AH	:ALTERNATIV-ZG
1622	E1AA	43		DB 43H	:MON-RET/RST 3BH
1623	E1AB	A7		DB 0A7H	:JMP 100H
1624					
1625					FUNKTIONSTASTEN-ADRESSTABELLE
1626					:
1627					FATAB: DW GRF
1628	E1AC	EOAC		DW SLOCK	
1629	E1AE	EOB4			

1630	E180	E080	C	DW	BEEP
1631	E181	E0CF	C	DW	F2MHZ
1632	E184	E005	C	DW	F4MHZ
1633	E186	E0C4	C	DW	B3264
1634	E188	E0C8	C	DW	ZGUM
1635	E18A	E0BE	C	DW	MOMRET
1636	E18C	E0A6	C	DW	J100
1637			C		
1638			C		;TASTENBELEGUNGSTABELLE
1639			C		;ERSTE SHIFTEBENE
1640			C		;
1641	E1B8	31	C	TAB:	DB 31H :E0/1
1642	E1BF	51	C		DB 51H :D1/Q
1643	E1C0	41	C		DB 41H :C1/A
1644	E1C1	59	C		DB 59H :B1/Y
1645	E1C2	32	C		DB 32H :E2/2
1646	E1C3	57	C		DB 57H :D2/W
1647	E1C4	53	C		DB 53H :C2/S
1648	E1C5	58	C		DB 58H :B2/X
1649	E1C6	33	C		DB 33H :E3/3
1650	E1C7	45	C		DB 45H :D3/E
1651	E1C8	44	C		DB 44H :C3/D
1652	E1C9	43	C		DB 43H :B3/C
1653	E1CA	34	C		DB 34H :E4/4
1654	E1CB	52	C		DB 52H :D4/R
1655	E1CC	46	C		DB 46H :C4/F
1656	E1CD	56	C		DB 56H :B4/V
1657	E1CE	35	C		DB 35H :E5/S
1658	E1CF	54	C		DB 54H :D5/T
1659	E1D0	47	C		DB 47H :C5/G
1660	E1D1	42	C		DB 42H :B5/B
1661	E1D2	36	C		DB 36H :E6/6
1662	E1D3	5A	C		DB 5AH :D6/Z
1663	E1D4	48	C		DB 48H :C6/H
1664	E1D5	4E	C		DB 4EH :A6/N
1665	E1D6	37	C		DB 37H :E7/7
1666	E1D7	55	C		DB 55H :D7/U
1667	E1D8	4A	C		DB 4AH :C7/J
1668	E1D9	40	C		DB 4DH :B7/M
1669	E1DA	38	C		DB 38H :E8/8
1670	E1D8	49	C		DB 49H :D8/I
1671	E1DC	4B	C		DB 4BH :C8/K
1672	E1DD	2C	C		DB 2CH :B8/
1673	E1DE	39	C		DB 39H :E9/9
1674	E1DF	4F	C		DB 4FH :D9/0
1675	E1E0	4C	C		DB 4CH :C9/L
1676	E1E1	2E	C		DB 2EH :B9/.
1677	E1E2	30	C		DB 30H :E10/0
1678	E1E3	50	C		DB 50H :D10/P
1679	E1E4	5C	C		DB 5CH :C10/OE
1680	E1E5	2D	C		DB 2DH :B10/-
1681	E1E6	7E	C		DB 7EH :E11/SZ
1682	E1E7	5D	C		DB 5DH :D11/UE
1683	E1E8	58	C		DB 5BH :C11/AE
1684	E1E9	3C	C		DB 3CH :E12/
1685	E1EA	28	C		DB 2BH :D12/
1686	E1EB	23	C		DB 23H :C12/DOPPELKREUZ
1687	E1EC	5E	C		DB 5EH :E00/CTRL-ZEICHEN
1688	E1ED	40	C		DB 40H :B0/KOMMERZIELLES A
1689	E1EE	3E	C		DB 3EH :E13/
1690	E1EF	0D	C		DB 0DH :B13/ENTER/CTRL-M
1691	E1F0	05	C		DB 5 :A1/CURSOR HOCH /CTRL-E
1692	E1F1	08	C		DB 8 :A10/CURSOR LINKS /CTRL-H
1693	E1F2	20	C		DB 20H :A5/LEERTASTE
1694	E1F3	00	C		DB 0 :D0/CAPS ON/OFF
1695	E1F4	18	C		DB 18H :A0/CURSOR RUNTER /CTRL-X
1696	E1F5	04	C		DB 4 :A11/CURSOR RECHTS /CTRL-D
1697	E1F6	00	C		
1698	E1F7	00	C		
1699	E1F8	00	C		
1700	E1F9	00	C		
1701	E1FA	00	C		
1702	E1FB	00	C		
1703	E1FC	00	C		
1704	E1FD	00	C		
1705	E1FE	00	C		
1706	E1FF	07	C		
1707	E200	09	C		
1708	E201	00	C		
1709	E202	18	C		
1710	E203	7B	C		
1711	E204	7D	C		
1712	E205	00	C		
1713	E206	00	C		
1714	E207	00	C		
1715	E208	00	C		
1716	E209	15	C		
1717	E20A	19	C		
1718	E20B	16	C		
1719	E20C	7F	C		
1720	E20D	03	C		
1721	E20E	06	C		
1722	E20F	00	C		
1723	E210	00	C		
1724	E211	00	C		
1725	E212	00	C		
1726	E213	00	C		

;B99,B11/SHIFT
;C0/CTRL-TASTE

;D13/ROTES C /DEL=µG
;E14/ROTES L /TAB
;F01/MONITORRUECKKEHR
;F02/ESCAPE
;F03/GESCHW.KLAMMER AUF
;F04/ -- ZU
;F06/2MHz
;F07/4MHz
;F08/32/64
;F09/2G-UMSCHALTUNG
;F10/CTRL-U
;F11/CTRL-Y
;F12/CTRL-R/INSERT=µV
;F13/CTRL-P/DEL
;F14/BREAKTASTE /CTRL-C
;F05/CTRL-F

;F53/GRAFIK ON/OFF
;E53
;D53/
;C53/PEEP ON/OFF

1727	E214	00	C	DB	0		
1728	E215	13	C	DB	13H	:E95/CTRL-S=STOP	
1729	E216	10	C	DB	1DH	:F95/CTRL-0	
1730	E217	00	C	DB	0		
1731	E218	00	C	DB	0		
1732	E219	00	C	DB	0		
1733	E21A	12	C	DB	12H	:D,C95/WIPPE + /pR	
1734	E21B	00	C	DB	0		
1735	E21C	03	C	DB	3	:B95/WIPPE - /pC	
1736	E21D	00	C	DB	0		
1737			C				
1738			C				
1739			C				
1740	E21E	21	C	TAB2:	DB	21H	:E1/!
1741	E21F	71	C		DB	71H	:D1/q
1742	E220	61	C		DB	61H	:C1/e
1743	E221	79	C		DB	79H	:B1/y
1744	E222	22	C		DB	22H	:E2/-
1745	E223	77	C		DB	77H	:D2/w
1746	E224	73	C		DB	73H	:C2/s
1747	E225	78	C		DB	78H	:B2/x
1748	E226	40	C		DB	40H	:E3/\$
1749	E227	65	C		DB	65H	:D3/e
1750	E228	64	C		DB	64H	:C3/d
1751	E229	63	C		DB	63H	:B3/c
1752	E22A	24	C		DB	24H	:E4/DOLLAR
1753	E22B	72	C		DB	72H	:D4/r
1754	E22C	66	C		DB	66H	:C4/f
1755	E22D	76	C		DB	76H	:B4/v
1756	E22E	25	C		DB	25H	:E5/%
1757	E22F	74	C		DB	74H	:D5/t
1758	E230	67	C		DB	67H	:C5/g
1759	E231	62	C		DB	62H	:B5/b
1760	E232	26	C		DB	26H	:E6/s
1761	E233	7A	C		DB	7AH	:D6/z
1762	E234	68	C		DB	68H	:C6/h
1763	E235	6E	C		DB	6EH	:B6/n
1764	E236	2F	C		DB	2FH	:E7//
1765	E237	75	C		DB	75H	:D7/u
1766	E238	6A	C		DB	6AH	:C7/j
1767	E239	6D	C		DB	6DH	:B7/m
1768	E23A	28	C		DB	28H	:E8/(
1769	E23B	69	C		DB	69H	:D8/i
1770	E23C	6B	C		DB	6BH	:C8/k
1771	E23D	3B	C		DB	3BH	:B8/;
1772	E23E	29	C		DB	29H	:E9/)
1773	E23F	6F	C		DB	6FH	:D9/o
1774	E240	6C	C		DB	6CH	:C9/1
1775	E241	3A	C		DB	3AH	:B9/:
1776	E242	3D	C		DB	3DH	:E10/*
1777	E243	70	C		DB	70H	:D10/p
1778	E244	7C	C		DB	7CH	:C10/o*
1779	E245	5F	C		DB	5FH	:B10/
1780	E246	3F	C		DB	3FH	:E11/
1781	E247	7D	C		DB	7DH	:D11/
1782	E248	78	C		DB	78H	:C11/B
1783	E249	58	C		DB	58H	:E12/ECK.KLAMMER AUF
1784	E24A	2A	C		DB	2AH	:D12/*
1785	E24B	27	C		DB	27H	:C12/
1786	E24C	7C	C		DB	7CH	:E0/SEKLRSTRICH
1787	E24D	5C	C		DB	5CH	:B0/NEG.SCHRAEGSTRICH
1788	E24E	5D	C		DB	5DH	:E13/ECK.KLAMMER ZU
1789	E24F	0D	C		DB	0DH	:C,B1/ENTER/CR
1790	E250	05	C		DB	05H	:A01/C.HOCH CPM
1791	E251	13	C		DB	13H	:A10/C.LINKS CPM
1792	E252	20	C		DB	20H	:A05/LEERTASTE
1793	E253	00	C		DB	0	:D00/SM-TAST. ON/OFF
1794	E254	18	C		DB	18H	:A00/C.RUNTER CPM
1795	E255	04	C		DB	04H	:A11/C.RECHTS CPM
1796	E256	00	C		DB	0	
1797	E257	00	C		DB	0	
1798	E258	00	C		DB	0	
1799	E259	00	C		DB	0	
1800	E25A	00	C		DB	0	
1801	E25B	00	C		DB	0	
1802	E25C	00	C		DB	0	
1803	E25D	00	C		DB	0	
1804	E25E	00	C		DB	0	
1805	E25F	07	C		DB	7H	:D13/ROTES C/CTRL-G
1806	E260	13	C		DB	13H	:E14/ROTES L/CTRL-S
1807	E261	00	C		DB	0	:F01/MON.RUECK.
1808	E262	18	C		DB	18H	:F02/ESCAPE
1809	E263	60	C		DB	60H	:F03/NEG.ANSTRICH
1810	E264	7E	C		DB	7EH	:F04
1811	E265	00	C		DB	0	:F06/JMP 100H
1812	E266	00	C		DB	0	:F07
1813	E267	00	C		DB	0	:F08
1814	E268	00	C		DB	0	:F09
1815	E269	0F	C		DB	0FH	:F10/CTRL-0
1816	E26A	18	C		DB	18H	:F11/CTRL-X
1817	E26B	11	C		DB	11H	:F12/CTRL-Q
1818	E26C	1F	C		DB	1FH	:F13
1819	E26D	03	C		DB	3H	:F14/BREAKTASTE /CTRL-C
1820	E26E	1E	C		DB	1EH	:F05/NL(SIF1000)
1821	E26F	00	C		DB	0	:F53/GRAFIK ON/OFF
1822	E270	00	C		DB	0	:E53
1823	E271	00	C		DB	0	

```

1824 E272 00 C DB 0 ;D53
1825 E273 00 C DB 0 ;IC53/PEEP ON/OFF
1826 E274 00 C DB 0
1827 E275 1C C DB 1CH ;E95/
1828 E276 1D C DB 1DH ;F95/
1829 E277 00 C DB 0
1830 E278 00 C DB 0
1831 E279 00 C DB 0
1832 E27A 19 C DB 19H ;0,C95/WIPPE +/CTRL-Y
1833 E27B 00 C DB 0
1834 E27C 18 C DB 18H ;B95/WIPPE - /CTRL-X
1835 E27D 00 C DB 0
1836 E27E 81 C ;ENDIF
1837 E27E 81 C SFELD: DB 81H
1838 E27F 41 3A 00 C DB 'A:',0DH
1839 E280 82 C DB 82H
1840 E281 42 3A 00 C DB 'B:',0DH
1841 E282 83 C DB 83H
1842 E283 43 3A 00 C DB 'C:',0DH
1843 E284 84 C DB 84H
1844 E285 44 49 52 00 C DB 'DIR',0DH
1845 E286 85 C DB 85H
1846 E287 45 58 49 54 C DB 'EXIT',0DH
1847 E288 00 C
1848 E289 00 C
1849 E289 91 C DB 91H ;Q
1850 E290 0B 51 4A C DB 0BH,'QJ'
1851 E291 86 C DB 86H ;S
1852 E292 53 54 41 54 C DB 'STAT',0DH
1853 E293 00 C
1854 E294 8A C DB 8AH ;$SIZE ****,0DH
1855 E295 53 49 5A 45 C DB 'SIZE ****,0DH
1856 E296 20 2A 2A 2A C
1857 E2A8 00 C
1858 E2A9 94 C DB 94H ;T
1859 E2AA 54 59 50 45 C DB 'TYPE'
1860 E2AB 90 C DB 90H ;P
1861 E2AF 50 4F 57 45 C DB 'POWER'
1862 E2B3 52 C
1863 E2B4 8F C DB 8FH ;O
1864 E2B5 43 4F 50 59 C DB 'COPY',0DH
1865 E2B9 00 C
1866 E2BA 80 C DB 80H ;-
1867 E2BB 04 04 04 04 C DB 4,4,4,4,4,4,4,4,4
1868 E2BF 04 04 04 04 C
1869 E2C3 04 04 C
1870 E2C5 88 C DB 88H ;-
1871 E2C6 08 08 08 08 C DB 8,8,8,8,8,8,8,8,8
1872 E2CA 08 08 08 08 C
1873 E2CE 08 08 C
1874 E2D0 00 C ENDSF: DB 0 ;ENDE SFELD
1875 E2D1 0000 C ;Arbeitszellen fuer Tastatur
1876 E2D3 00 C NEXTS: DW 0 ;$ ARBEITSZELLE
1877 E2D4 00 C MLZI: DB 0 ;MERKE LETZTES ZEICHEN TASTATUR
1878 E2D4 00 C MOFL: DB 0 ;MOFLAGS
1879 E2D4 00 C ;POINTERFELD
1880 E2D5 E18E C TAPIOI: DW TAB ;TASTATURTAB,-POINTER
1881 E2D7 E27E C POSTA: DW SFELD ;STRINGTAB,-POINTER
1882 E2D9 0009 C LFTB: DW FATAB-FTAB ;LAENGE FU-TASTENTAB.
1883 E2D8 E1A3 C FTPOI: DW FTAB ;FU-TASTENPOS,-TAB.POINTER
1884 E2D0 E1AC C FTAP0: DW FATAB ;FU-TASTERADR,-TAB.POINTER
1885 E2DF C9 C ton: db 0c9h ;Kurzschluss fuer Piep
1886 C
1887 C
1888 C
1889 C
1890 C
1891 C
1892 C
1893 C
1894 C ;Bildschirmausgabe alphanumerisch BS1K
1895 C ;IN: Zeichen in C
1896 C
1897 0040 C ZL EQU 40H ;ZEILENLAENGE
1898 0010 C ZZ EQU 10H ;ZEILENZAHL
1899 C
1900 E2E0 F5 C BS1CO: push af
1901 E2E1 C5 C push bc
1902 E2E2 D5 C push de
1903 E2E3 E5 C push hl
1904 E2E4 CD E2EC C call bcco
1905 E2E7 E1 C pop hl
1906 E2E8 D1 C pop de
1907 E2E9 C1 C pop bc
1908 E2EA F1 C pop af
1909 E2EB C9 C ret
1910 E2EC 21 E327 C bcco: LD HL,CO15
1911 E2EF E5 C PUSH HL
1912 E2F0 2A DAAE C LD HL,(CURS)
1913 E2F3 C8 BE C RES 7,M
1914 E2F5 3A DAB0 C LD A,(POS1) ;LAEUFT POSITIONIERUNG?
1915 E2F8 B7 C OR A
1916 E2F9 20 35 C JR NZ,M0044 ;JA
1917 E2FB 79 C LD A,C
1918 E2FC FE 20 C CP 20H ;STEUERZEICHEN?
1919 E2FE 38 55 C JR C,M0049
1920 E300 FE 7F C CP 7FH

