



SOFTWARE
DOKUMENTATION



Anleitung für den Systemprogrammierer



| Stand: |
| 5/88 |

Anwenderdokumentation

| System |
| DCP 3.30 |

ANLEITUNG
FÜR DEN
SYSTEMPROGRAMMIERER
DCP 3.30

VEB Robotron-Büromaschinenwerk
"Ernst Thälmann" Sommerda

VEB Robotron-Buchungsmaschinenwerk
Karl-Marx-Stadt

Die vorliegende 1. Auflage dieser Dokumentation entspricht dem Stand Mai 1988 und unterliegt nicht dem Änderungsdienst. Nachdruck, jegliche Vervielfältigung oder Auszüge daraus sind nicht zulässig.

Die Dokumentation wurde durch ein Kollektiv des

VEB ROBOTRON Büromaschinenwerk
"Ernst Thälmann"
Sömmerda

und

VEB ROBOTRON Buchungsmaschinenwerk
Karl-Marx-Stadt

ausgearbeitet.

Hinweise zur Verbesserung der Dokumentation und zur Fehlerkorrektur richten Sie bitte an:

VEB Robotron Büromaschinenwerk
"Ernst Thälmann" Sömmerda
Bereich Forschung und Entwicklung
Weissenseer Str. 52
Sömmerda
5230

VEB Robotron Buchungsmaschinenwerk
Karl-Marx-Stadt
PSF 129
Karl-Marx-Stadt
9010

Die "Anleitung für den System-Programmierer" besteht aus zwei Teilen:

Teil 1 enthält:

- Informationen über Eigenschaften, Struktur und Aufgaben des Betriebssystems DCP,
- die Systeminitialisierung, Datenträgerstruktur und Dateibehandlung,
- Angaben über installierbare Gerätetreiber und
- eine Beschreibung der BIOS-Interrupts.

Teil 2 enthält:

- die Beschreibung der DOS-Interrupts, der DOS-Funktionen,
- Informationen über die Struktur von COM- und EXE-Dateien und
- in der Anlage Zusammenstellungen der BIOS-/DOS-Interrupts, der DOS-Funktionen und eine Tastaturübersicht.

*** Inhaltsverzeichnis ***

Inhaltsverzeichnis	Seite
Teil 1	
1.	Einleitung 14
2.	Eigenschaften und Aufgaben des Betriebssystems DCP 15
3.	Struktur des Betriebssystems DCP 16
3.1.	Systembestandteile 16
3.1.1.	ROM-BIOS 16
3.1.2.	Der Bootrecord 17
3.1.3.	BIO.COM 17
3.1.4.	DOS.COM 18
3.1.5.	COMMAND.COM 18
3.2.	Schnittstellen für den Programmierer 21
3.3.	Speicherorganisation 22
4.	Systeminitialisierung 24
4.1.	Einschalttest, Laden Bootrecord 24
4.2.	Test System-Verfügbarkeit und Laden des RAM-BIOS 25
4.3.	Initialisierung DOS und Laden des Kommandoprozessors 25
4.4.	Kommandoprozessor 26
5.	Datenträgerstruktur und Dateibehandlung 27
5.1.	Merkmale der Dateiverwaltung 27
5.2.	Physischer Aufbau der Datenträger 27
5.2.1.	Aufbau und physische Adressräume 27
5.2.2.	Formatierung 28
5.3.	DCP-Bereiche auf der Diskette 28
5.3.1.	Einteilung der Diskette 28
5.3.2.	Der Bootrecord 29
5.3.3.	Die Dateizuordnungstabelle (FAT) 29
5.3.3.1.	Benutzung der Dateizuordnungstabelle für 12-Bit-Einträge 31
5.3.3.2.	Benutzung der Dateizuordnungstabelle für 16-Bit-Einträge 31
5.3.3.3.	Das Mediumkennzeichen 31
5.3.4.	Das Dateistammverzeichnis 32
5.4.	Organisation der Festplatte 35
5.4.1.	Arbeitsprinzipien 35
5.4.2.	Aufbau der Festplatte 35
5.4.3.	Master-Bootrecord 36
5.4.4.	Partition-Tabelle 36
5.4.4.1.	Aufbau einer Eintragung in der Partition-Tabelle 38
5.4.4.2.	Bedeutung des Boot-Indikator 38
5.4.5.	Aufbau einer erweiterten DCP-Partition 39
5.4.5.1.	Struktur der Tabelle der logischen Laufwerke 39
5.4.5.2.	Aufbau einer Eintragung in der Laufwerkstabelle 40
5.4.6.	Konfigurieren der Partitions 41
5.4.7.	Formatieren der Partitions 41
5.5.	Beschreibung der Datenträgerformate
5.5.1.	Der BIOS-Parameterblock (BPB)
5.5.2.	Parameter der unterstützten Formate

*** Inhaltsverzeichnis ***

	Seite	
5.6.	Dateibehandlung	43
5.6.1.	Arten des Dateiaufrufes	43
5.6.2.	Funktionsaufruf mit FCB	43
5.6.2.1.	Arbeitsweise	43
5.6.2.2.	Anzahl der eröffneten FCB-Dateien	44
5.6.2.3.	Struktur des Dateisteuerblocks (FCB)	44
5.6.3.	Funktionsaufruf über Händler	48
5.6.3.1.	Merkmale	48
5.6.3.2.	Vordefinierte Händler	49
5.6.3.3.	Datenstrom-Modi	50
5.6.3.4.	Anzahl aktiver Händler	51
6.	Installierbare Gerätetreiber	52
6.1.	Format eines Gerätetreibers	52
6.2.	Gerätetypen	52
6.2.1.	Zeichenorientierte Geräte	52
6.2.2.	Blockorientierte Geräte	52
6.3.	DEVICE HEADER	53
6.3.1.	Zeiger auf nächsten DEVICE HEADER in der Kette	53
6.3.2.	Attribut-Feld	54
6.3.3.	Zeiger auf Strategie- und Interruptroutinen	56
6.3.4.	Gerätebezeichnung	56
6.4.	Erstellung eines Gerätetreibers	56
6.5.	Installation des Gerätetreibers	57
6.5.1.	Installation von zeichenorientierten Geräten	58
6.5.2.	Installation von blockorientierten Geräten	58
6.6.	REQUEST HEADER	59
6.6.1.	Gerätekode	59
6.6.2.	Kommandokode	59
6.6.3.	Status-Feld	61
6.7.	Beschreibung der Gerätetreiberfunktionen	62
6.7.1.	INIT	62
6.7.2.	MEDIA CHECK	63
6.7.3.	Media-Descriptor-Byte	64
6.7.4.	Die Funktion BUILD BPB	Kommandokode =2 66
6.7.5.	INPUT oder OUTPUT	Kommandokodes=3,4,8,9,12 67
6.7.6.	NONDESTRUCTIVE INPUT NO WAIT	Kommandokode =5 68
6.7.7.	STATUS	Kommandokodes=6,10 69
6.7.8.	FLUSH	Kommandokodes=7,11 69
6.7.9.	OPEN oder CLOSE	Kommandokodes=13,14 70
6.7.10.	REMOVABLE MEDIA	Kommandokode =15 71
6.7.11.	GENERIC IOCTL REQUEST	Kommandokode =19 71
6.7.12.	GET LOGICAL DEVICE	Kommandokode.=23 72
6.7.13.	SET LOGICAL DEVICE	Kommandokode =24 72
6.8.	Das CLOCK8-Gerät (Zeitgebereinheit)	72
6.9.	Beispiel eines Gerätetreibers	73
6.10.	Zum System gehörende Gerätetreiber	74
6.10.1.	Der Gerätetreiber VDISK.SYS (virtuelle Diskette)	74
6.10.1.1.	Installation des Gerätetreibers VDISK.SYS	75
6.10.1.2.	Beispiel	76
6.10.2.	Der Gerätetreiber VE89.SYS	76
6.10.2.1.	Installation von VE89.SYS	76
6.10.2.2.	Die Escapefolgenunterstützung (ANSI-Teil)	77
6.10.2.2.1.	Wirkungsbereich	77

*** Inhaltsverzeichnis ***

	Seite	
6.10.2.2.2.	Syntax der Steuerfolgen	77
6.10.2.2.3.	Bildschirmsteuerfolgen	78
6.10.2.2.4.	Betriebsarten	80
6.10.2.2.5.	Umkodierung der Tastatur	81
6.10.2.2.6.	Beispiele	81
6.10.2.3.	Unterstützung des Zeichensatzwechsels	82
6.10.2.4.	Bildschirmmodi und Monitore	83
6.10.2.5.	Schnittstellen	85
7.	BIOS-Interrupts	89
7.1.	ROM-BIOS	89
7.2.	Erweiterung des ROM-BIOS	90
7.3.	DIVISION DURCH NULL	INT00 90
7.4.	EINZELSCHRITTVERARBEITUNG	INT01 90
7.5.	NMI-UNTERBRECHUNGEN	INT02 91
7.6.	PRÜFPUNKT	INT03 91
7.7.	ÜBERLAUFUNTERBRECHUNG	INT04 91
7.8.	BILDSCHIRMINHALT DRUCKEN	INT05 92
7.9.	Reservierter Interrupt	INT06 92
7.10.	Reservierter Interrupt	INT07 92
7.11.	ZEITGEBER	INT08 92
7.12.	TASTATUR	INT09 93
7.13.	Reservierter Interrupt	INT0A 93
7.14.	KOMMUNIKATION	INT0B 93
7.15.	KOMMUNIKATION	INT0C 94
7.16.	HARDDISK	INT0D 94
7.17.	DISKETTEN	INT0E 94
7.18.	DRUCKER	INT0F 94
7.19.	VIDEOSTEUERUNG	INT10 95
7.19.1.	Einführung	95
7.19.2.	Beschreibung der Funktionen	95
7.19.2.1.	BILDSCHIRMMODE SETZEN	INT10: 00h 95
7.19.2.2.	KURSORART SETZEN	INT10: 01h 96
7.19.2.3.	KURSORPOSITION SETZEN	INT10: 02h 97
7.19.2.4.	LESEN DER KURSORPOSITION	INT10: 03h 97
7.19.2.5.	LICHTSTIFTPOSITION ERMITTELN	INT10: 04h 98
7.19.2.6.	AKTUELLE SEITE SETZEN	INT10: 05h 98
7.19.2.7.	FENSTER NACH OBEN ROLLEN	INT10: 06h 99
7.19.2.8.	FENSTER NACH UNTEN ROLLEN	INT10: 07h 99
7.19.2.9.	ZEICHEN UND ATTRIBUT LESEN	INT10: 08h 100
7.19.2.10.	ZEICHEN UND ATTRIBUT SCHREIBEN	INT10: 09h 100
7.19.2.11.	ZEICHEN SCHREIBEN	INT10: 0Ah 101
7.19.2.12.	FARBE AUSWAHLEN	INT10: 0Bh 102
7.19.2.13.	PUNKT ZEICHNEN	INT10: 0Ch 102
7.19.2.14.	PUNKT LESEN	INT10: 0Dh 103
7.19.2.15.	ZEICHENAUSGABE	INT10: 0Eh 103
7.19.2.16.	BILDSCHIRMSTATUS ERMITTELN	INT10: 0Fh 104
7.19.2.17.	FARBPALETTE PROGRAMMIEREN	INT10: 10h 104
7.19.2.17.1.	Eine Farbe der Palette ändern	INT10: 10h 00h 105
7.19.2.17.2.	Palette ändern	INT10: 10h 02h 105
7.19.2.17.3.	Blinken erlauben/verbieten	INT10: 10h 03h 105
7.19.2.18.	ZEICHENGENERATOR PROGRAMMIEREN	INT10: 11h 106
7.19.2.18.1.	Nutzerzeichensatz laden ohne Veränderung der Systemeinstellungen	INT10: 11h 00h 106

*** Inhaltsverzeichnis ***

		Seite
7.19.2.18.2.	Standardzeichensatz 14x8 aktiv setzen ohne Veränderung der Systemeinstellungen	INT10: 11h 01h 106
7.19.2.18.3.	Standardzeichensatz 8x8 aktiv setzen ohne Veränderung der Systemeinstellungen	INT10: 11h 02h 107
7.19.2.18.4.	Standardzeichensatz 16x8 aktiv setzen ohne Veränderung der Systemeinstellungen	INT10: 11h 04h 107
7.19.2.18.5.	Nutzerzeichensatz laden mit Veränderung der Systemeinstellungen	INT10: 11h 10h 108
7.19.2.18.6.	Standardzeichensatz 14x8 aktiv setzen mit Veränderung der Systemeinstellungen	INT10: 11h 11h 108
7.19.2.18.7.	Standardzeichensatz 8x8 aktiv setzen mit Veränderung der Systemeinstellungen	INT10: 11h 12h 109
7.19.2.18.8.	Standardzeichensatz 16x8 aktiv setzen mit Veränderung der Systemeinstellungen	INT10: 11h 14h 109
7.19.2.18.9.	INT 1Fh ändern	INT10: 11h 20h 109
7.19.2.18.10.	Zeichengenerator für Grafikmodi ändern	INT10: 11h 21h 110
7.19.2.18.11.	Standardzeichensatz 14x8 für Grafikmodi setzen (nur Grafikmode 8 und 9 nutzen)	INT10: 11h 22h 110
7.19.2.18.12.	Standardzeichensatz 8x8 für Grafikmodi aktiv setzen	INT10: 11h 23h 111
7.19.2.18.13.	Standardzeichensatz 16x8 für Grafikmodi aktiv setzen (nur Grafikmode 8 und 9 nutzen)	INT10: 11h 24h 111
7.19.2.18.14.	Status ermitteln	INT10: 11h 30h 111
7.19.2.19.	ADAPTERSTATUS ERMITTELN	INT10: 12h 112
7.19.2.20.	ZEICHENKETTENAUSGABE	INT10: 13h 113
7.19.3.	Bildschirmmodi	114
7.19.3.1.	Alphamode 40x25 und 80x25	116
7.19.3.1.1.	Zeichendarstellung	116
7.19.3.1.2.	Organisation im Bildwiederholpeicher	116
7.19.3.2.	Grafikmode 320x200	117
7.19.3.2.1.	Punktendarstellung	117
7.19.3.2.2.	Organisation des Bildwiederholpeichers	117
7.19.3.2.3.	Zeichendarstellung	117
7.19.3.3.	Grafikmode 640x200	118
7.19.3.3.1.	Punktendarstellung	118
7.19.3.3.2.	Zeichendarstellung	118
7.19.3.4.	Grafikmode 640x400	118
7.19.3.4.1.	Punktendarstellung	118
7.19.3.4.2.	Zeichendarstellung	118
7.19.3.5.	Grafikmode 640x480	119
7.19.3.5.1.	Punktendarstellung	119
7.19.3.5.2.	GDC-Funktionen und Zugriff auf Bildwiederholpeicher	119
7.19.3.5.3.	Zeichendarstellung	119
7.19.4.	Reservierte Bytes im Kommunikationsbereich	120
7.19.5.	Steuerung des GDC des Farbgrafikadapters	121
7.19.5.1.	Portadressen	121
7.19.5.2.	Bitbelegung des GDC-Status-Ports	121
7.19.5.3.	Bitbelegung des Modussteuerregisters	121
7.19.5.4.	Bitbelegung des Farbauswahlregisters	122
7.19.5.5.	Bitbelegung des Statusregisters	122
7.19.5.6.	Bitbelegung des Steuerregisters	123
7.19.5.7.	Auswahl der Farbenen im Grafikmode 640x480	123

*** Inhaltsverzeichnis ***

		Seite
7.20.	SCHALTERBELEGUNG	INT11 124
7.21.	SPICHERGROSSE	INT12 125
7.22.	DISKETTEN UND HARDDISK	INT13 126
7.22.1.	Disketten-Interrupts	126
7.22.1.1.	Funktionen	127
7.22.1.1.1.	RÜCKSETZEN FD (RESET)	INT13: 00h 127
7.22.1.1.2.	LESEN DES DISKETTENSTATUS	INT13: 01h 128
7.22.1.1.3.	LESEN SEKTOREN	INT13: 02h 129
7.22.1.1.4.	SCHREIBEN SEKTOREN	INT13: 03h 130
7.22.1.1.5.	KONTROLL-LESEN	INT13: 04h 131
7.22.1.1.6.	FORMATIEREN EINER SPUR (FORMAT)	INT13: 05h 132
7.22.1.1.7.	LESEN LAUFWERK-PARAMETER	INT13: 08h 134
7.22.1.1.8.	ERKENNBARKEIT VON DISKETTENWECHSEL (READ DASD TYPE)	INT13: 15h 136
7.22.1.1.9.	DISKETTENWECHSEL-INFORMATION	INT13: 16h 136
7.22.1.1.10.	SETZEN FORMAT-PARAMETER (SET DASD TYPE)	INT13: 17h 137
7.22.1.1.11.	SETZEN DES DISKETTENTYPUS (SET MEDIA TYPE)	INT13: 18h 138
7.22.1.2.	Statustabelle	139
7.22.1.3.	Reservierte Bytes im Datensegment 40h	139
7.22.1.4.	Diskettenparameter-Block (DISK-BASE)	141
7.22.1.4.1.	DISK-BASE Arten	141
7.22.1.4.2.	DISK-BASE Aufbau	141
7.22.2.	Harddisk-Interrupts	143
7.22.2.1.	Funktionen	143
7.22.2.1.1.	RÜCKSETZEN HARDDISK UND FDC	INT13: 00h 143
7.22.2.1.2.	LESEN DES DISKSTATUS	INT13: 01h 143
7.22.2.1.3.	LESEN SEKTOREN	INT13: 02h 144
7.22.2.1.4.	SCHREIBEN SEKTOREN	INT13: 03h 145
7.22.2.1.5.	KONTROLL-LESEN	INT13: 04h 146
7.22.2.1.6.	FORMATIEREN EINER SPUR	INT13: 05h 147
7.22.2.1.7.	FORMATIEREN EINER SPUR MIT KENN- ZEICHNUNG DER DEFEKTEN SEKTOREN	INT13: 06h 148
7.22.2.1.8.	FORMATIEREN AB ADRESSIERTEM ZYLINDER	INT13: 07h 149
7.22.2.1.9.	LESEN LAUFWERK-PARAMETER	INT13: 08h 149
7.22.2.1.10.	INITIALISIERUNG LAUFWERK-PARAMETER	INT13: 09h 150
7.22.2.1.11.	LESEN LANG	INT13: 0Ah 151
7.22.2.1.12.	SCHREIBEN LANG	INT13: 0Bh 152
7.22.2.1.13.	POSITIONIEREN	INT13: 0Ch 153
7.22.2.1.14.	ALTERNATIVES RÜCKSETZEN	INT13: 0Dh 153
7.22.2.1.15.	LESEN SEKTORPUFFER	INT13: 0Eh 154
7.22.2.1.16.	SCHREIBEN SEKTORPUFFER	INT13: 0Fh 155
7.22.2.1.17.	TEST DER LAUFWERKBEREITSCHAFT	INT13: 10h 156
7.22.2.1.18.	REKALIBRIEREN	INT13: 11h 156
7.22.2.1.19.	RAM-DIAGNOSE	INT13: 12h 157
7.22.2.1.20.	LAUFWERK-DIAGNOSE	INT13: 13h 158
7.22.2.2.	Statustabelle	158
7.22.2.3.	Reservierte Bytes im Datensegment 40h	159
7.22.2.4.	Festplatten-Architektur	160
7.22.2.4.1.	Master-Bootrecord	160
7.22.2.4.1.1.	Funktion	161
7.22.2.4.1.2.	Partition-Tabelle	161
7.22.2.4.2.	Partitions	162
7.22.2.4.2.1.	Primäre DCP-Partition	163

*** Inhaltsverzeichnis ***

	Seite	
7.22.2.4.2.2.	Erweiterte DCP-Partition	163
7.22.2.4.2.2.1.	Logischer Aufbau der erweiterten DCP-Partition	163
7.22.2.4.2.2.2.	Erweiterter Partition-Bootrecord	164
7.22.2.4.2.2.3.	Logische Laufwerkstabelle des erweiterten Partition-Bootrecords	164
7.22.2.4.2.3.	Einrichten und Aktivieren der Partitions	167
7.22.2.4.2.4.	Formatieren Partition	167
7.22.2.4.3.	Neues INT19h BOOTSTRAP-LOADER	167
7.23.	EIN- UND AUSGABE ÜBER EINEN SERIELLEN ANSCHLUSS	169
	INT14	169
7.23.1.	INITIALISIERUNG SERIELLER ANSCHLUSS	169
	INT14: 00h	169
7.23.2.	SENDEN EINES ZEICHENS	170
	INT14: 01h	170
7.23.3.	EMPFANGEN EINES ZEICHENS	171
	INT14: 02h	171
7.23.4.	ABFRAGE STATUS	171
	INT14: 03h	171
7.24.	VERWALTUNGSFUNKTIONEN	173
	INT15	173
7.25.	TASTATUR	173
	INT16	173
7.25.1.	ZEICHEN LESEN	173
	INT16: 00h	173
7.25.2.	TEST ZEICHEN VERFÜGBAR	174
	INT16: 01h	174
7.25.3.	TASTATURSTATUS	175
	INT16: 02h	175
7.25.4.	TASTATURPUFFER SCHREIBEN	175
	INT16: 05h	175
7.25.5.	ZEICHEN LESEN ERWEITERT	176
	INT16: 10h	176
7.25.6.	TEST ZEICHEN VERFÜGBAR ERWEITERT	176
	INT16: 11h	176
7.25.7.	TASTATURSTATUS ERWEITERT	177
	INT16: 12h	177
7.26.	AUSGABE ÜBER PARALLELE SCHNITTSTELLE AN DRUCKER	178
	INT17	178
7.26.1.	DRUCKEN EINES ZEICHENS	178
	INT17: 00h	178
7.26.2.	DRUCKER INITIALISIEREN	179
	INT17: 01h	179
7.26.3.	DRUCKER-STATUS ABFRAGEN	179
	INT17: 02h	179
7.27.	Reserviert	180
	INT18	180
7.28.	URLADEN	180
	INT19	180
7.29.	TAGESZEIT	181
	INT1A	181
7.29.1.	TAGESZEIT LESEN	181
	INT1A: 00h	181
7.29.2.	TAGESZEIT SETZEN	181
	INT1A: 01h	181
7.30.	TASTATURUNTERBRECHUNG	182
	INT1B	182
7.31.	ZEITGEBER NUTZER	182
	INT1C	182
7.32.	ADRESSE BILDSCHIRMPARAMETER	182
	INT1D	182
7.33.	ADRESSE DISKETTENPARAMETER-BLOCK (DISK-BASE)	183
	INT1E	183
7.34.	ADRESSE GRAFISCHER ZEICHENSATZ	183
	INT1F	183

*** Inhaltsverzeichnis ***

Teil 2		Seite
8.	DOS-Interrupts	184
8.1.	PROGRAMM BEENDEN	185
8.2.	AUFRUF VON DOS-FUNKTIONEN	186
8.3.	ABSCHLUSSADRESSE	186
8.4.	BEHANDLUNG BEI CTRL-C	187
8.5.	VEKTOR DER BEHANDLUNGSRoutine FÜR KRITISCHE FEHLER	187
8.6.	ABSOLUTES DISKETTENLESEN	193
8.7.	ABSOLUTES DISKETTENSCHREIBEN	194
8.8.	BEENDEN, ABER RESIDENT BLEIBEN	196
8.9.	MULTIPLEX-INTERRUPT	197
8.9.1.	MULTIPLEX-INTERRUPT PRINT	197
8.9.1.1.	Ermittlung des installierten Status	198
8.9.1.2.	Submit-Datei	198
8.9.1.3.	Dateiabbruch	199
8.9.1.4.	Alle Dateien abrechnen	199
8.9.1.5.	Status	200
8.9.1.6.	Statusende	200
8.9.2.	MULTIPLEX-INTERRUPT ASSIGN	201
8.9.3.	MULTIPLEX-INTERRUPT SHARE	201
8.9.4.	MULTIPLEX-INTERRUPT APPEND	202
8.9.4.1.	Ermittlung des installierten Status	202
8.9.4.2.	Ausgabe Fehlertext und Programm beenden	203
8.9.4.3.	AX auf 0FFFFh setzen	203
8.9.4.4.	Ermitteln der Adresse der APPEND-INT21-Routine	204
8.9.4.5.	Zeiger auf APPEND-Liste	204
8.9.4.6.	Ermittlung der APPEND-Version	205
8.9.5.	Beispiel für eine 2Fh-Behandlungs- routine	205
8.10.	Reservierte Interrupts	206
9.	DOS-Funktionsaufrufe	207
9.1.	DOS-Funktionen nach Funktionsgruppen sortiert	207
9.1.1.	Ein- und Ausgabe für zeichenorientierte Standardgeräte	207
9.1.2.	Speicherbehandlung	208
9.1.3.	Prozessbehandlung	208
9.1.4.	Datei- und Verzeichnisbehandlung	209
9.1.4.1.	Händler	209
9.1.4.2.	Dateiabhängige Funktionsaufrufe	209
9.1.4.2.1.	Aufruf über Händler	209
9.1.4.2.2.	Aufruf über FCB	210
9.1.4.3.	Funktionsaufrufe für gemeinsamen Dateizugriff (Datei-Sharing)	210
9.1.4.4.	Geräteabhängige Funktionsaufrufe	211
9.1.4.5.	Verzeichnisabhängige Funktionsaufrufe	211
9.1.5.	Netzwerkaufrufe	212
9.1.6.	Sonstige Systemaufrufe	212
9.1.7.	Reservierte DOS-Funktionsaufrufe	213

*** Inhaltsverzeichnis ***

	Seite	
9.2.	DOS-Funktionen	213
9.2.1.	Allgemeine Grundsätze	213
9.2.1.1.	Aufruf der DOS-Funktionen	213
9.2.1.2.	Interner Stapel	213
9.2.1.3.	Dateispezifikation	214
9.2.1.4.	ASCII-Zeichenfolgen	215
9.2.1.5.	Netzwerkpfade	215
9.2.1.6.	Netzwerk-Zugriffsrechte	215
9.2.1.7.	Fehlerrückgabeformat	216
9.2.1.7.1.	Der Fehlercode	216
9.2.1.7.2.	Der erweiterte Fehler	218
9.2.1.7.2.1.	Fehlerklasse	218
9.2.1.7.2.2.	Nutzerhinweise	219
9.2.1.7.2.3.	Fehlerort	219
9.2.2.	Beschreibung der DOS-Funktionen	220
9.2.2.1.	PROGRAMM BEENDEN	INT21: 00h 220
9.2.2.2.	KONSOL-EINGABE MIT ECHO	INT21: 01h 220
9.2.2.3.	KONSOL-AUSGABE	INT21: 02h 221
9.2.2.4.	HILFSEINGABE	INT21: 03h 222
9.2.2.5.	HILFSAUSGABE	INT21: 04h 222
9.2.2.6.	AUSGABE AUF LISTGERAT	INT21: 05h 223
9.2.2.7.	DIREKTE KONSOL-EIN-/AUSGABE	INT21: 06h 223
9.2.2.8.	DIREKTE KONSOL-EINGABE OHNE ECHO	INT21: 07h 224
9.2.2.9.	KONSOL-EINGABE OHNE ECHO	INT21: 08h 225
9.2.2.10.	ZEICHENKETTENAUSGABE	INT21: 09h 225
9.2.2.11.	EINGABE KONSOL-PUFFER	INT21: 0Ah 226
9.2.2.12.	KONSOL-STATUS	INT21: 0Bh 227
9.2.2.13.	LÖSCHEN DES KONSOL-PUFFERS UND AUFRUF EINER STANDARD-EINGABE	INT21: 0Ch 227
9.2.2.14.	DISKETTE RÜCKSETZEN	INT21: 0Dh 228
9.2.2.15.	AUSWAHL BEZUGS-LAUFWERK	INT21: 0Eh 229
9.2.2.16.	DATEI ÖRÖFFNEEN	INT21: 0Fh 229
9.2.2.17.	DATEI SCHLIESSEN	INT21: 10h 230
9.2.2.18.	SUCHE NACH DEM ERSTEN EINTRAG	INT21: 11h 231
9.2.2.19.	SUCHE NACH DEM NÄCHSTEN EINTRAG	INT21: 12h 233
9.2.2.20.	DATEI LÖSCHEN	INT21: 13h 234
9.2.2.21.	SEQUENTIELLES LESEN	INT21: 14h 234
9.2.2.22.	SEQUENTIELLES SCHREIBEN	INT21: 15h 235
9.2.2.23.	DATEI ERSTELLEN	INT21: 16h 236
9.2.2.24.	DATEI UMBENENNEN	INT21: 17h 237
9.2.2.25.	ABFRAGE BEZUGS-LAUFWERK	INT21: 19h 238
9.2.2.26.	SETZEN DTA	INT21: 1Ah 239
9.2.2.27.	INFORMATIONEN DER ZUORDNUNGSTABELLE	INT21: 1Bh 239
9.2.2.28.	INFORMATIONEN DER ZUORDNUNGSTABELLE FÜR BELIEBIGES LAUFWERK	INT21: 1Ch 240
9.2.2.29.	WAHLFREIES LESEN	INT21: 21h 240
9.2.2.30.	WAHLFREIES SCHREIBEN	INT21: 22h 242
9.2.2.31.	DATEIGRÖSSE	INT21: 23h 243
9.2.2.32.	AKTUALISIEREN RELATIVE NUMMER DES DATENSATZES	INT21: 24h 244
9.2.2.33.	SETZEN INTERRUPTVEKTOR	INT21: 25h 244
9.2.2.34.	ERSTELLEN NEUER PSP	INT21: 26h 245
9.2.2.35.	WAHLFREIES BLOCKLESEN	INT21: 27h 245
9.2.2.36.	WAHLFREIES BLOCKSCHREIBEN	INT21: 28h 247