```

1921	E302	30 51	C	JR	NC,M0049
1922	E304	77	C	M0042:	LD M,A
1923	E305	E8	C		EX DE,HL
1924	E306	13	C	C09:	INC DE ;15H
1925	E307	7B	C		ld a,*
1926	E308	B2	C		or d
1927	E309	E8	C		EX DE,HL
1928	E30A	C0	C		ret nz
1929	E30B	11 FFBF	C		LD DE,BRAM +x1*zz-ZL-1
1930	E30E	D5	C	M0043:	PUSH DE ;scroll
1931	E30F	21 FC40	C		LD HL,bram+zz1
1932	E312	11 FC00	C		LD DE,BRAM
1933	E315	01 03C0	C		LD BC,zz*x1-ZL
1934	E318	ED 80	C		LDIR
1935	E31A	28	C		DEC HL
1936	E31B	36 00	C		LD M,00H
1937	E31D	11 FFFE	C		LD DE,bram+zz*x1-2
1938	E320	01 003F	C		LD BC,ZL-1
1939	E323	ED BB	C		LDDR
1940	E325	E1	C		POP HL
1941	E326	C9	C		RET
1942	E327	22 DAAE	C	C015:	LD (CURS),HL ;Ende der Ausgabe
1943	E32A	3A DAB1	C		LD A,(POS2)
1944	E32D	B6	C		OR M
1945	E32E	77	C		LD M,A
1946	E32F	C9	C		RET
1947			C		;CURSORPOSITIONIERUNG
1948			C		;IN: A ZEILE,C SPALTE mit gesetztem 7. bit
1949			C		; HL,BRAM,DE ZEILENLAENGE
1950			C	jout:	hl Cursorposition
1951	E330	CB B9	C	M0044:	RES 7,C
1952	E332	3D	C		DEC A
1953	E333	32 DAB0	C		LD (POS1),A
1954	E336	28 13	C		JR Z,spalte
1955	E338	3E OF	C		LD A,OFH
1956	E33A	89	C		CP C
1957	E33B	38 01	C		JR C,zeile
1958	E33D	79	C		LD A,C
1959	E33E	21 FC00	C	zeile:	LD HL,BRAM
1960	E341	B7	C		OR A
1961	E342	C8	C		RET Z
1962	E343	11 0040	C		LD DE,ZL
1963	E346	47	C		LD B,A
1964	E347	19	C	zeili:	ADD HL,DE
1965	E348	10 FD	C		DJNZ zeili
1966	E34A	C9	C		RET
1967	E34B	3E 3F	C	spalte:	LD A,ZL-1
1968	E34D	B9	C		CP C
1969	E34E	30 01	C		JR NC,spalti
1970	E350	4F	C		LD C,A
1971	E351	06 00	C	spalti:	LD B,00H
1972	E353	09	C		ADD HL,BC
1973	E354	C9	C		RET
1974			C		-----
1975			C		;STEUERZEICHENAUSWERTUNG
1976	E355	E8	C	M0049:	EX DE,HL
1977	E356	21 E40D	C		LD HL,bstab
1978	E359	01 000F	C		LD BC,lbstab ;ANZAHL STEUERZEICHEN
1979	E35C	E0 B1	C		CPIR
1980	E35E	28 0F	C		JR Z,M004B
1981	E360	E8	C		EX DE,HL
1982	E361	FE 84	C		CP 84H
1983	E363	D8	C		RET C
1984	E364	FE 88	C		CP 88H
1985	E366	38 03	C		JR C,M004A
1986	E368	FE 80	C		CP 080H
1987	E36A	D8	C		RET C
1988	E36B	CB BF	C	M004A:	RES 7,A
1989	E36D	18 95	C		M0042
1990	E36F	21 E40C	C	M004B:	LD HL,BSTAB-1 ;Steuerzeichen gefunden
1991	E372	AF	C		XOR A
1992	E373	ED 42	C		SBC HL,BC
1993	E375	ED 42	C		SBC HL,BC
1994	E377	46	C		LD B,M
1995	E378	2B	C		DEC HL
1996	E379	4E	C		LD C,M
1997	E37A	C5	C		PUSH BC
1998	E37B	6B	C		LD L,E
1999	E37C	62	C		LD H,D
2000	E37D	C9	C		RET
2001	E37E	01 0040	C	bcr:	LD BC,ZL ;Wagenruecklauf
2002	E381	21 FFFF	C		LD HL,bram+zz*xz-1
2003	E384	ED 42	C	bcri:	SBC HL,BC
2004	E386	E5	C		PUSH HL
2005	E387	ED 52	C		SBC HL,DE
2006	E389	E1	C		POP HL
2007	E38A	30 FB	C		JR NC,bcri
2008	E38C	23	C		INC HL
2009	E38D	C9	C		RET
2010	E38E	21 FFBF	C	blf:	LD ML,bram+zz*xz-zl-1 ;Zeilenverschub
2011	E391	ED S2	C		SBC HL,DE
2012	E393	DA E30E	C		JP C,M0043
2013	E396	21 0040	C		LD HL,ZL
2014	E399	19	C		ADD HL,DE
2015	E39A	C9	C		RET
2016	E39B	12	C	bdel:	LD (DE),A ;linkes Zeichen loeschen
2017	E39C	21 FC00	C	bbs:	LD ML,BRAM ;backspace

```

2018 E39F ED 52 C SBC HL,DE
2019 E3A1 EB C EX DE,HL
2020 E3A2 C8 C RET Z
2021 E3A3 28 C DEC HL
2022 E3A4 C9 C RET
2023 E3A5 11 FC00 C blhm: LD DE,BRAM ;Bildschirm loeschen
2024 E3A8 21 FFFF C C08: LD HL,bram+zz*zl-1 ;ab Cursor loeschen
2025 E3A8 EO 52 C SBC HL,DE
2026 E3AD EB C EX DE,HL
2027 E3AE 77 C LD M,A
2028 E3AF C8 C RET Z
2029 E3B0 E5 C PUSH HL
2030 E3B1 42 C LD B,D
2031 E3B2 48 C LD C,E
2032 E3B3 54 C LD D,H
2033 E3B4 5D C LD E,L
2034 E3B5 13 C INC DE
2035 E3B6 EO 80 C LDIR
2036 E3B8 E1 C POP HL
2037 E3B9 C9 C RET
2038 E3BA 21 FC00 C CHOME: LD HL,BRAM ;cursor home
2039 E3BD C9 C RET
2040 E3BE 21 FC3F C CUP: LD HL,BRAM+ZL-1 ;cursor hoch
2041 E3C1 EO 52 C SBC HL,DE
2042 E3C3 E8 C EX DE,HL
2043 E3C4 00 C RET NC
2044 E3C5 11 FF00 C LD DE,bram+zz*zl-zl
2045 E3C8 19 C ADD HL,DE
2046 E3C9 C9 C RET
2047 E3CA CO E37E C M004E: CALL bcr
2048 E3CD EB C EX DE,HL
2049 E3CE D5 C PUSH DE
2050 E3CF CD E37E C CALL bcr
2051 E3D2 09 C ADD HL,BC
2052 E3D3 AF C XOR A
2053 E3D4 EO 52 C SBC HL,DE
2054 E3D6 12 C M004F: LD (DE).A
2055 E3D7 13 C INC DE
2056 E3D8 2D C DEC L
2057 E3D9 20 FB C JR NZ,M004F
2058 E3D9 E1 C POP HL
2059 E3DC C9 C RET
2060 E3DD 3E 02 C
2061 E3DF 32 DAB0 C C011: LD A,02H
2062 E3E2 C9 C LD (POS1),A
2063 E3E3 3E 80 C RET
2064 E3E5 18 01 C C013: LD A,80H
2065 E3E7 AF C JR M0051
2066 E3E8 32 DAB1 C M0050: XOR A
2067 E3E8 C9 C M0051: LD (POS2),A
2068 E3EC C3 E2DF C RET
2069 bell: JP ton
2070 ;ADRESSEN
2071 E3EF E37E E38E C DW BCR,b17,bbs,blhm,cheme,CUP,BDEL,C08
2072 E3F3 E39C E3A5 C
2073 E3F7 E3BA E3BE C
2074 E3FB E398 E3AB C
2075 E3FF E306 E3CE C DW C09,C010,M004E,C011,bell,C013,M0050
2076 E403 E3CA E3D0 C
2077 E407 E3EC E3E3 C
2078 E408 E3E7 C
2079 ;STEUERZEICHEN
2080 E400 00 0A 08 0C C bstab: DB ODH,0AH,8,0CH,1,1AH,7FH,14H,15H,16H,18H,1BH,7,02H,83H
2081 E411 01 1A 7F 14 C
2082 E415 15 16 18 1B C
2083 E419 07 82 83 C
2084 000F C lbstab equ $-bstab
2085 ;Diskettenpuffer
2086 E41C hstact equ $ ;AKTIVIERUNGSFLAG
2088 E41D HSTWRT equ hstact+1 ;SCHREIBFLAG
2089 E41E UNACNT equ hstwrt+1 ;ZAehler FUER NICHTBEL. SAETZE
2090 E41F UNADSK equ unsact+1 ;LETZTES NICHTBEL. LAUFWERK
2091 E420 UNATRK equ unadsk+1 ;LETZTE NICHTBEL. SPUR
2092 E421 UNASEC equ unatrk+1 ;LETZTER NICHTBEL. SEKTOR
2093 E422 ERFFLAG equ unasec+1 ;FEHLERCODE
2094 E423 RSFLAG equ erfflag+1 ;LESEFLAG
2095 E424 READOP equ rsflag+1 ;1 BEI LESEOPERATIONEN
2096 E425 ALL0 equ readop+1
2097 E475 CHKO equ all0+80
2098 E4C5 ALL1 equ chko+80
2099 E515 CHK1 equ all1+80
2100 E565 all2 equ chk1+80
2101 E5B5 chk2 equ all2+32
2102 E5B0 DIRBF equ chk2+8 ;Inhaltsverzeichnis Diskette
2103 E60E HSTBUF equ dirbf+129 ;Zwischenraeber fuer Diskette
2104 EA0F kopie equ HSTBUF+1025 ;iccp-kopie
2105 E41C if gc
2106 gdconf equ $ .dphase
2107 .phase gdram
2108 include GRAFIK.MAC
2109 .z80
2110 ;Programm fuer Grafik-Display-Controller U82720/pD7220
2111 ;(c) Y23V0
2112 ;erstellt 17.9.87
2113 ;aktualisiert 19.5.88

```

```

2115
2116
2117
2118 F600 C3 F693 ;public domain
2119 F603 C3 F69F ;initialisierung fuer grafik-display-controller 82720/7220
2120 F606 C3 F94D ;gdcini: jp inigdc ;zeichnungsabgabe
2121 F609 C3 F968 ;jp pixel ;punkt zeichnen /loeschen
2122 F60C C3 F980 ;jp draw ;linie zeichnen
2123 F60F C3 F9C6 ;jp prbox ;rechteck zeichnen
2124 F612 C3 F9F9 ;jp hpr ;bildschirmzeile ruecklesen
2125
2126 ;jp hpw ;bildschirmzeile in speicher bringen
2127 ;ram-tabellen fuer graphic-display-controller 7220
2128 F615 00 00 ;Bereich mit Daten zum Zeichnen(draw,prbox,circle)
2129 F617 00 00 ;x1: db 0,0
2130 F619 0A 00 ;y1: db 0,0
2131 F61B 0A 00 ;x2: db 10,0
2132 F61D 00 ;y2: db 10,0
2133 F61E FBA9 ;rotat: db 0 ;fuer prbox
2134 F620 00 FD ;zadr: dw zganf ;Adresse des Zeichngenerators
2135 F622 01 20 ;phbr: db 0,0fh ;Adr. des Copybereichs im RAM(27h lang)
2136 F624 03 78 FF FF ;dwdat: db 1,20h ;Pattern-Pixel Linienart beim
2137 F628 04 49 ;dprom: db 3,78h,0ffh,0ffh;Zeichnen
2138 F62A 00 00 ;dcurs: db 4,49h
2139 F62C 00 ;dpas: db 0,0
2140 F62D 0C 4C ;dpix: db 0
2141 F62F 08 00 00 00 ;dfiga: db 0ch,4ch
2142 F633 00 00 00 00 ;drich: db 8,0,0,0,0,0,0,0,0ffh,0ffh
2143 F637 00 FF FF
2144 F63A 01 6C FF ;dx: db 1,6ch,0ffh
2145 F63D 00 00 ;dy: db 0,0
2146 F63F 00 00 ;;Datenbereich zur Ausgabe der ASCII-Zeichen (Pattern)
2147 F641 09 78 ;type: db 9,78h
2148 F643 00 FF 00 FF ;prmp: db 0,0ffh,0,0ffh,0,0ffh,0,0ffh
2149 F645 00 FF 00 FF
2150 F647 00 FF 00 FF
2151 F64B 03 4C 10 07 ;db 3,4ch,10h,7 ;8 Byte Laenge
2152 F64F 01 68 FF ;db 1,68h,0ffh ;Graphikzeichen ausgeben
2153 ;Datenbereich zum Loeschen von Bildschirmausschnitten_
2154 F652 01 22 09 78 ;cisc0: db 1,22h,9,78h,0ffh,0ffh,0ffh,0ffh,0ffh
2155 F656 FF FF FF
2156 F65A FF FF
2157 F65C FF FF 03 4C ;db Offh,0ffh,3,4ch,10h
2158 F660 10
2159 F661 2C 03 4A FF
2160 F665 FF 01 68 01 ;clspos: db 2ch,3,4ch,0ffh,0ffh,1,68h,1,20h,0ffh
2161 F669 20 FF
2162 F66B 01 ;zcls1: db 1
2163 F66C 20 04 49 ;smod: db 20h,4,49h ;Cursorpositionierung
2164 F66F 00 00 ;pos: db 0,0
2165 F671 00 FF ;pix: db 0,0ffh
2166 F673 03 78 FF FF ;zcls2: db 3,78h,0ffh,0ffh
2167 F677 03 4A FF FF ;db 3,4ch,0ffh,0ffh ;Maske auf fffff setzen
2168 F67B 04 4C 02 ;db 4,4ch,2 ;figs Kommando
2169 F67E 3F 01 ;leang: db 3fh,1 ;
2170 F680 03 22 FF FF ;db 3,22h,0ffh,0ffh ;wdat reset Pixel
2171 F684 01 20 FF ;curse2: db 1,20h,0ffh ;wdat Pattern Pixel
2172 ;Datenbereich zum Bildschirarollen
2173 F687 03 70 ;scroll: db 3,70h ;im PRAM Adr. des Bilds.anfangs aendern
2174 F689 00 00 FF ;pos0: db 0,0,0ffh
2175 ;System Daten
2176 F68C 01 ;flag: db 1 ;Curs.ein aus usw.
2177 F68D 00 ;xk: db 0 ;ix Koord.
2178 F68E 00 ;yk: db 0 ;iy Koord.
2179 F68F 00 00 ;curs0: db 0,0 ;Adr. des Bildanfangs fuer GDC
2180 F691 00 00 ;cursa: db 0,0 ;Adr. der Zeile des Cursors
2181 0093 lgdcram equ $-gdcini
2182
2183 0280 xmax equ 640 ;Anzahl Punkte
2184 00C8 ymax equ 200
2185
2186 ;Programm zur Initialisierung des GDC
2187 F693 21 0003 inigdc: ld hl,1oby7 ;auf gdc-Ausgabe umschalten
2188 F696 CB C6 ;set 0,(hl)
2189 ; ld hl,tabanf ;tabellenadresse im rom
2190 ; ld de,gdcram ;Adresse ramtabellen gdc
2191 ; ld bc,lgdcram
2192 ; ldir
2193 F698 21 FB64 ;ld hl,inid
2194 F69B CD F92D ;call ioup
2195 F69E C9 ;ret
2196 ;Hauptprogramm Zeichen wird in C uebergeben
2197 F69F F5 ;hp: push af
2198 F6A0 E5 ;push hl
2199 F6A1 C5 ;push bc
2200 F6A2 D5 ;push de
2201 F6A3 DD E5 ;push ix
2202 F6A5 79 ;ld e,c
2203 F6A6 D6 20 ;sub 020h
2204 F6A8 DA F777 ;jp c,stuP ;wenn Steuerzeichen Sprung
2205 F6AB FE 5F ;cp 5fh
2206 F6AD D2 F777 ;jp nc,stuP ;Steuerzeichen
2207 F6B0 CD F6C0 ;call hpi ;Zeichen ausgeben Cursor ueberschr.
2208 F6B3 CD F71A ;call test ;Test ob eine Pos. weiter moeglich
2209 F6B6 CD F6DA ;call cursb0 ;Cursor setzen
2210 F6B9 DD E1 ;ende: pop ix
2211 F6B8 D1 ;pop de

```

```

2212 F6BC C1
2213 F6BD E1
2214 F6BE F1
2215 F6BF C9
2216
2217 F6C0 26 00
2218 F6C2 6F
2219 F6C3 ED 5B F61E
2220 F6C7 29
2221 F6C8 29
2222 F6C9 29
2223 F6CA 19
2224 F6CB 11 F643
2225 F6CE 01 0008
2226 F6D1 ED 80
2227 F6D3 21 F641
2228 F6D6 CD F920
2229 F6D9 C9
2230
2231 F6DA 3A F6BC
2232 F6DD E6 01
2233 F6DF C8
2234 F6E0 FO E5
2235 F6E2 FO 21 F6BC
2236 F6E5 FO 36 00 21
2237 F6EA CD F6FC
2238 F6ED 3E 5F
2239 F6EF CD F6C0
2240 F6F2 FO 36 00 20
2241 F6F6 CD F6FC
2242 F6F9 FD E1
2243 F6FB C9
2244
2245 F6FC ED 5B F691
2246 F700 3A F68D
2247 F703 21 F671
2248 F706 36 00
2249 F708 A7
2250 F709 1F
2251 F70A C8 1E
2252 F70C 6F
2253 F70D 26 00
2254 F70F 19
2255 F710 22 F66F
2256 F713 21 F668
2257 F716 CD F920
2258 F719 C9
2259
2260
2261
2262 F71A 2A F68D
2263 F71D 2C
2264 F71E 22 F68D
2265 F721 3E 50
2266 F723 8D
2267 F724 C0
2268 F725 2E 00
2269 F727 24
2270 F728 22 F68D
2271 F72B 3E 19
2272 F72D BC
2273 F72E 28 1A
2274 F730 2A F691
2275 F733 D5
2276 F734 11 0140
2277 F737 19
2278 F738 D1
2279 F739 7C
2280 F73A E6 1F
2281 F73C 67
2282 F73D 22 F691
2283 F740 22 F66F
2284 F743 21 F668
2285 F746 CD F920
2286 F749 C9
2287 F74A 25
2288 F74B 22 F68D
2289 F74E 2A F68F
2290 F751 D5
2291 F752 11 0140
2292 F755 19
2293 F756 D1
2294 F757 7C
2295 F758 E6 1F
2296 F75A 67
2297 F758 22 F68F
2298 F75E 22 F689
2299 F761 21 F687
2300 F764 CD F920
2301 F767 CD F730
2302 F76A 21 F673
2303 F76D CD F920
2304 F770 21 F668
2305 F773 CD F920
2306 F776 C9
2307
2308

pop bc
pop hl
pop af
ret

;Prg.zur Zeichenausgabe (A)=ASCII - Zeichen-20h
hpi: ld h,0
      ld l,a
      ld de,(zgadr)
      add hl,hl ;"Zeichen"8
      add hl,hl
      add hl,de ;+ Zg.Anfangsadr.=Zeichenadr.
      ld de,pramp ;in Datenbereich zur Ausgabe
      ld bc,0008 ;kopieren
      ldir
      ld hl,type ;als Muster (pattern) uegeben
      call ioup
      ret

;Cursor auf aktuelle Position invertieren
curso0: ld s,(flag)
      and i
      ret x
      push iy
      ld iy,smod
      ld (iy),21h ;in Kompl.-Modus setzen
      call curso0 ;Curs.adr. und
      ld s,5th ;ASCII fuer Curs. ausgeben
      call hpi
      ld (iy),20h ;vom Kompl.-Zeichenmodus
      call curso0 ;in PATTERN-Modus
      pop iy
      ret

;Prg.zur Ausgabe der Cursoradr. fuer den GDC
curso1: ld de,(cursc)
      id s,(xk) ;aus Zeilenadr. und x Koord Curs.Adr.er-
      id l,(xk)
      id hl,pix
      id (hl),0h
      and s ;Carry = 0
      rra (hl) ;ix Koord. durch 2 dividieren
      rr (hl) ;i=Wortadr.und Punktsadr.
      ld l,a
      ld m,0
      add hl,de ;errechnen
      ld (pos),hl ;und ausgeben
      ld hl,zclsi
      call ioup
      ret

;Prg. zum Test ob neue Position ,auf die der GDC-Curs nach
;letzter Ausgabeoperation steht,moeglich ist ,sonst neue Zeilenadr.
;bestimmen und Bildschirmrollen wenn noetig
test: ld hl,(xk)
      inc l ;seine Position weiter
      ld (xk),hl
      ld s,xmax/8 ;Spaltenzahl
      cp l ;letzte Spalte?
      ret nz ;nicht-Ruecksprung
      ld l,0 ;neue Zeile,Zeilenanfang
      inc h
      ld (xk),hl ;Zeilenzahl
      ld s,ymax/8 ;neue Adr.des Zeilenanfangs
      cp h ;letzte Zeile?
      jr x,scr1 ;ja Bildschirm rollen
      ld hl,(cursc) ;neue Adr.des Zeilenanfangs
      push de ;errechnen(alte=320 Worte)
      ld de,xmax/2 ;(da obere Adr.bits nicht
      add hl,de ;ausgewertet werden auch
      pop de ;Adr.'8k Worte moeglich)
      ld a,h ;hier auf
      and 01fh ;8k Worte begrenzt
      ld m,a
      ld (cursc),hl ;neue Cursor pos
      ld (pos),hl ;ausgeben
      ld hl,zclsi
      call ioup -
      ret

scr1: dec h
      ld (xk),hl ;Anfangsadr. fuer Bildschirm
      ld hl,(cursc) ;bestimmen (alte >320)
      push de
      ld de,320
      add hl,de
      pop de
      ld s,h ;hier Adr.
      and 01fh ;auf max.8k Worte begrenzt
      ld h,s
      ld (curs0),hl ;und ausgeben bzw zwischen-
      ld (pos0),hl ;speichern
      ld hl,scroll
      call ioup
      call m003 ;Zeilenadr. bestimmen
      ld hl,zclsi2 ;Zeile loeschen
      call ioup
      ld hl,zclsi ;letzte Curs.pos neu einschr.
      call ioup
      ret

=====
;Steuerzeichenauswertung