*** Inhaltsverzeichnis ***

			Seite
9.2.2.37.	DATEIBEZEICHNUNG ANALYSIEREN	INT21: 29h	248
9.2.2.38.	DATUM ABFRAGEN	INT21: 2Ah	249
9.2.2.39.	DATUM SETZEN	INT21: 2Bh	250
9.2.2.40.	ZEIT ABFRAGEN	INT21: 2Ch	251
9.2.2.41.	ZEIT SETZEN	INT21: 2Dh	251
9.2.2.42.	PRÜFLESEN SETZEN/RÜCKSETZEN	INT21: 2Eh	252
9.2.2.43.	DTA HOLEN	INT21: 2Fh	252
9.2.2.44.	DCP-VERSIONSNUMMER ABFRAGEN	INT21: 30h	253
9.2.2.45.	PROZESS BEENDEN UND RESIDENT BLEIBEN	INT21: 31h	253
9.2.2.46.	PRÜFUNG AUF CTRL-C	INT21: 33h	254
9.2.2.47.	INTERRUPTVEKTOR ABFRAGEN	INT21: 35h	255
9.2.2.48.	ORGANISATION DES DATENTRAGERS ABFRAGEN	INT21: 36h	255
9.2.2.49.	LANDESSPEZIFISCHE INFORMATIONEN ABFRAGEN/SETZEN	INT21: 38h	256
9.2.2.49.1.	ABFRAGE DER AKTUELLEN LANDES- SPEZIFISCHEN INFORMATION	INT21: 3800h	256
9.2.2.49.2.	ABFRAGE EINER BELIEBIGEN LANDES- SPEZIFISCHEN INFORMATION	INT21: 38xxh	258
9.2.2.49.3.	SETZEN DER LANDESSPEZIFISCHEN INFORMATION	INT21: 38xxh	258
9.2.2.50.	VERZEICHNIS ERSTELLEN (MKDIR)	INT21: 39h	259
9.2.2.51.	VERZEICHNIS LÖSCHEN (RMDIR)	INT21: 3Ah	260
9.2.2.52.	AKTUELLES VERZEICHNIS WECHSELN (CHDIR)	INT21: 3Bh	261
9.2.2.53.	HANDLER ERSTELLEN (CREAT)	INT21: 3Ch	262
9.2.2.54.	HANDLER ERÖFFNEN	INT21: 3Dh	264
9.2.2.55.	HANDLER SCHLIESSEN	INT21: 3Eh	269
9.2.2.56.	LESEN VON DATEI/EINHEIT	INT21: 3Fh	270
9.2.2.57.	SCHREIBEN AUF DATEI/EINHEIT	INT21: 40h	271
9.2.2.58.	LÖSCHEN EINER DATEI AUS ANGEGEBENEM VERZEICHNIS (UNLINK)	INT21: 41h	272
9.2.2.59.	VERSCHIEBEN DES LESE-/SCHREIBZEIGERS (LSEEK)	INT21: 42h	273
9.2.2.60.	DATEIATTRIBUT ABFRAGEN/VERÄNDERN (CHMOD)	INT21: 43h	274
9.2.2.61.	E/A-STEUERUNG FÜR EINHEITEN (IOCTL)	INT21: 44h	275
9.2.2.61.1.	Abfrage von Händlerinformationen	INT21: 44h 00h	275
9.2.2.61.2.	Setzen von Händlerinformationen	INT21: 44h 01h	277
9.2.2.61.3.	Lesen vom Zeichengerät	INT21: 44h 02h	278
9.2.2.61.4.	Schreiben auf Zeichengerät	INT21: 44h 03h	279
9.2.2.61.5.	Lesen vom Blockgerät	INT21: 44h 04h	280
9.2.2.61.6.	Schreiben auf Blockgerät	INT21: 44h 05h	281
9.2.2.61.7.	Eingabestatus abfragen	INT21: 44h 06h	281
9.2.2.61.8.	Ausgabestatus abfragen	INT21: 44h 07h	282
9.2.2.61.9.	Abfrage, ob ein einzelnes Blockgerät auswechselbar ist	INT21: 44h 08h	283
9.2.2.61.10.	Abfrage, ob ein logisches Gerät lokal oder entfernt ist	INT21: 44h 09h	284
9.2.2.61.11.	Abfrage, ob ein Händler lokal oder entfernt ist	INT21: 44h 0Ah	285
9.2.2.61.12.	Verändern des Zählers für Wieder- holungsversuche bei gemeinsamer Dateinutzung	INT21: 44h 0Bh	286

*** Inhaltsverzeichnis ***

	Seite
9.2.2.61.13. Art und Weise der E/A-Steuerung für Händler	INT21: 44h 0Ch 287
9.2.2.61.13.1. Beginn der Vorbereitung von Zeichensatztabellen	INT21: 44h 0Ch 4Ch 287
9.2.2.61.13.2. Ende der Vorbereitung von Zeichensatztabellen	INT21: 44h 0Ch 4Dh 289
9.2.2.61.13.3. Auswahl bzw. Aktivierung einer Zeichensatztabelle	INT21: 44h 0Ch 4Ah 290
9.2.2.61.13.4. Anzeige der aktuellen Zeichensatztabelle	INT21: 44h 0Ch 6Ah 291
9.2.2.61.13.5. Anzeige der vorbereiteten Zeichensatztabelle	INT21: 44h 0Ch 6Bh 292
9.2.2.61.14. Art und Weise der E/A-Steuerung für Blockgeräte	INT21: 44h 0Dh 293
9.2.2.61.14.1. Abfrage der Geräteparameter	INT21: 44h 0Dh 60h 293
9.2.2.61.14.2. Setzen der Geräteparameter	INT21: 44h 0Dh 40h 295
9.2.2.61.14.3. Lesen einer Spur vom logischen Gerät	INT21: 44h 0Dh 61h 296
9.2.2.61.14.4. Schreiben einer Spur auf logisches Gerät	INT21: 44h 0Dh 41h 297
9.2.2.61.14.5. Formatieren und Überprüfen einer Spur auf logischem Gerät	INT21: 44h 0Dh 42h 298
9.2.2.61.14.6. Überprüfen einer Spur auf logischem Gerät	INT21: 44h 0Dh 62h 299
9.2.2.61.15. Abfrage des logischen Laufwerkes	INT21: 44h 0Eh 300
9.2.2.61.16. Setzen des logischen Laufwerkes	INT21: 44h 0Fh 301
9.2.2.62. HANDLER DUPLIZIEREN (DUP)	INT21: 45h 302
9.2.2.63. HANDLERDUPLIKAT ERSTELLEN (FORCDUP)	INT21: 46h 303
9.2.2.64. AKTUELLES VERZEICHNIS HOLEN	INT21: 47h 304
9.2.2.65. SPEICHER ZUWEISEN	INT21: 48h 304
9.2.2.66. ZUGEWIESENEN SPEICHER FREIGEBEN	INT21: 49h 305
9.2.2.67. ZUGEWIESENEN SPEICHERBLOCK VERANDERN (SETBLOCK)	INT21: 4Ah 306
9.2.2.68. PROGRAMM LADEM MIT UND OHNE AUSFÜHREN (EXEC)	INT21: 4Bh 307
9.2.2.68.1. Programm laden und ausführen (EXEC)	INT21: 4Bh 00h 307
9.2.2.68.2. Programm laden	INT21: 4Bh 03h 309
9.2.2.69. PROZESS BEENDEN (EXIT)	INT21: 4Ch 310
9.2.2.70. BEENDIGUNGSKODE LIEFERN (WAIT)	INT21: 4Dh 311
9.2.2.71. ERSTE DATEIEINTRAGUNG FINDEN (FIND FIRST)	INT21: 4Eh 312
9.2.2.72. NÄCHSTE DATEIEINTRAGUNG FINDEN (FIND NEXT)	INT21: 4Fh 313
9.2.2.73. STATUS PRÜFLESEN ABFRAGEN	INT21: 54h 314
9.2.2.74. DATEI UMBENENNEN	INT21: 56h 315
9.2.2.75. DATUM UND UHRZEIT EINER DATEI ABFRAGEN/SETZEN	INT21: 57h 316
9.2.2.75.1. Datum und Uhrzeit einer Datei abfragen	INT21: 57h 00h 316
9.2.2.75.2. Datum und Uhrzeit einer Datei setzen	INT21: 57h 01h 317
9.2.2.76. ERWEITERTE FEHLERMELDUNG	INT21: 59h 317
9.2.2.77. EINZIGARTIGE DATEI ERSTELLEN	INT21: 5Ah 319
9.2.2.78. NEUEN HANDLER ERSTELLEN	INT21: 5Bh 320
9.2.2.79. DATEIZUGRIFF VERBIETEN/ERLAUBEN	INT21: 5Ch 321
9.2.2.80. MASCHINENNAME ABFRAGEN	INT21: 5Eh 00h 323

*** Inhaltsverzeichnis ***

		Seite
9.2.2.81.	MASCHINENNAME SETZEN	INT21: 5Eh 01h 324
9.2.2.82.	DRUCKER-SETUP SETZEN	INT21: 5Eh 02h 324
9.2.2.83.	DRUCKER-SETUP ABFRAGEN	INT21: 5Eh 03h 325
9.2.2.84.	EINTRAGUNGEN DER ZUWEISUNGSTABELLE ABFRAGEN	INT21: 5Fh 02h 326
9.2.2.85.	ZUWEISUNG EINES GERATES	INT21: 5Fh 03h 327
9.2.2.86.	ZUWEISUNG AUFHEBEN	INT21: 5Fh 04h 329
9.2.2.87.	ERMITTELN AKTUELLER PSP	INT21: 62h 330
9.2.2.88.	ERMITTLUNG EINER ERWEITERTEN LANDESINFORMATION	INT21: 65h 330
9.2.2.89.	ERMITTELN/SETZEN DER GLOBALEN ZEICHENSATZTABELLE	INT21: 66h 334
9.2.2.89.1.	Ermitteln der Zeichensatztabelle	INT21: 66h 01h 334
9.2.2.89.2.	Setzen der Zeichensatztabelle	INT21: 66h 02h 335
9.2.2.90.	SETZEN DER HANDLERANZAHL	INT21: 67h 336
9.2.2.91.	ÜBERGABE AN HANDLER	INT21: 68h 337
10.	Strukturen von COM- und EXE-Dateien	338
10.1.	PSP (Programm-Segment-Präfix)	338
10.2.	Struktur einer COM-Datei	339
10.3.	Struktur einer EXE-Datei	340
10.4.	Relokationstabelle	342
 Anlagen		
Anlage 1:	Die BIOS-Interrupts im Überblick	346
Anlage 2:	Die DOS-Interrupts im Überblick	358
Anlage 3:	Die DOS-Funktionen im Überblick	361
Anlage 4:	Speicherbelegung	374
Anlage 5:	Tastatur-Übersicht	376
	A5.1. Tastaturpositionen	376
	A5.2. Deutsche Tastatur	377
	A5.2.1. Tastaturbelegung	377
	A5.2.2. Codes (INT16 00h)	378
	A5.2.3. Codes (INT16 10h)	381
Anlage 6:	Kode-Tabelle	384
	A6.1. Kode - Tabelle 850	384
	A6.2. Kode - Tabelle 437	385
Anlage 7:	Umrechnungstabelle (dezimal <-> hexadezimal)	386

1. Einleitung

Das im VEB Kombinat ROBOTRON entwickelte Betriebssystem DCP (Disk Control Program) ist ein disketten- bzw. harddisk-orientiertes Betriebssystem, welches kompatibel ist zu dem international eingeführten Betriebssystem PC-DOS der Firma IBM. Durch Verwendung gleicher BIOS- bzw. DOS-Interrupts und Verarbeitung gleicher Floppy-Disk-/Harddisk-Formate wird eine Lauffähigkeit der Anwenderprogramme gewährleistet.

Die vorliegende Dokumentation ist besonders für den Systemprogrammierer gedacht, der vorhandene Standardsoftware modifizieren (installieren) muss bzw. spezielle Probleme zu lösen hat, welche die Nutzung der internen Betriebssystemkomponenten erfordern. Letzteres ist z.B. dann der Fall, wenn periphere Geräte an den EC 1834 anzuschliessen sind, welche nicht vom Gerätehersteller softwaremässig unterstützt werden.

Im Kapitel 2 dieser Dokumentation werden Eigenschaften, Aufgabe und Arbeitsweise des Betriebssystems DCP dargestellt.

Die Struktur des Systems mit den Bestandteilen Bootrecord, ROM-BIOS, RAM-BIOS, DOS und COMMAND sowie die Nutzerschnittstellen und die Speicheraufteilung sind aus dem Kapitel 3 ersichtlich.

Das Kapitel 4 behandelt den Komplex der Systeminitialisierung mit dem Umladen des Systems vom Floppy-Disk/Harddisk, dem Einbinden von Treibern und gibt Hinweise zu den Dateien CONFIG.SYS und AUTOEXEC.BAT.

Informationen über die Datenträgerstruktur und die Dateibehandlung (Disketten-/Harddisk-Aufbau, Dateiverzeichnis, Pfadorganisation, Dateizuordnungstabelle und Dateisteuerblöcke) stehen im Kapitel 5.

Das Kapitel 6 enthält die Treiber-Struktur und die Beschreibung der externen Treiber ANSI und VDISK.

BIOS-Interrupts werden im Kapitel 7, DOS-Interrupts im Kapitel 8 und die verfügbaren DOS-Funktionen im Kapitel 9 behandelt.

Das Kapitel 10 enthält Erläuterungen zur Struktur von .COM- und .EXE-Files sowie Informationen über die Relokationstabelle. Der Umgang mit der vorliegenden Dokumentation setzt Kenntnisse der Hardware des EC 1834 und der Bedienung des Betriebssystems DCP voraus.

Dazu steht folgende Dokumentation zur Verfügung:

- Anleitung für den Bediener/ Hardware-Installationshandbuch
- Betriebsvorschrift EC 1834
- Systemhandbuch DCP - Anleitung für den Bediener.

2. Eigenschaften und Aufgaben des Betriebssystems DCP

DCP ist ein leistungsfähiges disketten- bzw. harddisk-orientiertes Betriebssystem und unterstützt gleichermaßen den Programmierer und Anwender.

Dieses Einzelnutzerbetriebssystem bietet einfache Möglichkeiten zur Behandlung von Daten, unterstützt eine geräteunabhängige Ein- und Ausgabe und hat eine leistungsfähige Kommandosprache zur Erstellung von Kommando-Dateien (Stapelverarbeitung). Bedeutungsvoll ist weiterhin die komfortable Unterstützung bei der Nutzung der seriellen Schnittstellen und der Farbgrafik.

Spezifische Merkmale aus der Sicht des Anwenders:

- Kommandos zur Magnetplattenverwaltung
- Möglichkeiten der Stapelverarbeitung, mit welcher oft benutzte Kommandofolgen zusammengefasst werden können
- Verwaltung von Datum/Uhrzeit
- erweiterte Ein-/Ausgabefunktionen (Umlenken von Datenströmen, Nebenbei-Textausgabe, Hardcopy-Funktion für Bildschirminhalt)
- Cursorsteuerung nach ANSI und Änderung von Tastenfunktionen
- Händler-Technik
- hierarchische Dateistruktur.

Spezifische Merkmale aus der Sicht des Programmierers:

- Dienstprogramme für zeichenweise Ein-/Ausgabe (durch Systemfunktionen aufrufbar)
- Systemfunktionen zur Dateiverwaltung (Anlegen, Lesen, Schreiben und Löschen von Dateien)
- Programme zur Unterstützung der Softwareentwicklung (Editor, Assembler, Compiler, Linker, Debugger)
- Systemfunktionen zum Setzen und Abfragen von Datum und Uhrzeit
- Standardtechnik zum Starten, Beenden und Unterbrechen von Programmen.

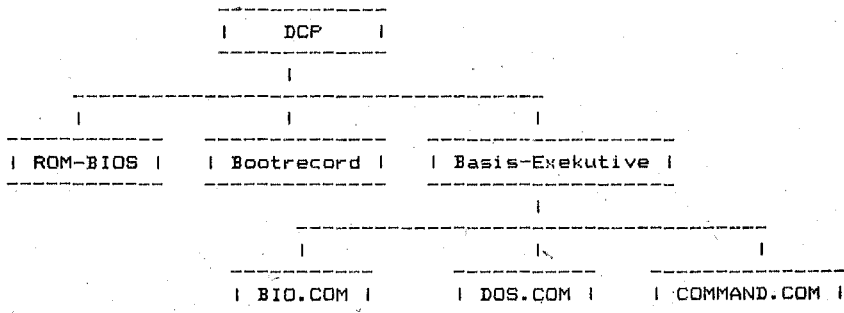
Grundsätzlich lassen sich die Aufgaben des Betriebssystems DCP in folgenden Punkten zusammenfassen:

- (1) Bearbeitung der bereitgestellten Kommandos
- (2) Laden, Starten und Ausführen von Programmen
- (3) Steuerung und Überwachung der angeschlossenen externen Geräte
- (4) Verwalten von Daten und Programmen auf Diskette und Harddisk
- (5) Bereitstellung von notwendigen Dienstfunktionen für den Nutzer
- (6) Erkennen, Anzeigen und Behandeln von aufgetretenen Fehlern.

3. Struktur des Betriebssystems DCP

3.1. Systembestandteile

Das Grundsystem DCP setzt sich aus folgenden Bestandteilen zusammen:



3.1.1. ROM-BIOS

Das ROM-BIOS (ROM = Read Only Memory

BIOS = Basic Input Output System)

ist ein Betriebssystembestandteil, der sich fest auf ROM-Bausteinen befindet und demzufolge Bestandteil der Hardware des Rechners ist.

ROM-BIOS ist ein einfaches Ein- und Ausgabesystem und enthält grundlegende Funktionen, welche von den Systemfunktionen des DCP benutzt werden. Dabei sind einige ROM-BIOS-Funktionen leistungsfähiger als die entsprechenden übergeordneten DCP-Funktionen. Die Funktionen des ROM-BIOS sind nur über Programme nutzbar und nicht vom Anwender (z.B. über Kommandos) aufrufbar. Schnittstellen zur Dateiverwaltung sind nicht vorhanden.

Mit dem ROM-BIOS wird eine Verbindung zum System hergestellt (die BIOS-Verbindung trennt den Benutzer von der Hardware), so daß die Programme relativ unabhängig von den Eigenschaften der Hardware geschrieben werden können. Nutzerprogramme bleiben somit bei Hardwareänderungen und Systemerweiterungen transparent.

Hauptbestandteile des ROM-BIOS sind:

- Initialisierungsroutinen für Mikroprozessor- und Controller-schaltkreise
- Einschalttest
- Hardwaretestroutinen
- die physischen Gerätetreiber für Tastatur, Bildschirm, Drucker, Floppy-Disk und Harddisk-Routinen für Datum und Uhrzeit
- Suchen und Laden des Bootrecords.

Nach einem Systemstart werden vom ROM-BIOS die Hardwaretest- und Initialisierungsroutinen abgearbeitet und der Bootrecord von der Systemdiskette bzw. vom Systemharddisk in den Speicher geladen und gestartet. ROM-BIOS belegt Speicheradressen ab F8000 und hat eine Grösse von 32 KByte.

Das ROM-BIOS kann durch zusätzliche Moduln erweitert werden. Diese Erweiterungen sind auf ROM's abgespeichert, die sich auf Adapterkarten zur Ansteuerung spezieller Geräte (z.B. Harddisk) befinden.

3.1.2. Der Bootrecord

Vom ROM-BIOS wird ein spezielles Programm geladen, dass sich ab Spur 0, Sektor 1, Seite 0 auf jeder Diskette bzw. dem ersten Sektor der Harddisk befindet und als Bootrecord bezeichnet wird. Dieser Satz wird durch das DCP-Kommando FORMAT erzeugt und hat die Aufgabe,

- das Vorhandensein der Systemdateien BIO.COM und DOS.COM zu prüfen,
- eine Fehlermeldung auszugeben, falls die Dateien nicht vorhanden sind,
- die Datei BIO.COM zu laden und dem Programm die Steuerung zu übergeben.

3.1.3. BIO.COM

Die Datei BIO.COM muss sich als erste Datei auf der Systemdiskette (Laufwerk A) bzw. dem Systemharddisk befinden und wird vom Umlade-Programm (Bootrecord) in den Speicher geladen.

Diese DCP-Komponente stellt den logischen Überbau des ROM-BIOS dar und beinhaltet:

- die logischen Standardgerätetreiber und
- die Initialisierungsroutine.

Nach dem Laden von BIO.COM wird die Initialisierungsroutine dieser Datei ausgeführt.

Anschliessend wird

- die Systemkonfiguration eingestellt,
- die Konfigurationsdatei CONFIG.SYS verarbeitet,
- die Datei DOS.COM und die Datei COMMAND.COM geladen und
- die Steuerung an den Kommandoprozessor COMMAND.COM übergeben.

Danach steht der Initialisierungsteil von BIO nicht mehr zur Verfügung. Die Standardgerätetreiber verbleiben für die weitere Systemarbeit im Speicher. Diese Treiber realisieren die durch DOS in einem Request Header übergebenen Aufgaben. Nach Ausführung der entsprechenden Funktion wird DOS über das Ergebnis informiert. Anforderungen, die in den Treibern nicht enthalten sind, werden abgewiesen und ein entsprechender Status gesetzt.

Für folgende Geräte sind standardmässig Treiber vorhanden:

- Tastatur, Bildschirm: CON
- Parallelerinterface: PRN (LPT1, LPT2, LPT3)
- Asynchronadapter: AUX (COM1, COM2)
- Systemuhr: CLOCK
- Diskette/Harddisk

3.1.4. DOS.COM

Die Datei DOS.COM (DOS = Disk Operating System) muss sich unmittelbar nach der Datei BIO.COM auf der Systemdiskette befinden. Nach dem Neustart von DOS werden Arbeitstabellen und Interrupt-Vektoren initialisiert.

DOS.COM liefert ein Interface für Nutzerprogramme auf höherer Stufe und beinhaltet das logische Ein- und Ausgabesystem (z.B. Dateiverwaltungsroutinen, Dateiblockung/-entblockung für Diskettenroutinen).

Daneben existieren im DOS eine Reihe von Funktionen, die vom Kommando-Prozessor bzw. von Nutzerprogrammen durch Übergabe von Parametern in Registern und Kontrollblöcken aufgerufen werden können. Wenn diese Funktionen Geräteoperationen betreffen, dann werden durch DOS entsprechende Aufrufe an das Programm BIO.COM ausgelöst.

Eine wesentliche Aufgabe von DOS besteht darin, von gerätespezifischen Unterschieden zu abstrahieren und einheitliche Programmierschnittstellen zu liefern.

3.1.5. COMMAND.COM

Der Kommandoprozessor bildet jenen Bestandteil des Betriebssystems DCP, der den direkten Dialog mit dem Bediener des Computers führt.

Er nimmt die Wünsche des Anwenders in Form von Kommandos entgegen und sorgt dafür, dass für deren Ausführung entsprechende Anweisungen an DCP weitergegeben werden.

Gleichzeitig führt er den Bediener und informiert ihn genau über den Zeitpunkt der Beendigung und die Qualität der Ausführung eines aktivierten Kommandos!

Die drei Bestandteile des COMMAND.COM sind

- Residentteil (ca. 3,7 KByte)
- Initialisierungsteil (ca. 2,1 KByte)
- Transientteil (ca. 19,7 KByte).

Bei jedem Neuanlauf des Systems DCP, d.h. nach Einschalten des Computers (Kaltstart) oder nach Drücken der Tastenkombination CTRL-ALT-DEL (Warmstart), wird der Kommandoprozessor durch die Systemkomponente BIO.COM so geladen, dass sein Residentteil sich im Speicher an DOS.COM anlagert.

Gleichzeitig erfolgt das Starten des Kommandoprozessors durch Sprung in seinen Initialisierungsteil.

*** Struktur des Betriebssystems DCP ***

Hier testet COMMAND.COM als erstes durch Abfrage der DCP-Versionnummer, ob er zu den schon geladenen DCP-Komponenten BIO.COM und DOS.COM passt. Ist das nicht der Fall, erfolgt die Fehlermeldung:

**Falsche DCP-Version
Falsch oder fehlend; Befehlsinterpretier**

Ist COMMAND.COM innerhalb des Betriebssystems arbeitsfähig, laufen weitere Initialisierungsarbeiten ab:

- Der Transiententeil wird an das Ende des noch verfügbaren Speicherbereiches verschoben.
- Die Umgebung des Kommandoprozessors wird unmittelbar am Ende seines Residentteils angelegt. Darin wird die COMSPEC-Zeichenkette definiert, die für das Betriebssystem den Suchpfad für COMMAND.COM darstellt.
- Der Speicherbereich des Initialisierungsteils wird an den DCP-Speicherverwalter zurückgegeben und kann damit durch das erste zu ladende Programm (.COM oder .EXE-Datei) überlagert werden.
- Falls die Datei AUTOEXEC.BAT im Stammdateiverzeichnis des aktuellen Laufwerks vorhanden ist, wird ihre spätere Abarbeitung im Transiententeil vorbereitet, indem dafür Speicherbereich angefordert und eine Kommandozeile mit aktuellem Laufwerk, Pfad und der Dateibezeichnung AUTOEXEC.BAT eingerichtet wird.
- Läuft die Initialisierung ohne AUTOEXEC.BAT, wird die DCP-Systemmeldung "ROBOTRON DCP Version 3.30/1.0" auf dem Bildschirm ausgegeben und die Eingabeanforderung nach Systemdatum und Systemzeit aktiviert.
- Mit einem Sprung an die INT22h-Routine (Abschlussadresse) wird der Initialisierungsteil beendet. Die nun erstmalig erscheinende DCP-Systemanfrage signalisiert dem Anwender, daß das soeben geladene Betriebssystem voll funktionsfähig und arbeitsbereit ist.

Der Residentteil des Kommandoprozessors bleibt während der Arbeit des Betriebssystems beständig im RAM. Er enthält die Routinen für die Interrupts (s. Punkt 8 "DOS-Interrupts")

22h = Abschlussadresse

23h = Behandlung bei CTRL-C

24h = Vektor der Behandlungsroutine für kritische Fehler

und einen Programmteil, der für das Laden und Starten von .COM- und .EXE-Dateien verantwortlich ist. Nach Ausführung einer .COM- oder .EXE-Datei durch DOS.COM erfolgt dadurch immer ein Sprung an die Interrupt-Routine 22h. Hier wird durch Prüfsummenbildung kontrolliert, ob der Transiententeil des Kommandoprozessors noch vollständig im RAM steht. Ist das nicht der Fall, der Transiententeil könnte durch eine grosse .EXE-Datei teilweise überschrieben worden sein, wird er durch eine spezielle Laderoutine unter Verwendung der COMSPEC-Zeichenkette in der Umgebung nachgeladen.

*** Struktur des Betriebssystems DCP ***

Der Interrupt 22h verzweigt schliesslich in den Transiententeil und mündet dort in den Kommando-Eingabehaltepunkt des Systems; die DCP-Systemanfrage wird gesetzt.

Die Behandlung von Peripherie-Fehlern, sogenannten Einheiten-Fehlern (z.B. Diskettenfehler, Druckerfehler), überlässt DCP ebenfalls dem Residentteil des Kommandoprozessors. Dieser übergibt mittels Interrupt 24h dem Anwender Fehlermeldungen auf dem Bildschirm und zeigt gleichzeitig Möglichkeiten, wie auf den Fehler reagiert werden kann.

So könnte es z.B. beim Ausführen des residenten Kommandos DIR A: zu einem Diskettenlesefehler kommen. Daraufhin würde COMMAND.COM folgende Fehlerausschrift veranlassen:

**Sektor nicht gefunden. Fehler
bei Lesen von Laufwerk A
A(bbruch), W(liederholen), I(gnорieren) ?**

Der Interrupt 23h stellt die CTRL-C-Beendigungsadresse dar. Bei Erkennung der Tastenkombination CTRL-PAUSE verzweigt DCP an diese Adresse. Damit wird eine vom Anwender gewünschte Unterbrechung des gerade laufenden Programms ausgelöst und zum übergeordneten Prozess zurückgekehrt. Der übergeordnete Prozess ist standardmässig der Kommandoprozessor, der die Ausführung des Programms (.COM-, .EXE- oder .BAT-Datei) veranlasst hat.

Mit ca. 19 KByte stellt der Transiententeil die Hauptkomponente des Kommandoprozessors dar. In ihm sind alle internen Kommandos enthalten (s. Punkt 13 im "Systemhandbuch DCP - Anleitung für den Bediener").

Folgende wichtige Aufgaben werden ausgeführt:

Kommandobehandlung

Es werden die vom Bediener eingegebenen Kommandos mittels einer 128-Byte-Kommandozeile entgegengenommen und in einem 128-Byte-Zeilenpuffer zwischengespeichert. Das heisst, ein über die Kommandozeile eingegebenes und angezeigtes Bedienerkommando, das mit der <ENTER>-Taste quittiert wurde, wird durch COMMAND.COM - im Zeilenpuffer bis zur nächsten Kommandozeile aufbewahrt und - als internes Kommando sofort im Transiententeil ausgeführt oder als externes Kommando zu einer für DOS.COM verständlichen Kommandozeile aufbereitet und dann an DOS.COM zur Ausführung übergeben.

Das durch COMMAND.COM gut organisierte Zusammenspiel von Kommandozeile und Zeilenpuffer gestattet den Einsatz leistungsfähiger Editiertasten. Die Kommandozeile kann Datenumleitungen und Kommandoverknüpfungen enthalten, die ebenfalls durch COMMAND.COM ausgeführt werden.

Stapeldateiverarbeitung

Handelt es sich bei dem eingegebenen Kommando um eine .BAT-Datei, so übernimmt der ebenfalls im Transiententeil implementierte Stapeldateiverarbeitungsprozessor die Steuerung für die schrittweise Abarbeitung dieser Datei.

DCP-Systemanfrage setzen

Es wird entweder die Standard-DCP-Systemanfrage der Form "Aktuelles Laufwerk: >" (z.B. C: >) veranlasst oder bei Benutzung des internen Kommandos PROMPT die durch den Anwender veränderte DCP-Systemanfrage angezeigt.

3.2. Schnittstellen für den Programmierer

Das Betriebssystem DCP bietet dem Nutzer die Möglichkeit, durch Systemaufrufe bestimmte DOS- bzw. BIOS-Funktionen zu nutzen. Diese Systemaufrufe werden durch programmgesteuerte Unterbrechungen (Software-Interrupts) realisiert.

BIOS-Funktionen werden durch die Interrupts 10h..20h und DOS-Funktionen durch die Interrupts 21h..FFh aktiviert.

Eine besondere Bedeutung hat dabei der Interrupt 21h, über welchen die DOS-Funktionsaufrufe eingeleitet werden.

Die Systemaufrufe von Funktionen des DOS bzw. BIOS werden durch Übergabe der entsprechenden Funktionsnummer, (und eventuellen Parametern) an eine Systemroutine über Register ausgeführt.

Diese Systemaufrufe lassen sich in folgende Gruppen unterteilen:

- SCP-ähnliche, zeichenorientierte Ein-/Ausgabefunktionen
- SCP-ähnliche Dateiverwaltung
- DCP-Systemsteuerung
- DCP-Speicherverwaltung
- MUTOS-ähnliche Dateiverwaltung
- MUTOS-ähnliche Speicherverwaltung.

Aus dieser Gruppierung lässt sich die enge Verbindung zwischen DCP und den Betriebssystemen SCP bzw. MUTOS erkennen.

Prinzipiell sind sowohl SCP- als auch MUTOS-Programme unter Steuerung des Betriebssystems DCP lauffähig. Obwohl eine vollständige Kompatibilität zwischen den jeweiligen Systemen nicht gewährleistet ist, sind erforderliche Änderungen relativ leicht zu realisieren.

Die verfügbaren DOS-Funktionen und die Möglichkeiten des Aufrufs werden im Kapitel 9 dieser Dokumentation ausführlich behandelt.

Die Behandlung der BIOS- bzw. DOS-Interrupts erfolgt in den Kapiteln 7 und 8 dieser Dokumentation.

3.3. Speicherorganisation

Der EC 1834 verfügt in der maximalen Ausstattung über einen Hauptspeicher von 640 KByte-RAM, 32 KByte-ROM (EPROM) und 128 KByte für den Grafik- bzw. 8 KByte für den Monochrom-Monitor.

Der 1 MByte-Adressbereich wird wie folgt verwendet:

0 0000	0K	
2 0000	120K	
4 0000	250K	RAM
		(128, 256, 384
6 0000	384K	oder 640 KByte)
8 0000	512K	
A 0000	640K	Bildwiederholungspeicher für Grafik- bzw. Monochrom-Monitor
C 0000	768K	ROM-
		Erweiterungs-
E 0000	806K	bereich
F 8000	992K	ROM-BIOS (EPROM)

*** Struktur des Betriebssystems DCP ***

In der nachfolgenden Tabelle wird der RAM-Speicherbereich (0000..9FFF) und seine Nutzung durch das Betriebssystem bzw. das Anwenderprogramm dargestellt, wobei besonders der untere Adressraum betrachtet werden soll:

00000	Unterbrechungsvektoren 00..0F	
	BIOS-Vektoren 10..1F	Interrupt- Vektoren
00080	DOS-Vektoren 20..3F	
00100	Verfügbare Vektoren	
00400	Datenbereich ROM-BIOS	
00500	Datenbereich DOS	
00600	BIOS (BIO.COM)	D C P
	DOS (DOS.COM)	
	DOS - Puffer	
	Steuerbereiche und installierte Gerätetreiber	Benutzer- spezifischer Teil
	Residenter Teil von COMMAND.COM	
	Nutzer- programm	
	System-Stack (256 Byte)	Benutzer- Programm
	Nichtresidenter Teil von COMMAND.COM	

Die Speicherbelegung im einzelnen ist in der Anlage 4 dargestellt.