```

2309	F777	C5	C	stup:	push	bc		
2310	F778	CD F6DA	C		call	cursb0	jelten Cursor loeschen	
2311	F778	C1	C		pop	bc		
2312	F77C	79	C		ld	s,c	;Steuerzeichen neu laden	
2313	F77D	F5	C		push	af		
2314	F77E	ED 48 F680	C		ld	bc,(xk)	;Koordin.laden	
2315	F782	17	C		rla	nc,m0288	;Zeichen groesser 80h?	
2316	F783	30 3D	C		jr	e,(flag)	;nein,Steuerz "20h u Sprung)	
2317	F785	3A F68C	C		ld			
2318	F788	F5	C		push			
2319	F789	17	C		rla			
2320	F78A	30 35	C		jr	nc,m0289	;Tset ob x o,y Koordin.	
2321	F78C	17	C		rla		;erwartet werden (zuvor 18h)	
2322	F78D	30 21	C		jr	nc,m028a	;Verzweigung y o. y Koordin.	
2323	F78F	F1	C		pop	af	;Sprung bei x Koordin	
2324	F790	C8 B7	C		res	6,a		
2325	F792	32 F68C	C		ld	(flag),a	;speichern, dass x Koordin.	
2326	F795	F1	C		and	a	;noch erwartet wird	
2327	F796	E6 1F	C		pop	af		
2328	F798	32 F68E	C		and	Oifh		
2329	F798	47	C		ld	(yk),a		
2330	F79C	2A F68F	C		ld	b,a		
2331	F79F	11 0140	C		hl,(curs0)		;aktuelle Cursoradr.	
2332	F7A2	A7	C		ld	de,320	;errechnen und ausgeben	
2333	F7A3	28 03	C		and	a		
2334	F7A5	19	C	m0301:	add	x,m0300		
2335	F7A6	10 FD	C		djnz	hi,de		
2336	F7AB	22 F691	C	m0300:	ld	m0301		
2337	F7AB	CD F6FC	C		call	(cursa),hl		
2338	F7AE	18 20	C		jr	cursc0		
2339						m0280		
2340	F7B0	F1	C					
2341	F7B1	CB BF	C	m028a:	pop	af		
2342	F7B3	32 F68C	C		res	7,a	;Steuerzeichenfolge (1Bh)	
2343	F7B6	F1	C		ld	(flag),a	;beenden	
2344	F7B7	E6 7F	C		pop	af		
2345	F7B9	32 F68D	C		and	07fh		
2346	F7BC	CD F6FC	C		ld	(xk),a	;aktuelle Adr. ausgeben	
2347	F7BF	18 0F	C		call	cursc0		
2348					jr	m0280		
2349	F7C1	F1	C					
2350	F7C2	F1	C	m0289:	pop	af		
2351	F7C3	21 F6FF	C		pop	af		
2352	F7C6	CD F648	C	m0288:	id	hl,jptab	;Steuerzeichen hohlen	
2353	F7C9	38 05	C		call	FCGA	;Tabellenanfang laden	
2354	F7CB	11 F700	C		jr	c,m0280		
2355	F7CE	D5	C		ld	de,m0280	;Sprung wenn Zeichen kein Steuerzei.	
2356	F7CF	E9	C		push	de	;Rt-Adr. fuer Unterprg. auf Stack	
2357					jp	(hl)	;schaffen	
2358	F7D0	CD F6DA	C	m0280:	call	cursb0		
2359	F7D3	C3 F689	C		jp	ende		
2360	F7D6	01 0000	C	ghome:	ld	bc,0		
2361	F7D9	ED 43 F680	C		ld	(xk),bc	;Zeile u Spalte 0 ausgeben	
2362	F7D0	ED 48 F68F	C		ld	bc,(curs0)		
2363	F7E1	ED 43 F691	C		ld	(cursa),bc		
2364	F7E5	CD F6FC	C		call	cursc0		
2365	F7E8	C9	C		ret			
2366	F7E9	CD F8AA	C		call	cull		
2367	F7EC	AF	C		xor	a		
2368	F7ED	CD F6C0	C		call	hp1	;Leerzeichen ausgeben	
2369	F7F0	CD F6FC	C		call	cursc0		
2370	F7F3	C9	C		ret			
2371	F7F4	3E 4F	C		if:	ld	;rechten Rand als x Koord.	
2372	F7F6	32 F680	C			a,79		
2373	F7F9	C5	C			(xk),a		
2374	F7FA	CD F71A	C		push	bc		
2375	F7FD	C1	C		call	test	;=nsechste Position	
2376	F7FE	79	C		pop	bc		
2377	F7FF	32 F68D	C		ld	e,c		
2378	F802	CD F6FC	C		ld	(xk),a		
2379	F805	C9	C		call	cursc0	;alte x Position einsetzen	
2380	F806	21 1FFF	C		ret			
2381	F809	22 F67E	C	bloeh:	ld	h1,01ffff	;Byteanzahl der wdat-Funktion	
2382	F80C	21 F673	C		ld	(laeng),h1	;auf ges.Bilds.erhoehen	
2383	F80F	CD F920	C		ld	h1,zcl12		
2384	F812	21 0140	C		call	ioup		
2385	F815	22 F67E	C		ld	h1,320	;und auf Zeilenlaenge setzen	
2386	F818	01 0000	C		ld	(laeng),h1		
2387	F81B	ED 43 F689	C		ld	bc,0	;Adr. Bildenfang auf 00	
2388	F81F	ED 43 F68F	C		ld	(pos0),bc	;;z.B.def.Zustand fuer	
2389	F823	21 F687	C		ld	(curs0),bc	;Graphicarbeiten	
2390	F826	CD F920	C		ld	hl,scroll		
2391	F829	18 AB	C		call	ioup		
2392	F82B	0E 00	C		jr	ghome		
2393	F82D	ED 43 F68D	C	cr:	ld	c,0h	;linken Rand eintragen	
2394	F831	CD F6FC	C		ld	(xk),bc		
2395	F834	C9	C		call	cursc0		
2396	F835	3E 4F	C		ret			
2397	F837	91	C	clszc:	ld	a,4fh	;Differenz zum rechten Rand	
2398	F838	28 16	C		sub	c		
2399	F83A	1F	C		jr	z,clszc3	;Cursor am rechten Rand	
2400	F83B	38 06	C		rra		;durch 2 div.	
2401	F83D	F5	C		jr	c,clszc2	;ungerade Diff.=?xKoord. war gerade	
2402	F83E	CD F850	C		push	af		
2403	F841	F1	C		call	clszc3	;Leerzeichen ausgeben, aktive Cursor-	
2404	F842	3D	C		pop	af		
2405	F843	32 F661	C		dec	a		
				clszc2:	ld	(clspos),a	;aktive Koord.ist gerade	

```

2408 F846 21 F652 C ld hl,clszc0 ;a*16 Bit werden in x Richt.
2407 F849 CD F92D C call ioup ;geloeschett
2408 F84C CD F6FC C call cursc0
2409 F84F C9 C
2410 F850 F5 C clszc3: push af
2411 F851 AF C xor a
2412 F852 CD F6C0 C call hpi1 ;Leerzeichen ausgeben
2413 F855 F1 C pop af
2414 F856 C0 C ret nz
2415 F857 CD F6FC C call cursc0
2416 F85A C9 C ret
2417 F85B CD F71A C cuir: call test ;Cursor eine Position weiter.
2418 F85E CD F6FC C call cursc0
2419 F861 C9 C
2420 F862 CD F828 C clsz1: call er ;Cursor an Zeilenanfang
2421 F865 21 F673 C id hl,zcls2 ;zu Zeile loeschen
2422 F868 CD F92D C call ioup
2423 F86B CD F6FC C call cursc0
2424 F86E C9 C ret
2425 F86F 78 C cuih: ld s,b
2426 F870 D6 01 C sub 1 ;Carry wird registriert
2427 F872 47 C ld b,a
2428 F873 30 0A C jr nc,cuin0 ;Cursor nicht Oben,Sprung
2429 F875 06 00 C ld b,0
2430 F877 ED 43 F6BD C cuih1: ld (xk).bc
2431 F878 CD F6FC C call cursc0
2432 F87E C9 C ret
2433 F87F 2A F691 C cuih0: ld hl,(curse) ;neue Zeilenadr. errechn.
2434 11 0140 C id de,320
2435 F885 A7 C and a ;Carry auf 0 setzen
2436 F886 ED 52 C sbc hl,da
2437 F888 22 F691 C id (curse).hl
2438 F88B 18 EA C jr culhi
2439 F88D 3A F68C C xyein: ld a,(flag)
2440 F890 CD F7 C set 6,a ;folgendes Zeichen or 80 = y Koordin.
2441 F892 C0 FF C set 7,a
2442 F894 32 F68C C id (flag).a
2443 F897 C9 C ret
2444 F898 3A F68C C cuein: ld a,(flag) ;Cursor ein(ca 3*langsammer
2445 F899 CB C7 C set 0,a ;als ohne Cursor)
2446 F89D 32 F68C C id (flag).a
2447 F8A0 C9 C ret
2448 F8A1 3A F68C C cuseus: ld a,(flag) ;Cursor aus
2449 F8A4 CB 87 C res 0,a
2450 F8A6 32 F68C C id (flag).a
2451 F8A9 C9 C ret
2452 F8AA 79 C cuil: ld a,c
2453 F8AB D6 01 C sub 1
2454 F8AD 4F C ld c,e
2455 F8AE 30 06 C jr nc,cu110 ;Sprung wenn Cu.nicht links
2456 F8B0 0E 4F C ld c,4fh ;Cursor zum rechten Rand
2457 F8B2 CD F86F C call cuih ;Cu.eine Zeile hoch
2458 F8B5 C9 C ret
2459 F8B6 ED 43 F6BD C cuil0: ld (xk).bc
2460 F8BA CD F6FC C call cursc0
2461 F8BD C9 C
2462 F8BE 2A F691 C clabs: ld hl,(curse) ;ab Cursor bis Bildende
2463 F8C1 E5 C push hl ;loeschen,
2464 F8C2 CD F835 C call clszc ;erst bis Zeilenende
2465 F8C5 ED 48 F6BD C id bc,(xk)
2466 F8C9 C5 C push bc
2467 F8CA 3E 18 C id a,24
2468 F8CC B8 C cp b ;Sprung wenn
2469 F8CD 28 23 C jr z,clabs2 ;Cu.in der letzten Zeile
2470 F8CF 0E 00 C ld c,0 ;Cu.eine Zeile tiefer
2471 F8D1 CD F7F4 C call if ;und zum linken Rand
2472 F8D4 3A F68E C id a.(yk) ;
2473 F8D7 2F C cpl
2474 F8D8 3C C inc a ;Zeilenanzahl bis
2475 F8D9 C6 19 C add a,19h ;zum Bildende
2476 F8D8 47 C ld b,a
2477 F8DC 11 0140 C ld de,320
2478 F8DF 21 0000 C ld hl,0
2479 F8E2 19 C clabs0: add hl,de ;Anzahl der zu loeschenden
2480 F8E3 10 FD C djnz clabs0 ;Worte bilden
2481 F8E5 22 F67E C clabs1: id (laeng).hl ;und Ausgabebereich ein-
2482 F8E8 21 F673 C id hl,zcls2 tragen
2483 F8EB CD F92D C call ioup
2484 F8EE ED 53 F67E C id (laeng).de ;Laenge zum Zeilenloeschen
2485 F8F2 C1 C clabs2: pop bc ;alte Cursoradr.
2486 F8F3 ED 43 F6BD C ld (xk).bc ;wieder ausgeben
2487 F8F7 E1 C pop hl
2488 F8FB 22 F691 C ld (curse).hl
2489 F8FB CD F6FC C call cursc0
2490 F8FE C9 C ret
2491 C ===== ;Tabelle der Bildschirmsteuerzeichen und Adressen dazu
2492 C ;ptab: db 01h
2493 F8FF 01 C dw ghome
2494 F900 F7D6 C db 02h
2495 F902 07 C dw ton
2496 F903 E2DF C db 08h
2497 F905 08 C dw cul1
2498 F906 F8AA C db 09h
2499 F908 0A C dw lf
2500 F909 F7F4 C db 0Ch
2501 F90B 0C C dw bloch