4. Systeminitialisierung

Das System wird sowohl durch System-Reset als auch bei Einschalten der Betriebsspannung initialisiert. Dabei werden folgende Schritte durchlaufen:

- Einschalttest, Laden des Bootrecords
- Test, ob System verfügbar und Laden des RAM-BIOS
- Initialisierung von DOS und Laden des Kommandoprozessors
- Starten des Kommandoprozessors.

4.1. Einschalttest, Laden Bootrecord

Zunächst wird ein Einschalttest durchgeführt. Hierbei werden unter anderem verschiedene Hardwarebausteine des EC 1834 getestet, die Ausstattung des Gerätes bestimmt, die Hardwarebausteine initialisiert und die Interruptvektoren für die BIOS- und DOS-Funktionen eingetragen.

Im einzelnen werden folgende Tests ausgeführt und Initialisierungen bzw. Einstellungen vorgenommen:

- Mikroprozessor-Test (Flags, Register, bedingte Sprünge)
- RAM-Test
- DMA-Controller-Test (Register-Test)
- Initialisierung Interrupt-Controller
- Einstellen von Interruptadressen
- Einlesen des Konfigurationsschalters
- Test CRT-Controller (Bestimmung Video-Typ, Einstellen Hintergrundadresse, Einstellen Cursor)
- Test Interrupt-Controller
- Test Timer
- Tastatur-Test (Tastatur-Reset, hängende Tasten)
- RAM-Test mit Bildschirmanzeige
- Test der Disketten-Laufwerke (Motor ON, verschiedene Spuren anwählen)
- Einstellen Tastaturpuffer
- Einstellen TIME OUT Zeiten für Drucker und serielle Schnittstelle
- Bildschirm löschen
- Kontrolle, ob Drucker-Karte im Gerät vorhanden (wenn ja, Eintragung in Merkmallen)
- Kontrolle, ob serielle Karte vorhanden (wenn ja, Eintragung in Merkmallen)
- Freigabe NMI-Interrupt
- Umladen des Systems (INT19).

Nachdem der Einschalttest erfolgreich durchlaufen ist, sucht das ROM-BIOS zuerst den Bootrecord auf LW A. Der Bootrecord steht auf der Spur 0, Sektor 1, Seite 0 einer jeden Diskette oder Festplatte, die durch das DOS-Kommando-Format formatiert wurde. Wird dieser dort nicht gefunden (LW nicht bereit oder Lesefehler), wird versucht, den Master-Bootrecord vom ersten Sektor der ersten Festplatte zu lesen. Verläuft die Suche auch hier erfolglos, wird eine Fehlermeldung ausgegeben. Der Bootrecord wird in den Speicher gelesen und erhält die Steuerung.

4.2. Test_System-Verfügbarkeit_und_Laden_des_RAM-BIOS

Der Bootrecord überprüft, ob im Wurzelverzeichnis des Datenträgers als erste Datei BIO.COM und als zweite Datei DOS.COM eingetragen ist.

Ist eine dieser Dateien nicht vorhanden, wird die Meldung

Diskette: Kein System oder fehlerhaft
Austauschen und eine Taste betätigen

ausgegeben.

Sind beide Dateien vorhanden, wird BIO.COM in den Speicher geladen und dem Initialisierungsteil die Steuerung übertragen.

4.3. Initialisierung_DOS_und_Laden_des_Kommandoprozessors

Im Initialisierungsteil von BIO.COM werden folgende Schritte abgearbeitet:

Systemkonfiguration einstellen

- Ermittlung der Speichergrösse
- Initialisieren der Geräte COM, LPT
- Analyse des Diskettensystems
- Analyse des Festplattensystems
- Bestimmung des Endes des residenten BIOS (Gerätetreiber)

Laden und Initialisierung von DOS

- Laden der Datei DOS.COM
- Aufruf der Initialisierungsroutine des DOS (Danach sind die Systemfunktionen von DOS verfügbar.)
- Das Laufwerk, von welchem das Umladen erfolgt, wird als aktuelles Laufwerk gesetzt.

Verarbeitung der Konfigurationsdatei CONFIG.SYS

- Test, ob die Datei CONFIG.SYS (siehe Anleitung für den Bediener Teil 1 - Kapitel 13) auf dem aktuellen Laufwerk vorhanden ist;
- Verarbeiten der Konfigurationsdatei;
- Falls keine Konfigurationsdatei vorhanden ist bzw. wenn entsprechende Kommandos fehlen, werden folgende Standardwerte gesetzt:
 - BREAK = OFF
 - BUFFERS = 2 (720 KB-Diskette = 3)
 - COUNTRY = 037
 - FCBS = 4,0
 - FILES = 8
 - LASTDRIVE = E
- Eröffnen der Standardgerätetreiber.

*** Systeminitialisierung ***

Laden und Starten des Kommandoprozessors

- Ermittlung des Kommandoprozessors (SHELL-Kommando)
- Laden des Kommandoprozessors
- Übergabe der Steuerung an den Kommandoprozessor.

4.4.1. Kommandoprozessor

Der Standard-Kommandoprozessor (COMMAND.COM) führt zunächst eine Initialisierung durch und kontrolliert anschliessend, ob die Stapeldatei AUTOEXEC.BAT (siehe Anleitung für den Bediener Teil 1 - Kapitel 10) auf dem aktuellen Laufwerk vorhanden ist. Falls diese vorhanden ist, wird sie abgearbeitet.

Ansonsten wird vom System

- zur Eingabe des Datums aufgefordert,
- zur Eingabe der Zeit aufgefordert,
- die DOS-Version angezeigt,
- das Systemprompt (z.B. A>) ausgegeben und
- die Kommandoeingabe über die Tastatur erwartet.

5. --- Datenträgerstruktur und Dateibehandlung

5.1. --- Merkmale der Dateiverwaltung

Ein Kriterium für die Leistungsfähigkeit eines Betriebssystems ist die Unterstützung der Dateiverwaltung. Durch die Organisation der Daten in Dateien werden so logisch zusammenhängende Datenbestände gespeichert, verwaltet und angesprochen. Unter anderem wird damit der Zugriff auf die in der Datei zusammengefasste Datenmenge für den Nutzer transparent. Es werden auch die An- und Abkopplung der Datenströme an den Prozess sowie die Ein- und Ausgabe von nutzerdefinierten Datenquanten möglich. Durch die Dateiverwaltung werden ausserdem die Schutzmechanismen für die Datenbestände gewährleistet. Durch Einführung eines hierarchischen, baumstrukturierten Dateiverzeichnisaufbaus können vom Nutzer Datenbestände nach seinen Gesichtspunkten zusammengefasst werden. Dadurch wird es möglich, die Datenbestände z.B. nach Nutzergruppen, Anwendungsfällen, anzuwendenden Schutzmechanismen und Zugriffsrechten oder projektorientiert zu strukturieren. Man kann entsprechend den Erfordernissen noch beliebig tiefe Baumunterstrukturen errichten. Die Baumstruktur kann auf jedem Datenträger in beliebiger Form vom Nutzer selbst eingerichtet und dynamisch modifiziert werden. Der Nutzer hat sofortigen Zugriff auf alle Dateien des Datenträgers. Es besteht aber auch die Möglichkeit, lokal in einem Unterverzeichnis und den angegliederten Unterstrukturen zu arbeiten.

Von DCP werden die magnetischen Datenträger Diskette (Floppy Disk) und Festplatte (Harddisk) unterstützt. Sie haben zwei Funktionen. Einmal kann sich auf ihnen das gesamte Betriebssystem des Gerätes befinden. Beim Initialisieren des Rechners (Boot-Vorgang) wird das Betriebssystem, gegebenenfalls über mehrere Stufen, in den Speicher geladen und dann aktiviert. Nicht residente Teile des Betriebssystems können später bei Bedarf von hier wieder nachgeladen werden. Der grössere Teil des Datenträgers wird aber zur Speicherung von Daten verwendet. Der Zugriff auf diese Daten erfolgt über das Betriebssystem. Der Datenträger wird dazu durch Dienstprogramme eingerichtet, strukturiert und verwaltet.

5.2. --- Physischer Aufbau der Datenträger

5.2.1. --- Aufbau und physische Adressräume

Die magnetischen Datenträger sind rotierende Platten bzw. Plattenstapel mit ein oder mehreren Köpfen. Jeder Plattenoberfläche wird ein Kopf zugeordnet. Die Datenspuren sind auf jeder Oberfläche in konzentrischen Kreisen um den Mittelpunkt angeordnet. Durch die Rotation der Platten befindet sich bei stillstehendem Kopf genau eine Spur im Zugriff des Kopfes. Der Spurwechsel erfolgt durch radiale Bewegung des Kopfes. Die Spuren aller Oberflächen mit identischer Spuradresse befinden sich auf dem gleichen Zylinder. Innerhalb der Spuren sind die Daten in Sektoren aufgeteilt. Auf jeder Spur befindet sich eine gleiche Anzahl

*** Datenträgerstruktur und Dateibehandlung ***

von Sektoren. Alle Sektoren des Mediums beinhalten die gleiche Anzahl von Datenbytes. Auf dem Datenträger ist die kleinste physisch verfügbare Datenmenge der Sektorinhalt. Die Sektoren werden über die Parameter

- Laufwerkadresse
 - Zylinderadresse
 - Kopfadresse
 - Sektoradresse in der Spur
- mehrdimensional adressiert.

Neben dieser mehrdimensionalen Adressierung ist im DCP auch eine eindimensionale Adresse gebräuchlich. Zur Zuordnung der Sequenznummer, mit der die Sektoren adressiert werden, findet folgendes Prinzip der Aufreihung Anwendung:

Auf dem Datenträger wird mit dem ersten Sektor des ersten Kopfes des ersten Zylinders begonnen. Dieser erhält die Sequenznummer Null. Dann folgen alle Sektoren dieser Spur in aufsteigender Reihenfolge. Die Zählung wird auf der gleichen Spur der nächsten Oberfläche analog fortgesetzt und im weiteren über alle Oberflächen des gesamten Zylinders fortgeführt. Auf allen folgenden Zylindern wird in der gleichen Art verfahren. Dadurch erhalten alle Sektoren eine eindeutige Sequenznummer zur Adressierung.

5.2.2. Formatierung

Vor Belegung des Datenträgers mit Nutzerdaten ist dieser zu initialisieren. Dazu werden über den gesamten Datenträger entsprechend seiner gewünschten Struktur (physische Formate) die Sektoren angelegt und eine Grundbelegung (00h) eingeschrieben. Anschliessend werden die einzelnen Sektoren adressiert und ihr Dateninhalt wieder gelesen. Damit wird überprüft, ob auf dem Datenträger in diesem Format Schreib-/Lesevorgänge auf allen Sektoren durchführbar sind. Andererseits wird ein entsprechendes Softformat angelegt. Beim Datenaustausch wird dann in diesen formatierten Bereichen gearbeitet. Die von DCP unterstützten Formate sind in Punkt 5.3.3.3. tabellarisch erfasst.

5.3. DCP-Bereiche auf der Diskette

5.3.1. Einteilung der Diskette

Von DCP werden alle Diskettenformate mit einer Sektorgrösse von 512 Byte erzeugt.

Die DCP-Bereiche (die Eintragungen auf einer Diskette oder die DCP-Partition auf einer Festplatte) befinden sich in folgender Reihenfolge auf dem Datenträger:

Sektor 0	Bootrecord
ab Sektor 1	Dateizuordnungstabelle (FAT 1)
	Kopie der Dateizuordnungstabelle (FAT 2)
	Stammverzeichnis
	Datenbereich

5.3.2. Der Bootrecord

Beim Initialisieren des Betriebssystems (BOOT-Vorgang) wird unter Steuerung des ROM-BIOS der Bootrecord in den Speicher eingelesen und dieses geladene Programm aktiviert. Die Abarbeitung dieses Programms stellt die zweite Stufe des BOOT-Vorganges dar.

Dieses Programm überprüft, ob die Dateien BIO.COM sowie DOS.COM auf diesem Datenträger vorhanden sind, d.h. ob dieser Datenträger eine Systemdiskette ist. Trifft dieses nicht zu, wird der Boot-Vorgang unterbrochen und eine Fehlermeldung abgegeben. Andernfalls wird mit Laden und Aktivieren der Betriebssystemdatei BIO.COM die nächste Stufe der Systeminitialisierung eingeleitet.

Der Bootrecord wird vom FORMAT-Kommando auf jeder formatierten Diskette auf Seite 0/Spur 0/Sektor 1 angelegt. Auf Festplatten befindet sich der Bootrecord auf dem ersten Sektor der DOS-Partition. Damit ist gesichert, dass dieses Programm auf allen DCP-Datenträgern vorhanden ist.

5.3.3. Die Dateizuordnungstabelle (FAT)

Dateien werden nicht unbedingt sequentiell auf eine Diskette geschrieben. Die Belegungsverwaltung des Datenbereiches erfolgt dynamisch, d.h., werden während der Dateiarbeit (Schreiboperationen) weitere Datenbereiche benötigt, wird ein freier Datenbereich gesucht. Der erste auf der Diskette verfügbare Platz wird der Datei zugewiesen und somit belegt.

Beim Löschen von Dateien werden die zugehörigen Datenbereiche wieder freigegeben und damit für spätere Anforderungen zur Verfügung gestellt.

Die Belegung der Bereiche wird in der FAT registriert. Die FAT gibt zu jeder Zeit einen Überblick über die Belegung des Datenträgers.

Die kleinste Einheit des Datenträgers, der auf diese Weise verwaltet wird, ist die Gruppe (Cluster). Sie wird aus ein oder mehreren aufeinanderfolgenden Sektoren gebildet. Dadurch entsteht eine geordnete Reihenfolge gleichgrosser Gruppen. Die Gruppennummer wird zu deren Adressierung herangezogen. Die Gruppen 0 und 1 existieren nicht. Der erste Sektor des Datenbereiches ist der erste Sektor der Gruppe 2. Die Anzahl Sektoren, die eine Gruppe bilden, sind formatabhängig.

Die Dateizuordnungstabelle wird von DCP zur Verkettung von Diskettenbereichen für eine Datei benutzt, wobei die entstehende Sequenz den Datenstrom der Datei darstellt.

Durch die Verkettung der Gruppen ist sowohl sequentieller als auch wahlfreier Zugriff auf die Dateien realisierbar.

Der Zugriff auf eine Datei erfolgt in folgender Weise:

In der FAT ist jedem Tabellenelement indexiell eine Gruppennummer im Datenbereich zugeordnet. In der Dateiverzeichniseintragung steht die Nummer der ersten von der Datei belegten Gruppe. Über diesen Index erreicht man das zugeordnete FAT-Element. Dort steht die Nummer der nächsten von dieser Datei belegten Gruppe.

*** Datenträgerstruktur und Dateibehandlung ***

Auf diese Weise wird die Sequenz der Datenbereiche beschrieben, die zur Datei gehören. Erreicht man das Endekennzeichen statt eines weiterverweisenden Zeigers in der FAT, so ist diese Gruppe der letzte Datenbereich dieser Datei.

Die Elemente der Dateizuordnungstabelle bestehen aus einem 12-Bit-Eintrag (1,5 Bytes bzw. 3 Hex-Zeichen) für jede Gruppe auf der Diskette. Wenn ein Datenträger mehr als 4095 Gruppen besitzt, dann besteht die Dateizuordnungstabelle aus 16-Bit-Einträgen (2 Byte bzw. 4 Hex-Zeichen).

Die Eintragungen haben folgende Bedeutung:

- (0)000 Gruppe ist nicht belegt und damit z.Zt. verfügbar
- (F)FF0-(F)FF7 werden für die Anzeige von reservierten Gruppen benutzt
- (F)FF7 gibt eine fehlerhafte Gruppe an, wenn sie nicht Teil einer Belegungskette ist
- (F)FF8-(F)FFF Angabe der letzten Gruppe der Datei (Endekennzeichen)
- (X)XXX beliebiges anderes Hexadezimalzeichen
Dabei handelt es sich um die Gruppennummer der nächsten Gruppe in der Datei. Die Gruppennummer der ersten Gruppe in der Datei steht in dem Eintrag des Verzeichnisses für diese Datei.

Die Dateizuordnungstabelle beginnt immer in dem logischen Sektor 1 (dies ist der zweite aktuelle Sektor auf einer Diskette oder in einer Partition auf der Festplatte) unmittelbar nach dem Bootrecord.

Belegt die Dateizuordnungstabelle mehr als einen Sektor, so sind die folgenden Sektoren belegt.

Aus Sicherheitsgründen wird noch eine Kopie der Dateizuordnungstabelle geführt, wobei FAT1 unmittelbar auf FAT2 folgt.

Die Grösse von FAT1 und FAT2 hängt von der Anzahl Gruppen ab, die die Kapazität des Datenträgers beschreiben.

Die Dateizuordnungstabelle wird bei Dateiverwaltungsarbeiten (beim Eröffnen, Zuweisen von weiterem Speicherplatz usw.) in einen der DCP-Puffer hoher Priorität gelesen, um die Tabelle aus Leistungsgründen so lange wie möglich im Speicher halten zu können.

Zur Umwandlung der Gruppennummer in eine logische Sektoradresse (relativer Sektor, wie beispielsweise der von INT 25 und 26 sowie von DEBUG benutzte Sektor) wird wie folgt vorgegangen:

1. Von der Gruppennummer wird 2 subtrahiert.
2. Das Ergebnis wird mit der Anzahl von Sektoren pro Gruppe multipliziert.
3. Die logische Sektoradresse des Anfangs des Datenbereichs wird hinzuaddiert.

5.3.3.1. Benutzung der Dateizuordnungstabelle für 12-Bit-Einträge

Jeweils 3 zusammenhängende Byte (24 Bit) stellen 2 aufeinanderfolgende Einträge der FAT dar. Die unteren 12 Bit werden der geraden Gruppennummer, die oberen 12 Bit der folgenden ungeraden Gruppennummer zugeordnet.

Die Startgruppe der Datei wird aus dem Verzeichniseintrag entnommen. Um jede nachfolgende, der Datei zugeordnete Gruppe lokalisieren zu können, wird wie folgt vorgegangen:

1. Multiplizieren der gerade benutzten Gruppennummer mit 1,5 (jeder FAT-Eintrag hat eine Länge von 12 Bit).
2. Der ganze Teil dieses Produktes ist ein Offset in die Dateizuordnungstabelle und adressiert ein Datenwort. Dieses enthält die Gruppennummer der nächsten Gruppe der Datei.
3. War die zur Adressierung benutzte Gruppe eine gerade Zahl, so sind die 12 niederwertigen Bit (0FFFh) des Registers gültig. Ansonsten stellen die 12 höherwertigen Bit (FFF0h) den entsprechenden Eintrag dar.
4. Ergeben die 12 Bit einen Wert aus dem Bereich FF8h-FFFh, so sind keine weiteren Gruppen in der Datei enthalten. Ansonsten enthalten die 12 Bit die Gruppennummer der nächsten Gruppe in der Datei.

5.3.3.2. Benutzung der Dateizuordnungstabelle für 16-Bit-Einträge

Jeweils ein Datenwort in der FAT entspricht einer Gruppeneintragung.

Die Startgruppe der Datei muss aus dem Verzeichniseintrag entnommen werden. Um jede nachfolgende Gruppe der Datei lokalisieren zu können, wird wie folgt vorgegangen:

1. Multiplizieren der gerade benutzten Gruppennummer mit 2 (jeder FAT-Eintrag hat eine Länge von 16 Bit).
2. Das Ergebnis ist ein Offset in die FAT.
3. Das damit adressierte Datenwort enthält die nachfolgende Gruppennummer.
4. Stellen die sich ergebenden 16 Bit die Werte FFF8h-FFFFh dar, so sind keine weiteren Gruppen in der Datei enthalten. Ansonsten enthalten die 16 Bit die Gruppennummer der nächsten Gruppe in der Datei.

5.3.3.3. Das Medienkennzeichen

Auf den Tabellenplätzen für die Gruppen 0 und 1 werden in der FAT andere Systeminformationen eingetragen.

Diese FAT-Einträge enthalten Hinweise auf die Grösse und das Format des Datenträgers; sie identifizieren diesen.

*** Datenträgerstruktur und Dateibehandlung ***

Es sind

- bei 12-Bit-Eintrag
 - Gruppe 0: Fmm h
 - Gruppe 1: FFF h
 - bei 16-Bit-Eintrag
 - Gruppe 0: FFmm h
 - Gruppe 1: FFFF h
- eingetragen.

Das 1. Byte der FAT stellt das Mediumkennzeichen (mm) dar und charakterisiert folgende Formate:

FFh	Doppelseitige Diskette:	40	Spuren mit je	8	Sektoren
FEh	Einseitige Diskette:	40	Spuren mit je	8	Sektoren
FDh	Doppelseitige Diskette:	40	Spuren mit je	9	Sektoren
FCh	Einseitige Diskette:	40	Spuren mit je	9	Sektoren
F9h	Doppelseitige Diskette:	80	Spuren mit je	9	Sektoren
F8h	Festplatte				

5.3.4. Das Dateistammverzeichnis

Das Dateiverzeichnis enthält für jede Datei auf dem Datenträger genau eine Eintragung. Diese enthält alle Dateiparameter zur vollständigen Beschreibung der Datei. Dieser Eintrag wird bei Dateizugriffen herangezogen und modifiziert.

Mit dem Dienstprogramm FORMAT wird das Stammverzeichnis entsprechend dem Diskettenformat eingerichtet und mit Initialwerten belegt. Seine Speicherposition (logische Sektoradresse) und seine Größe sind formatabhängig. Speziell gekennzeichnete Eintragungen stellen keinen Dateieintrag dar, sondern werden als Datenträgerkennsatz, Zeiger auf ein Unterverzeichnis, Zeiger auf das übergeordnete Verzeichnis usw. interpretiert.

Da es sich bei den Verzeichnissen, mit Ausnahme des Stammverzeichnisses, in Wirklichkeit um spezielle Dateien handelt, gibt es keine Begrenzung für die Anzahl von Einträgen in diesen Verzeichnissen. Untergeordnete Verzeichnisse können als Dateien gelesen werden. Dabei wird ein erweiterter Dateisteuerblock mit dem entsprechenden Attributbyte benutzt.

Werden neue Dateieinträge in das Stammverzeichnis aufgenommen, wird der erste unbelegte Platz dafür benutzt. Beim Aufsuchen einer Datei im Verzeichnis wird sequentiell gesucht. Durch diese gepackte Belegung des Verzeichnisses wird die Suchsequenz verkleinert.

Struktur des Dateieintrages

Sämtliche Verzeichniseinträge haben eine Länge von 32 Bytes und weisen das folgende Format auf (Angabe aller Offset's in Hexadezimalform):

00h-07h Eintrag-Kennzeichen

Das erste Byte dieses Feldes gibt den Status an:

00h Eintragplatz wurde bisher nie benutzt. Dadurch wird die Länge der Suchsequenz beschränkt.

E5h Platz wurde schon benutzt; die Datei ist jedoch gelöscht worden. Der Platz kann für neuen Eintrag genutzt werden.

2Eh Dieser Eintrag ist für ein Verzeichnis bestimmt. Ist das zweite Byte ebenfalls 2Eh, so enthält das Gruppenfeld die Gruppennummer des übergeordneten Verzeichnisses (0000h, wenn es sich bei dem übergeordneten Verzeichnis um das Stammverzeichnis handelt).

Bei jedem anderen Zeichen handelt es sich um das erste Zeichen eines Dateinamens.

00h-07h Dateiname

Wenn das erste Zeichen kein vorgenanntes Sonderzeichen ist, stellen diese acht Zeichen, linksbündig und gegebenenfalls mit Leerzeichen aufgefüllt, den Dateinamen dar.

08h-0Ah Dateierweiterung

08h Dateiattribut

Das Attributbyte ist folgendermassen zu interpretieren:

01h Kennzeichen für Dateiattribut Nur Lesen
Wird versucht, die Datei auch für Ausgabezwecke zu eröffnen, so wird ein Fehlercode zurückgegeben. Auch das Löschen der Datei führt zum Fehlerstatus.

02h Kennzeichen für Verborgene Datei
Diese Datei ist von der normalen Verzeichnisdurchsuchung ausgeschlossen und für den Nutzer im Dateiverzeichnis nicht sichtbar.

04h Kennzeichen für Systemdatei
Diese Datei ist von der normalen Verzeichnisdurchsuchung ausgeschlossen.

08h Kennzeichen für Datenträgerkennsatz
Dieser Eintrag enthält den Datenträgerkennsatz in den ersten 11 Bytes. Der Eintrag enthält keine weiteren nützlichen Informationen und darf nur im Stammverzeichnis vorhanden sein.

*** Datenträgerstruktur und Dateibehandlung ***

10h Kennzeichen für Untergeordnetes Verzeichnis
Dieser Eintrag definiert ein untergeordnetes Verzeichnis und ist aus der normalen Verzeichnisdurchsuchung ausgenommen.

20h Kennzeichen für Archiv
Dieses Bit wird gesetzt, nachdem eine Datei geschrieben und geschlossen wurde. Es wird von den Dienstprogrammen benutzt (abgefragt und verwaltet), z.B. um festzustellen, ob die Datei seit der letzten Sicherung geändert wurde oder nicht. Dieses Bit kann zusammen mit den anderen Attributbits benutzt werden.

Hinweis: Die Systemdateien (BIO.COM und DOS.COM) können nur gelesen werden. Ferner sind sie als **Verborgene** und **Systemdateien** gekennzeichnet. Dateien können schon beim Erstellen als verborgen gekennzeichnet werden. Die Attribute **Nur Lesen**, **Verborgene Dateien** und **Systemdateien** sowie die Archivierungsattribute können über den Funktionsaufruf 43h geändert werden.

0Ch-15h Reserviert

16h-17h Zeit

Zeitpunkt der Dateierstellung oder der letzten Aktualisierung! Die Zeit wird innerhalb des Datenwortes in den Bits wie folgt abgebildet:

```
<          17h          > <          16h          >  
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0  
<    hh    > <    mm    > <    ss    >
```

Dabei ist:

hh die binäre Zahl für Stunden (00h bis 17h).
mm die binäre Zahl für Minuten (00h bis 3Bh).
ss die binäre Zahl für 2-Sekunden-Intervalle.

18h-19h Datum

Datum der Dateierstellung oder der letzten Aktualisierung. Die Datumsangabe in Form von tt.mm.jj wird im Datenwort wie folgt abgebildet:

```
<          19h          > <          18h          >  
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0  
<    jj    > <    mm    > <    tt    >
```

*** Datenträgerstruktur und Dateibehandlung ***

Dabei ist :

tt die binäre Zahl für den Tag (01h bis 1Fh)
mm die binäre Zahl für den Monat (1h bis Ch)
jj die binäre Zahl für die Jahre 1980-2079
(00h bis 77h)

1Ah-1Bh Startgruppe

Relative Gruppennummer der ersten Gruppe der Datei
Hier muss darauf geachtet werden, dass die erste Gruppe
des Datenbereiches auf allen Datenträgern immer die
Gruppe 002 ist. Die Gruppennummer steht als Datenwort.

1Ch-1Fh Dateigrösse

Die 4 aufeinanderfolgenden Bytes sind als Doppelwort zu
interpretieren (niederwertigstes Bit ist das unterste
Bit des untersten Bytes).

5.4. Organisation der Festplatte

5.4.1. Arbeitsprinzipien

Die Festplatte kann in bis zu vier unabhängige Teile (Partitions) unterteilt werden. Jede dieser Partition ist dann wie eine Diskette (siehe Punkt 5.3.) organisiert.

Zu einer Partition gehören immer ein oder mehrere aufeinanderfolgende Zylinder auf dem Datenträger. Jeder Zylinder darf nur einer Partition zugeordnet werden. Eine Partition kann allerdings auch die gesamte Festplatte belegen.

Es ist immer nur eine der eingerichteten Partitions aktiv und dem aktuellen Betriebssystem zugeordnet.

Da die Partition wie eine Diskette im System behandelt wird, kann das Betriebssystem auch von der aktivierten Partition gebootet werden.

Zur Unterstützung von Festplatten mit sehr grosser Kapazität kann eine erweiterte DCF-Partition eingerichtet werden. Innerhalb dieser Partition können mehrere logische Einheiten mit der Unterstruktur eines logischen Laufwerkes angelegt werden. Neben der aktiven primären Partition kann über diese zusätzlichen Datenbereiche verfügt werden. Von erweiterten Partitions kann keine Systeminitialisierung erfolgen.

Die Dienst- und Nutzerprogramme sind so aufzubauen, dass sie nur mit den zugehörigen Partitions arbeiten.

5.4.2. Aufbau der Festplatte

Jede der Partitions beginnt am Anfang eines Zylinders (Adresse Zyl, x, Kopf 0, Sektor 1). Eine Ausnahme bildet die erste Partition, die erst auf der 2. Spur des 1. Zylinders (Adresse Zyl 0, Kopf 1, Sektor 1) einsetzt. Die 1. Spur der Festplatte ist für den Master-Bootrecord reserviert, der eine Tabelle zur Einteil-

*** Datenträgerstruktur und Dateibehandlung ***

lung der Festplatte in Partitions sowie ein Programm enthält, das während der Initialisierung aktiviert werden kann. Unterschieden werden primäre DCP-Partitions und erweiterte DCP-Partitions, die durch den Systemindikator in der Partition-Tabelle gekennzeichnet sind.

Eine primäre DCP-Partition ist wie eine Diskette strukturiert. Sie enthält einen Bootrecord und einen Datenbereich.

Die erweiterten DCP-Partitions enthalten ein oder mehrere erweiterte logische Einheiten, die alle am Anfang eines Zylinders beginnen (Adresse Zylinder x, Kopf 1, Sektor 1).

Jede dieser erweiterten logischen Einheiten hat eine Struktur ähnlich dem physischen Aufbau der gesamten Festplatte. In der 1. Spur der logischen Einheit (Adresse Zylinder x, Kopf 0, Sektor 1) befindet sich ein erweiterter Partition-Bootrecord. Dieser enthält eine logische Laufwerktafel mit analoger Struktur zur Partition-Tabelle. Durch sie wird das logische Laufwerk, das auf dem gleichen Zylinder beginnt (Adresse Zylinder x, Kopf 1, Sektor 1), beschrieben. Ausserdem enthält sie einen Verkettungszeiger mit der Adresse der nachfolgenden erweiterten logischen Einheit innerhalb der DCP-Partition.

Die logischen Laufwerke sind wie Disketten organisiert; der 1. Sektor enthält einen Bootrecord.

5.4.3. Master-Bootrecord

Die Informationen über den Festplatten-Aufbau befinden sich im ersten physischen Sektor, im Master-Bootrecord.

Er enthält ein Programm zur Übergabe der Steuerung an die aktive Partition der Festplatte, das während der Systeminitialisierung von der Festplatte geladen und abgearbeitet wird.

Weiterhin ist im Master-Bootrecord eine Partition-Tabelle enthalten, in der sich die Partition-Parameter befinden.

Ein analog aufgebauter Bootrecord existiert am Anfang jedes logischen Blockgerätes in einer erweiterten Partition.

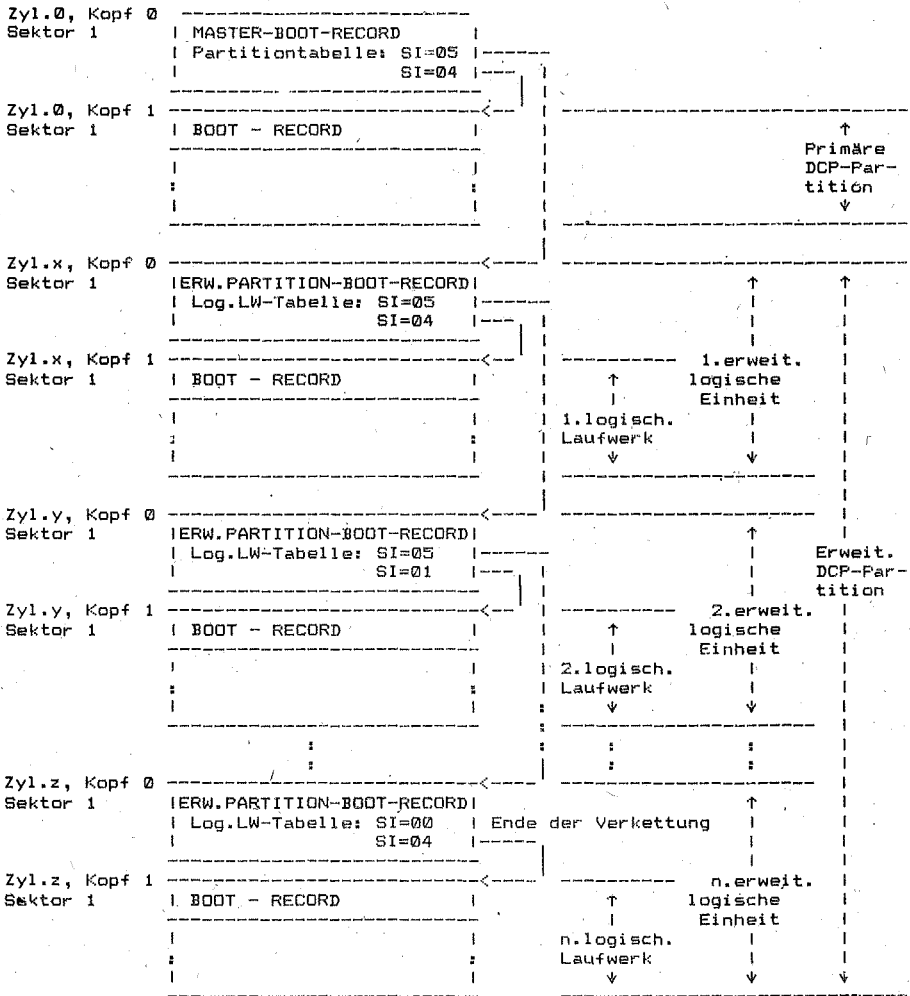
5.4.4. Partition-Tabelle

Die Partition-Tabelle enthält 4 Tabelleneintragungen mit jeweils 16 Byte (eine Eintragung für jede der 4 Partitions).

Am Ende der Partition-Tabelle sind Signaturbytes zur Kennung eines gültigen Master-Bootrecords eingetragen.

Offset	Inhalt
1BEh - 1CDh	Parameter der 1. Partition
1CEh - 1DDh	Parameter der 2. Partition
1DEh - 1EDh	Parameter der 3. Partition
1EEh - 1FDh	Parameter der 4. Partition
1FEh - 1FFh	2 Signaturbytes (55AAh)

*** Datenträgerstruktur und Dateibehandlung ***



Struktur der Festplatte

SI = Systemindikator

5.4.4.1. Aufbau einer Eintragung in der Partition-Tabelle

Die Parameter sind so in der Tabelle organisiert, dass durch einfache Ladebefehle die Register zum Aufruf des INT13 vorbereitet werden können.

Die 16 Byte einer Eintragung sind folgendermassen aufgebaut:

- Boot-Indikator:
 - 1. Byte 00h = Partition beinhaltet kein ladbares Betriebssystem
 - 80h = Partition beinhaltet ladbares Betriebssystem
- Startadresse der Partition (mehrdimensional):
 - 2. Byte Kopfadresse des Partition-Beginns
 - 3. Byte Sektoradresse des Partition-Beginns (Bit 6 und 7 sind die höchsten 2 Bit der Zylinder)
 - 4. Byte Zylinderadresse des Partition-Beginns (die höchsten 2 Bit befinden sich im Sektoradressen-Byte)
- System-Indikator:
 - 5. Byte 00h = unbekannte Partition
 - 01h = primäre DCP-Partition mit 12-Bit-FAT
 - 04h = primäre DCP-Partition mit 16-Bit-FAT
 - 05h = erweiterte DCP-Partition
- Endadresse der Partition (mehrdimensional):
 - 6. Byte Kopfadresse des Partition-Endes
 - 7. Byte Sektoradresse des Partition-Endes (Bit 6 und 7 sind die höchsten 2 Bit der Zylinderadresse)
 - 8. Byte Zylinderadresse des Partition-Endes (die höchsten 2 Bit befinden sich im Sektoradressen-Byte)
- Kopfadresse des Partition-Beginns (eindimensional)
 - 9./12. Byte Anzahl der Sektoren, die der Partition vorangehen (als Doppelwort)
- Länge der Partition (eindimensional)
 - 13./16. Byte Anzahl der Sektoren, die die Partition belegt (als Doppelwort).

5.4.4.2. Bedeutung des Boot-Indikator

Der Boot-Indikator 80h wird durch FDISK.COM für eine primäre Partition gesetzt, wenn diese als aktiv bzw. ladbar gekennzeichnet werden soll. Die Boot-Indikatoren der restlichen 3 Partitions werden gleichzeitig auf 0 (inaktiv) gesetzt.

Bei Einbeziehung der Festplatte zum Booten des Systems wird das Programm im Master-Bootrecord aktiviert. Dieses überprüft, ob für eine primäre Partition der Boot-Indikator gesetzt ist. Wenn das nicht der Fall ist, ist die Festplatte kein Medium zum Booten. Andernfalls wird die Steuerung an das Programm weitergegeben, das sich im Bootrecord der aktiven Partition befindet und dessen Sektoradresse als Startadresse der aktiven Partition in der Partition-Tabelle eingetragen ist.

5.4.5. Aufbau einer erweiterten DCP-Partition

Die erweiterte DCP-Partition ist zur Unterstützung von Festplatten mit hoher Kapazität eingeführt worden. Dieser Partition-Typ ist in der Partition-Tabelle durch den Systemindikator (05h) gekennzeichnet. Diese Partition dient einzig der Erweiterung der Datenkapazität, die durch DCP im Zugriff steht. Sie ist nicht bootbar. Deshalb darf der Boot-Indikator in der Partition-Tabelle für diesen Typ nicht setzbar sein.

Auf einem bootbaren Laufwerk kann eine erweiterte DCP-Partition nur angelegt werden, wenn bereits eine primäre DCP-Partition existiert. Kann vom betreffenden Laufwerk nicht gebootet werden, ist die Existenz einer primären DCP-Partition zum Anlegen einer erweiterten DCP-Partition nicht erforderlich. Von DCP werden nur erweiterte Partitions bis zu einer Kapazität von 32 MB unterstützt.

In der erweiterten DCP-Partition können mehrere erweiterte logische Einheiten eingerichtet werden. Diese Einheiten bestehen jeweils aus einem erweiterten Partition-Bootrecord und einem logischen Laufwerk. Sie sind analog der gesamten Festplatte strukturiert. Dieser Bootrecord entspricht dem Master-Bootrecord, und das logische Laufwerk entspricht der DCP-Partition. Das logische Laufwerk beginnt mit dem DCP-Bootsektor und steht nach dem erweiterten Partition-Bootrecord auf der physischen Platte.

Jede erweiterte Einheit enthält einen erweiterten Partition-Bootrecord in seinem ersten Sektor (gekennzeichnet durch die Signaturbytes 55AAh). Hiermit wird eine Kompatibilität zu Programmen erreicht, die an dieser Stelle einen Master-Bootrecord erwarten. Dieser Record enthält eine Tabelle für das logische Laufwerk, analog der Partition-Tabelle des Master-Bootrecords der Festplatte.

5.4.5.1. Struktur der Tabelle der logischen Laufwerke

Die Laufwerkstabelle enthält 4 Tabelleneintragungen mit jeweils 16 Byte. Am Ende der Tabelle sind zur Kennzeichnung als Bootrecord Signaturbytes eingetragen. Die Tabelle hat folgende Form:

Offset	Inhalt
1BEh - 1CDh	Parameter des 1. logischen Laufwerkes
1CEh - 1DDh	Parameter des 2. logischen Laufwerkes
1DEh - 1EDh	Parameter des 3. logischen Laufwerkes
1EEh - 1FDh	Parameter des 4. logischen Laufwerkes
1FEh - 1FFh	2 Signaturbytes (55AAh)

Die Belegung der anderen Bytes innerhalb des Bootrecords ist unbedeutend, da die erweiterten Partitions nicht bootfähig sind.

5.4.5.2. Aufbau einer Eintragung in der Laufwerkstabelle

Die 16 Byte beschreiben die Lage und Grösse der logischen Laufwerke innerhalb der erweiterten Partition. Ausserdem enthalten sie einen Verkettungszeiger zum nächsten erweiterten Volume. Die 16 Byte für jeweils eine Laufwerkeintragung haben folgende Bedeutung:

- Boot-Indikator:

1. Byte ohne Bedeutung, da nicht bootbar

- System-Indikator:

5. Byte 00h = diese logische Laufwerk existiert nicht
01h = DCF-Laufwerk mit 12-Bit-FAT
04h = DCF-Laufwerk mit 16-Bit-FAT
05h = Eintragung weist auf nächstes erweitertes Volume
06h = reserviert

- für System-Indikator mit Werten 01h, 04h oder 06h gilt:

-- Startadresse des Laufwerkes (mehrdimensional)

relativ zum Beginn der erweiterten Partition:

2. Byte Kopfadresse des Laufwerk-Beginns
3. Byte Sektoradresse des Laufwerk-Beginns (Bit 6 und 7 sind die höchsten 2 Bit der Zylinder)
4. Byte Zylinderadresse des Laufwerk-Beginns (die höchsten 2 Bit befinden sich im Sektoradressen-Byte)

-- Endadresse des Laufwerkes (mehrdimensional)

relativ zum Beginn der erweiterten Partition:

6. Byte Kopfadresse des Laufwerk-Endes
7. Byte Sektoradresse des Laufwerk-Endes (Bit 6 und 7 sind die höchsten 2 Bit der Zylinderadresse)
8. Byte Zylinderadresse des Laufwerk-Endes (die höchsten 2 Bit befinden sich im Sektoradressen-Byte)

-- Anfangsadresse des Laufwerkes (eindimensional)

relativ zum Beginn der erweiterten Partition:

9./12. Byte Anzahl der Sektoren bis zum Beginn des Laufwerkes (als Doppelwort)

-- Kapazität des Laufwerkes (eindimensional):

13./16. Byte Anzahl der Sektoren, die das Laufwerk belegt (als Doppelwort).

- für System-Indikator 00h

sind alle Bytes des Eintrages mit 00h belegt.

***** Datenträgerstruktur und Dateibehandlung *****

- für System-Indikator 05h gilt:
 - Startadresse der nächsten erweiterten Partition (mehrdimensional) relativ zum Beginn der physischen Festplatte:
 - 2. Byte Kopfadresse des Partition-Beginns
 - 3. Byte Sektoradresse des Partition-Beginns (Bit 6 und 7 sind die höchsten 2 Bit der Zylinder)
 - 4. Byte Zylinderadresse des Partition-Beginns (die höchsten 2 Bit befinden sich im Sektoradressen-Byte)
 - Endadresse der nächsten erweiterten Partition (mehrdimensional) relativ zum Beginn der physischen Festplatte:
 - 6. Byte Kopfadresse des Partition-Endes
 - 7. Byte Sektoradresse des Partition-Endes (Bit 6 und 7 sind die höchsten 2 Bit der Zylinderadresse)
 - 8. Byte Zylinderadresse des Partition-Endes (die höchsten 2 Bit befinden sich im Sektoradressen-Byte)
 - Anfangsadresse der nächsten erweiterten Partition (eindimensional) relativ zum Beginn der physischen Festplatte:
 - 9./12. Byte Anzahl der Sektoren bis zum Beginn des Laufwerkes (als Doppelwort)
 - Kapazität der nächsten erweiterten Partition (eindimensional):
 - 13./16. Byte ist mit 00h aufzufüllen

5.4.6. Konfigurieren der Partitions

Die Konfiguration der Partitions und damit der gesamten Festplatte wird durch die Partition-Tabelle beschrieben. Mit dem Dienstprogramm FDISK werden die Partitions auf dem Harddisk eingerichtet. Durch Vorgabe der Größe der Partition wird, wenn eine ausreichende Anzahl zusammenhängender freier Zylinder existieren, eine entsprechende Partition eingerichtet und die zugehörigen Eintragungen in der Partition-Tabelle vorgenommen.

Bei Löschung einer Partition geht auch der Dateninhalt für den Nutzer verloren.

Bei der Formatierung durch FDISK wird der Master-Bootrecord initialisiert und somit eine bestehende Einteilung in Partitions und damit auch alle Daten in den einzelnen Partitions zerstört. Mit dem Programm FDISK kann eine der eingerichteten primären Partitions auch aktiviert werden.

5.4.7. Formatieren der Partitions

Mit dem Dienstprogramm FORMAT wird immer nur die aktive Partition formatiert und alle entsprechenden Operationen auf dieses Diskettenäquivalent angewendet.

5.5. Beschreibung der Datenträgerformate

Die Datenträgerformate werden durch die Parameter

- Mediumkennzeichen (s. Punkt 5.3.3.3.) und
- BIOS-Parameterblock beschrieben.

Diese Parameter werden beim Zugriff des Betriebssystems auf die Datenträger ausgewertet.

5.5.1. Der BIOS-Parameterblock (BPB)

Der BPB enthält alle Parameter über die logische Organisation einer Diskette bzw. Partition auf der Festplatte.

Struktur des BPB:

BPB	FMT	Inhalt
+00	dw	Anzahl Byte pro Sektor
+02	db	m; wenn die Anzahl Sektoren in der Gruppe 2^m ist
+03	dw	vom System reservierte Sektoreanzahl
+05	dw	Anzahl der FAT's
+06	dw	Anzahl der maximal möglichen Dateieintragen im Stammverzeichnis
+08	dw	Gesamtkapazität der Partition (Anzahl Sektoren)
+0A	db	Datenträgerkennzeichen
+0B	dw	Anzahl Sektoren für eine FAT
+0D	dw	Sektoren pro Spur
+0F	dw	Anzahl Köpfe im Zylinder
+11	dw	Anzahl verborgener Sektoren
+13	db	Länge der Gruppennummern in der FAT (00: 12 Bit; 01: 16 Bit)

5.5.2. Parameter der unterstützten Formate

Durch DCP werden folgende logische Formate der Datenträger unterstützt:

Mediumkennzeichen	FDh	F9h	F8h	FCh	FEh	FFh
Byte/Sektor	512	512	512	512	512	512
Sektor/Gruppe	2	2	4	1	1	2
Verschiebefaktor	1	1	2	0	0	1
reserv. Sektoren	1	1	1	1	1	1
Anzahl FATs	2	2	2	2	2	2
Zahl Stammeinträge	112	112	512	64	64	112
Beginn Datenbereich	12	14	161	9	7	10
Anzahl Gruppen	354	713	16326	351	313	315
Sektoren/FAT	2	3	64	2	1	1
Beginn Stammver- zeichnis	5	7	129	5	3	3

5.6. Dateibehandlung

5.6.1. Arten des Dateiaufrufes

Zum Erzeugen, Eröffnen, Schliessen, Lesen, Schreiben, Umbenennen, Finden und Löschen von Dateien existieren DOS-Funktionsaufrufe. Es werden von DOS zwei Arten unterstützt.

- Funktionsaufruf mit Dateisteuerblock (FCB) (Funktionsaufrufe 0Fh-24h)

Dieser Steuerblock wird vom Anwender angelegt und bildet den Verständigungsbereich zwischen DCP und dem Nutzer. Das DCP verwaltet bei jedem Funktionsaufruf den FCB.

Als Einschränkung gilt, dass nur Dateien im eingestellten Unterverzeichnis ansprechbar sind.

- Erweiterte Funktionsaufrufe (Händler-Funktionsaufrufe 39H-62H).

Händler-Funktionsaufrufe sind leichter und komfortabler nutzbar als FCB-Funktionsaufrufe.

Bei Dateiarbeit mit Händleraufruf können Dateien aller Verzeichnisse angesprochen werden. Die Dateien werden durch eine Zeichenkette mit den Merkmalen Laufwerk, Pfad und Dateibezeichnung angesprochen, so dass alle Dateien eines Datenträgers erreichbar sind. Bei Eröffnung des Händlers wird vom System eine Händlernummer vergeben, die für die weitere Arbeit mit der Datei als Identifikator dient.

Der Händler stellt ein einheitliches Niveau für Dateien und Geräte dar.

5.6.2. Funktionsaufruf mit FCB

5.6.2.1. Arbeitsweise

Die FCB-Funktionsaufrufe erfordern die Benutzung eines Dateisteuerblockes für jede eröffnete Datei. Dieser wird vom Anwendungsprogramm und von DOS verwaltet.

Das Anwendungsprogramm stellt beim Funktionsaufruf einen Zeiger auf den FCB zur Verfügung. Es sind dazu die entsprechenden Parameterfelder, die bei einem spezifischen Funktionsaufruf gefordert werden, im FCB voreinzustellen.

Da im FCB der Pfad nicht angebbar ist, bezieht sich die Bezeichnung immer auf den aktuellen Pfad.

Die sequentiellen E/A-Operationen Lesen und Schreiben verwalten die entsprechenden Felder (aktueller Block, aktueller Satz, Satzlänge) selbst.

Bei wahlfreien E/A-Operationen muss der Nutzer das Feld für die Satznummer selbst verwalten. Ein Übergang vom wahlfreien auf den sequentiellen Zugriff ist automatisch möglich, da auch bei wahlfreiem Zugriff die Felder für sequentiellen Zugriff vom DCP mitgeführt werden.

Die FCB-Struktur wird im Punkt 5.3.6.2.3. beschrieben.

*** Datenträgerstruktur und Dateibehandlung ***

Der Anwender muss für jede eröffnete Datei einen eigenen FCB zur Verfügung stellen. Erst nach Schliessen der Datei kann dieser Speicherbereich wieder freigegeben werden.

Die vom DCP im FCB verwalteten Felder dürfen vom Nutzer nicht modifiziert werden, da sie betriebssysteminterne Informationen enthalten. Sie sind als reserviert gekennzeichnet.

Es ist zu empfehlen, nach Beendigung aller E/A-Operationen alle Dateien zu schliessen. Eine bereits eröffnete Datei sollte nicht gelöscht bzw. umbenannt werden, da dies ebenfalls einen Fehler zur Folge hat.

5.6.2.2. Anzahl der eröffneten FCB-Dateien

Bei Dateiarbeit ohne Mehrfachzugriff auf eine Datei ist die Anzahl der gleichzeitig eröffnenbaren Dateien unbeschränkt, da der Nutzer für jede Datei einen FCB anlegt.

Bei gemeinsamem Dateizugriff, unterstützt durch das Programm SHARE, ist die Anzahl der maximal über FCB eröffneten Dateien beschränkt. Diese Standardanzahl kann durch das FCBS-Kommando in der CONFIG.SYS-Datei variiert werden. Als Parameter sind in diesem Kommando die Gesamtzahl der eröffnenbaren Dateien sowie die Anzahl der zuerst eröffneten Dateien, die vor automatischem Schliessen geschützt sind, angebar.

Wenn dann während der Programmabarbeitung die maximale Anzahl von zugelassenen Dateieröffnungen überschritten wird, so schliesst DCP automatisch die älteste nicht vom Schliessen geschützte Datei.

Jeder Zugriffsversuch auf eine solche Datei erzeugt die kritische Fehlermeldung "FCB nicht verfügbar". Abhilfe kann geschaffen werden durch Spezifikation einer grösseren Gesamtzahl eröffnenbarer Dateien unter Anwendung des FCBS-Kommandos.

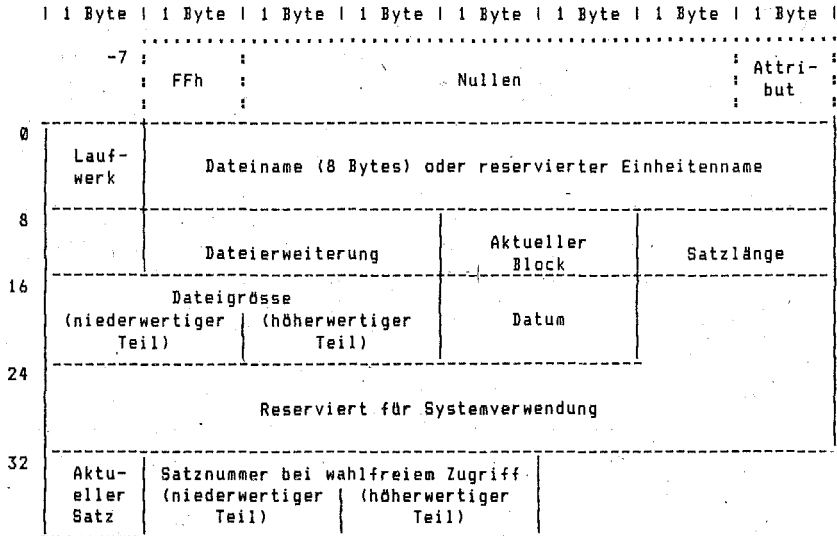
5.6.2.3. Struktur des Dateisteuerblocks (FCB)

FCB (Dateisteuerblock)

Für die allgemeine Dateiarbeit genügt die Verwendung eines Standard-FCB. Beim Aufruf der entsprechenden DCP-Funktion wird die Adresse des Standard-FCB als Parameter übergeben.

Sollen Dateibefehle ausgeführt werden, die als Eingabeparameter auch das Dateiattribut benötigen, wird ein erweiterter FCB notwendig, dessen Adresse wieder als Parameter übergeben wird. Das Betriebssystem interpretiert einen FCB mit der Belegung FFh des ersten Bytes als erweiterten FCB.

*** Datenträgerstruktur und Dateibehandlung ***



Bemerkungen zur Abbildung:

- Bereiche -7 bis -1 FCB-Erweiterung
- Bereiche 0 bis 36 Standard-FCB
- Bereiche 16 bis 31 von DOS gefüllt, dürfen nicht verändert werden.
- Ubrige Bereiche sind vom Benutzerprogramm auszufüllen.

Standard-FCB

Der Standard-Dateisteuerblock (FCB) hat folgenden Aufbau:

Offset Funktion

00h Laufwerkadresse

z.B.

Vor Eröffnen: 0 - Standardlaufwerk
 1 - Laufwerk A
 2 - Laufwerk B
 usw.

Nach Eröffnen: 0 - Laufwerk A
 1 - Laufwerk B
 usw.

Das Standardlaufwerk (0) wird während des Eröffnens durch die aktuelle Laufwerkadresse ersetzt.

*** Datenträgerstruktur und Dateibehandlung ***

Offset	Funktion
01h-08h	Dateiname wird linksbündig mit führenden Leerstellen aufgefüllt dargestellt. Wird hier ein reservierter Einheitenname eingesetzt (wie beispielsweise LPT1), so wird der wahl-freie Doppelpunkt nicht mit aufgenommen.
09h-0Bh	Dateierweiterung wird linksbündig mit führenden Leerstellen aufgefüllt und kann auch nur aus Leerzeichen bestehen, wenn die Erweiterung nicht spezifiziert ist.
0Ch-0Dh	Aktueller Block wird relativ zum Beginn der Datei mit Null gestartet (wird von Eröffnungs-Funktionsaufruf auf Null gesetzt). Ein Block besteht immer aus 128(80h) Sätzen, wobei jeder die in dem Feld für die Satzlänge angegebene Grösse aufweist. Die Nummer des aktuellen Blockes wird in Verbindung mit dem aktuellen Satzfeld für sequen-tielle Lese- und Schreiboperationen benutzt.
0Eh-0Fh	Satzlänge wird durch den Aufruf der DOS-Funktion "Datei eröffnen" auf 80h gesetzt. Ist dies nicht richtig, so muss der Wert vom Benutzer modifiziert werden. DOS benutzt ihn zum Auffinden der richtigen Speicherplätze in der Datei für alle Lese- und Schreiboperationen auf der Diskette.
10h-13h	Dateigrösse (in Byte)
14h-15h	Datum Datum der Dateierstellung oder der letzten Aktualisie-rung. Die Angabe tt.mm.jj wird wie folgt in den Bits abgebil-det:
	<pre> < 15h > < 14h > 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 < jj > < mm > < tt > </pre>
	Dabei ist: mm gleich 1 bis 12 (01h-0Ch) tt gleich 1 bis 31 (01h-1Fh) jj gleich 0 bis 119 f)r 1980-2099 (00h-77h)
16h-1Fh	Reserviert für Systemverwendung
20h	Aktueller Satz Relative Nummer (0 bis 127) innerhalb des aktuellen Blockes. Dieses Feld muss gesetzt werden, bevor sequentielle Lese-/Schreiboperationen auf der Diskette durchgeführt werden. (Dieses Feld wird nicht von der DOS-Funktion "Datei eröffnen" initialisiert.)

Offset	Funktion
21h-24h	Satznummer bei wahlfreiem Zugriff Relative Nummer, die am Dateianfang mit 0 beginnt. Dieses Feld muss gesetzt werden, bevor wahlfreie Lese-/Schreiboperationen auf der Diskette durchgeführt werden. (Dieses Feld wird nicht von der DOS-Funktion "Datei eröffnen" initialisiert.) Ist die Satzlänge kleiner als 64 Bytes, so werden beide WORDS benutzt. Ansonsten werden nur die ersten 3 Bytes benutzt. Hier ist zu beachten, dass bei Verwendung des Dateisteuerblocks im Programmsegment (ab 5Ch) das letzte Byte des Dateisteuerblocks das erste Byte des nicht formatierten Parameterbereichs überlappt.

Hinweise:

1. Ein nicht eröffneter FCB besteht aus dem Vorsatz des FCB (sofern benutzt), der Laufwerkadresse und der Dateibezeichnung.
Bei einem eröffneten FCB wurden die restlichen Felder von den DOS-Funktionen 16h und 0Fh (Datei erstellen und eröffnen) initialisiert.
2. Die Bytes 00h-0Fh und 16h-24h müssen vom Benutzerprogramm verwaltet werden.
Die Bytes 10h-1Fh werden von DOS gesetzt und dürfen von den Benutzerprogrammen nicht geändert werden.
3. Bei allen Wortfeldern wird das niederwertige Byte zuerst gespeichert.
Eine Satzlänge von 128 wird beispielsweise als 80h in Offset 0Eh und 00h in Offset 0Fh gespeichert.

Erweiterter FCB

Der erweiterte FCB wird dazu benutzt, um Dateien mit besonderen Attributen in dem Diskettenverzeichnis zu erstellen oder zu suchen.

In diesem Fall erhält der FCB einen 7 Byte-Vorsatz, der wie folgt formatiert ist:

Offset	Funktion
-7	Kennzeichenbyte FFh - erweiterter FCB
-6	Reserviert
-2	Attributbyte
-1	Für die Definition der Attributbits wird auf Punkt 5.2.4.1. verwiesen.

Für Einzelheiten über die Benutzung von Attributbits während des Durchsuchens von Verzeichnissen wird ausserdem auf die DOS-Funktion 11h verwiesen. Mit dieser Funktion können Anwendungen ihre eigenen Dateien als geschützte Dateien definieren und sie somit aus den Verzeichnisdurchsuchungen ausschliessen. Ausserdem ermöglicht diese Funktion das selektive Durchsuchen von Verzeichnissen.

Wird in den DOS-Funktionsaufrufen auf einen eröffneten oder nicht eröffneten FCB Bezug genommen, so kann es sich entweder um einen Standard-FCB oder einen erweiterten FCB handeln.

Wird ein erweiterter FCB benutzt, so muss der entsprechende Adresszeiger auf das erste Byte des Vorsatzes und nicht auf das Feld mit der Laufwerkadresse gesetzt werden.

5.4.3. Funktionsaufruf über Händler

5.4.3.1. Merkmale

Eine allgemeinere Form der Dateiarbeit wird durch die Verwendung der Händler-Funktionen möglich:

Merkmale und Folgerungen daraus sind:

1. Sequentieller E/A-Zugriff wird von den Händlern unterstützt. Dazu können die DCP-Funktionen 3Fh und 40h genutzt werden.
2. Der wahlfreie Zugriff wird nicht über Satznummern unterstützt, sondern der Nutzer muss über die DCP-Funktion 42h (LSEEK) einen Offset-Zeiger in der Datei, der bytengenau arbeitet, verwalten und kann damit eine Satzstruktur im wahlfreien Zugriff nachbilden.
3. Es können über die Händler-Funktionen Dateioperationen nicht nur in der lokalen Umgebung (aktuelles Dateiunterverzeichnis) durchgeführt werden, sondern es ist durch Angabe des Pfades ein Zugriff auf den ganzen Datenträger möglich. Die Dateibezeichnung besteht aus einer Kette von ASCII-Zeichen, die neben dem Dateinamen und der Erweiterung optional auch Laufwerk und Pfad enthalten können. Fehlen diese optionalen Angaben, wird die entsprechende lokale Umgebung angenommen. Diese Zeichenkette wird durch das Endekennzeichen 00h abgeschlossen; die Syntax entspricht der des Kommandointerpreters COMMAND.
4. Wenn eine Datei angelegt wird (CREATE), können auch vom Standard abweichende Dateiattribute (r/o, verborgen, System, Unterverzeichnis, Volume-Kennzeichen) für diese Datei festgelegt werden.
5. Bei Eröffnung einer Datei können Zugriffsbeschränkungen für konkurrierenden Zugriff anderer Prozesse (SHARING) sowie auch ein Zugriffsmodus (read-only, write-only, read-write) festgelegt werden.

*** Datenträgerstruktur und Dateibehandlung ***

Die Zugriffsattribute werden bei Dateizugriffen in einem angelegten Tochterprozess mit übernommen und sind dort gültig. Die Attribute des Mehrfachzugriffes sind nur gültig, wenn das residente Steuerprogramm SHARE bei Dateieröffnung aktiv ist.