```

```

2503 F90E DD C db Odh
2504 F90F F82B C dw cr
2505 F911 16 C db 16h
2506 F912 F835 C dw clscz ;loescht ab Cu Zeile
2507 F914 15 C db 15h
2508 F915 F858 C dw cuir
2509 F917 14 C db 14h
2510 F918 F88E C dw clbs ;loeschen ab Cu ges Bildsch
2511 F91A 18 C db 18h
2512 F91B F862 C dw clxi ;loeschen Zeile ,Cu nach links
2513 F91D 1A C db 1ah
2514 F91E F86F C dw cuhn
2515 F920 1B C db ibn
2516 F921 F88D C dw xyein
2517 F923 82 C db 82h ;Es ist darauf zu achten,dass
2518 F924 F898 C dw cuein ;ibn-Folgen beendet sind, da
2519 F926 83 C db 83h ;sonst diese Kommandos als Bild-
2520 F927 F8A1 C dw cuas ;koordinaten gewertet werden!
2521 F929 7F C db 7fh
2522 F92A F7E9 C dw del
2523 F92C FF C db 0ffh ;Ende der Tab
2524 ;Prg. zur Ausgabe zum GDC
2525 ;(HL)=Datenbereich ;(Laenge,Kommando,Parameter)
2526 F92D 46 C ioup: ld b,(hl)
2527 F92E 04 C inc b
2528 F92F C8 C ret z ;wenn (hl)=ffh dann Sprung
2529 F930 05 C dec b
2530 F931 23 C inc hl
2531 F932 0E 8F C ld c,gddc
2532 F934 ED A3 C puti
2533 F936 F5 C push af
2534 F937 DB 8E C ioup0: in a,(gdcc) ;Test ob letzte Operation ausge-
2535 F939 E6 04 C end 4 ;fuehrt wurde
2536 F93B 28 FA C jr z,ioup0
2537 F93D F1 C pop af
2538 F93E 28 ED C jr z,ioup
2539 F940 DD C dec c
2540 F941 DB 8E C ioup1: in a,(gdcc) ;Test ob FIFO-Puffer nicht
2541 F943 E6 02 C and 2 ;voll ist
2542 F945 20 FA C jr nz,ioup1
2543 F947 ED A3 C outi
2544 F949 20 F6 C jr nz,ioup1
2545 F94B 18 EO C jr ioup
2546 ;=====
2547 ;Unterprogramm Punkt setzen/ruecksetzen
2548 F94D F5 C pixel: push af
2549 F94E E5 C push hl
2550 F94F C5 C push bc
2551 F950 DS C push de
2552 F951 DD E5 C push ix
2553 F953 CD FAF8 C call upcup
2554 F956 21 0000 C ld hl,0h
2555 F959 22 F62F C ld (drich),hl
2556 F95C 22 F631 C ld hl,dwdat
2557 F95F 21 F622 C id hl,dwdat
2558 F962 CD F920 C call ioup
2559 F965 C3 F6B9 C jp ende
2560 ;Programm zum Zeichnen von Strecken entsp.x1 y1 x2 y2
2561 F968 F5 C draw: push af
2562 F969 E5 C push hl
2563 F96A C5 C push bc
2564 F96B DS C push de
2565 F96C DD E5 C push ix
2566 F96E CD FA34 C call upkpt
2567 F971 CD FA8D C call upprg
2568 F974 CD FAF8 C call upcup
2569 F977 21 F622 C id hl,dwdat
2570 F97A CD F920 C call ioup
2571 F97D C3 F6B9 C jp ende
2572 ;Zeichnung eines Rechtecks x1,y1 Bezugskoordin.
2573 ;x2,y2 entsprechen Seitenlaenge, rotat Rotation
2574 ;um je 45 grd
2575 F980 F5 C prbox: push af
2576 F981 E5 C push hl
2577 F982 C5 C push bc
2578 F983 DS C push de
2579 F984 DD E5 C push ix
2580 F986 CD FAF8 C call upcup
2581 F989 DD 21 F62F C ld ix,drich
2582 F98D 06 40 C ld b,40h
2583 F98F 3A F61D C ld a.(rotat)
2584 F992 B0 C or b
2585 F993 DD 77 00 C ld (ix),a
2586 F996 DD 36 01 03 C ld (ix+1),3
2587 F99A 2A F61B C ld hl,(y2)
2588 F99D ED 58 F619 C ld de,(x2)
2589 F9A1 2B C dec hl
2590 F9A2 1B C dec de
2591 F9A3 DD 75 03 C ld (ix+3),1
2592 F9A6 DD 74 04 C ld (ix+4),h
2593 F9A9 DD 73 05 C ld (ix+5),e
2594 F9AC DD 72 06 C ld (ix+6),d
2595 F9AF DD 36 07 FF C ld (ix+7),0ffh
2596 F9B3 DD 36 08 3F C ld (ix+8),3fh
2597 F9B7 DD 75 09 C ld (ix+9),l
2598 F9B8 DD 74 0A C ld (ix+0eh),h
2599 F9BD 21 F622 C ld hl,dwdat

```

```

2600 F9C0 CD F92D C call ioup
2601 F9C3 C3 F689 C jp ende
2602 C ;Zeile des aktuellen Cursors in den
2603 C ;RAM-Bereich schreiben
2604 F9C6 F5 C hpr: push af
2605 F9C7 E5 C push hl
2606 F9C8 C5 C push bc
2607 F9C9 D5 C push de
2608 F9CA DD E5 C push ix
2609 F9CC CD F60A C call cursb0
2610 F9CF CD FB37 C call cursa
2611 F9D2 21 F883 C ld hl,cmask
2612 F9D5 CD F92D C call ioup
2613 F9D8 21 F892 C ld hl,figar
2614 F9DB CD F92D C call ioup
2615 F9DE 3E A1 C ld a,0sh
2616 F9E0 D3 8F C out (gcdc).a
2617 F9E2 2A F620 C ld hl,(pber)
2618 F9E5 D6 08 C ld b,8 ;Pixelzeilen
2619 F9E7 C5 C hpri: push bc
2620 F9E8 D6 50 C ld b,xmax/8 ;Zeichen je Zeile
2621 F9EA CD FB2A C call inup
2622 F9ED C1 C pop bc
2623 F9EE 10 F7 C djnz hpr1
2624 F9F0 CD FB37 C call cursa
2625 F9F3 CD F60A C call cursb0
2626 F9F6 C3 F689 C jp ende
2627 C ;Zeile auf Bildschirm schreiben,
2628 C ;es wird die aktuelle Cursorzeile be-
2629 C ;nutzt
2630 F9F9 F5 C hpw: push af
2631 F9FA E5 C push hl
2632 F9FB C5 C push bc
2633 F9FC D5 C push de
2634 F9FD DD E5 C push ix
2635 F9FF CD F60A C call cursb0
2636 FA02 21 F888 C ld hl,syncs
2637 FA05 CD F92D C call ioup
2638 FA08 CD FB37 C call cursa
2639 FA0B 21 F883 C ld hl,cmask
2640 FA0E CD F92D C call ioup
2641 FA11 3E 20 C ld a,20h
2642 FA13 D3 8F C out (gcdc).a
2643 FA15 D6 08 C ld b,8 ;Pixel
2644 FA17 2A F620 C ld hl,(pber)
2645 FA1A C5 C hpw1: push bc
2646 FA1B D6 50 C ld b,xmax/8 ;Zeichen/Zeile
2647 FA1D D6 8E C ld c,gdcc
2648 FA1F CD FB1F C call ioup3
2649 FA22 C1 C pop bc
2650 FA23 10 F5 C djnz hpw1
2651 FA25 21 F898 C ld hl,syncb
2652 FA28 CD F92D C call ioup
2653 FA2B CD FB37 C call cursa
2654 FA2E CD F60A C call cursb0
2655 FA31 C3 F689 C jp ende
2656 C ;Zeichnungsrichtung (8-tel) sowie deltax und delty
2657 C ;fuer flags-Kommando aus x1,y1,x2,y2 errechnen
2658 FA34 2A F619 C upokt: ld hl,(x2)
2660 FA37 ED 4B F615 C ld bc,(xi)
2661 FA3B A7 C and a
2662 FA3C ED 42 C sbc hl,bc
2663 FA3E 38 22 C jr c,a1
2664 FA40 22 F63D C ld (dx),hl
2665 FA43 ED 4B F617 C ld bc,(y1)
2666 FA47 2A F61B C ld hl,(y2)
2667 FA4A A7 C and a
2668 FA4B ED 42 C sbc hl,bc
2669 FA4D 38 06 C jr c,a2
2670 FA4F 22 F63F C ld (dy),hl
2671 FA52 3E 03 C ld a,3
2672 FA54 C9 C ret
2673 FA55 E5 C a2: push hl
2674 FA56 C1 C pop bc
2675 FA57 21 0000 C ld hl,0
2676 FA5A A7 C and a
2677 FA5B ED 42 C sbc hl,bc
2678 FA5D 22 F63F C ld (dy),hl
2679 FA60 AF C xor a
2680 FA61 C9 C ret
2681 FA62 E5 C m1: push hl
2682 FA63 C1 C pop bc
2683 FA64 21 0000 C ld hl,0
2684 FA67 A7 C and a
2685 FA68 ED 42 C sbc hl,bc
2686 FA6A 22 F63D C ld (dx),hl
2687 FA6D ED 4B F617 C ld bc,(y1)
2688 FA71 2A F61B C ld hl,(y2)
2689 FA74 A7 C and a
2690 FA75 ED 42 C sbc hl,bc
2691 FA77 38 06 C jr c,a3
2692 FA79 22 F63F C ld (dy),hl
2693 FA7C 3E 02 C ld a,2
2694 FA7E C9 C ret
2695 FA7F E5 C m3: push hl
2696 FA80 C1 C pop bc

```

```

2697 FAB1 21 0000 C id hl,0
2698 FAB4 A7 C and s
2699 FAB5 ED 42 C sbc hl,bc
2700 FAB7 22 F63F C ld {dy},hl
2701 FABA 3E 01 C ld s,i
2702 FABC C9 C ret
2703 C ;figs-Kommando fuer Zeichenoperation vorbereiten,
2704 ;spezifische Kommandodaten errechnen
2705 FABD ED 4B F63D C upprg: ld bc,(dx)
2706 F9A1 2A F63F C ld hl,(dy)
2707 F9A4 A7 C and s
2708 F9A5 ED 42 C sbc hl,bc
2709 F9A7 2A F63D C ld hl,(dx)
2710 F9A9 ED 58 F63F C ld de,(dy)
2711 F9AE 38 03 C jr c,a4
2712 F9A0 EB C ex de,hl
2713 F9A1 C6 04 C add s,a
2714 F9A3 ED 53 F63D C m4: ld (dx),de
2715 F9A7 22 F63F C ld (dy),hl
2716 F9AA 21 FBA1 C ld hl,okti
2717 F9AD 85 C add s,l
2718 F9AE 30 01 C jr nc,m5
2719 F9B0 24 C inc h
2720 F9B1 6F C m5: ld l,a
2721 F9B2 7E C ld s,(hl)
2722 F9B3 F6 08 C or 8
2723 F9B5 32 F62F C ld (drich),s
2724 F9B8 2A F63F C ld hl,(dy)
2725 F9B9 7C C ld s,h
2726 F9BC E6 3F C and 3fh
2727 F9BE 67 C ld h,s
2728 F9BF 22 F630 C ld (drich+1),hl
2729 FAC2 ED 5B F63D C ld de,(dx)
2730 FAC6 A7 C and s
2731 FAC7 CB 13 C rl e
2732 FAC9 CB 12 C rl d
2733 FACB EB C ex de,hl
2734 FACC A7 C and s ;theor. Carry nicht gesetzt
2735 FACD ED 52 C sbc hl,de
2736 FACF 7C C ld s,h
2737 FADO E6 3F C and 3fh
2738 FA02 67 C ld h,s
2739 FA03 22 F632 C ld (drich+3),hl
2740 FA06 2A F63D C ld hl,(dx)
2741 FA09 ED 58 F63F C ld de,(dy)
2742 FA0D A7 C and s
2743 FA0E ED 52 C sbc hl,de
2744 FA0F A7 C and s
2745 FAE1 CB 15 C rl l
2746 FAE3 CB 14 C rl h
2747 FAE5 7C C ld s,h
2748 FAE6 E6 3F C and 3fh
2749 FAE8 67 C ld h,s
2750 FAE9 22 F634 C ld (drich+5),hl
2751 FAEC 2A F630 C ld hl,(dx)
2752 FAEF A7 C and s
2753 FAFO CB 15 C rl l
2754 FAF2 CB 14 C rl h
2755 FAF4 22 F636 C ld (drich+7),hl
2756 FAF7 C9 C ret
2757 C ;aus xi,yi Cursoradr. errechnen und in Ausgabe-
2758 ;datenbereich eintragen
2759 FAF8 2A F617 C upcup: ld hl,(y1)
2760 FAFB 29 C upcup1: add hl,hl
2761 F AFC 29 C add hl,hl
2762 FAFD 29 C add hl,hl
2763 FAFE E5 C push hl
2764 FAFF 29 C add hl,hl
2765 FB00 29 C add hl,hl
2766 FB01 D1 C pop de
2767 FB02 19 C add hl,de
2768 FB03 ED 58 F615 C ld de,(x1)
2769 FB07 AF C xor s
2770 FB08 06 04 C ld b,4
2771 FB0A CB 1A C m6: rr d
2772 FB0C CB 1B C rr e
2773 FB0E CB 1F C rr a
2774 FB10 10 F8 C djnz m6
2775 FB12 19 C add hl,de
2776 FB13 ED 5B F6BF C ld de,(cursor)
2777 FB17 19 C add hl,de ;errechnete Adr.+Bildanf..
2778 FB18 22 F62A C ld (dpos),hl
2779 FB19 32 F62C C ld (dpix),s
2780 FB1E C9 C ret
2781 FB1F DB 8E C ioup3: in s,(gdcc)
2782 FB21 E6 02 C and 2
2783 FB23 20 FA C jr nz,ioup3
2784 FB25 ED A3 C outi
2785 FB27 20 F6 C jr nz,ioup3
2786 FB29 C9 C ret
2787 FB2A 0E 8F C inup3: in s,(80h)
2788 FB2C DB 80 C inup1: in s,(80h)
2789 FB2E E6 01 C and 1
2790 FB30 28 FA C jr z,inup1
2791 FB32 ED A2 C ini
2792 FB34 20 F6 C jr nz,inup1
2793 FB36 C9 C ret

```

```

2794
2795 FB37 2A F691 C curse: ld hl,(curse)
2796 FB3A 22 F66F C ld (pos),hl
2797 FB3D AF C xor a
2798 FB3E 32 F671 C ld (pix),a
2799 FB41 21 F66B C ld hl,zcsl
2800 FB44 CD F920 C call ioup
2801 FB47 C9 C ret

2802 C ;SUBROUTINE VERZWEIGUNG
2803 C ;IN: ZEICHEN IN A
2804 C ;TABELLENADRESSE IN HL
2805 C ;OUT: CY=0:HL=ADR,CY=1:Nicht gefunden
2806 C
2807 FB48 D5 C FCGA1: PUSH DE
2808 FB49 5F C LD E,A ;ZEICHEN
2809 FB4A 7E C FCGA1: LD A,(HL)
2810 FB4B FE FF C CP 0fh ;tabellende
2811 FB4D 28 11 C JR Z,FCGA3 .
2812 FB4F BB C CP E
2813 FB50 28 05 C JR Z,FCGA2 ;gefunden
2814 FB52 23 C INC HL
2815 FB53 23 C INC HL
2816 FB54 23 C INC HL
2817 FB55 18 F3 C JR FCGA1
2818 FB57 7B C FCGA2: LD A,E
2819 FB58 23 C INC HL
2820 FB59 5E C LD E,(HL)
2821 FB5A 23 C INC HL
2822 FB5B 56 C LD D,(HL)
2823 FB5C 23 C INC HL
2824 FB5D EB C EX DE,HL
2825 FB5E 18 02 C JR FCGA4 ;nicht gefunden
2826 FB60 37 C FCGA3: SCF
2827 FB61 7B C LD A,E
2828 FB62 D1 C FCGA4: POP DE
2829 FB63 C9 C RET

2830 C
2831 C
2832 C ;Initialisierungsparameter fuer den GDC (Länge, Kommando, Parameter...)
2833 C ;Daten koennen auch in ROM stehen
2834 FB64 09 00 12 26 C inid: defm 9,0,12h,26h ;Reset GDC und Bildschirm-
2835 FB65 48 1D 07 37 C defm 48h,1dh,7h,37h ;parameter 640*200 Pixel
2836 FB6C C8 90 C defm Dc8h,98h ;kein Refresh, Graphikbetrieb
2837 FB6E 01 6F C defm 1,6h ;Vertikalsync. durch GDC ein
2838 FB70 09 70 00 00 C defm 9,70h,0,0 ;PRAM auf 640*200 festlegen
2839 FB74 80 0C FF FF C defm 80h,0ch,Offh,Offh
2840 FB78 FF FF C defm 0fh,0fh
2841 FB7A 02 47 28 C defm 2,47h,28h ;Pitch (Worte pro Zeile)
2842 FB7D 02 46 00 C defm 2,46h,00 ;Zoom 00
2843 FB80 01 6B FF C defm 1,6bh,Offh ;Bildschirmsfreigabe
2844 C
2845 FB83 03 4A FF FF C :Zeichenparameter
2846 FB87 FF C cmask: db 3,4ch,Offh,Offh,Offh
2847 FB88 02 0E 10 C
2848 FB88 02 73 4F C syncs: db 2,0eh,10h
2849 FB8E 02 4C 02 FF C mpram: db 2,73h,4fh
2850 FB92 04 4C 02 40 C figsw: db 2,4ch,02h,Offh
2851 FB96 01 FF C figar: db 4,4ch,02h,040h,1,Offh
2852 FB98 02 0E 12 C
2853 FB9B 02 73 0F 01 C syncb: db 2,0eh,12h
2854 FB9F 00 FF C gpram: db 2,73h,0fh,1,0dh,Offh
2855 FBAA 02 05 06 01 C okti: db 2,5,6,1,3,4,7,0
2856 FBAA 03 04 07 00 C
2857 C
2858 C include GDCZG2.MAC
2859 FBAA 00 00 00 00 C zgenf: db 0,0,0,0,0,0,0
2860 FBAD 00 00 00 00 C
2861 FB81 00 00 00 5F C db 0,0,0,5fh,5fh,0,0,0
2862 FB85 5F 00 00 00 C
2863 FB89 00 00 07 07 C db 0,0,7,7,0,7,7,0
2864 FB8D 00 07 07 00 C
2865 FB8C 00 14 7F 7F C db 0,14h,7fh,7fh,14h,7fh,7fh,14h
2866 FB85 14 7F 7F 14 C
2867 FB89 00 12 2A 7F C db 0,12h,2ah,7fh,7fh,2ah,24h,0
2868 FB8D 7F 2A 24 00 C
2869 FBBD 00 62 66 0C C db 0,62h,66h,0ch,18h,30h,66h,46h
2870 FBBD 18 30 66 46 C
2871 FBBD 00 48 7A 37 C db 0,48h,7eh,37h,5dh,4fh,7ah,30h
2872 FBBD 50 4F 7A 30 C
2873 FB81 00 00 00 03 C db 0,0,0,3,7,4,0,0
2874 FB85 07 04 00 00 C
2875 FB89 00 00 41 63 C db 0,0,41h,63h,3eh,1ch,0,0
2876 FBED 3E 1C 00 00 C
2877 FBF1 00 00 1C 3E C db 0,0,1ch,3eh,63h,41h,0,0
2878 FBF5 63 41 00 00 C
2879 FBF9 08 2A 3E 1C C db 8,2ah,3eh,1ch,1ch,3eh,2ah,8
2880 FBFD 1C 3E 2A 08 C
2881 FC01 00 08 08 3E C db 0,8,8,3eh,3eh,8,8,0
2882 FC05 3E 08 08 00 C
2883 FC09 00 00 00 60 C db 0,0,0,60h,0e0h,80h,0,0
2884 FC0D E0 80 00 00 C
2885 FC11 00 08 08 08 C db 0,8,8,8,8,8,8,0
2886 FC15 08 08 08 00 C
2887 FC19 00 00 00 60 C db 0,0,0,60h,60h,0,0,0
2888 FC1D 60 00 00 00 C
2889 FC21 00 01 03 06 C db 0,1,3,6,0ch,18h,30h,60h
2890 FC25 NC 18 30 60 C