6. Gelöscht werden kann eine jede Datei des Datenträgers in beliebigem Unterverzeichnis.
7. Umbenannt werden kann eine jede Datei des Datenträgers in beliebigem Unterverzeichnis. Eine Transformation der Datei in ein anderes Unterverzeichnis durch Angabe einer entsprechenden allgemeinen neuen Dateibezeichnung ist nicht möglich.
8. Beim Erzeugen oder Eröffnen einer Datei über Händlerfunktion wird bei fehlerfreier Funktionsausführung eine Händlernummer vom DCP an den Nutzer übergeben. Diese Nummer wird von DCP intern verwaltet und vergeben. Bei allen weiteren Dateioperationen mit dieser eröffneten Datei wird nur noch als Dateiidentifikator diese Händlernummer benutzt. Nach Schliessen der Datei wird diese Referenz aufgelöst.
9. Die Datenströme, die über einen Händler laufen, können auf einen beliebigen anderen Händler umgelenkt werden.
10. Auf Händlerniveau können die Datenströme, die mit Dateihändlern bzw. Gerätehändlern gearbeitet werden, gleich behandelt werden.
11. Für Mehrfachzugriff auf eine Datei oder ein Gerät innerhalb eines Prozesses kann ein Händler kopiert werden. Dann wird über mehrere, allerdings voneinander abhängige Händler gearbeitet.

5.4.3.2. Vordefinierte Händler

Von DCP sind beim Booten des Systems bereits 5 Händler vordefiniert und eröffnet. Sie können sofort für die Datenströme im Bediendialog der Anwenderprogramme benutzt werden. Werden diese vordefinierten Händler nicht benötigt, können sie auch geschlossen werden.

Für diese 5 vordefinierten Händler sind auch entsprechende Händlernummern vergeben:

#1	STDIN	(Standard-Eingabe)
#2	STDOUT	(Standard-Ausgabe)
#3	STDERR	(Standard-Fehlerausgabe)
#4	STDAUX	(sequentielle Daten-Ein-/Ausgabe)
#5	STDPRN	(parallele Daten-Ein-/Ausgabe)

Auf Grund ihrer vorgesehenen Verwendung haben diese Händler auch bestimmte Zugriffsattribute:

- Über STDIN ist die Tastatur angeschlossen. Der Händler hat das Attribut read-only.

*** Datenträgerstruktur und Dateibehandlung ***

- Über STDOUT läuft die Ausgabe auf die Konsole. Der Händler hat das Attribut write-only.
- Über STDERR werden die Fehlermeldungen auf Konsole ausgeführt. Der Händler hat das Attribut write-only.
- STDAUX ist als bidirektionaler sequentieller Datenhändler angelegt. Der Händler hat das Attribut read-write.
- STDPAN ist als bidirektionaler paralleler Datenhändler angelegt. Der Händler hat das Attribut read-write.

Der Datenstrom für STDIN und STDOUT kann bereits auf Kommandoebene vom Nutzer aufgerufen werden. Enthält die Kommandozeile, die von COMMAND.COM interpretiert wird, eine derartige Anweisung, wird der Datenstrom während der gesamten Ausführung des Kommandos umgelenkt. Der Händler STDERR ist auf diesem Niveau nicht umlenkbar.

Die vordefinierten Händler arbeiten standardmässig im ASCII-Mode (siehe Punkt 5.3.6.3.3.).

5.6.3.3. Datenstrom-Modi

Unterschieden wird zwischen dem Binär- und dem ASCII-Modus. Beim ASCII-Mode werden bestimmte Zeichen gefiltert, als Steuerzeichen interpretiert und zur Steuerung des Datenstromes verwendet. Die Ein- und Ausgabe mit Dateien erfolgt standardmässig im Binär-Modus; die Daten werden ohne Modifikation verarbeitet. Die Ein- und Ausgabe auf Geräte erfolgt standardmässig im ASCII-Modus.

Ein Wechsel der Modi kann über die DCP-Funktion 44h (IOCTL) erfolgen, wenn das Gerät dafür vorgesehen ist. Dateien können nicht im ASCII-Mode bearbeitet werden.

Die Konsolenhändler STDIN, STDOUT und STDERR sind duplizierte Händler. Bei Umschalten des Modes eines dieser Händler wird der Mode der anderen mit umgeschaltet.

Im Binär-Mode gilt bei Eingabe:

- Es erfolgt keine Echo-Ausgabe auf dem Händler STDOUT.
- Bei Eingaben über den Händler STDIN wird keine Zeilenedition mit Korrekturmöglichkeiten unterstützt.
- Die Steuerzeichen ^S, ^P und ^C haben keine Steuerwirkung, sondern werden wie alle anderen Zeichen behandelt.
- Vor der Eingabe ist die gewünschte Bytezahl zu spezifizieren. Die Eingabe ist abgeschlossen, wenn diese Zahl oder das Ende der Datei erreicht ist.

Im ASCII-Mode gilt bei Eingabe:

- Die eingegebenen Zeichen werden als Echo auf dem Händler STDOUT ausgegeben.
- Wenn der Eingabepuffer leer ist, wird eine Zeile (abgeschlossen mit 0Ah) in diesen Puffer eingelesen. Diese Zeile kann editiert werden. Ist die Länge der Eingabekette kleiner als vom Nutzer angefordert, wird die Abschlussfolge 0DH, 0Ah vom Händler angehängt. Ist die Zahl der anstehenden Zeichen grösser als angefordert, wird nur die geforderte Menge angenommen und keine Abschlussfolge angehängt. Die noch anstehenden Zeichen können mit folgenden Eingaben behandelt werden.

*** Datenträgerstruktur und Dateibehandlung ***

- Wenn ein EOF (^Z) gefunden wird, wird die Eingabe sofort und ohne Ergänzung der Folge 0Dh, 0Ah abgeschlossen.
- Die Steuerzeichen ^S(Scroll lock), ^P(Print screen) und ^C(Control break) sind wirksam, die Kodierung ^I(Tab) wird in entsprechende Anzahl Leerzeichen aufgelöst.

Im Binär-Mode gilt bei Ausgabe:

- Es erfolgt keine Echo-Ausgabe auf dem Händler STDOUT.
- Die Steuerzeichen ^S, ^P, ^C und ^I erhalten keine Sonderbehandlung.
- Die Bytezahl der Ausgabe ist zu spezifizieren.

Im ASCII-Mode gilt bei der Ausgabe:

- Die Steuerzeichen ^S, ^P, ^C kommen zur Wirkung. Die Kodierung ^I wird in entsprechende Anzahl von Leerzeichen aufgelöst, um eine 8-Zeichen-Raster-Tabulation zu erzeugen.
- Die anderen Steuerzeichen aus der Menge 00h bis 1Fh werden als 2 abbildbare Zeichen dargestellt. (Zum Beispiel das Steuerzeichen ^A als die beiden Zeichen ^ und A.)

5.6.3.4. Anzahl aktiver Händler

Die Anzahl der gleichzeitig von allen Prozessen eröffneten Händler wird durch das Kommando FILES in der CONFIG.SYS festgelegt. Die Anzahl der Händler, die ein einzelner Prozess aktivieren kann, hängt auch von dieser getroffenen Vereinbarung ab. Wenn der Parameter im FILES-Kommando die Anzahl nicht einschränkt, können 20 Dateien eröffnet werden. Zu beachten ist dabei, dass 3 Dateien durch die vordefinierten Händler (eine für die Konsole mit STDIN/STDOUT/STDERR und je einer für die bidirektionalen Händler STDAUX sowie STDPN) im allgemeinen schon aktiviert sind. Unter diesen Bedingungen kann ein einzelner Prozess also noch maximal 17 Händler gleichzeitig aktivieren.

6. Installierbare Gerätetreiber

Dieses Kapitel enthält Informationen zu folgenden Themen:

- Aufbau eines Gerätetreibers
- Erstellen eines Gerätetreibers
- Installieren eines Gerätetreibers.

Das Kapitel informiert ferner über die Typen von Gerätetreibern, den Request Header und die Zeitgeber-Einheit CLOCKR.

Das DCP-Geräteinterface verbindet alle Gerätetreiber zu einer Kette. Damit wird es möglich, neue Gerätetreiber für zusätzliche Geräte hinzuzufügen.

6.1. Format eines Gerätetreibers

Ein Gerätetreiber ist eine Datei mit Speicherabbild oder eine EXE-Datei. Sie enthält alle notwendigen Informationen zur Implementation des Gerätes. Vorangestellt ist eine spezielle Datenstruktur, welche die Datei als Gerätetreiber kennzeichnet, die Eintrittspunkte für Strategie- und Interruptroutine festlegt und die verschiedenen Geräteeigenschaften beschreibt.

Beachte: Als Gerätetreiber darf eine Speicherabbild-Datei kein ORG 100h enthalten. Da der PSP (Programm-Segment-Präfix) nicht verwendet wird, wird der Gerätetreiber einfach nur geladen. Die Datei muss deshalb auf Null abgebunden sein (ORG 0 oder gar keine ORG-Anweisung).

6.2. Gerätetypen

Es gibt 2 Grundtypen von Geräten:

- zeichenorientierte Geräte (zeichenweise Ein-/Ausgabe)
- blockorientierte Geräte (Ein-/Ausgabe von Datenblöcken).

6.2.1. Zeichenorientierte Geräte

Diese Geräte realisieren eine serielle Ein-/Ausgabe in der gleichen Art und Weise wie CON, AUX und PRN. Sie haben Bezeichnungen wie CON, AUX, CLOCKR, und es können Kanäle eröffnet werden (über Händler oder FCB's) für die Ein-/Ausgabeoperationen mit diesen Geräten. Da diese zeichenorientierten Geräte nur eine einzige Bezeichnung besitzen, können sie auch nur genau ein Gerät unterstützen.

6.2.2. Blockorientierte Geräte

Blockorientierte Geräte im System sind Festplatte oder Diskettenlaufwerk. Sie unterstützen wahlfreie Ein-/Ausgabeoperationen mit einer Datenmenge, die als ein Block bezeichnet wird. Als Blockgrösse ist gewöhnlich die physische Sektorgrösse der Diskette vereinbart.

*** Installierbare Gerätetreiber ***

Diese Geräte erhalten keine solchen Bezeichnungen wie die zeichenorientierten Geräte und können nicht direkt eröffnet werden. Sie werden stattdessen durch die Verwendung der Laufwerkbezeichnungen A, B, C usw. aktiviert. Blockorientierte Geräte können mehrere Geräte einschliessen. Damit kann ein einzelner Treiber für blockorientierte Geräte eine oder mehrere Disketten bzw. Laufwerke bedienen.

Zum Beispiel könnte der erste Treiber die Laufwerke A, B, C und D bedienen, d.h., für jedes der vier Geräte wird ein Buchstabe als Laufwerkbezeichnung festgelegt. Die Position des Treibers in der Kette aller Treiber legt die Zuordnung der Laufwerkbezeichnung zum Gerät fest. Ist z.B. der Gerätetreiber der erste Treiber für blockorientierte Geräte in der Kette aller Gerätetreiber und definiert vier Geräte, dann sind das die Geräte A, B, C und D. Definiert ein zweiter Treiber für blockorientierte Geräte drei Geräte, dann sind das die Geräte E, F und G. Auf diese Weise können bis zu 26 Geräte mit den Buchstaben A bis Z den Treibern zugewiesen werden.

4.3. DEVICE_HEADER

Jeder Gerätetreiber muss mit einem Device Header (Gerätekensatz) beginnen.

Der Device Header hat folgenden Inhalt:

Offset	Feldinhalt	Länge
+ 0	Zeiger auf nächsten Device Header in der Kette	DWORD
+ 4	Attribut (Treiber-Eigenschaften)	WORD
+ 6	Zeiger auf Strategie-Routine	WORD
+ 8	Zeiger auf Interrupt-Routine	WORD
+10	Gerätebezeichnung	8 Bytes

4.3.1. Zeiger auf nächsten DEVICE_HEADER in der Kette

Dieses Feld enthält einen Zeiger auf den nächsten Device Header des nächsten Gerätetreibers. Es ist ein DWORD-Feld und wird beim Laden des Gerätetreibers von DCP initialisiert. Das erste WORD ist der Offset und das zweite WORD das Segment. Der letzte Treiber in der Kette enthält -1 (FFFF/FFFFh) als Endekennzeichen in diesem Feld.

Wird nur ein Treiber geladen, dann muss vor dem Laden in diesem Feld eine -1 eingetragen werden. Wenn mehrere Treiber geladen werden, wird im ersten WORD dieses Feldes die Offsetadresse des Device Headers vom nächsten Gerätetreiber eingetragen. Im letzten Treiber ist wieder -1 als Folgeadresse einzutragen.

*** Installierbare Gerätetreiber ***

6.3.2.1 Attribut-Feld

Dieses Feld, in der Länge von zwei Bytes, gibt dem System Auskunft über die Eigenschaften des Gerätetreibers. Folgende Eigenschaften können kodiert werden:

Rit	Belegung	Bedeutung
0	1	Aktuelles Standard-Eingabe-Gerät
	0	Kein aktuelles Standard-Eingabe-Gerät
1	1	Aktuelles Standard-Ausgabe-Gerät
	0	Kein aktuelles Standard-Ausgabe-Gerät
2	1	Aktuelles NUL-Gerät
	0	Kein aktuelles NUL-Gerät
3	1	Aktuelles Zeitgebergerät
	0	Kein aktuelles Zeitgebergerät
4-5	0	Reserviert für DCF
6	1	Unterstützung Abfragen/Setzen log. Gerät
	0	Abfragen/Setzen logisches Gerät nicht unterstützt
	1	Unterstützung Generic-IOCTL-Funktionsaufrufe
	0	Generic-IOCTL-Funktionsaufrufe werden nicht unterstützt.
7-10	0	Reserviert für DCF
11	1	Unterstützung austauschbarer Datenträger
	0	Keine Unterstützung austauschbarer Datenträger
13	1	Für blockorientierte Geräte:
		kein IBM-Format
	0	IBM-Format
		Für zeichenorientierte Geräte:
1	unterstützt Ausgaben bis Besetztstatus	
0	unterstützt Ausgaben bis Besetztstatus nicht	
14	1	Unterstützung von IOCTL
	0	Keine Unterstützung von IOCTL
15	1	Zeichenorientiertes Gerät
	0	Blockorientiertes Gerät

*** Installierbare Gerätetreiber ***

Bit 0 und 1

Diese Bits dienen der Kennzeichnung des Standardein- bzw. -ausgabegerätes und haben nur für zeichenorientierte Geräte Bedeutung. Verwendet werden sie, um dem DCP mitzuteilen, dass der Treiber für ein zeichenorientiertes Gerät das neue Standardein- oder -ausgabegerät bedient.

Bit 2

Bit 2 kennzeichnet das NUL-Gerät. Dieses Attribut wird nur für zeichenorientierte Geräte verwendet. Über Bit 2 erhält DCP die Information, ob der Gerätetreiber ein NUL-Gerät bedient. Obwohl es ein solches Bit gibt, kann die Zuweisung des NUL-Gerätes nicht zurückgesetzt werden. Ferner hat es nur Bedeutung für DCP, denn damit kann DCP den Zugriff auf ein NUL-Gerät melden.

Bit 3

Dieses Bit kennzeichnet das Zeitgeber-Gerät und hat nur für zeichenorientierte Geräte eine Bedeutung. Dem DCP wird mitgeteilt, ob der Gerätetreiber das neue Zeitgeber-Gerät bedient.

Bit 6

Dieses Bit gilt sowohl für zeichen- als auch für blockorientierte Geräte. Ist dieses Bit gesetzt, dann unterstützt der Gerätetreiber Generic-IOCTL-Funktionsaufrufe.

Bit 11

Mit diesem Bit wird mitgeteilt, ob ein Datenträgeraustausch behandelt wird.

Bit 13

Das Bit 13 kennzeichnet für blockorientierte Geräte die Methode, mit der der Gerätetreiber den Datenträgertyp feststellt. Bei zeichenorientierten Geräten wird mit diesem Bit gekennzeichnet, ob der Treiber die Ausgabe bis zum Besetztstatus des Gerätes unterstützt oder nicht.

Verwendet der Gerätetreiber für ein blockorientiertes Gerät zur Bestimmung des Datenträgertyps die Informationen aus dem BFB, dann muss Bit 13 auf 1 gesetzt werden. Wird vom Gerätetreiber aber das Datenträger-Kennzeichenbyte zur Bestimmung des Datenträgertyps benutzt, dann muss Bit 13 auf 0 gesetzt sein.

Die Ausgabe bis zum Besetztstatus wird für Druckertreiber benutzt. Bei dieser Art der Ausgaben sendet der Treiber Zeichen an das Gerät, wenn dieses bereit ist. Unterstützt der Treiber diese Ausgaben bis zum Besetztstatus, und das Gerät ist nicht bereit, meldet der Treiber sofort einen Fehler zurück.

Bit 14

Dieses Bit ist das IOCTL-Bit. Es wird sowohl für zeichen- als auch für blockorientierte Geräte genutzt. Es informiert DCP darüber, ob der Gerätetreiber Steuerfolgen über den Aufruf der IOCTL-Funktion (44h) behandelt oder nicht. Bit 14 beeinflusst nur die IOCTL-Aufrufe mit AL=2 bis AL=5.

Wenn der Gerätetreiber Steuerfolgen nicht verarbeiten kann, muss Bit 14 auf 0 gesetzt werden. DCP kann dann eine Fehlermeldung generieren, wenn beim Aufruf der IOCTL-Funktion (44h) Steuerfolgen den Treiber passieren. Kann ein Gerät Steuerfolgen verarbeiten, dann muss Bit 14 auf 1 gesetzt werden. In diesem Fall führt das DCP die Funktionsaufrufe zum Senden und Empfangen von Steuerfolgen aus.

Mit den IOCTL-Funktionen ist es möglich, Daten an das Gerät zu senden oder von ihm zu empfangen, ohne die normalen Lese-/Schreiboperationen auszuführen. Damit kann der Treiber die Daten auch für seine Zwecke verwenden (z.B. zum Setzen der Baudrate oder Stopp-Bits). Es obliegt dann dem Gerätetreiber selbst, wie die Daten zu interpretieren sind, aber die Information wird nicht als normale I/O-Anforderung betrachtet.

Bit 15

Bit 15 kennzeichnet den Gerätetyp. Unterschieden wird zwischen Gerätetreibern für zeichen- und blockorientierte Geräte.

6.3.3. Zeiger auf Strategie- und Interruptroutinen

Diese beiden Felder beinhalten die Adressen für die Eintrittspunkte der beiden Routinen. Angegeben werden nur die Offsets, d.h., die Routinen müssen sich im gleichen Segment wie der Device Header befinden.

6.3.4. Gerätebezeichnung

Handelt es sich um einen Gerätetreiber für ein zeichenorientiertes Gerät, dann wird der Geräteiname linksbündig in das Feld eingetragen und eventuell freie Plätze mit Leerzeichen aufgefüllt. Bei blockorientierten Geräten kann im ersten Byte die Anzahl der Geräte eingetragen werden, die dieser Treiber bedient. DCP trägt in dieses Feld den Rückkehrcode der INIT-Routine des Treibers ein.

6.4. Erstellung eines Gerätetreibers

Zur Erstellung eines Gerätetreibers, der in das DCP eingebunden werden kann, sind folgende Schritte auszuführen:

- Generieren einer COM- oder EXE-Datei, die mit einem Device Header beginnt;
- Abbinden des Treibers auf 0 (einschliesslich dem Device Header), nicht auf 100h;

***** Installierbare Gerätetreiber *****

- Setzen des Zeigers auf den nächsten Device Header (vgl. dazu Abschnitt 4.3.1.);
- Setzen des Attributfeldes im Device Header (vgl. dazu Abschnitt 4.3.2.);
- Setzen der Eintrittspunkte für Strategie- und Interruptroutinen;
- Eintragen der Gerätebezeichnung oder der Anzahl von Geräten.

DCP arbeitet immer zuerst mit den installierbaren Gerätetreibern für zeichenorientierte Geräte, bevor auf die Standardgeräte zurückgegriffen wird. Um ein neues logisches Gerät CON zu installieren, muss im Device Header nur der Name CON eingetragen werden. Die Bits 0 und 1 des Attributfeldes sind entsprechend zu setzen. Die Suche in der Treiberkette nach dem entsprechenden Gerät wird beendet, wenn das erste zutreffende gefunden wird. Dadurch haben die installierbaren Gerätetreiber Vorrang.

Beachte: Da das DCP den Treiber an beliebiger Stelle im Speicher ablegt, muss mit FAR-Speicherzugriffen sorgfältig umgegangen werden. Man darf sich auch nicht darauf verlassen, daß der Treiber immer an die gleiche Stelle geladen wird.

6.5. Installation des Gerätetreibers

DCP installiert neue Gerätetreiber dynamisch zum Zeitpunkt des Kaltstartes durch Abarbeitung der Device-Kommandos in der Datei CONFIG.SYS. Soll zum Beispiel der Gerätetreiber TREIBER1 in das System eingebunden werden, dann ist in der CONFIG.SYS-Datei der folgende Befehl anzugeben:

```
DEVICE=TREIBER1
```

DCP ruft zuerst den Strategie-Eintrittspunkt des Gerätetreibers auf. Dabei wird die Adresse eines Kommandofeldes - Request Header - übergeben, indem das DCP den Auftrag an den Treiber formuliert. Die Strategieroutine selbst führt den Auftrag nicht aus, sie ordnet ihn entweder in eine Warteschlange ein oder sichert die Adresse des Request Header. Der Interrupt-Aufruf wird damit angedeutet. Der zweite Eintrittspunkt ist die Interruptroutine. DCP ruft diesen Eintrittspunkt unmittelbar nach der Rückkehr aus der Strategieroutine auf. Die Interruptroutine wird ohne Parameter aufgerufen. Der angemeldete Auftrag ist nun vom Treiber entsprechend den Angaben im Request Header auszuführen und eine Rückmeldung an DCP zu übergeben.

DCP übergibt die Adresse des Request Header in den Registern ES:BX. Diese Datenstruktur (Request Header) umfasst einen Teil konstanter Länge, dem ein Teil variabler Länge folgt. Aufbau und Länge des variablen Teils werden durch die auszuführende Operation bestimmt.

*** Installierbare Gerätetreiber ***

Beachte: Der Gerätetreiber ist für die Sicherung des Maschinenzustandes verantwortlich: So sind z.B. am Eintrittspunkt alle Register zu retten und beim Verlassen des Treibers wieder zurückzuspeichern.

Der vom DCP benutzte Stapel bietet genügend Platz, um alle Register ablegen zu können. Sollte mehr Platz im Stapelspeicher erforderlich sein, dann ist der Gerätetreiber selbst dafür verantwortlich, sich einen Stapelspeicher anzulegen und diesen zu verwalten. Alle Aufrufe zum Gerätetreiber sind vom Typ FAR (gegebenenfalls in einem anderen Segment). Die Steuerung wird mit entsprechenden RET FAR wieder an DCP zurückgegeben.

6.5.1. Installation von zeichenorientierten Geräten

Eine notwendige Treiberfunktion ist INIT. Diese darf nur einmal bei der Installation des Treibers aufgerufen werden. Die INIT-Routine gibt an das DCP folgendes zurück:

- Die Byteadresse des ersten Bytes nach dem initialisierten Treiber wird in den Request Header eingetragen. Damit kann die INIT-Routine nach ihrer Abarbeitung den von ihr belegten Speicherbereich wieder freigeben.
- Danach wird das Status-WORD übertragen und die Steuerung an DCP zurückgegeben.

6.5.2. Installation von blockorientierten Geräten

Blockorientierte Geräte werden gleichermaßen installiert wie zeichenorientierte. Der Unterschied besteht darin, dass Treiber für blockorientierte Geräte zusätzliche Informationen an das DCP zurückgeben.

Diese Treiber müssen folgende Informationen liefern:

- Anzahl der Geräte, die dem Treiber zugeordnet sind. Über diese Anzahl werden die den logischen Geräten zugeordneten Namen bestimmt. Ist z.B. der Name des letzten logischen Gerätes C und die INIT-Routine gibt als Anzahl der dem Treiber zugeordneten Geräte den Wert 2 zurück, dann erhalten diese Geräte die logischen Namen D und E zugeordnet. Die Zuordnung wird durch die Position des Gerätes in der Verkettungsliste und die Anzahl der Einheiten für die einzelnen Geräte bestimmt. Die Anzahl der Einheiten wird in das Feld für die Gerätebezeichnung im Device Header eingetragen.
- Zeiger auf das Zeigerfeld des BFB (BIOS-Parameter-Block). Dieses Feld enthält die Zeiger auf die BPB's der einzelnen Geräte. Für alle Geräte mit gleichen Eigenschaften muss nur ein BFB angelegt werden; diese Zeiger im Zeigerfeld sind dann identisch. Der BFB enthält zu einem Gerät gehörende Informationen, wie Sektorgrösse, Anzahl Sektoren pro Block usw. Die Sektorgrösse im BFB kann nicht grösser sein als der während der DCP-Initialisierung maximal zugewiesene Wert. Dieses Feld muss im geschützten Bereich innerhalb des Treibers angelegt werden.

*** Installierbare Gerätetreiber ***

- Das Datenträgerkennzeichen und damit die physische Organisation des Datenträgers wird an die Geräte weitergegeben, so dass dort Informationen vorliegen, welche aktuellen Parameter das DCP für ein bestimmtes Laufwerk benutzt.

6.6. REQUEST HEADER

Der Request Header enthält die Informationen, die beim Aufruf durch DCP übergeben werden müssen.

Er hat folgenden Aufbau:

Offset	Größe	Bedeutung
+ 0	Byte	Länge des gesamten Kopfes (fester und variabler Teil) in Byte
+ 1	Byte	Kode für Subgerät, auf das sich die Anforderung bezieht (ohne Bedeutung für zeichenorientierte Geräte)
+ 2	Byte	Kommando-Kode
+ 3	WORD	Status
+ 5	8 Bytes	Für DCP reserviert
+13	variabel	Von der Operation abhängige Daten (variabler Teil)

6.6.1. Gerätecode

In diesem Feld wird angegeben, für welches Gerät des blockorientierten Gerätetreibers die Anforderung bestimmt ist. Besitzt z.B. ein Gerätetreiber drei zugeordnete Geräte, dann sind die möglichen Werte für dieses Gerätecode-Feld 0, 1 und 2.

6.6.2. Kommandocode

In diesem Feld des Request Header können folgende Werte stehen:

Kode	Funktion	Bemerkungen
0	INIT Initialisierung des Gerätes	
1	MEDIA CHECK Datenträgerprüfung	nur für blockorientierte Geräte; bei zeichenorientierten Geräten NOP
2	BUILD BPB Anlegen des BPB	nur für blockorientierte Geräte; bei zeichenorientierten Geräten NOP

*** Installierbare Gerätetreiber ***

Kode	Funktion	Bemerkungen
3	IOCTL-Input Steuerfolgen- Eingabe	wird nur gerufen, wenn IOCTL-Bit = 1
4	INPUT Eingabe	
5	NON DESTRUCTIVE INPUT NO WAIT Abruf des Zeichens ohne Entnahme aus dem Eingabekanal	nur für zeichenorien- tierte Geräte
6	INPUT STATUS	nur für zeichenorien- tierte Geräte
7	INPUT FLUSH Leeren Eingabe- puffer	nur für zeichenorien- tierte Geräte
8	OUTPUT Ausgabe	
9	OUTPUT WITH VERIFY Ausgabe mit Prüfung	
10	OUTPUT STATUS	nur für zeichenorien- tierte Geräte
11	OUTPUT FLUSH Leeren Ausgabe- puffer	nur für zeichenorien- tierte Geräte
12	IOCTL OUTPUT Ausgabe von Steuerfolgen	nur wenn IOCTL-Bit=1 gesetzt
13	DEVICE OPEN	nur wenn OPEN/CLOSE/RM-Bit gesetzt
14	DEVICE CLOSE	nur wenn OPEN/CLOSE/RM-Bit gesetzt
15	REMOVABLE MEDIA wechselbarer Datenträger	nur für blockorien- tierte Geräte und wenn OPEN/CLOSE/RM-Bit=1
19	GENERIC IOCTL REQUEST	
23	GET LOGICAL DEVICE Abfrage log. Gerät	
24	SET LOGICAL DEVICE Setzen log. Gerät	

6.6.3. Status-Feld

Das Statusfeld im Request Header ist beim DCP-Aufruf = Null gesetzt und wird durch die Interruptroutine des Gerätetreibers eingestellt.

Das Status-WORD hat folgende Struktur:

Bit	Bedeutung
0-7	Fehlerkode (Bit 15=1)
8	Auftrag abgeschlossen
9	Bereitschaft
10-14	Reserviert
15	Fehlerindikator

Bit 15 ist der Fehlerindikator. Ist es gesetzt, trat ein Fehler auf, der in Bit 0-7 spezifiziert ist. Bit 8 ist gesetzt, wenn die Operation abgeschlossen ist. Bit 9 zeigt die Bereitschaft eines Gerätes an. Es wird nur bei Statusabfragen und beim Kommando REMOVABLE MEDIA gesetzt.

Zum Wert des Fehlerbytes (Bit 0-7) gehören folgende Fehlermeldungen:

Fehlerkode	Bedeutung
00	Verletzung des Schreibschutzes
01	Unbekanntes Gerät
02	Gerät nicht bereit
03	Unbekanntes Kommando
04	CRC-Fehler
05	Ungültige Längenangabe
06	Spur nicht gefunden
07	Unbekannter Datenträger
08	Sektor nicht gefunden
09	Papierende
0A	Schreibfehler
0B	Lesefehler
0C	Allgemeiner Fehler
0D	Reserviert
0E	Reserviert
0F	Ungültiger Diskettenwechsel

6.7. Beschreibung der Gerätetreiberfunktionen

Beim Aufruf der Strategieroutinen übergibt DCP in ES:BX die Adresse des Request Header. Die Interruptroutinen erhalten diese Adressen aus der Warteschlange, in die beim Aufruf der Strategieroutinen die Gerätetreiber die Adresse des Request Header speichern. Der Kommandokode im Request Header teilt dem Treiber mit, welche Funktion auszuführen ist. Es ist zu beachten, dass bei allen DWORD-Zeigern zuerst die Offset- und dann die Segmentadresse abgelegt wird. Nachfolgend werden die einzelnen Funktionen mit ihrem Aufruf und dem Leistungsumfang dargestellt.

6.7.1. INIT

Kommandokode = 0

Initialisieren des Gerätes

ES:BX	Länge	Feld
+ 0	13 Byte	Request Header
+13	BYTE	Anzahl von Geräten, die der Treiber bedient (nur für blockorientierte Geräte)
+14	DWORD	Endadresse des residenten Treibers
+18	DWORD	Zeiger auf BFB-Feld (wird bei zeichenorientierten Geräten nicht gesetzt) / Zeiger auf den Rest des Arguments
+22	BYTE	Laufwerkcode (0=A, 1=B, ... ; nur für blockorientierte Geräte)

Der Treiber muss folgende Funktionen erfüllen:

- Setzen der Anzahl von Geräten (nur bei Treibern für blockorientierte Geräte)
- Eintragen des Zeigers zum BFB-Feld (ebenfalls nur bei Treibern für blockorientierte Geräte)
- Durchführung der erforderlichen Initialisierung der Hardware (z.B. Modem, Drucker usw.) und interner Arbeitsbereiche
- Setzen der Endadresse des residenten Teils vom Treiber
- Setzen des Status-WORD im Request Header.

Damit dem Gerätetreiber die Informationen aus der Datei CONFIG.SYS zum Zeitpunkt der Initialisierung zur Verfügung stehen, zeigt das BFB-Zeigerfeld auf einen Pufferbereich. Dieser enthält die Informationen, die in der Datei CONFIG.SYS nach dem Gleichheitszeichen stehen. In diesem Puffer steht eine Zeichenkette, die entweder durch Wagenrücklauf 0Dh oder Zeilenschaltung 0Ah abgeschlossen sein muss. Diese Informationen sind nur zum Auslesen. Es können nur die Systemrufe 01h bis 0Ch sowie 30h von der Initroutine ausgegeben werden.

Das letzte Parameterbyte enthält den Laufwerkcode für das erste Gerät, das der blockorientierte Treiber bedient (z.B. 0=A, 1=B usw.).

*** Installierbare Gerätetreiber ***

Wenn die Initialisierung abgebrochen werden muss, ohne dass der Treiber eingebunden wird (es soll kein Speicherplatz belegt werden), dann muss der Treiber folgende Aufgaben ausführen:

- Anzahl der Geräte auf Null setzen
- Offset der Endadresse des Treibers auf Null setzen
- Segment der Endadresse auf das Codesegment (CS) setzen.

Wenn mehrere Gerätetreiber in einer Datei zusammengefasst sind, berücksichtigt DCP die Endadresse des zuletzt initialisierten Gerätetreibers. Deshalb empfiehlt es sich, dass alle Gerätetreiber in einer Datei die gleiche Endadresse zurückgeben.

6.7.2. MEDIA CHECK

Kommandokode = 1

Überprüfung des Datenträgers

ES:BX	Länge	Feld
+ 0	13 Bytes	Request Header (residenter Teil)
+13	BYTE	Datenträgerkennzeichen vom DCP
+14	BYTE	Rückgabewert
+15	DWORD	Zeiger auf vorhergehendes ID-Feld (wenn Bit 11=1 und Diskettenwechsel vorgenommen wurde)

Steht im Kommandofeld eine 1, dann ruft DCP die Funktion MEDIA CHECK für ein Laufwerk und liefert das aktuelle Datenträger-Kennzeichenbyte (vgl. zum Aufbau dieses Bytes den Abschnitt "Datenträger-Kennzeichenbyte" innerhalb dieses Kapitels).

Die Funktion MEDIA CHECK gibt nach Überprüfung des Datenträgers eine der folgenden Informationen zurück:

- der Datenträger wurde nicht gewechselt
- der Datenträger wurde gewechselt
- keine sichere Aussage über Datenträgerwechsel möglich
- Fehlerkode.

Der Treiber muss folgende Aufgaben realisieren:

- Status im Request Header setzen
- Rückkehrkode setzen:
 - 1 Datenträger gewechselt
 - 0 keine Aussage möglich
 - 1 Datenträger nicht gewechselt.

Es wird vom Kommando folgende Methode zur Bestimmung des Rückkehrkodes benutzt:

- Ist der Datenträger eine Festplatte (also ein nicht wechselbarer Datenträger), dann wird im Rückkehrkode "Datenträger nicht gewechselt" eingetragen.

*** Installierbare Gerätetreiber ***

- Wenn der letzte erfolgreiche Zugriff weniger als zwei Sekunden zurückerliegt, dann wird im Rückkehrkode "Datenträger nicht gewechselt" eingetragen.
- Ist das Signal "changeline" nicht verfügbar, dann wird im Rückkehrkode "möglicherweise Datenträgerwechsel" eingetragen.
- Ist das Signal "changeline" verfügbar, aber nicht aktiv, dann wird im Rückkehrkode "Datenträger nicht gewechselt" eingetragen.
- Wenn das Datenträgerkennzeichen im neuen BFB nicht dem aus dem alten entspricht, dann wird im Rückkehrkode "Datenträger gewechselt" eingetragen.
- Entspricht das aktuelle ID-Feld dem vorhergehenden, dann wird im Rückkehrkode "möglicherweise Datenträgerwechsel" eingetragen.

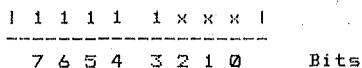
Wurde im Attribut-Feld des Device Header das Bit 11 gesetzt und der Treiber hat einen Datenträgerwechsel erkannt, muss der Zeiger auf das vorhergehende ID-Feld eingetragen werden. Wenn DCP erkennt, dass der Wechsel ein Fehler ist, generiert es den Fehlerkode 0Fh (ungültiger Diskettenwechsel) seitens des Gerätes. Wenn der Treiber keine Volume-Identifikation unterstützt, aber das Bit 11 gesetzt ist, muss der Treiber einen Zeiger auf die Zeichenkette 'KEIN NAME', 00h setzen.

6.7.3. Media-Descriptor-Byte

Datenträger-Kennzeichenbyte

Im allgemeinen wird das Datenträger-Kennzeichenbyte so festgelegt, dass es für mehrere Datenträgertypen gilt. Dieses Byte muss identisch sein mit dem Media Byte, für den Fall, dass für das Gerät das Bit "Nicht-IBM-Format" = Null ist. Die vorgegebenen Werte sind:

Datenträger-Kennzeichenbyte



Bit	Bedeutung	
0	1= zweiseitig	0= nicht zweiseitig
1	1= 8 Sektoren	0= keine 8 Sektoren
2	1= austauschbar	0= nicht austauschbar
3-7	müssen 1 sein	

Eine Ausnahme für die oben angegebene Bit-Bedeutung ist der Wert F0 vom Datenträgerkennzeichen, mit dem ein beliebiger, nicht definierter Datenträgertyp gekennzeichnet wird.

*** Installierbare Gerätetreiber ***

Beispiele aktueller DCP-Datenträger-Kennzeichenbytes:

Disketten- typ	Anzahl der Seiten	Anzahl Sektoren/ Spuren	Datenträger- Kennzeichen
Festplatte	-	-	F8h
5 1/4 Zoll	2	15	F9h
5 1/4 Zoll	1	9	FCh
5 1/4 Zoll	2	9	FDh
5 1/4 Zoll	1	8	FEh
5 1/4 Zoll	2	8	FFh
8 Zoll	1	26	FEh
8 Zoll	2	26	FDh
8 Zoll	2	8	FEh

Anmerkung:

Bei den zwei übereinstimmenden 8-Zoll-Diskettentypen (FEh) kann durch Lesen der zweiten Seite festgestellt werden, ob es sich um eine ein- oder zweiseitige Diskette handelt. Wird beim Lesen der zweiten Seite ein Fehler festgestellt, handelt es sich um eine einseitige Diskette.

1. Das Datenträgerkennzeichen FDh muss für die oben nicht aufgeführten Disketten verwendet werden.
2. Die Werte für das Datenträgerkennzeichen sind nur zur Information angegeben. Sie dürfen nicht in Programmen verwendet werden.
3. Um den Datenträgertyp von Disketten im IBM-Format zu bestimmen, nutzen DCP-interne Routinen die Informationen aus dem BIOS-Parameter-Block (BPB) anstelle des Datenträgerkennzeichens. Mit diesem Datenträgerkennzeichen kann ein Datenträgertyp nicht eindeutig gekennzeichnet werden.

Für 8-Zoll-Disketten gilt:

- FEh einseitig, einfache Dichte, 128 Bytes pro Sektor, softsektoriert, 4 Sektoren pro zugewiesene Einheit, 1 reservierter Sektor, 2 FAT's, 68 Verzeichniseintragen, 77*26*1 Sektoren
- FDh zweiseitig, einfache Dichte, 128 Bytes pro Sektor, softsektoriert, 4 Sektoren pro zugewiesene Einheit, 4 reservierte Sektoren, 2 FAT's, 68 Verzeichniseintragen, 77*26*2 Sektoren
- FEh zweiseitig, doppelte Dichte, 1024 Bytes pro Sektor, softsektoriert, 1 Sektor pro zugewiesene Einheit, 1 reservierter Sektor, 2 FAT's, 192 Verzeichniseintragen, 77*8*2 Sektoren.

Erstellen eines BIOS-Parameter-Blocks für blockorientierte Geräte

ES:BX	Länge	Feld
+0	13 Byte	Request Header
+13	BYTE	Datenträgerkennzeichen von DCP
+14	DWORD	Übertragungsadresse (Pufferadresse)
+18	DWORD	Zeiger auf BPB-Tabelle

DCP ruft BUILD BPB unter folgenden Bedingungen auf:

- wenn "Datenträgerwechsel" zurückgemeldet wird
- wenn "Datenträgerwechsel nicht sicher" zurückgemeldet wird (es gibt dann keine benutzten Puffer). Benutzte Puffer sind Puffer mit geänderten Daten, die noch nicht auf Disketten geschrieben wurden.

Der Treiber muss folgendes ausführen:

- Setzen des Zeigers auf BPB
- Setzen des Status im Request Header.

Der Treiber muss exakt den Datenträgertyp bestimmen, der sich gerade im Gerät befindet, um den Zeiger zur BPB-Tabelle zurückzugeben. Der Treiber überprüft den Urladesektor und liefert von dort den BPB zurück.

Die Information, die sich auf den BPB für einen bestimmten Datenträger bezieht, wird im Urladesektor des Datenträgers abgelegt.

Ein Urladesektor besitzt folgenden Aufbau:

2 Byte kurzer Sprung (Ebh), gefolgt von einem NOP (90h)
8 Byte OEM Name und Version

WORD	Bytes pro Sektor	+
BYTE	Sektoren pro Block (Potenz von 2)	
WORD	reserv. Sektoren (beginnend mit dem log. Sektor 0)	
BYTE	Anzahl FAT's	B
WORD	Anzahl (max.) der Eintragungen im Wurzelverzeichnis	P
WORD	Zahl der Sektoren im logischen Abbild (Summe der Sektoren im Datenträger einschl. Bootsektor, Verzeichnisse usw.)	B
BYTE	Datenträgerkennzeichen	
WORD	Anzahl der durch eine einzelne FAT belegten Sektoren	+
WORD	Sektoren pro Spur	
WORD	Anzahl von Köpfen	
WORD	Anzahl der verborgenen Sektoren	

*** Installierbare Gerätetreiber ***

Die drei WORD-Parameter am Ende unterstützen den Gerätetreiber bei der Erkennung des Datenträgers. Die Anzahl der Köpfe ist nützlich für die Unterstützung verschiedener Multikopf-Laufwerke, die die gleiche Speicherkapazität, aber eine unterschiedliche Anzahl Oberflächen besitzen.

Die Zahl der verborgenen Sektoren wird zur Unterstützung der Unterteilung von Laufwerken genutzt.

Ein Treiber, der die Datenträgeridentifikation und den Diskettenwechsel unterstützt, generiert bei diesem Aufruf eine neue Datenträgeridentifikation. Dieser Aufruf zeigt zudem auch den ordnungsgemässen Wechsel der Diskette an.

6.7.5. INPUT oder OUTPUT

Kommandokodes = 3,4,8,9,12

Ein- und Ausgaben

ES:BX	Länge	Feld
+0	13 Bytes	Request Header
+13	BYTE	Datenträgerkennzeichen
+14	DWORD	Pufferadresse
+18	WORD	Byte- bzw. Sektorzähler
+20	WORD	Nummer des Startsektors (keine Bedeutung für zeichenorient. Geräte)
+22	DWORD	Zeiger auf Datenträgeridentifikation, wenn der Fehlercode 0Fh zurückgegeben wird.

Der Treiber muss folgendes ausführen:

- Setzen Status im Request Header
- Ausführen der angeforderten Funktion
- Setzen der tatsächlich übertragenen Anzahl von Sektoren (oder Bytes).

Anmerkung:

Bei einem IOCTL-Ein-/Ausgabe-Aufruf wird keine Fehlerkontrolle ausgeführt. Der Treiber muss jedoch den Zähler für Sektoren oder Bytes auf die Anzahl der tatsächlich übertragenen Bytes setzen.

Anwendung für blockorientierte Gerätetreiber:

Unter bestimmten Umständen kann ein Gerätetreiber aufgefordert werden, eine Schreiboperation von 64 KByte auszuführen, die einen Überlauf der Transferadresse im Request Header bewirken würde. Das geht aus einer Optimierung zum Schreibcode in DCF hervor. Nur bei WRITE-Operationen innerhalb einer Sektorgröße von 64 KByte in Dateien, die über das aktuelle Ende (EOF) erweitert worden sind, tritt dieser Effekt auf. Wird von einem Gerätetreiber eine solche Funktion verlangt, dann darf er die Bytes, bei deren Übertragung die Transferadresse überlaufen würde, ignorieren. Sollen z.B. in einer Schreiboperation 10000h Bytes (in Sektoren angeordnet) mit einer Transferadresse XXXX:1 übertragen werden, dann werden die letzten beiden Bytes bei der Übertragung ignoriert, also nicht geschrieben.

*** Installierbare Gerätetreiber ***

Anmerkung:

Ein Programm, das mit DCP-Funktionsaufrufen arbeitet, kann niemals eine Ein- oder Ausgabeoperation von mehr als FFFFh Bytes fordern. Deshalb kann ein Überlauf im Transfer(Puffer)-Segment nicht auftreten. Das ist der Grund dafür, dass der Treiber die Bytes, die im Transfersegment einen Überlauf erzeugen würden, vernachlässigen kann.

Gibt der Treiber den Fehler-Kode 0Fh (unerlaubter Diskettenwechsel) zurück, dann muss er einen DWORD-Zeiger auf eine Zeichenkette einstellen, welche die korrekte Datenträgeridentifikation beinhaltet und den Nutzer auffordert, die gewechselte Diskette wieder einzulegen.

Der Referenzzähler geöffneter Dateien auf der Diskette (verwaltet durch Funktionsaufrufe für OPEN und CLOSE) erlaubt dem Treiber zu bestimmen, wann ein Fehler 0Fh zurückzugeben ist. Wenn keine eröffneten Dateien vorliegen (d.h. Referenzzähler=0), und die Diskette ist gewechselt worden, dann sind Ein- und Ausgaben erlaubt und der Fehler 0Fh wird nicht zurückgemeldet. Existieren aber eröffnete Dateien (Referenzzähler>0), und die Diskette wurde gewechselt, dann wird damit eine Situation erzeugt, die den Fehler 0Fh provoziert.

6.7.6. NONDESTRUCTIVE INPUT NO WAIT

Kommandokode = 5

Abruf eines Zeichens aus dem Eingabekanal (ohne Entnahme)

ES:BX	Länge	Feld
+0	13. Byte	Request Header
+13	BYTE	vom Gerät gelesenes Byte

Der Treiber muss folgendes leisten:

- Rückgabe eines Bytes vom Gerät
- Setzen des Status im Request Header.

Wenn das zeichenorientierte Gerät das Besetztbit=0 (Zeichen im Puffer) zurückgibt, dann wird das nächste gelesene Zeichen zurückgegeben. Dieses Zeichen wird nicht aus dem Eingabepuffer entfernt, deshalb die Bezeichnung 'zerstörungsfreie Eingabe'. Dieser Funktionsruf erlaubt DCP, an das darauffolgende Zeichen heranzukommen.

6.7.7. STATUS

Kommandokodes = 6,10

Status-Ein-, bzw. -Ausgabe zeichenorientierter Geräte

ES:BX Länge Feld

+0 13 Byte Request Header

Der Treiber muss folgendes leisten:

- Ausführung der verlangten Funktion
- Setzen Besetztbit
- Setzen des Status im Request Header.

Das Besetztbit wird wie folgt gesetzt:

Ausgabe auf ein zeichenorientiertes Gerät:

Wird Besetztbit=1 zurückgegeben, dann wartet die Anforderung für eine Schreiboperation auf die Beendigung der zur Zeit ausgeführten Funktion, d.h., das Gerät ist besetzt. Ist das Besetztbit=0, dann ist das Gerät bereit, und eine Schreibenanforderung könnte sofort ausgeführt werden.

Eingabe von einem zeichenorientierten Gerät mit Puffer:

Wird Besetztbit=1 zurückgegeben, dann geht eine Lese-Anforderung an das physische Gerät. Ist das Besetztbit=0, dann sind Zeichen im Gerätepuffer, und eine Leseanforderung kehrt sofort zurück. Es wird damit auch darauf hingewiesen, dass der Nutzer etwas eingegeben hat. DCP geht davon aus, dass alle zeichenorientierten Geräte einen sequentiellen Eingabepuffer haben. Geräte, die diesen Puffer nicht haben, müssen immer Besetztbit=0 zurückmelden. Andernfalls würde DCP ständig darauf warten, Daten in einen Puffer zu übertragen, den es gar nicht gibt.

6.7.8. FLUSH

Kommandokodes = 7,11

Leeren des Ein- bzw. Ausgabe-Puffers

ES:BX Länge Feld

+0 13 Byte Request Header

Dieser Funktionsaufruf weist den Treiber an, alle noch offenen Forderungen, die zuvor angenommen wurden, zu löschen. Diese Funktion wird hauptsächlich dazu genutzt, die Warteschlange für Eingabeanforderungen an zeichenorientierte Geräte zu leeren. Der Treiber setzt den Status im Request Header bei Rückkehr aus der Funktion.

6.7.9. OPEN oder CLOSE

Kommandokodes = 13,14

Eröffnen bzw. Schliessen der Geräte

```
-----  
ES:BX Länge   Feld  
-----  
+0    13 Byte Request Header  
-----
```

Mit dem Aufruf dieser Funktionen werden bei gesetztem Bit 11 im Attributbyte Informationen über die momentane Dateiarbeit des Gerätes bereitgestellt. Bei Blockgeräten können diese Kommandos zur Verwaltung der lokalen Pufferung verwendet werden. Jedes Gerät kann einen Referenzzähler führen. Bei jedem OPEN wird der Referenzzähler inkrementiert und bei jedem CLOSE dekrementiert. Wenn der Referenzzähler Null ist, dann ist auf dem Gerät keine Datei eröffnet. In diesem Fall sind alle internen Puffer des Gerätes, in die geschrieben wurde, wegzuschreiben und freizugeben, damit der Nutzer jetzt den Datenträger wechseln kann. Wenn der Datenträger gewechselt wurde, ist es empfehlenswert, den Referenzzähler auf Null zu setzen, ohne dabei die Puffer zu leeren. Das kann vernachlässigt werden, weil die letzte CLOSE-Operation ein Leeren der Puffer veranlasst.

Diese Funktionen sind insbesondere für zeichenorientierte Geräte nützlich. Mit dem Kommando OPEN kann eine Initialisierungsfolge an das Gerät übertragen werden. Für einen Drucker können mit einer Initialisierungsfolge die Schriftart, das Papierformat usw. eingestellt werden. Damit befindet sich der Drucker für die nachfolgenden Ein- und Ausgaben in einem definierten Zustand. In gleicher Weise kann mit dem Aufruf des CLOSE-Kommandos eine abschliessende Zeichenfolge an den Drucker gesendet werden, wie z.B. ein Form-Feed, d.h. Blattvorschub auf den Anfang der nächsten Seite. Die Verwendung von IOCTL zur Ausgabe dieser einleitenden und abschliessenden Zeichenfolgen bietet die Möglichkeit einer flexiblen Steuerung von seriellen Ein- und Ausgaben.

Anmerkung:

Es besteht für alle Prozesse Zugriff auf STDIN, STDOUT, STDERR, STDAUX und STDP RN (0,1,2,3,4) der logischen Geräte CON, AUX und PRN, denn diese Geräte sind immer eröffnet. DCP nutzt diese Funktion bei Aufruf von 0Fh, 10h, 3Dh und 3Eh des Interrupt 21h nur, wenn Datei-Sharing (SHARE, EXE) geladen wurde.

6.7.10. REMOVABLE MEDIA

Kommandokode = 15

Kontrolle der Möglichkeit des Datenträger austausches

ES:BX	Länge	Feld
+0	13 Byte	Request Header

Wenn diese Funktion genutzt werden soll, dann ist das Bit 11 im Attributfeld auf 1 zu setzen. Blockorientierte Geräte können diese Funktion nur über den Aufruf einer Unterfunktion des IOCTL-Funktionsaufrufes (44h) benutzen. Über den Aufruf dieser Funktion kann ein Dienstprogramm herausfinden, ob für das angesprochene Laufwerk ein Datenträger austausch möglich ist oder nicht. Eine solche Information ist z.B. für das Dienstprogramm FORMAT interessant.

Die Information wird im Besetzbit des Status zurückgegeben:
Besetzbit = 1 ---> Datenträger austausch nicht möglich
Besetzbit = 0 ---> Datenträger austausch möglich

Anmerkung:

Es wird keine Fehlerbit-Kontrolle ausgeführt, da angenommen wird, dass diese Funktion immer erfolgreich abläuft.

6.7.11. GENERIC IOCTL REQUEST

Kommandokode = 19

ES:BX	Länge	Feld
+0	13 Bytes	Request Header
+13	BYTE	Hauptfunktion
+14	BYTE	Unterfunktion
+15	WORD	Inhalt von SI
+17	WORD	Inhalt von DI
+19	DWORD	Zeiger auf das Generic-IOCTL- Anforderungspaket

Der Treiber muss folgendes leisten:

- Unterstützung der unter Generic-IOCTL-Anforderung beschriebenen Funktionen
- Verwalten der eigenen Spurtabelle (Track Layout).

DCP verwendet sowohl zum Setzen und Abfragen von Geräteparametern als auch zum Formatieren und Prüfen einer Spur auf einem logischen Gerät den Aufruf AL=0Dh. Weiterhin verwendet DCP den Aufruf AL=0Ch für das Umschalten der Zeichensatztabelle. Die von Generic IOCTL unterstützten Funktionen sind im Kapitel 9 unter IOCTL 44h erklärt.

6.7.12. GET LOGICAL DEVICE

Kommandokode = 23

Abfrage des logischen Gerätes

ES:BX	Länge	Feld
+0	13 Bytes	Request Header
+13	BYTE	Eingabe (Gerätekode)
+14	BYTE	Kommandokode
+15	WORD	Status
+17	DWORD	Reserviert

6.7.13. SET LOGICAL DEVICE

Kommandokode = 24

Setzen des logischen Gerätes

ES:BX	Länge	Feld
+0	13 Bytes	Request Header
+13	BYTE	Eingabe (Gerätekode)
+14	BYTE	Kommandokode
+15	WORD	Status
+17	DWORD	Reserviert

6.8. Das CLOCKX-Gerät (Zeitgeberereinheit)

Eine stark gefragte Eigenschaft stellt die Echtzeituhr dar. Um diese Eigenschaft für TIME und DATE (Uhrzeit und Datum) in das System einzubinden, gibt es ein spezielles Gerät (definiert im Attribut-WORD), welches als CLOCKX-Gerät bezeichnet wird. Dieses Gerät arbeitet wie jedes andere zeichenorientierte Gerät. Bei einer Lese- oder Schreiboperation werden genau 6 Bytes übertragen.

Bedeutung der einzelnen Bytes:

Die ersten 2 Bytes stellen den Tageszähler dar, in dem die Tage ab 1.1.1980 bis zum aktuellen Datum gezählt werden. Das dritte Byte enthält die Minuten, das vierte die Stunden, das fünfte die 1/100 Sekunden und das sechste die Sekunden. Beim Lesen des CLOCKX-Gerätes erhält man Datum und Uhrzeit, und beim Schreiben werden Datum und Uhrzeit gesetzt.

6.9. Beispiel eines Gerätetreibers

Als Beispiel für den Aufbau eines Device Header und eines Request Header soll der blockorientierte Gerätetreiber VDISK dienen. Mit VDISK wird im RAM-Speicher des PC ein Diskettenlaufwerk nachgebildet.

Der Device Header hat folgenden Aufbau:

```

;DEVICE HEADER
  DD  -1                ;Zeiger auf nächsten
                        ;Device Header
  DW  0800h            ;Attribut (IBM-Format, block-
                        ;orientiertes Gerät,
                        ;unterstützt OPEN/CLOSE/RM)
  DW  OFFSET STRATEGIE ;Zeiger auf Strategieroutine
  DW  OFFSET INTERRUPT ;Zeiger auf Interruptroutine
  DB  1                ;Anzahl blockorientierter Geräte
  DB  7 DUP (?)       ;Restbyte der Gerätebezeichnung
;Ende DEVICE HEADER

```

Aus dem angegebenen Attribut lassen sich folgende Informationen entnehmen:

- Es handelt sich um einen Treiber für ein blockorientiertes Gerät (Bit 15=0).
- Es werden keine IOCTL-Funktionen unterstützt (Bit 14=0).
- Es handelt sich um einen Gerätetreiber mit IBM-Format (Bit 13=0).
- Datenträgeraustausch wird unterstützt (Bit 11=1).
- Es werden keine Generic-IOCTL-Funktionsaufrufe unterstützt (Bit 6=0).

Die Bits 4-5, 7-10 und 12 sind durch DCP reserviert, und die Bits 0-3 haben nur für zeichenorientierte Geräte eine Bedeutung. Über Request Header werden dem Treiber die zur Ausführung eines Kommandos notwendigen Daten übermittelt. Gleichzeitig hat aber auch der Treiber selbst die Möglichkeit, über diese Struktur Informationen (z.B. über den Erfolg der Kommandoausführung) an das DCP zurückzugeben.

Der Request Header für das Kommando INIT sieht wie folgt aus:

```

-----
; Request Header für das Kommando INIT
-----
RHI  STRUC
      DB  23          ;Gesamtlänge Request Header
      DB  0           ;Gerätekodex
RHI_CMD  DB  0       ;Kommandokode für INIT
RHI_STA  DB  0       ;Status (=0 bei Aufruf des Kommandos)
          DD  ?       ;für DCP reservierter Bereich

; Ende des Teiles konstanter Länge (13 Bytes)
;
RHI_NUM  DB  ?       ;Anzahl von Geräten
                        ;=0, wenn Fehler bei Initialisierung

```

*** Installierbare Gerätetreiber ***

```
                                ; < > 0, wenn Initialisierung
                                ; erfolgreich
RHI_ENDO DW ?                   ; Offset Endadresse Treiber
RHI_ENDS DW ?                   ; Segment Endadresse Treiber
RHI_BPBO DW ?                   ; Offset Adresse BPB-Feld
RHI_BPBS DW ?                   ; Segment Adresse BPB-Feld
RHI_LWK  DB 2                   ; Laufwerkcode
; Ende des variablen Teiles
RHI     ENDS
```

Die Adresse des Request Header wird bei Aufruf des Strategie-Eintrittspunktes im Treiber abgespeichert. Erst der anschließende Aufruf des Interrupt-Eintrittspunktes führt schliesslich zur Ausführung des Kommandos. Am Interrupt-Eintrittspunkt werden zunächst alle Register gerettet, anschliessend die zuvor gespeicherte Adresse des Request Header geladen und das Kommandobyte ausgewertet. Die Auswertung resultiert im Verzweigen zu dem Teil des Treibers, der die für das verlangte Kommando notwendigen Arbeiten vornimmt. Nach der Ausführung eines Kommandos wird der Status gesetzt, alle am Beginn geretteten Register zurückergeben und wieder zum DCP verzweigt.

6.10. Zum System gehörende Gerätetreiber

Zum System werden die Gerätetreiber VDISK.SYS und VE89.SYS zur Verfügung gestellt.

6.10.1. Der Gerätetreiber VDISK.SYS (virtuelle Diskette)

Die VDISK.SYS-Datei auf der DCP-Diskette ist ein Einheitentreiber, der ein Plattenlaufwerk simuliert, wobei ein Teil des Hauptspeichers als Speichermedium verwendet wird. Diese nicht physisch vorhandenen Laufwerke werden als virtuelle Laufwerke bezeichnet.

Für diese virtuellen Laufwerke gilt folgendes:

- Virtuelle Laufwerke sind schnell, da sie mit Hauptspeichergeschwindigkeit arbeiten.
- Es können mehrere virtuelle Laufwerke installiert werden, wobei jedes mit einem Laufwerksbuchstaben angesprochen wird (wie Diskettenlaufwerke). Hat der Computer beispielsweise zwei Diskettenlaufwerke (aber keine Festplatten), dann werden die Diskettenlaufwerke mit A: und B: und die virtuellen Laufwerke mit C:, D: usw. angesprochen.
- Für jedes virtuelle Laufwerk können die gewünschte Speicherkapazität ("Plattengrösse"), Sektorgrösse und Anzahl Verzeichniseinträge angegeben werden.
- Jedes virtuelle Laufwerk wird mit einem Namen versehen.
- Durch jedes virtuelle Laufwerk wird der residente Teil von DCP um ca. 800 Byte für den VDISK.SYS-Einheitentreiber und um die für das virtuelle Laufwerk angegebene Puffergrösse erweitert, wenn der Treiber im niedrigen Speicherbereich installiert ist.
- Bei erneutem Starten des Systems bzw. bei Stromausfall geht der Inhalt des virtuellen Laufwerkes verloren.

*** Installierbare Gerätetreiber ***

- Virtuelle Laufwerke können nicht formatiert werden, da sie mit der Installation bereits formatiert sind.

6.10.1.1. Installation des Gerätetreibers VDISK.SYS

Die Installation des Gerätetreibers VDISK.SYS erfordert die Aufnahme des nachfolgenden Kommandos in die Datei CONFIG.SYS:

```
DEVICE=[zweig]VDISK.SYS ["zeichenfolge"] [bbb] ["zeichenfolge"]  
[sss] ["zeichenfolge"] [ddd] [/E[:n]]
```

Folgende Parameter können angegeben werden, um die Eigenschaften des virtuellen Laufwerkes festzulegen:

- [zweig] kennzeichnet das Laufwerk und den Verzeichnispfad der Datei VDISK.SYS.
- ["zeichenfolge"] enthält ASCII-Zeichen im Bereich zwischen 32 und 126 bzw. 20h und 7Eh (ausser dem Schrägstrich (/)) zur Angabe von Bemerkungen.
- [bbb] stellt die Grösse des virtuellen Laufwerkes in KByte dar und ist als Dezimalwert anzugeben. Der Standardwert ist 64 KByte. Der Wert für das virtuelle Laufwerk kann zwischen 1 KByte und der verfügbaren Hauptspeichergrösse liegen. Durch VDISK wird die für das virtuelle Laufwerk benutzte Speichergrösse wie folgt angeglichen:
 - (a) Wenn bei der Installation von VDISK weniger als 64 KByte Speicher verfügbar sind, dann gibt VDISK eine Fehlermeldung aus und erstellt kein virtuelles Laufwerk.
 - (b) Ist der angegebene Wert kleiner als 1 KByte oder grösser als der verfügbare Speicherbereich, dann werden von VDISK standardmässig 64 KByte angenommen.
 - (c) Bleiben durch die angegebene Grösse weniger als 64 KByte verfügbarer Hauptspeicher übrig, dann gleicht VDISK die Grösse des virtuellen Laufwerkes so an, dass auch nach deren Installation mindestens 64 KByte Speicherbereich zur Verfügung stehen.
 - (d) In die Grösse des virtuellen Laufwerkes gehen ein Umladesektor, die Dateizuordnungstabelle und die Verzeichniseinträge mit ein, so dass für Dateien weniger Platz, als durch bbb angegeben, zur Verfügung steht.
- [sss] gibt die Sektorgrösse in Byte an. Zulässig sind die Werte 128, 256 und 512. Wird keine oder eine falsche Angabe gemacht, dann werden von VDISK standardmässig 128 Byte verwendet. Soll das virtuelle Laufwerk vorrangig kleine Dateien aufnehmen, dann wird eine geringere Sektorgrösse empfohlen, um Platz zu sparen. Die Verarbeitung erfolgt schneller bei einem hohen Wert für die Sektorgrösse.