```

2891	FC29	00 3E 7F 45	C	db	0,3eh,7fh,45h,49h,51h,7fh,3eh
2892	FC2D	49 51 7F 3E	C	db	0,40h,40h,7fh,7fh,42h,40h,0
2893	FC31	00 40 40 40	C	db	0,66h,6fh,49h,49h,7bh,72h,0
2894	FC35	7F 42 40 00	C	db	0,36h,7fh,49h,49h,63h,22h,0
2895	FC39	00 66 6F 49	C	db	0,50h,7fh,7fh,53h,16h,1ch,18h
2896	FC3D	49 78 72 00	C	db	0,33h,79h,49h,49h,6fh,2fh,0
2897	FC41	00 36 7F 49	C	db	0,32h,7bh,49h,49h,7fh,3eh,0
2898	FC45	49 63 22 00	C	db	0,7,0fh,79h,71h,3,3,0
2899	FC49	00 50 7F 7F	C	db	0,36h,7fh,49h,49h,7fh,36h,0
2900	FC4D	53 16 1C 18	C	db	0,3eh,7fh,49h,49h,6fh,26h,0
2901	FC51	00 33 79 49	C	db	0,0,0,6ch,6ch,0,0,0
2902	FC55	49 6F 2F 00	C	db	0,0,0,6ch,0ech,80h,0,0
2903	FC59	00 32 78 49	C	db	0,0,41h,63h,36h,1ch,8,0
2904	FC5D	49 7F 3E 00	C	db	0,24h,24h,24h,24h,24h,0
2905	FC61	00 07 0F 79	C	db	0,0,8,1ch,36h,63h,41h,0
2906	FC65	71 03 03 00	C	db	0,6,0fh,59h,51h,7,6,0
2907	FC69	00 36 7F 49	C	db	0,1eh,5fh,5dh,5dh,41h,7fh,3ch
2908	FC6D	49 7F 36 00	C	db	0,7ch,7eh,13h,13h,7eh,7ch,0
2909	FC71	00 3E 7F 49	C	db	0,36h,7fh,49h,49h,7fh,7fh,41h
2910	FC75	49 6F 26 00	C	db	0,22h,63h,41h,41h,63h,3eh,1ch
2911	FC79	00 00 00 6C	C	db	0,1ch,3eh,63h,41h,7fh,7fh,41h
2912	FC7D	6C 00 00 00	C	db	0,63h,41h,5dh,40h,7fh,7fh,41h
2913	FC81	00 00 00 6C	C	db	0,3,1,1dh,40h,7fh,7fh,41h
2914	FC85	EC 80 00 00	C	db	0,72h,73h,51h,41h,63h,3eh,1ch
2915	FC89	00 00 41 63	C	db	0,7fh,7fh,8,8,7fh,7fh,0
2916	FC8D	36 1C 08 00	C	db	0,41h,41h,7fh,7fh,41h,41h,0
2917	FC91	00 24 24 24	C	db	0,41h,41h,7fh,7fh,41h,41h,0
2918	FC95	24 24 24 00	C	db	0,1,3fh,7fh,41h,40h,70h,30h
2919	FC99	00 00 01 1C	C	db	0,63h,77h,1ch,8,7fh,7fh,41h
2920	FC9D	36 63 41 00	C	db	0,70h,60h,40h,41h,7fh,7fh,41h
2921	FCA1	00 06 0F 59	C	db	0,7fh,7fh,0eh,1ch,0eh,7fh,7fh
2922	FCA5	51 07 06 00	C	db	0,7fh,7fh,18h,0ch,6,7fh,7fh
2923	FCA9	00 1E 5F 50	C	db	0,1ch,3eh,63h,41h,63h,3eh,1ch
2924	FCA0	50 41 7F 3E	C	db	0,6,0fh,9,49h,7fh,7fh,41h
2925	FCB1	00 7C 7E 13	C	db	0,5ch,6eh,33h,51h,43h,7eh,3ch
2926	FCB5	13 7E 7C 00	C	db	0,66h,7fh,19h,9,7fh,7fh,41h
2927	FCB9	00 36 7F 49	C	db	0,32h,7fh,49h,49h,6fh,26h,0
2928	FCBD	49 7F 41 01	C	db	0,3,41h,7fh,41h,40h,70h,30h
2929	FCC1	00 22 63 41	C	db	0,1,1fh,31h,60h,60h,31h,1fh,0
2930	FCC5	41 63 3E 1C	C	db	0,7fh,7fh,0eh,1ch,0eh,7fh,7fh
2931	FCC9	00 1C 3E 63	C	db	0,6,0fh,9,49h,7fh,7fh,41h
2932	FCC0	41 7F 7F 41	C	db	0,7fh,7fh,18h,0ch,6,7fh,7fh
2933	FCD1	00 63 41 50	C	db	0,1ch,3eh,63h,41h,63h,3eh,1ch
2934	FCD5	49 7F 7F 41	C	db	0,6,0fh,9,49h,7fh,7fh,41h
2935	FCD9	00 03 01 10	C	db	0,1ch,3eh,63h,41h,63h,3eh,1ch
2936	FCDD	49 7F 7F 41	C	db	0,72h,73h,51h,41h,63h,3eh,1ch
2937	FCE1	00 72 73 51	C	db	0,7fh,7fh,8,8,7fh,7fh,0
2938	FCE5	41 63 3E 1C	C	db	0,41h,41h,7fh,7fh,41h,41h,0
2939	FCE9	00 7F 7F 00	C	db	0,1,3eh,63h,41h,7fh,7fh,41h
2940	FCEO	08 7F 7F 00	C	db	0,41h,41h,7fh,7fh,41h,41h,0
2941	FCF1	00 41 41 7F	C	db	0,1,3eh,63h,41h,7fh,7fh,41h
2942	FCF5	7F 41 41 00	C	db	0,6,0fh,9,49h,7fh,7fh,41h
2943	FCF9	00 01 3F 7F	C	db	0,1,3fh,7fh,41h,40h,70h,30h
2944	FCFD	41 40 70 30	C	db	0,63h,77h,1ch,8,7fh,7fh,41h
2945	FD01	00 63 77 1C	C	db	0,70h,60h,40h,41h,7fh,7fh,41h
2946	FD05	08 7F 7F 41	C	db	0,7fh,7fh,0eh,1ch,0eh,7fh,7fh
2947	FD09	00 70 60 40	C	db	0,7fh,7fh,18h,0ch,6,7fh,7fh
2948	FD00	41 7F 7F 41	C	db	0,1ch,3eh,63h,41h,63h,3eh,1ch
2949	FD11	00 7F 7F 0E	C	db	0,6,0fh,9,49h,7fh,7fh,41h
2950	FD15	1C 0E 7F 7F	C	db	0,7fh,7fh,18h,0ch,6,7fh,7fh
2951	FD19	00 7F 7F 18	C	db	0,1ch,3eh,63h,41h,63h,3eh,1ch
2952	FD1D	0C 06 7F 7F	C	db	0,6,0fh,9,49h,7fh,7fh,41h
2953	FD21	00 01 3E 63	C	db	0,7fh,7fh,0eh,1ch,0eh,7fh,7fh
2954	FD25	41 63 3E 1C	C	db	0,6,0fh,9,49h,7fh,7fh,41h
2955	FD29	00 06 0F 09	C	db	0,5ch,6eh,33h,51h,43h,7eh,3ch
2956	FD2D	49 7F 7F 41	C	db	0,66h,7fh,19h,9,7fh,7fh,41h
2957	FD31	00 5C 6E 33	C	db	0,32h,7fh,49h,49h,6fh,26h,0
2958	FD35	51 43 7E 3C	C	db	0,3,41h,7fh,41h,40h,70h,30h
2959	FD39	00 60 7F 19	C	db	0,3,41h,7fh,41h,40h,70h,30h
2960	FD3D	09 7F 7F 41	C	db	0,7fh,7fh,0eh,1ch,0eh,7fh,7fh
2961	FD41	00 32 78 49	C	db	0,7fh,7fh,18h,0ch,6,7fh,7fh
2962	FD45	49 6F 26 00	C	db	0,1,3eh,63h,41h,7fh,7fh,41h
2963	FD49	00 03 41 7F	C	db	0,3,41h,7fh,41h,40h,70h,30h
2964	FD4D	7F 41 03 00	C	db	0,3,41h,7fh,41h,40h,70h,30h
2965	FD51	00 3F 7F 40	C	db	0,1,1fh,31h,60h,60h,31h,1fh,0
2966	FD55	40 7F 3F 00	C	db	0,7fh,7fh,0eh,1ch,0eh,7fh,7fh
2967	FD59	00,1F 3F 60	C	db	0,6,0fh,9,49h,7fh,7fh,41h
2968	FD5D	60 3F 1F 00	C	db	0,7fh,7fh,0eh,1ch,0eh,7fh,7fh
2969	FD61	00 7F 7F 30	C	db	0,7fh,7fh,30h,18h,30h,7fh,7fh
2970	FD65	18 30 7F 7F	C	db	0,61h,73h,1eh,0ch,1eh,73h,61h
2971	FD69	00 61 73 1E	C	db	0,7,4fh,78h,78h,4fh,7,0
2972	FD6D	0C 1E 73 61	C	db	0,73h,67h,4dh,59h,71h,63h,47h
2973	FD71	00 07 4F 78	C	db	0,0,41h,41h,7fh,7fh,0,0,0
2974	FD75	78 4F 07 00	C	db	0,0,41h,41h,7fh,7fh,41h,41h,0
2975	FD79	03 73 67 4D	C	db	0,4,6,7fh,7fh,6,4,0
2976	FD7D	59 71 63 47	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2977	FD81	00 00 41 41	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2978	FD85	7F 7F 00 00	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2979	FD89	80 80 80 80	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2980	FD8D	80 80 80 80	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2981	FD91	00 00 07 7F	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2982	FD95	41 41 00 00	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2983	FD99	00 04 06 7F	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2984	FD9D	7F 06 04 00	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2985	FDA1	C0 C0 C0 C0	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2986	FDA5	C0 C0 C0 C0	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h
2987	FDA9	00 00 04 06	C	db	0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h,0c0h

2988	F0AD	03 01 00 00	C		
2989	F0B1	00 40 78 3C	C	db	0,40h,78h,3ch,54h,54h,74h,20h
2990	F0B5	54 54 74 20	C	db	0,38h,7ch,44h,44h,3fh,7fh,41h
2991	F0B9	00 38 7C 44	C	db	0,28h,6ch,44h,44h,7ch,38h,0
2992	F0BD	44 7F 41	C	db	0,40h,7fh,3fh,45h,44h,7ch,38h
2993	F0C1	00 28 6C 44	C	db	0,18h,5ch,54h,54h,7ch,38h,0
2994	F0C5	44 7C 38 00	C	db	0,2,3,49h,7fh,7fh,48h,0
2995	F0C9	00 40 7F 3F	C	db	0,7ch,0fch,0a4h,0a4h,0bch,98h,0
2996	F0CD	45 44 7C 38	C	db	0,78h,7ch,4,8,7fh,7fh,41h
2997	F0D1	00 18 5C 54	C	db	0,0,40h,7dh,7dh,44h,0,0
2998	F0D5	54 7C 38 00	C	db	0,7dh,0fch,84h,80h,0e0h,60h,0
2999	F0D9	00 02 03 49	C	db	0,44h,6ch,38h,10h,7fh,7fh,41h
3000	F0DD	7F 7E 48 00	C	db	0,0,40h,7fh,7fh,41h,0,0
3001	FDE1	03 7C FC A4	C	db	0,8,0ch,4,4ch,78h,7ch,44h
3002	FDE5	A4 8C 98 00	C	db	0,3ch,7ch,24h,0a4h,0fch,84h
3003	FDE9	00 78 7C 04	C	db	0,84h,0fch,0fch,0a4h,24h,3ch,18h
3004	FDED	08 7F 7F 41	C	db	0,8,0ch,4,4ch,78h,7ch,44h
3005	FDF1	00 40 70	C	db	0,20h,74h,54h,54h,5ch,48h,0
3006	FDF5	7D 44 00 00	C	db	0,0,40h,7dh,7dh,44h,0,0
3007	FDF9	00 7D FD 84	C	db	0,1ch,3ch,60h,60h,3ch,1ch,0
3008	FDFD	80 E0 60 00	C	db	0,3ch,7ch,60h,38h,60h,7ch,3ch
3009	FE01	00 44 6C 38	C	db	0,44h,6ch,38h,10h,7fh,7fh,41h
3010	FE05	10 7F 7F 41	C	db	0,78h,7ch,4,8,7fh,7fh,41h,0,0
3011	FE09	00 00 40 7F	C	db	0,78h,7ch,4,4,78h,7ch,4
3012	FE0D	7F 41 00 00	C	db	0,38h,7ch,44h,44h,7ch,38h,0
3013	FE11	00 78 7C 0C	C	db	0,18h,3ch,24h,0a4h,0fch,84h
3014	FE15	3C 0C 7C 78	C	db	0,8,0ch,4,4ch,78h,7ch,44h
3015	FE19	00 78 7C 04	C	db	0,3ch,7ch,24h,0a4h,0fch,84h
3016	FE1D	04 78 7C 04	C	db	0,8,0ch,4,4ch,78h,7ch,44h
3017	FE21	00 38 7C 44	C	db	0,20h,74h,54h,54h,5ch,48h,0
3018	FE25	44 7C 38 00	C	db	0,20h,64h,44h,7fh,3fh,4,0
3019	FE29	00 18 3C 24	C	db	0,7ch,40h,40h,7ch,3ch,0
3020	FE2D	A4 F8 FC 84	C	db	0,1ch,3ch,60h,60h,3ch,1ch,0
3021	FE31	00 84 FC F8	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3022	FE35	A4 24 3C 18	C	db	0,1,3,2,3,1,3,2
3023	FE39	00 08 0C 04	C	db	0,41h,41h,77h,3eh,8,8,0
3024	FE3D	4C 78 7C 44	C	db	0,0,0,7fh,7fh,0,0,0
3025	FE41	00 20 74 54	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3026	FE45	54 5C 48 00	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3027	FE49	00 20 64 44	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3028	FE4D	7F 3F 04 00	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3029	FE51	00 7C 7C 40	C	db	0,7ch,4,4,78h,7ch,4
3030	FE55	40 7C 3C 00	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3031	FE59	00 1C 3C 60	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3032	FE5D	60 3C 1C 00	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3033	FE61	00 3C 7C 60	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3034	FE65	38 60 7C 3C	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3035	FE69	00 44 6C 38	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3036	FE6D	10 38 6C 44	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3037	FE71	00 7C FC A0	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3038	FE75	A0 BC 9C 00	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3039	FE79	00 64 4C 5C	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3040	FE7D	74 64 4C 00	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3041	FE81	00 41 41 77	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3042	FE85	3E 08 00 00	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3043	FE89	00 00 00 7F	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3044	FE8D	7F 00 00 00	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3045	FE91	00 08 08 3E	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3046	FE95	77 41 41 00	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3047	FE99	00 01 03 02	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3048	FE9D	03 01 03 02	C	db	0,64h,4ch,5ch,74h,64h,4ch,0
3049	FEA1	FF FF FF FF	C	db	0ffh,0ffh,0ffh,0ffh,0ffh,0ffh,0ffh,0ffh
3050	FEA5	FF FF FF FF	C	db	0ffh,0ffh,0ffh,0ffh,0ffh,0ffh,0ffh,0ffh
3051			C	lgdc	equ
				.dephase	
				endif	
				end	

ADRE	539	543	547#		
ALL0	117	2096#	2097		
ALL1	125	2098#	2099		
ALL2	133	2100#	2101		
ALLOC	404	411	416	421	435#
ASIO	911#	914			
AUSF	1218	1220	1222#		
AUSF1	1232	1234#			
AUSF2	1236	1239	1242#		
AUSF3	1243#				
AUSG8	1089	1283#			
AUSGE	1290	1293#			
B3264	1231#	1633			
BBS	2017#	2071			
BCC0	1904	1910#			
BCR	2001#	2047	2050	2071	
BCR1	2003#	2007			
BOEL	2016#	2073			
BDOS	23#	303			
BEEP	1219#	1630			
BELL	2068#	2076			
BIOS	19#	23	24	83	88
BLF	2010#	2071			
BLHM	2023#	2072			
BLKSIZE	70#	394			
BLOEM	2380#	2502			
BOOT	90	295#			
BOOT1	339	355#			