*** Installierbare Gerätetreiber ***

- [ddd] gibt die Anzahl der Verzeichniseinträge (Anzahl Dateien) an, die das virtuelle Laufwerk aufnehmen soll. Dabei ist für jede Datei ein Verzeichniseintrag erforderlich. Der Standardwert ist 64. Die gültigen Werte liegen zwischen 2 und 512.
VDISK nimmt folgende Korrekturen vor:
 - (a) Der Wert wird an die nächsthöhere Grenze der Sektorgröße angeglichen. Wenn z.B. für ddd=10 angegeben wird und die Sektorgröße 128 ist, dann erstellt VDISK 12 Verzeichniseinträge (12 Einträge zu je 32 Byte ergeben ein Vielfaches der Sektorgröße).
 - (b) Wurde die Größe des virtuellen Laufwerkes sehr klein angegeben, um die Dateizuordnungstabelle, das Verzeichnis und zwei zusätzliche Sektoren aufzunehmen, dann wird die Verzeichnisgröße solange um jeweils einen Sektor reduziert, bis die notwendigen Bedingungen erfüllt sind. Erreicht die Verzeichnisgröße 1 Sektor, und die Bedingungen sind immer noch nicht erfüllt, gibt VDISK eine Fehlermeldung aus, und das virtuelle Laufwerk wird nicht erstellt.
 - (c) VDISK benutzt einen Verzeichniseintrag für den Namen des virtuellen Laufwerkes.
- [/E[im]] ist ein Parameter für Speichererweiterungen, welcher beim EC 1834 nicht verwendet wird.

6.10.1.2. Beispiel

Durch den Befehl

```
DEVICE=VDISK.SYS 160 512 64           oder  
DEVICE=VDISK.SYS Speicher=160 Sektor=512 Anzeintr=64
```

wird ein virtuelles Laufwerk von 160 KByte mit 512 Byte-Sektoren und 64 Verzeichniseinträgen erstellt.

6.10.2. Der Gerätetreiber VE89.SYS

Der Treiber VE89.SYS unterstützt zusätzliche Funktionen der Bildschirm- und Tastatursteuerung in DCP, die Grafikmodi 8 (640x480) und 9 (640x400) sowie die Funktionen 10h, 11h und 12h des INT 10h. Mit Hilfe von Schaltern ist es möglich, gewisse Funktionen zu unterstützen und andere nicht. Der Treiber verlangt einen ROM-BIOS-Stand ab 3/88. Ist das ROM-BIOS älteren Datums, wird automatisch nur der ANSI-Teil (Escape-folgenunterstützung) des Treibers installiert.

6.10.2.1. Installation von VE89.SYS

Zur Installation von VE89.SYS muss in die Datei CONFIG.SYS folgendes Kommando aufgenommen werden:

```
DEVICE = [zweig]VE89.SYS [schalter[...]]
```

*** Installierbare Gerätetreiber ***

Mögliche Schalter sind:

- /=ANSI nur die Escapefolgeunterstützung des VE89.SYS wird installiert (2480 Byte des Hauptspeichers werden belegt)
- /ANSI- VE89.SYS wird installiert, die Escapefolgenunterstützung wird unterdrückt (ist aber mit VMODE wieder einstellbar).
- /CP- VE89.SYS wird installiert, der Zeichensatzwechsel wird unterdrückt. (ist aber mit VMODE wieder einstellbar).
- /CPE:Zahl Für den Zeichensatzwechsel werden entsprechend der angegebenen Zahl Puffer reserviert (pro Puffer ca. 10112 Byte des Hauptspeichers). Es sind damit so viele Zeichensätze vorbereitbar (mit MODE CON:CP PREPARE), wie die Zahl angibt. Standard: 3 Die Zahl muss im Bereich zwischen 0 und 12 liegen.
- /INT42 Der INT 42h wird unterstützt (Anderung über Dienstprogramm VMODE möglich).

Werden mehrere Schalter angegeben, sind diese durch Leerzeichen voneinander zu trennen.

6.10.2.2. Die Escapefolgenunterstützung (ANSI-Teil)

6.10.2.2.1. Wirkungsbereich

Ist VE89.SYS installiert und die Escapefolgenunterstützung ausgeschaltet, wirken eventuell umkodierte Tasten weiterhin.

6.10.2.2.2. Syntax der Steuerfolgen

Jede Bildschirmsteuerfolge besitzt folgendes Format:

ESC [parameter Kommando

Dabei gilt:

ESC	Symbol für 1Bh bzw. ^[
[Das Zeichen [
parameter	Das Zeichen # ist ein numerischer Parameter, d.h. ein mit ASCII-Zeichen dargestellter ganzzahliger Wert. Das Zeichen # muss durch einen numerischen Wert ersetzt werden. Wird kein Parameter oder der Wert 0 angegeben, so wird für den Parameter der Standardwert eingesetzt.

*** Installierbare Gerätetreiber ***

ESC | Symbol für IBh bzw. ^[

Kommando | Bestimmte Anzahl von Buchstaben, die das Kommando
darstellen.

Beispiel:

ESC [5;16H

Diese Steuerfolge könnte in BASIC folgendermassen programmiert werden:

```
OK
open "kurz" for output as #1
OK
print #1, CHR$(27);"[5;16H";"Kursor auf Zeile 5 Spalte 16"
OK
close #1
OK
```

Dabei ist "CHR\$(27)" die Darstellung von ESC.

4.10.2.2.3. Bildschirmsteuerfolgen

Der folgende Abschnitt beschreibt die Steuerfolgen, die für die Positionierung des Kursors benutzt werden können:

Steuerfolge	Beschreibung

ESC [#;#H	Kursorpositionierung (Zeile und Spalte) Bewegt den Kursor auf die festgelegte Position. Der erste numerische Parameter kennzeichnet die Nummer der Zeile und der zweite die Nummer der Spalte. Als Trennzeichen dient ;. Der Standardwert der numerischen Parameter ist 1. Keine Parameterangabe setzt den Kursor auf die linke obere Ecke (HOME Position).
ESC [#AH	Kursor nach oben Bewegt den Kursor eine oder mehrere Zeilen nach oben, ohne die Spaltenposition zu verändern. Der Wert von # bestimmt die Anzahl der Zeilen, um die der Kursor bewegt wird. Der Standardwert ist 1. Diese Steuerfolge wird ignoriert, wenn der Kursor bereits in der obersten Zeile steht.
ESC [#BH	Kursor nach unten Bewegt den Kursor eine oder mehrere Zeilen nach unten, ohne die Spaltenposition zu verändern. Der Wert von # bestimmt die Anzahl der Zeilen, um die der Kursor bewegt wird. Der Standardwert ist 1. Diese Steuerfolge wird ignoriert, wenn der Kursor bereits in der untersten Zeile steht.

*** Installierbare Gerätetreiber ***

- ESC [#C** **Kursor vorwärts**
Bewegt den Kursor in der jeweiligen Zeile eine oder mehrere Spalten vorwärts (nach rechts). Der Wert von # bestimmt die Anzahl der Spalten, um die der Kursor bewegt wird. Der Standardwert ist 1. Diese Steuerfolge wird ignoriert, wenn der Kursor bereits in der äussersten rechten Spalte steht.
- ESC [#D** **Kursor rückwärts**
Bewegt den Kursor in der jeweiligen Zeile eine oder mehrere Spalten rückwärts (nach links). Der Wert von # bestimmt die Anzahl der Spalten, um die der Kursor bewegt wird. Der Standardwert ist 1. Diese Steuerfolge wird ignoriert, wenn der Kursor bereits in der äussersten linken Spalte steht.
- ESC [#;##** **Kursorpositionierung (Zeile und Spalte)**
Bewegt den Kursor auf die festgelegte Position. Der erste numerische Parameter kennzeichnet die Nummer der Zeile und der zweite die Nummer der Spalte. Als Trennzeichen dient ;. Der Standardwert der numerischen Parameter ist 1. Keine Parameterangabe setzt den Kursor auf die linke obere Ecke (HOME Position).
- ESC [#;#R** **Kursor Positionsmeldung**
Diese Steuerfolge gibt die augenblickliche Kursorposition über die Standardeingabeeinheit aus. Der erste Parameter steht für die aktuelle Zeile und der zweite für die aktuelle Spalte.
- ESC [6n** **Treiber Status Meldung**
Der Konsoltreiber gibt eine Kursor-Positionsmeldung bei Empfang dieser Steuerfolge aus.
- ESC [s** **Kursorposition sichern**
Die aktuelle Kursorposition wird gespeichert.
- ESC [u** **Gesicherte Kursorposition anfordern**
Setzt den Kursor an die Stelle, die gesichert ist.
- ESC [2J** **Bildschirm löschen**
Löscht den gesamten Bildschirm. Der Kursor wird auf die linke obere Ecke (HOME Position) gesetzt.
- ESC [K** **Löschen in einer Zeile**
Löscht einschliesslich der Kursorposition bis zum Ende der Zeile.

6.10.2.2.4. Betriebsarten

Die folgenden Aufstellungen enthalten die Steuerfolgen für eine Veränderung der Bildschirm-Betriebsart.

Steuerfolge	Beschreibung																																														
ESC [#;...;#m	<p>Darstellungseigenschaften setzen (SGR - Set Graphics Rendition) Setzt entsprechend der angegebenen Parameter die Zeichenattribute. Alle nachfolgenden Zeichen besitzen das Attribut bis zum nächsten SGR.</p> <table border="1"> <thead> <tr> <th>Parameter</th> <th>Bedeutung:</th> </tr> </thead> <tbody> <tr><td>0</td><td>Alle Attribute aus (normal, Weiss auf Schwarz)</td></tr> <tr><td>1</td><td>Intensiv ein</td></tr> <tr><td>4</td><td>Unterstreichen ein (nur bei IBM monochrom Bildschirm)</td></tr> <tr><td>5</td><td>Blinken ein</td></tr> <tr><td>7</td><td>Invers ein</td></tr> <tr><td>8</td><td>Unsichtbar ein</td></tr> <tr><td>30</td><td>Vordergrund Schwarz</td></tr> <tr><td>31</td><td>" Rot</td></tr> <tr><td>32</td><td>" Grün</td></tr> <tr><td>33</td><td>" Gelb</td></tr> <tr><td>34</td><td>" Blau</td></tr> <tr><td>35</td><td>" Magenta</td></tr> <tr><td>36</td><td>" Cyan</td></tr> <tr><td>37</td><td>" Weiss</td></tr> <tr><td>40</td><td>Hintergrund Schwarz</td></tr> <tr><td>41</td><td>" Rot</td></tr> <tr><td>42</td><td>" Grün</td></tr> <tr><td>43</td><td>" Gelb</td></tr> <tr><td>44</td><td>" Blau</td></tr> <tr><td>45</td><td>" Magenta</td></tr> <tr><td>46</td><td>" Cyan</td></tr> <tr><td>47</td><td>" Weiss</td></tr> </tbody> </table>	Parameter	Bedeutung:	0	Alle Attribute aus (normal, Weiss auf Schwarz)	1	Intensiv ein	4	Unterstreichen ein (nur bei IBM monochrom Bildschirm)	5	Blinken ein	7	Invers ein	8	Unsichtbar ein	30	Vordergrund Schwarz	31	" Rot	32	" Grün	33	" Gelb	34	" Blau	35	" Magenta	36	" Cyan	37	" Weiss	40	Hintergrund Schwarz	41	" Rot	42	" Grün	43	" Gelb	44	" Blau	45	" Magenta	46	" Cyan	47	" Weiss
Parameter	Bedeutung:																																														
0	Alle Attribute aus (normal, Weiss auf Schwarz)																																														
1	Intensiv ein																																														
4	Unterstreichen ein (nur bei IBM monochrom Bildschirm)																																														
5	Blinken ein																																														
7	Invers ein																																														
8	Unsichtbar ein																																														
30	Vordergrund Schwarz																																														
31	" Rot																																														
32	" Grün																																														
33	" Gelb																																														
34	" Blau																																														
35	" Magenta																																														
36	" Cyan																																														
37	" Weiss																																														
40	Hintergrund Schwarz																																														
41	" Rot																																														
42	" Grün																																														
43	" Gelb																																														
44	" Blau																																														
45	" Magenta																																														
46	" Cyan																																														
47	" Weiss																																														
ESC [=#h	Mode setzen																																														
ESC [?7h	<p>Legt durch den Parameter den Bildschirmmode fest.</p> <table border="1"> <thead> <tr> <th>Parameter</th> <th>Bedeutung:</th> </tr> </thead> <tbody> <tr><td>0</td><td>40* 25 alphanumerisch</td></tr> <tr><td>1</td><td>40* 25 alphanumerisch</td></tr> <tr><td>2</td><td>80* 25 alphanumerisch</td></tr> <tr><td>3</td><td>80* 25 alphanumerisch</td></tr> <tr><td>4</td><td>320*200 Grafik</td></tr> <tr><td>5</td><td>320*200 Grafik</td></tr> <tr><td>6</td><td>640*200 Grafik</td></tr> <tr><td>7</td><td>Zeilenumbruch (wird über das Zeilenende hinaus eingegeben, so wird eine neue Zeile angefangen)</td></tr> <tr><td>8</td><td>640x480 Grafik (falls unterstützt)</td></tr> <tr><td>9</td><td>640x400 Grafik (falls unterstützt)</td></tr> </tbody> </table>	Parameter	Bedeutung:	0	40* 25 alphanumerisch	1	40* 25 alphanumerisch	2	80* 25 alphanumerisch	3	80* 25 alphanumerisch	4	320*200 Grafik	5	320*200 Grafik	6	640*200 Grafik	7	Zeilenumbruch (wird über das Zeilenende hinaus eingegeben, so wird eine neue Zeile angefangen)	8	640x480 Grafik (falls unterstützt)	9	640x400 Grafik (falls unterstützt)																								
Parameter	Bedeutung:																																														
0	40* 25 alphanumerisch																																														
1	40* 25 alphanumerisch																																														
2	80* 25 alphanumerisch																																														
3	80* 25 alphanumerisch																																														
4	320*200 Grafik																																														
5	320*200 Grafik																																														
6	640*200 Grafik																																														
7	Zeilenumbruch (wird über das Zeilenende hinaus eingegeben, so wird eine neue Zeile angefangen)																																														
8	640x480 Grafik (falls unterstützt)																																														
9	640x400 Grafik (falls unterstützt)																																														

*** Installierbare Gerätetreiber ***

ESC [=#1 Mode rücksetzen
ESC [?7] Legt durch den Parameter den Bildschirmmode fest.
Es gelten analog die Parameter wie oben beschrieben. Der Parameter 7 bewirkt das Zurücksetzen des Zeilenumbruchs.

6.10.2.2.5. Umkodierung der Tastatur

Die folgende Tabelle enthält die Steuerfolgen für die Umkodierung der Tastaturbelegungen.

Steuerfolge	Beschreibung
ESC [#;#;...;#p	Der erste ASCII-Kode in der Steuerfolge definiert den neu zu belegenden Kode. Die restliche Anzahl Zeichen definiert die Zeichenfolge, die von der Taste neu geliefert werden soll.
ESC ["Kette"p	Beachte: Ist der erste Kode in der Steuerfolge 0 (00h), dann bilden der erste und der zweite Kode zusammen den erweiterten ASCII-Kode.
ESC [#;"Kette";#; #;"Kette";#p	
oder jede beliebige Kombination von Zeichenkette und Dezimalwert eines Zeichens	

6.10.2.2.6. Beispiele

Die Ausführung der Beispiele kann erfolgen durch:

- Erstellen einer Datei, die die Anweisungen enthält, und Darstellung dieser auf dem Bildschirm mit dem Kommando TYPE;
- Ausführung der Kommandos nach der DCP-Bereitschaftsanzeige.

1. Umdefinierung der Taste D bzw. d zu Z bzw. z sowie umgekehrt:

. Erstellen einer Datei

```
ESC [65;81p            A wird Q  
ESC [97;113p           a wird q  
ESC [81;65p            Q wird A  
ESC [113;97p           q wird a
```

Nach DCP-Bereitschaftsanzeige

```
prompt %e[65;81p        A wird Q  
prompt %e[97;113p       a wird q  
prompt %e[81;65p        Q wird A  
prompt %e[113;97p       q wird a
```

2. Belegung der Taste F10 mit der Zeichenkette DIR und einem nachfolgenden Wagenrücklauf (0Dh):

Erstellen einer Datei

```
ESC [0;68;"dir";13p
```

*** Installierbare Gerätetreiber ***

Nach DCP-Bereitschaftsanzeige

prompt [0;68;"dir";13p

Das Metasymbol [0 ist die Darstellung von ESC (1Bh) im DCP PROMPT-Kommando. Der erweiterte ASCII-Kode für die Taste F10 ist 0 und 68. Für den Wagenrücklauf steht 13.

- Das folgende Beispiel setzt das aktuelle Inhaltsverzeichnis des Standardlaufwerks in die oberste Zeile des Bildschirms und gleichzeitig die Standardlaufwerkbezeichnung in die aktuelle Zeile:

prompt [1;1f[2K[3p[4u[5d

6.10.2.3. Unterstützung des Zeichensatzwechsels

Wird diese Funktion nicht unterdrückt, können mit dem Kommando MODE Zeichensätze vorbereitet werden. Der Zeichensatz 437 ist schon als sogenannter Hardwarezeichensatz vorhanden und braucht nicht vorbereitet werden. Gleichzeitig sind so viele Zeichensätze vorbereitbar, wie mit dem Schalter /CPE bei der Installation angegeben wurden (Standard: 3).

Mit der Systemdiskette wird die Datei COL.CPI geliefert. Diese Datei enthält verschiedene Zeichensätze. In der dazugehörigen INFO.DOC ist nachzulesen, welche Zeichensätze enthalten sind. Der Nutzer kann eigene Zeichensatzdateien erstellen und diese Zeichensätze über das Kommando MODE vorbereiten und aktivieren. Eine solche Zeichensatzdatei hat folgenden Aufbau:

```
ORG 0
DB 0FFh
DB "FONT " ;Kennzeichen (7 Zeichen)
DB 8 DUP (0) ;reserviert
DW 1 ;reserviert
DB 1 ;reserviert
DW OFFSET INFO
DW SEG INFO ;relative Segmentadresse
; (mit 0 beginnend)
INFO:
DW 1 ;Anzahl der in der Datei
; enthaltenen Zeichen-
; sätze
INFO_KOPF:
DW INFO_KOPF_LANGE
DW 0 ;Offset des nächsten
; INFO-Kopfes der Datei
DW 0 ;relative Segmentadresse
; des nächsten INFO-
; Kopfes der Datei
DW 1 ;Gerätetyp
; (1=Bildschirm)
DB "COL " ;Gerätebezeichner
; (8 Zeichen)
DW 999 ;Zeichensatznummer
```

*** Installierbare Gerätetreiber ***

```
DW 3 DUP (0) ;reserviert
DW OFFSET DATENKOPF
DW SEG DATENKOPF ;relative Segmentadresse
INFO_KOPF_LANGE EQU 8-INFO_KOPF
```

DATENKOPF:

```
DW 1 ;reserviert
DW 3 ;Anzahl der unter-
;stützten Zeichen-
;größen
DW DATENLANGE
```

FONTKOPF_16:

```
DB 16,8 ;Zeichenbox (Zeilen,
;Spalten)
DB 0,0 ;reserviert
DW 256 ;Anzahl der Zeichen
;256x16 Byte Zeichen-
;muster
:
```

FONTKOPF_14:

```
DB 14,8 ;Zeichenbox (Zeilen,
;Spalten)
DB 0,0 ;reserviert
DW 256 ;Anzahl der Zeichen
;256x14 Byte Zeichen-
;muster
:
```

FONTKOPF_8:

```
DB 8,8 ;Zeichenbox
;reserviert
DW 256 ;Anzahl der Zeichen
;256x8 Byte Zeichen-
;muster
:
```

```
DATENLANGE EQU 8 - FONTKOPF_16
```

6.10.2.4. Bildschirmmodi und Monitore

Mit Hilfe des Dienstprogrammes VMODE lassen sich Modetabellen ändern. Diese Tabellen enthalten die Synchronparameter für die angeschlossenen Bildschirme.

Soll an einem Displayadapter ein Bildschirm angeschlossen werden, der andere Synchronparameter als die Standardbildschirme verlangt, kann der Nutzer eine Modetabellen-Datei mit den entsprechenden Synchronparametern erstellen und durch VMODE aktivieren.

Der Aufbau einer solchen Datei wird nun beschrieben.

```
DB 0FFh
DB "MODESET" ;Kennzeichen (7 Zeichen)
DB 8 DUP (0) ;reserviert
DW 1 ;reserviert
DB 1 ;reserviert
DW 1 ;Anzahl der in der Datei
```


*** Installierbare Gerätetreiber ***

			; enthaltenen Modetabellen
DW	TAB_LANGE		
DB	0		; Kennung (0 bis 99 ; möglich)
DB	7		; Mode (0 bis 9 sinnvoll, ; 7=für Monochromadapter)
DB	0		; Submode (s. unten- ; stehende Tabelle)
Tabelle:			
DW	80		; Spaltenzahl für Zeichen
DW	4096		; Länge einer Seite für ; die Standardzeichen- ; größe
DW	0B0Ch		; Cursorart
DW	3B4h		; Grundpostadresse ; (Farbgrafikadapter: 3D4h, ; Monochromadapter 3B4h)
DB	0FFh		; Ausgabewert auf Modus- ; steuerregister
DB	30h		; Ausgabewert auf ; Farbauswahlregister
DB	0		; Steuerpostausgabewert
DB	60h		; 3. Parameter von CCHAR
DB	28h		; 2. Parameter von CCHAR
DB	8Dh		; 1. Parameter von CCHAR
DB	0		; 1. Parameter von ZOOM
DB	15h		; 4. Parameter von PRAM
DB	0E0h		; 3. Parameter von PRAM
DB	50h		; 1. Parameter von PITCH
DB	31h		; 8. Parameter von SYNC
DB	5Eh		; 7. Parameter von SYNC
DB	1		; 6. Parameter von SYNC
DB	6		; 5. Parameter von SYNC
DB	4		; 4. Parameter von SYNC
DB	0C2h		; 3. Parameter von SYNC
DB	4Eh		; 2. Parameter von SYNC
DB	20h		; 1. Parameter von SYNC
DW	350		; Anzahl der Pixelzeilen

Der Submode kennzeichnet beim alphanumerischen Displayadapter den Bildschirm- und Adaptertyp, wie aus nachstehender Tabelle ersichtlich ist.

Submode	Adapter	Bildschirm	Statusport.Bit
			4 2 1
0	4MHz	K7229.21, K7229.22, K7229.23, K7229.24, ALPHA 1	1 0 0

 Submode | Adapter | Bildschirm | Statusport.Bit

*** Installierbare Gerätetreiber ***

			4	2	1
1	4MHz	3.20	1	0	1
2	4MHz	K7228.	1	1	0
3	4MHz	-	1	1	1
4	2MHz	K7229.21 bis K7229.24, ALPHA 1	0	0	0
5	2MHz	3.20	0	0	1
6	2MHz	K7228	0	1	0
7	2MHz	-	1	1	1

6.10.2.5. Schnittstellen

Im Segment des VE89.SYS (auch wenn nur die Escapefolgenunterstützung installiert ist) liegen verschiedene Merkmale, die nachfolgend beschrieben werden:

Offset	Bytezahl	Bedeutung
0EEh	1	Funktion (Wert für AH) zum Lesen der Tastatur über INT 16h für die Konsoleneingabe (Standardwert: 0)
0EFh	2	Anzahl der Pixelzeilen im aktuellen Mode
0F1h	1	Anzahl Pixelzeilen pro Zeichen
0F5h	1	Schalter 1: Bit0 = 1 Unterstützung des Zeichensatzwechsels, Bit4 = 0 Rollen über Anfangsadresse im Mode 640x480
0F6h	1	Schalter 2: Bit0 = 1 Unterstützung des Grafikmodes 640x480, Bit1 = 1 Unterstützung des Grafikmodes 640x400, Bit2 = 1 kein Löschen des Bildwiederholers bei Einstellen des Modes 640x480
0F7h	1	Schalter 3: Bit0 = 1 Funktion 10h des INT 10h unterstützt, Bit1 = 1 Funktion 11h des INT 10h unterstützt, Bit2 = 1 Funktion 12h des INT 10h unterstützt, Bit3 = 1 Programmieren des Zeichengenerators bei Modewechsel, Bit4 = 1

*** Installierbare Gerätetreiber ***

		Modewechsel ohne Löschen des Bildschirm, wenn Modenummer grösser 128, Bit5 = 1 Programmieren der Farbpalette bei Modewechsel
0F8h	1	Fehlermerkmale: Bit0 = 1 Fehler beim Lesen vom GDC, Bit1 = 1 Fehler beim Schreiben zum GDC, Bit2 = 1 Fehlende Bereitschaftsmeldung vom GDC, Bit4 = 1 Pufferverwaltungsfehler
0FAh	1	Auto-LF-Kennung: wenn >0, so kein automatischer Übergang zur nächsten Zeile, wenn Zeilenende erreicht ist
0FFh	1	aktueller Bildschirmcode
100h	1	Spaltenzahl - 1
101h	1	aktuelle Kursorposition (Spalte)
102h	1	aktuelle Kursorposition (Zeile)
109h	2	Offset des Puffers für die umkodierte Tastenkodes
115h	1	aktuelles Attribut

Die Segmentadresse von VE89.SYS ist bei Installation identisch mit der des INT 10h.

Ist der INT 42h unterstützt, sind folgende Funktionen möglich:

AH = 0 bis AH = 13h Funktionen des INT 10h
 AH = 24 Programmieren des Zeichengenerators

Aufruf:

Register	Inhalt/Bedeutung
AH	24 = 18h
CX	Anzahl zu programmierender Zeichen
AL	Kode des ersten zu programmierenden Zeichens
DL	Anzahl Pixelzeichen pro Zeichen
DS:BX	Adresse des Zeichengenerators

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AX	Anzahl nicht progr. Zeichen (>0 heisst: Fehler)

*** Installierbare Gerätetreiber ***

AH = 31 Farbpalette programmieren

Aufruf:

Register	Inhalt/ Bedeutung
AH	31 = 1Fh
CX	Anzahl zu programmierender Farben
AL	Nummer der ersten zu progr. Farbe
DS:BX	Tabelle der Farbkodes (CX x 3 Bytes) in der Reihenfolge rot, grdn, blau mit jeweils Bit7 = 0 keine Neuprogrammierung, Bit7 = 1 neu programmieren entsprechend Bits 0 bis 3

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

AH = 32

Aufruf:

Register	Inhalt/Bedeutung
AH	32 = 20h
AL	0 Versionsnummer abfragen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Hauptversionsnummer von VE89
AH	Unterversionsnummer von VE89
CX	Name ('89')

Aufruf:

Register	Inhalt/Bedeutung
AH	32 = 20h
AL	1

*** Installierbare Gerätetreiber ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CH	maximale Modenummer
CL	maximale Funktionsnummer für
	INT 42h

Z. BIOS-Interrupts

Z.1. ROM-BIOS

Das auf ROM-Schaltkreisen abgespeicherte Basis-Ein-/Ausgabe-System (ROM-BIOS) dient zur Steuerung der Standard-Ein-/Ausgabegeräte des EC 1834. Es kann in die beiden folgenden Hauptbestandteile gegliedert werden:

- Initialisierungsroutinen
- Ein-/Ausgabe-Routinen.

Die Initialisierungsroutinen werden nach dem Einschalten oder einem Software-Reset aktiviert. Sie schliessen den Einschalttest, das Initialisieren der Hardware und das Laden und Starten des Bootrecords ein (vgl. Kapitel 3).

Die Ein-/Ausgabe-Routinen übernehmen die physische Steuerung folgender Systemkomponenten:

- Disketten
- Bildschirm
- Tastatur
- asynchrone Kommunikation (V.24, IFSS)
- Drucker (Centronics)
- Zeitgeber.

Sie befreien die logischen Routinen des Betriebssystems bzw. der Nutzerprogramme von der physischen E/A-Programmierung. Ausserdem garantieren die E/A-Routinen bei eventuellen Hardwareänderungen stets gleiche Schnittstellen zur Bedienung der Ein-/Ausgabegeräte.

Auf die BIOS-Bestandteile wird über die Interruptvektoren 00h bis 1Fh des K1810 WMB6 zugegriffen. Im BIOS werden folgende Gruppen von Interrupts verarbeitet:

- Prozessorinterrupts
- hardwaregesteuerte Interrupts
- Softwareinterrupts
- Zeiger auf Tabellen.

Die Prozessorinterrupts werden bei der Abarbeitung von speziellen CPU-Befehlen ausgelöst.

Hardwaregesteuerte Interrupts werden über den Interrupt-Controller des EC 1834 ausgelöst. In den zugehörigen Interrupt-Routinen werden Aktionen zur Daten- und Status-Steuerung durchgeführt.

Die Softwareinterrupts bilden die eigentliche Schnittstelle zwischen dem ROM-BIOS und dem Nutzerprogramm. Der Datenaustausch zwischen BIOS-Routinen und Nutzerprogramm erfolgt über Prozessorregister.

Mit den Zeigern auf Tabellen werden für verschiedene BIOS-Routinen erforderliche Parameter adressiert.

Im ROM-BIOS ist die Möglichkeit der Einbindung weiterer Ein-/Ausgaberroutinen vorgesehen. Im Kapitel 6 wird beschrieben, welche Bedingungen dabei zu beachten sind.

7.2. Erweiterung des ROM-BIOS

Die E/A-Steuerungen des EC 1834 können BIOS-Erweiterungen auf ROM's enthalten. Ihr Aufbau und ihre Integration in das ROM-BIOS werden nachfolgend beschrieben.

Nach Abarbeitung der Initialisierungsroutinen des ROM-BIOS werden die Speicherbereiche von C8000 bis F0000 und F8000 bis FC000 in 2K-Schritten nach gültigen Erweiterungen zum ROM-BIOS durchsucht. Eine gültige ROM-Erweiterung muss an einer 2KByte-Speicheradresse beginnen und folgenden Aufbau besitzen:

Byte 0: 055h

Byte 1: 0AAh

Byte 2: Längenindikator

Byte 3: Startadresse der Initialisierungsroutine der BIOS-Erweiterung

Mit dem Längenindikator wird die Speichergröße der Erweiterung als Anzahl von 512Byte-Blöcken angegeben. Jedes Byte der Erweiterung wird modulo 100h addiert. Das Ergebnis muss 0 sein. Wird eine gültige Erweiterung erkannt, dann wird vom ROM-BIOS ein CALL FAR zur Startadresse der Initialisierungsroutine ausgeführt.

Die Initialisierungsroutine übernimmt die Initialisierung der zugehörigen Hardware, aktualisiert die entsprechenden Interruptvektoren und gibt mit einem RET FAR die Steuerung an das ROM-BIOS zurück.

Ein Beispiel für eine ROM-BIOS-Erweiterung ist die Steuerung des HD-Adapters.

7.3. DIVISION DURCH NULL

INT00

Funktion:

Dieser Interrupt bietet die Möglichkeit, eine bei einem Mikrobefehl (DIV, IDIV) aufgetretene Division durch Null innerhalb einer Interruptroutine zu behandeln.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

7.4. EINZELSCHRITTVERARBEITUNG

INT01

Funktion:

Bei gesetztem TRAP-Flag (TF=1) wird immer nach Abarbeitung eines Mikrobefehls in die angeschlossene Interruptroutine verzweigt.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

7.5. NMI-UNTERBRECHUNGEN

INT02

Funktion:

Über diesen Interrupt werden sowohl Paritätsfehler lokalisiert und angezeigt als auch verschiedene Hardwarekomponenten gesteuert.

Bemerkung:

Aufgrund der Steuerung von Hardwarekomponenten darf keine andere Interruptroutine angeschlossen werden!