	90#										
BOT	26#	197	1929	1931	1932	1937	1959	2002	2010		
BRAM	2017	2023	2024	2038	2040	2044					
BREAK	897	911	992#								
BS1CO	890	1900#									
BTSTAB	1977	1990	2080#	2084							
CBAWF	82	86#									
CCI	93#	713									
CCO	94#										
CCP	24#	321	355	361	372						
CFDC	56#	598	674	694	722	759					
CHKO	116	2097#	2098								
CHK1	124	2099#	2100								
CHK2	132	2101#	2102								
CHKUNA	392	402#									
CHOME	2038#	2072									
CLSBS	2462#	2510									
CLSBS0	2479#	2480									
CLSBS1	2481#										
CLSBS2	2469	2485#									
CLSPDOS	2159#	2405									
CLSZC	2396#	2464	2506								
CLSZC0	2154#	2406									
CLSZC2	2400	2405#									
CLSZC3	2398	2402	2410#								
CLSZT	2420#	2512									
CNAHK	2611	2639	2845#	966							
CNTR	14#	839									
CO10	2049#	2075									
CO11	2060#	2076									
CO13	2063#	2077									
CO15	1910	1942#									
COEIN	2444#	2518									
CUP	2040#	2073									
CURS	197#	1912	1942								
CURSO	2179#	2289	2297	2330	2362	2388	2776				
CURSA	2180#	2245	2274	2282	2336	2363	2433	2437	2462		
CURSS	2610	2624	2638	2653	2795#						
DCURS	2137#										
DEL	2366#	2522									
DEL1	736#	736									
DELAY	623	707	721	734#	740						
DFDC	57#	601	656	727	742						
DFIGS	2140#										
DIRBF	114	122	130	2102#	2103						
DHAADR	193#	290	500	570							
DPBA	115	123	138#								
DPBAS	108#	246									
DPBB	148#										
DPBC	131	157#									
DPHA	110#										
DPHB	118#										
DPHC	126#										
OPIX	2139#	2779									
DPOS	2138#	2778									
DRAM	2136#										
DRAW	2121	2561#									
DREAD	103	330	375#								
DRICH	2141#	2555	2556	2581	2723	2728	2739	2750	2755		
DSKBUF	194#	655									
DSKIN1	576#	586									
DSKIN2	580#	581									
DSKINI	307	573#									
DTL	175#										
DWDAT	2135#	2557	2569	2599							
DX	2145#	2664	2686	2705	2709	2714	2729	2740	2751		
DY	2146#	2670	2678	2692	2700	2706	2710	2715	2724		
	2741										
END\$F	1874#										
ENDE	2210#	2359	2559	2571	2601	2626	2655				
ENTPR	1142#	1149									
EOT	173#										
ERFLAG	441	523	531	665	2093#	2094					
ERM	767#										
ES10	897#	900									
F\$	1259	1265	1272#								
F\$1	1276#	1281									
F2MHZ	1237#	1631									
F4MHZ	1240#	1632									
FATAB	1628#	1883	1885								
FCGA	2352	2807#									
FCGA1	2809#	2817									
FCGA2	2613	2818#									
FCGA3	2811	2826#									
FCGA4	2825	2828#									
FIGR	2613	2850#									
FIGSW	2849#										
FILHST	456	475	477#								
FTKZU	1186	1192#									
FLAG	2176#	2231	2317	2325	2342	2439	2442	2444	2446		
FLGIO	2448	2450									
	202#										

FTAB	1616#	1883	1884						
FTAPO	1200	1885#							
FTP0I	1192	1197	1884#						
FZUE	1165#	1210							
GADR	537#	562	564	567	575	578	580	584	
GC	12#	860	2105						
GDCANF	861	2106#							
GDCC	43#	857	2534	2540	2647	2781			
GDCO	2119#								
GDCD	44#	2531	2616	2642	2787				
GDCINI	865	2118#	2181	3052					
GDCRAM	25#	862	889	2108					
GHOME	2360#	2391	2494						
GPL	174#								
GPRAM	2853#								
GRF	1217#	1628							
HEAD	185#								
HEADS	64#	775#							
HED	170#	786	793						
HOME	98	279#							
HP	2119	2197#							
HP1	2207	2217#	2239	2368	2412				
HPR	2123	2604#							
HPR1	2619#	2623							
HPW	2124	2630#							
HPW1	2645#	2650							
HSTACT	311	365	452	2087#	2088				
HSTBLK	73#	74	75						
HSTBUF	194	494	2103#	2104					
HSTDOSK	187#	459	479	778					
HSTSEC	171#	469	483						
HSTSIZ	71#	73							
IOPU	2194	2228	2257	2285	2300	2303	2305	2383	2390
IOPU	2407	2422	2483	2526#	2538	2545	2558	2570	2600
IOPU	2612	2614	2637	2640	2652	2800			
IOUPO	2534#	2536							
IOUP1	2540#	2542	2544						
IOUP3	2648	2781#	2783	2785					
IRDY	741	759#	761						
J100	1215#	1636							
JPTAB	2351	2493#							
K7652	1023#	1383							
K7659	1022#	1612							
KDPIE	356	360	2104#						
LAENG	2169#	2381	2385	2481	2484				
LATCH	34#	1046	1086	1097	1154	1347	1353	1367	1371
LBSTAB	1978	2084#							
LESE	377	539#							
LF	2371#	2471	2500						
LFTB	1193	1883#							
LGDC	863	3052#							
LGDCRAM	2181#								
LIST	95	923#							
LISTST	105	263#							
LPI0A	872#								
LPI0B	810	874#							
LSI0A	827	878#							
LSI0B	836	881#							
M003	2274#	2301							
M0042	1922#	1989							
M0043	1930#	2012							
M0044	1916	1951#							
M0049	1919	1921	1976#						
M004A	1985	1988#							
M0048	1980	1990#							
M004E	2047#	2075							
M004F	2054#	2057							
M0050	2065#	2077							
MLZ	1049	1079	1096	1099	1181	1244	1274	1335	1877#
MNULL	1273	1278	1299#						
MODE	645	680#							
MOFL	1047	1087	1206	1287	1299	1365	1878#		
MONRET	1228#	1635							
MPO	1140	1150#							
MP1	1101	1112	1126	1139#					
MP2	1152	1154#							
MP21	1195	1249#							
MP3	1156#								
MP4	1159	1170#							
MP5	1151#	1163							
MPE1	1350#	1351							
MPE2	1356#	1357							
MPE3	1341#	1359							
MPE4	1345	1347#							
MPRAM	2848#								
MR0	1103	1116#	1132						
MR1	1117#	1124							
MR2	1118#	1123							
MR3	1121	1127#							
MR4	1128#	1135							
MR5	1129#	1134							
MSG1	205#	308							
MSG2	212#	711							

MSG3	224#	1002						
HSG5	218#							
HSM1	1302	1304	1306	1312#				
MSTA1	1093	1099#						
MTS	1320#							
MT61	1321	1325#						
MTE	1313	1316	1324	1326	1328	1334#		
N	172#							
NOISK	67#	238						
NEXT8	1282	1283	1286	1876#				
NOMATC	461	466	472#					
NOOVF	426	432#						
NROY	701#	710	715					
OKT1	2716	2855#						
P1AC	63#							
P1AD	61#							
P1BC	64#							
P1BD	62#							
PBER	2134#	2617	2644					
PEEP	1340#							
PIOBC	33#	809						
PIOBD	32#	1090	1118	1129	1143	1156	1182	1257
PIX	2165#	2247	2798					1263
PIXEL	2120	2548#						
POSTA	1275	1882#						
POS	2164#	2255	2283	2796				
POS0	2174#	2298	2387					
POS1	198#	1914	1953	2061				
POS2	199#	1943	2066					
PRAMP	2149#	2224						
PRBOX	2122	2575#						
PUNCH	96	909#						
RBYTE	611	614	688	740#	746	751		
RDOI	770#							
RDY	607	654	692	699#	771			
ROY1	705	717#						
READER	97	895#						
READOP	379	388	502	2095#	2096			
REAHST	486	641#						
RECAL	260	606#						
RESL1	751#	754						
RESLT	176#	747						
RESTR	332	716	767	1001#				
ROTAT	2132#	2583						
RRSLT	659	745#	772					
RSFLAG	380	433	438	484	2094#	2095		
RW	657	671#						
RW1	674#	676	681					
RW2	672#	683						
RWERR	635	664#						
RWIT	639	644#						
RWMOVE	504	508#	513					
RWDOP	649#	663						
RWOPER	383	434	440#					
SC1	444#	448						
SC2	445	446	449#					
SCHR	306	543#						
SCR1	2273	2287#						
SCROLL	2173#	2299	2389					
SD1152	945#							
SDR1	949#	957						
SDR2	954#	956						
SDR3	948	951	958#	960				
SDR4	953	963#						
SDS	633	686#	702					
SDSK1	255	261#						
SECHSK	75#	490						
SECTRAN	106	268#						
SEEK	647	691#						
SEKDSK	183#	240	375	384	396	408	458	477
SEKHST	186#	450	468	482				
SEKSEC	192#	275	400	418	442	489	519	554
SEKTRK	184#	284	398	414	463	480	517	547
SELDISK	99	236#	316					
SENSE	609#	616	693					
SETO	1069	1077#						
SETDMA	102	288#	329	369				
SETFF	1227	1364#						
SETSEC	101	275#	326					
SETTRK	100	283#	319	349				
SFELD	1838#	1882						
SIOAC	52#	828	946	954	958			
SIOAD	50#	949	962					
SIOBC	53#	837	898	912				
SIOBD	51#	901	916					
SKBSY	694#	696						
SLOAD01	322#	343	353					
SLOCK	1221#	1629						
SMOD	2163#	2235						
SPALT1	1969	1971#						
SPALTE	1954	1967#						
SPEZI	600	603#						
STAB	603	618#						
STR	309	712	1003	1010#	1016			
STUP	2204	2206	2309#					
SYNCA	2636	2847#						
SYNCB	2651	2852#						

TAB	1641#	1881							
TAB2	1740#								
TAPO1	1249	1881#							
TAST	1067	1085#							
TC	58#	622	624						
TCPLS	595	622#	658						
TEST	2209	2262#	2374	2417					
TON	1886#	2068	2496						
TP10A	871#	872							
TP10B	808	873#	874						
TRCK	169#	784	791						
TS10A	876#	878							
TS10B	879#	881							
TYPE	2148#	2227							
UNACNT	312	366	395	402	407	436	2089#	2090	
UNADSK	397	409	516	2090#	2091				
UNASEC	401	419	520	2092#	2093				
UNATRK	399	413	429	431	518	2091#	2092		
UNIT	168#	259	782	789					
UNTEN	780	788#							
UPCUP	2553	2568	2580	2759#					
UPCUP1	2760#								
UPOKT	2566	2659#							
UPPRG	2567	2705#							
V24	13#	924							
WBOOT	91	360#							
WBOTE	91#	300							
WCOM	610	687	719#						
WCOM1	605	720#	725	730					
WRDIR	77#	522							
WRFLG	190#								
WRHST1	636#								
WRHMST	476	530	633#						
WRIT	104	384#							
WRTYPE	189#	382	390	521					
WRUAL	78#	381	391						
X1	2128#	2660	2768						
X2	2130#	2588	2659						
XX	2177#	2246	2262	2264	2270	2288	2314	2345	2361
	2372	2377	2393	2430	2459	2465	2486		
XMAX	2183#	2265	2276	2620	2646				
XYEIN	2439#	2516							
Y1	2129#	2665	2687	2759					
Y2	2131#	2587	2666	2688					
YK	2178#	2328	2472						
YMAX	2184#	2271							
Z1013	15#	18	31	795	807				
ZCLSL1	2162#	2256	2284	2304	2799				
ZCLSL2	2166#	2302	2382	2421	2482				
ZEIL1	1964#	1965							
ZEILE	1957	1959#							
ZGADR	2133#	2219							
ZGANF	2133	2859#							
ZGUM	1233#	1634							
ZK	200#								
ZKNEG	1021#	1127							
ZKPOS	1020#	1141							
ZL	1697#	1929	1929	1931	1933	1933	1937	1938	1962
	1967	2001	2002	2010	2010	2013	2024	2040	2044
ZS11	1143#	1148							
ZZ	1898#	1929	1933	1937	2002	2010	2024	2044	

Das Betriebssystem CP/M ist nicht auf einen bestimmten Mikrorechner festgelegt, sondern kann auf jedem Computer mit dem U 880 und mehr als 20 kByte RAM-Speicher laufen. Vorhandene Software, wie „Wordstar“, „Turbo-Pascal“, „Dbase“ usw., bietet auch dem Amateur interessante Anwendungsmöglichkeiten.

Die Broschüre soll allen Interessenten, auch Arbeitsgemeinschaften und Besitzern von Mikrorechnern der KC-Typen, am konkreten Beispiel des Z 1013 die notwendigen Informationen und Kenntnisse über die innere Struktur des Betriebssystems zur Anpassung eines Eigenbau- oder industriellen Heimcomputers vermitteln.