Ein Paritätsfehler wird wie folgt gemeldet:

PARITY CHECK 2 (Fehler auf der Systemkarte)

PARITY CHECK 1 (Fehler auf der RAM-Erweiterungskarte)

Zusätzlich wird noch die Adresse des ersten fehlerhaften 16KByte-Speicherblockes ausgegeben. Dahinter steht ein 'S' für Systemkarte bzw. 'E' für Erweiterungskarte. Wird die Fehleradresse nicht gefunden, dann werden anstelle der Adresse Fragezeichen ausgegeben,

7.6. PRÜFPUNKT

INT03

Funktion:

Dieser Interrupt ist reserviert für einen 1Byte-Software-Unterbrechungsbefehl.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

7.7. ÜBERLAUFSUNTERBRECHUNG

INT04

Funktion:

Mit dem Mikrobefehl INTO kann bei Überlaufkennzeichenbit OF=1 die angeschlossene Interruptroutine aktiviert werden.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

7.8. BILDSCHIRMINHALT DRUCKEN

INT05

Funktion:

Mit diesem Interrupt wird der Bildschirminhalt ausgedruckt.

Bemerkung:

Das Ergebnis der Interruptroutine wird im Statusbyte auf der Adresse 50:0 wie folgt zurückgemeldet:

Statusbyte = 0: Bildschirminhalt erfolgreich ausgedruckt
= 1: Bildschirminhalt wird bereits gedruckt, erneute Aufforderung wird ignoriert.
= 255: Beim Ausdruck des Bildschirminhaltes ist ein Fehler aufgetreten.

7.9. Reservierter Interrupt

INT06

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

7.10. Reservierter Interrupt

INT07

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

7.11. ZEITGEBER

INT08

Funktion:

Die Routine behandelt den Interrupt vom Kanal Null des Zeitgeberschaltkreises.

Ferner werden die Systemzellen für die Uhrzeit und Motorsteuerung der Diskettenlaufwerke verwaltet.

Bemerkung:

Mit dem Interrupt 1Ah besteht die Möglichkeit, die Uhrzeit abzufragen bzw. zu setzen.

Bei jedem Zeitinterrupt wird die zum Interrupt 1Ch gehörige Routine aufgerufen. Damit ist es dem Nutzer möglich, zeitabhängige Routinen einzubinden.

7.12. TASTATUR

INT09

Funktion:

Diese Routine behandelt einen Interrupt, der durch die Tastatursteuerung ausgelöst wird. In der Interruptroutine werden folgende Aufgaben realisiert:

- Übernahme der SCAN-Kodes von der Tastatursteuerung
- Umsetzen der SCAN-Kodes in Zeichenkodes und Abspeichern der beiden Kodes im BIOS-Tastaturpuffer
- Auswertung der Umschalttasten (Shift, CTRL, ALT, ALT1, CAPS LOCK, NUM LOCK, INS und SYS) und Setzen der entsprechenden Zustände in den Systemzellen
- Erkennen von Tastaturüberlauf und Ausgabe eines akustischen Signals
- Bei Erkennen einer Tastaturunterbrechung (Tastenkombination CTRL/PAUSE oder CTRL/SCROLL LOCK) wird ein Interrupt 1Bh ausgelöst.
- Wird die Tastenkombination CTRL/ALT/DEL erkannt, dann wird ein System-Reset veranlasst.
- Bei Erkennen der Taste PRT SC wird ein Interrupt 05h ausgelöst (Ausgabe des Bildschirminhalts auf das List-Gerät).
- Wenn die PAUSE-Taste oder die Tastenkombination CTRL/NUM LOCK erkannt wird, dann wird bis zur Betätigung einer anderen Taste in einer Warteschleife verblieben.
- Bei Erkennen des Niederdrückens und Loslassens der SYS-Taste wird ein Interrupt 15h mit Übergabe einer entsprechenden Information im Register AX ausgelöst.

Bemerkung:

Der Interrupt 16h bietet die Möglichkeit, den Füllungsstand des BIOS-Tastaturpuffers zu testen, die dort gespeicherten Kodes zu lesen und den Zustand der Umschalttasten zu ermitteln. Die im transienten Kommando KEYB enthaltenen Interruptroutinen für die Tastatur lenken den Interruptvektor 09h auf diese Routinen um. Vom System werden damit länderspezifische Tastaturen unterstützt.

7.13. Reservierter Interrupt

INT0A

Funktion:

Dieser Interrupt ist reserviert.

7.14. KOMMUNIKATION

INT0B

Funktion:

Dieser Interrupt ist reserviert für serielle Datendübertragung.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

*** BIOS-Interrupts ***

Die Interruptleitung des Adapters für serielle Kommunikation ist mit dem Interrupt-Controller verbunden.

7.15. KOMMUNIKATION

INT0C

Funktion:

Dieser Interrupt ist reserviert für serielle Datendübertragung.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

Die Interruptleitung des Adapters für serielle Kommunikation ist mit IRQ3 bzw. IRQ4 des Interrupt-Controllers verbunden.

7.16. HARDDISK

INT0D

Funktion:

Diese Routine registriert Interrupts bei der Arbeit mit dem HARDDISK.

Bemerkung:

Die Interrupts werden über IRQ5 des Interrupt-Controllers ausgelöst.

7.17. DISKETTEN

INT0E

Funktion:

Diese Routine registriert Interrupts des FD-Schaltkreises bei Diskettenarbeit.

Bemerkung:

Die Interruptleitung des FD-Schaltkreises ist mit IRQ6 des Interrupt-Controllers verbunden.

7.18. DRUCKER

INT0F

Funktion:

Dieser Interrupt ist reserviert für Druckausgabe über Parallelschnittstelle (CENTRONICS).

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

Das Signal ACKNOWLEDGE der CENTRONICS-Platte ist mit IRQ7 des Interrupt-Controllers verbunden.

7.19. VIDEOSTEUERUNG

INT10

7.19.1. Einföhrung

Die Systemschnittstelle zur Bildschirmarbeit ist der INT 10h. Die Funktionen 0 bis 13h des INT 10h sind im ROM-BIOS enthalten und können stets angesprochen werden. Ausnahmen bilden die Funktionen 10h, 11h und 12h, die nur wirksam sind, wenn der Treiber VE89.SYS installiert ist (s. Punkt 6.10.2.)

Bei Aufruf des INT 10h wird im Register AH die Funktionsnummer übergeben, in anderen Registern können weitere Daten übergeben werden. Der Wert des Registers AX wird beim INT 10h verändert, alle anderen Register, sofern mit ihnen keine Daten zurückgemeldet werden, bleiben unverändert.

7.19.2. Beschreibung der Funktionen

7.19.2.1. BILDSCHIRMMODE SETZEN

INT10: 00h

Funktion:

Diese Funktion initialisiert den Bildschirm auf den ausgewählten Mode und löscht den Bildwiederholpeicher. Die Angabe mit Blauanteil bei den Modi 0, 2 und 5 betrifft Bit 2 des Modussteuerregisters (s. Punkt 7.19.5.3.).

Aufruf:

Register	Inhalt/Bedeutung
AH	00h
AL	Einzustellender Mode

Im Register AL muss der einzustellende Mode wie folgt übergeben werden:

- AL = 0 Alphamode 40 x 25 mit Blauanteil
- = 1 Alphamode 40 x 25 16-farbig
- = 2 Alphamode 80 x 25 mit Blauanteil
- = 3 Alphamode 80 x 25 16-farbig
- = 4 Grafikmode 320 x 200 4-farbig
- = 5 Grafikmode 320 x 200 mit Blauanteil
- = 6 Grafikmode 640 x 200 2-farbig
- = 7 Alphamode 80 x 25 bei Monochromadapter (intern)
- = 8 Grafikmode 640 x 480 16-farbig (nur bei installiertem VE89.SYS)
- = 9 Grafikmode 640 x 400 2-farbig (nur bei installiertem VE89.SYS)

Die Alphamodi 0 und 2 (mit Blauanteil) werden nur bei Anschluss von Composite-Monitoren unterstützt. Für den EC 1834 ist der Anschluss von Composite-Monitoren allerdings nicht vorgesehen.

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.2. KURSORART SETZEN

INT10: 01h

Funktion:

Diese Funktion bestimmt die Art der Kursordarstellung in den Alphamodi. In den Grafikmodi ist der Cursor immer unsichtbar.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
CX	Kursorart

Im Registerpaar CX ist das Kursorformat in folgender Art und Weise zu übergeben:

CX = Kursorart

CH = Anfangszeile

(Bits 4-0, gezählt von oben ab Zeile 0;
ist Anfangszeile > max. mögliche Zeile,
so ist der Cursor unsichtbar;
Bit 5=1 und Bit 6=0: sichtbarer Cursor)

CL = Endzeile

(Bits 4-0, gezählt von oben ab Zeile 0;
ist Endzeile > max. mögliche Zeile, so
fällt der Cursor das ganze Zeichenfeld)

max. mögliche Zeile = 0Dh bei Monochromadapter

= Pixelzeilen pro Zeichen
minus eins bei Farbadapter
im direkten Mode

= Pixelzeilen pro Zeichen
durch zwei minus eins bei
Farbadapter im kompatiblen
Mode

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.3. KURSORPOSITION SETZEN

INT10: 02h

Funktion:

Diese Funktion setzt den Cursor an die übergebene Position der spezifizierten Seite.

Aufruf:

Register	Inhalt/Bedeutung
AH	02h
DH	Zeile des Cursors
DL	Spalte des Cursors
BH	Seite, auf der der Cursor gesetzt werden soll

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.4. LESEN DER KURSORPOSITION

INT10: 03h

Funktion:

Diese Funktion liest die Cursorposition auf der spezifizierten Seite und liefert die aktuelle Art der Kursordarstellung.

Aufruf:

Register	Inhalt/Bedeutung
AH	03h
BH	Seite

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
DH	Zeile der Cursorposition
DL	Spalte der Cursorposition
CX	Aktuelle Cursorart

Bemerkungen:

Die Darstellung der aktuellen Cursorart im Register CX ist unter Punkt 7.19.2.2. (Cursorart setzen) nachzulesen.

7.19.2.5. LICHTSTIFTPOSITION ERMITTELN

INT10: 04h

Funktion:

Diese Funktion gibt fest AH=0 zurück, da kein Lichtstiftanschluss vorhanden ist.

Aufruf:

Register	Inhalt/Bedeutung
AH	04h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AH	0

7.19.2.6. AKTUELLE SEITE SETZEN

INT10: 05h

Funktion:

Diese Funktion setzt eine neue aktive Seite und ist nur bei den Alphamodi sinnvoll.

Aufruf:

Register	Inhalt/Bedeutung
AH	05h
AL	Neue aktive Seite

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.7. FENSTER NACH OBEN ROLLEN

INT10: 06h

Funktion:

Diese Funktion rollt ein Rechteck auf der aktiven Seite nach oben.

Aufruf:

Register	Inhalt/Bedeutung
AH	06h
AL	Anzahl Zeilen
CH	Oberste Zeile des Rechtecks
CL	Linke Spalte des Rechtecks
DH	Unterste Zeile des Rechtecks
DL	Rechte Spalte des Rechtecks
BH	Attribut für zu löschende Zeilen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

Bemerkungen:

Im Register wird die Anzahl der Zeilen eingetragen, um die gerollt werden soll. Bei AL=0 wird das ganze Rechteck gelöscht. Die Register CX und DX beschreiben das Rechteck, in dem die Funktion ausgeführt werden soll.

Im Register BH wird das Attribut für zu löschende Zeilen übergeben. In den Grafikmodi enthält BH direkt das einzuschreibende Byte (im Mode 640 x 480 mit dem Attribut 0Fh).

7.19.2.8. FENSTER NACH UNTEN ROLLEN

INT10: 07h

Funktion:

Diese Funktion rollt ein Rechteck auf der aktiven Seite nach unten.

Aufruf:

Register	Inhalt/Bedeutung
AH	07h
AL	Anzahl Zeilen
CH	Oberste Zeile des Rechtecks
CL	Linke Spalte des Rechtecks
DH	Unterste Zeile des Rechtecks
DL	Rechte Spalte des Rechtecks
BH	Attribut für zu löschende Zeilen

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

Bemerkungen:

Für die Bereitstellung der Registerinhalte gilt das unter Punkt 7.19.2.7. Angegebene.

7.19.2.9. ZEICHEN UND ATTRIBUT LESEN

INT10: 08h

Funktion:

Diese Funktion liest das auf der Cursorposition der angegebenen Seite befindliche Zeichen und das zugehörige Attribut. In den Grafikmodi wird das Pixelmuster der Zeichenposition mit dem Font verglichen. Bei Übereinstimmung wird der Zeichenkode zurückgegeben, ansonsten wird 0 zurückgemeldet.

Aufruf:

Register	Inhalt/Bedeutung
AH	08h
BH	Seite

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Gelesenes Zeichen
AH	Gelesenes Attribut (nur in Alphamodi und im Grafikmode 640 x 480)

7.19.2.10. ZEICHEN UND ATTRIBUT SCHREIBEN

INT10: 09h

Funktion:

Diese Funktion schreibt an die Cursorposition der angegebenen Seite ein Zeichen mit zugehörigem Attribut. Der Wiederholungsfaktor lässt ein mehrfaches Schreiben desselben Zeichens zu, ohne die Cursorposition zu verändern. Bei den Grafikmodi ist die Wiederholung nur innerhalb derselben Zeile möglich.

Aufruf:

Register	Inhalt/Bedeutung
AH	09h
BH	Seite
AL	Auszugebendes Zeichen
BL	Auszugebendes Attribut
CX	Wiederholungsfaktor

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

Bemerkungen:

Das auszugebende Attribut ist im Register BL zu übergeben. Dieses Register enthält in den Grafikmodi 320x200 und 640x480 die Zeichenfarbe. In allen Grafikmodi wird bei Bit 7=1 im Attributwert der bisherige Wert mit dem auszugebenden Zeichen durch exklusives Oder verknüpft.

7.19.2.11. ZEICHEN SCHREIBEN

INT10: 0Ah

Funktion:

Diese Funktion schreibt an die Cursorposition der spezifizierten Seite ein Zeichen, ohne das zugehörige Attribut zu verändern. Der Wiederholungsfaktor lässt ein mehrfaches Schreiben desselben Zeichens zu, ohne die Cursorposition zu verändern. In den Grafikmodi ist diese Funktion mit der vorigen identisch.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Ah
BH	Seite
AL	Auszugebendes Zeichen
BL	Zeichenfarbe

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

Bemerkungen:

Es gelten die Bemerkungen wie unter Punkt 7.19.2.10.

*** BIOS-Interrupts ***

7.19.2.12. FARBE AUSWAHLEN

INT10: 0Bh

Funktion:

Diese Funktion vermittelt die Fortausgabe auf das Farbauswahlregister. Es können die Hintergrundfarbe und die Farbpalette im Grafikmode 320x200 sowie die Vordergrundfarbe in den Grafikmodi 640x200 und 640x400 gesetzt werden. Für die Alphamodi und den Grafikmode 640x480 hat diese Funktion keine Wirkung.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Bh
BX	Farbauswahl

Das Programm muss entsprechend folgender Übersicht die Register BH und BL setzen:

- BH = 0: Hintergrund- bzw. Vordergrundfarbe setzen und Alternativ-Paletten-Auswahl
- BL = Bits 0 bis 4: Bits 0 bis 4 des Farbauswahlregisters
 - Bits 0 bis 3: Farbenkode (0 bis 15)
 - Bit 4: = 0: normaler Farbensatz
 - = 1: alternativer Farbensatz
- BH = 1: Palettenauswahl
- BL = Bit 0: Bit 5 des Farbauswahlregisters
- BL = 0: Palette mit den Farben 2, 4 und 6 (normal) bzw. 10, 12 und 14 (alternativ)
- BL = 1: Palette mit den Farben 3, 5 und 7 (normal) bzw. 11, 13 und 15 (alternativ)

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.13. PUNKT ZEICHNEN

INT10: 0Ch

Funktion:

Diese Funktion zeichnet einen Punkt an der spezifizierten Position in der angegebenen Farbe.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Ch
DX	Punktzeile
CX	Punktspalte
AL	Farbe des Punktes

*** BIOS-Interrupts ***

Der Wert für die Punktzeile liegt zwischen 0 und 479 und der Wert für die Punktspalte zwischen 0 und 639. Für die Angabe der Farbe im Register AL werden 1, 2 oder 4 Bits benötigt. Ist Bit 7 vom Register AL=1, dann wird eine Exklusiv-Oder-Verknüpfung der bisherigen Punktfarbe mit der in AL angegebenen ausgeführt.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.14. PUNKT LESEN

INT10: 0Dh

Funktion:

Diese Funktion ermittelt die Farbe des Punktes an der spezifizierten Position.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Dh
DX	Punktzeile
CX	Punktspalte

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Farbe des Punktes

7.19.2.15. ZEICHENAUSGABE

INT10: 0Eh

Funktion:

Diese Funktion gibt Alphazeichen auf den Bildschirm aus. Die Codes 07h (Piepton), 08h (eine Cursorposition nach links), 0Ah (Zeilenvorschub) und 0Dh (Wagenrücklauf) werden als Steuerzeichen behandelt. Die Cursorposition wird weitergezählt. Nach Zeilenende erfolgt ein Übergang auf den Beginn der nächsten Zeile. Nach dem Ende der letzten Zeile wird der Bildschirminhalt um eine Zeile nach oben gerollt. Das Attribut der neuen letzten Zeile ergibt sich in den Alphamodi aus dem des ersten Zeichens der alten letzten Zeile; in den Grafikmodi ist es immer 0.

*** BIOS-Interrupts ***

Aufruf:

Register	Inhalt/Bedeutung
AH	0Eh
AL	Auszugebendes Zeichen
BL	Zeichenfarbe

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

Bemerkungen:

Der Inhalt des Registers BL ist der Beschreibung der Funktion 10 zu entnehmen (s. Punkt 7.19.2.11.). Der Registerinhalt wird nur in den Grafikmodi berücksichtigt.

7.19.2.16. BILDSCHIRMSTATUS ERMITTELN

INT10: 0Fh

Funktion:

Die Funktion meldet Zeilenkapazität, aktuellen Mode und aktive Seite zurück.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Fh

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AH	Zeichenzahl pro Zeile
AL	Aktueller Bildschirmmode
BH	Aktuelle Seite

7.19.2.17. FARBPALETTE PROGRAMMIEREN

INT10: 10h

Funktion:

Mit dieser Funktion kann jeder Farbnummer ein Farbwert aus 64 möglichen zugeordnet werden. Diese 64 Farben können mittels des Dienstprogrammes PCOLOR aus 4096 ausgewählt werden (vergleiche Anleitung für den Bediener).

*** BIOS-Interrupts ***

7.19.2.17.1. Eine Farbe der Palette ändern INT10: 10h 00h

Aufruf:

Register	Inhalt/Bedeutung
AH	10h
AL	00h
BL	Farbnummer (0 bis 15)
BH	Farbwert (0 bis 63)

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.17.2. Palette ändern INT10: 10h 02h

Aufruf:

Register	Inhalt/Bedeutung
AH	10h
AL	02h
ES:DX	Adresse eines Puffers mit 17 Farbwerten (die ersten 16 werden genutzt für die Farben)

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.17.3. Blinken erlauben/verbieten INT10: 10h 03h

Aufruf:

Register	Inhalt/Bedeutung
AH	10h
AL	03h
BL	00 Blinken verboten
BL	01 Blinken erlaubt

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18. ZEICHENGENERATOR PROGRAMMIEREN

INT10h: 11h

Funktion:

Mit dieser Funktion kann der Zeichengenerator verändert werden. Es ist in jedem Fall zu überprüfen, ob der Zeichengenerator mit den Systemeinstellungen verträglich ist.

7.19.2.18.1. Nutzerzeichensatz laden ohne Veränderung der Systemeinstellungen

INT10: 11h 00h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	00h
ES:BP	Adresse der Zeichenmuster
CX	Auswahl zu programmierender Zeichen
DX	Kode des ersten zu programmierenden Zeichens
BH	Anzahl Byte (Pixelzeilen) pro Zeichen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.2. Standardzeichensatz 14x8 aktiv setzen ohne Veränderung der Systemeinstellungen

INT10: 11h 01h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	01h

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.3. Standardzeichensatz 8x8 aktiv setzen
ohne Veränderung der
Systemeinstellungen

INT10: 11h 02h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	02h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.4. Standardzeichensatz 16x8 aktiv setzen
ohne Veränderung der
Systemeinstellungen

INT10: 11h 04h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	04h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

*** BIOS-interrupts ***

7.19.2.18.5. Nutzerzeichensatz laden mit Veränderung der Systemeinstellungen INT10: 11h 10h

Übergabeparameter wie bei Funktion AL = 0

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	10h
ES:BP	Adresse der Zeichenmuster
CX	Auswahl zu programmierender Zeichen
DX	Kode des ersten zu programmierenden Zeichens
BH	Anzahl Byte (Pixelzeilen) pro Zeichen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.6. Standardzeichensatz 14x8 aktiv setzen mit Veränderung der Systemeinstellungen INT10: 11h 11h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	11h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

*** BIOS-Interrupts ***

7.19.2.18.7. Standardzeichensatz 8x8 aktiv setzen
mit Veränderung der
Systemeinstellungen

INT10: 11h 12h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	12h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.8. Standardzeichensatz 16x8 aktiv setzen
mit Veränderung der
Systemeinstellungen

INT10: 11h 14h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	14h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.9. INT 1Fh ändern

INT10: 11h 20h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	20h
ES:BP	neue INT 1Fh-Adresse

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.10. Zeichengenerator für Grafikmodi ändern

INT10: 11h 21h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	21h
ES:BP	Adresse des Zeichengenerators
CX	Byte (Pixelzeile) pro Zeichen
BL	0 Zeilenzahl wird in DL übergeben
DL	Zeilenzahl
BL	01 14 Zeilen
BL	02 25 Zeilen
BL	03 43 Zeilen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.11. Standardzeichensatz 14x8 für Grafikmodi setzen (nur Grafikmode 8 und 9 nutzen)

INT10: 11h 22h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	22h
BL	wie in Funktion AL=21h
DL	wie in Funktion AL=21h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

*** BIOS-Interrupts ***

7.19.2.18.12. Standardzeichensatz 8x8 für
Grafikmodi aktiv setzen

INT10: 11h 23h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	23h
BL	wie in Funktion AL=21h
DL	wie in Funktion AL=21h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.13. Standardzeichensatz 16x8 für
Grafikmodi aktiv setzen
(nur Grafikmode 8 und 9 nutzen)

INT10: 11h 24h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	24h
BL	wie in Funktion AL=21h
DL	wie in Funktion AL=21h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.2.18.14. Status ermitteln

INT10: 11h 30h

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
AL	30h
BH	Auswahl der Zeichengeneratoradresse

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CX	Byte (Pixelzeilen) pro Zeichen
DL	Zeilenzahl
ES:BP	Adresse von
	- bei Eingangswert
	BH=0: Adresse des INT 1Fh (2. Teil Grafik)
	BH=1: Adresse des INT 43h (Grafik)
	BH=2: Adresse des Standardzeichengenerators 14 x 8
	BH=3: Adresse des Standardzeichengenerators 8 x 8
	BH=4: Adresse des 2. Teils (Kodes ab 128) des Standardzeichengenerators 8 x 8
	BH=6: Adresse des Standardzeichengenerators 16 x 8

7.19.2.19. ADAPTERSTATUS ERMITTELN

INT10: 12h

Aufruf:

Register	Inhalt/Bedeutung
AH	12h
BL	10h Adaptereinstellung abfragen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
BH	0 (Adapter unterstützt keine alpha- numerischen Bildschirme)
AL	3 (Adapter enthält 256K-RAM-Bildwiederhol- speicher)
BL	3 (Adapter enthält 256K-RAM-Bildwiederhol- speicher)
CX	(0) (Schalterstellung)

7.19.2.20. ZEICHENKETTENAUSGABE

INT10: 13h

Funktion:

Mit dieser Funktion können Zeichenketten mit Attribut ab einer spezifizierten Kursorposition auf der angegebenen Seite ausgegeben werden. Die Steuerzeichen 07h (Piepton), 08h (Rückschritt), 0Ah (Zeilenvorschub) und 0Dh (Wagenrücklauf) werden entsprechend behandelt. In den Grafikmodi wird das Attribut wie in Funktion 9 (AH=9) verwendet.

Aufruf:

Register	Inhalt/Bedeutung
AH	0h
ES:BP	Adresse der Zeichenkette
CX	Zeichenzahl der Zeichenkette
DX	Kursorposition für die Zeichenkette
BH	Seite
AL	Modifikationswert
BL	Attribut

Es besteht folgender Zusammenhang zwischen dem Modifikationswert im Register AL und dem Attribut im Register BL:

- AL = 0: BL = Attribut
Kursorposition wird nicht verändert
- 1: BL = Attribut
Kursorposition wird verändert
- 2: Zeichenkette besteht aus Zeichen (Low-Byte) und Attribut (High-Byte)
Kursorposition wird nicht verändert
- 3: Zeichenkette besteht aus Zeichen (Low-Byte) und Attribut (High-Byte)
Kursorposition wird verändert

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

7.19.3. Bildschirmmodi

Bei Anschluss eines Farbmonitors an den Farbgrafikadapter sind gleichzeitig maximal 16 Farben, die aus 4096 Farben ausgewählt werden können, nutzbar. Die Standardzuordnung der Farben, die nach jedem Reset gegeben ist, ist aus der nachfolgenden Tabelle ersichtlich.

Bei Anschluss eines Monochrommonitors an den Farbgrafikadapter sind gleichzeitig maximal 16 Helligkeitsabstufungen nutzbar. Mit dem Dienstprogramm PCOLOR (Programmieren des Grünstrahles) sind Helligkeitsstufen und Farbkodes beliebig zuordenbar, sofern der Hardwarestand des Farbgrafikadapter das zulässt.

Farbenkode	Standardzuordnung
0	Schwarz (black)
1	Blau (blue)
2	Grün (green)
3	Kobaltblau (cyan)
4	Rot (red)
5	Violett (magenta)
6	Braun (brown)
7	Hellgrau (light grey)
8	Dunkelgrau (dark grey)
9	Hellblau (light blue)
10	Hellgrün (light green)
11	Hellkobaltblau (light cyan)
12	Hellrot (light red)
13	Hellviolett (light magenta)
14	Gelb (yellow)
15	Weiss (white)

Bei der Arbeit mit dem Farbgrafikadapter sind die in nachstehender Tabelle aufgeführten Formate möglich:

waagerecht	senkrecht	Farben	Anzahl nutzbarer Seiten	Bildwiederhol-Speicher
40 Zeichen	25 Zeilen	16	8	16 KByte
80 Zeichen	25 Zeilen	16	4	16 KByte
320 Punkte	200 Punkte	4	1	16 KByte
640 Punkte	200 Punkte	2	1	16 KByte
640 Punkte	400 Punkte	2	1	32 KByte
640 Punkte	480 Punkte	16	1	256 KByte

Vom ROM-BIOS werden nur die alphanumerischen Formate (40x25 und 80x25) sowie die Grafikmodi 320x200 und 640x200 unterstützt. Die Formate 640x400 und 640x480 sind nur bei installiertem VE89.SYS logisch (über INT 10h) nutzbar.

*** BIOS-Interrupts ***

Bei den alphanumerischen Formaten wird eine Zeichenbox von 8x16 Punkten benutzt. Nach Reset wird der Standardzeichensatz geladen. Die folgende Tabelle betrifft die Darstellung von Alphazeichen:

Format	140x25	180x25	320x200	640x200	640x400	640x480
Zeichenbox	8x16	8x16	8 x 8	8 x 8	8x16	8x16
Vordergrundfarbe für jedes Zeichen wählbar aus wieviel Farben	16	16	4	1	1	16
Hintergrundfarbe für jedes Zeichen wählbar aus wieviel Farben	8	8	1	1	1	1
Hintergrundfarbe wählbar aus wieviel Farben	oder 16	oder 16	16	1	1	1

In allen Formaten, mit Ausnahme von 640x480, befindet sich der Bildwiederholpeicher im Adressbereich der CPU ab der physischen Adresse 0B8000h.

Schreiben und Lesen von Daten des Bildwiederholspeichers ohne Störungen auf dem Monitor sind in den Modi 40x25, 320x200 und 640x200 möglich. In den Modi 80x25 und 640x400 werden die Zugriffe prinzipiell dunkelgetastet, wobei die Möglichkeit des Zugriffs in den Displayrücklaufphasen besteht.

Tabelle der physischen Adressen der ersten Zeilen (in den Grafikmodi Punktzeilen) der Seite 0:

Format	40 x 25	80 x 25	320 x 200	640 x 200	640 x 400
1. Zeile	0B8000h	0B8000h	0B8000h	0B8000h	0B8000h
2. Zeile	0B8050h	0B80A0h	0BA000h	0BA000h	0BA000h
3. Zeile	0B80A0h	0B8140h	0B8050h	0B8050h	0BC000h
4. Zeile	0B80F0h	0B81E0h	0BA050h	0BA050h	0BE000h
5. Zeile	0B8140h	0B8280h	0B80A0h	0B80A0h	0B8050h
6. Zeile	0B8190h	0B8320h	0BA0A0h	0BA0A0h	0BA050h
7. Zeile	0B81E0h	0B83C0h	0B80F0h	0B80F0h	0BC050h
8. Zeile	0B8230h	0B8460h	0BA0F0h	0BA0F0h	0BE050h
9. Zeile	0B8280h	0B8500h	0B8140h	0B8140h	0B80A0h

Der physische Adressbereich 0A0000h bis 0A0FFFh der CPU wird durch einen ROM belegt, der den Font des gesamten Zeichensatzes (Kodes 0 bis 256) in einer 8x16-Box enthält.

Z.19.3.1. ... 01bbmode_40_x_25_und_80_x_25

Z.19.3.1.1. Zeichendarstellung

Jedes Zeichen belegt im Bildwiederholtspeicher ein WORD (16 Bit), wobei das niederwertige Byte des WORD den Zeichenkode und das höherwertige Byte das Attribut beinhalten. Die niederwertigen 4 Bit (0 bis 3) des Attributes ergeben den Farbenkode der Vordergrundfarbe; entweder beinhalten die höherwertigen 4 Bit (4 bis 7) oder die Bits 4 bis 6 den Farbenkode der Hintergrundfarbe; im letzten Falle fungiert Bit 7 als Blinkbit.

Standardzuordnung des Attributbytes (Bit 5 des Modussteuerregisters = 1):

```
+-----+-----+-----+-----+
|Blink| Hintergrundfarbe|   Vordergrundfarbe   |
+-----+-----+-----+-----+
```

Alternativzuordnung des Attributbytes (Bit 5 des Modussteuerregisters = 0):

```
+-----+-----+-----+-----+
|   Hintergrundfarbe   |   Vordergrundfarbe   |
+-----+-----+-----+-----+
```

Jedes Zeichen wird standardmässig in einer Box von 8 Punkten waagerecht und 16 Punkten senkrecht dargestellt. Bei installiertem VE87.SYS sind auch andere Boxgrössen möglich, insbesondere sind 8x14 und 8x8 unterstützt. Durch Umprogrammierung des Zeichengenerators ist ein beliebiger Zeichensatz in diesen Boxgrössen einstellbar. Im Mode 40x25 wird durch den GDC jedes Zeichen auf die doppelte Breite gebracht.

Z.19.3.1.2. Organisation im Bildwiederholtspeicher

Die Zuordnung der Zeichen zum Bildwiederholtspeicher ist fortlaufend, d.h., das Zeichen in der linken oberen Bildschirmcke (dem Koordinatenursprung) hat die niedrigste Bildwiederholtspeicheradresse, es folgt das rechts daneben liegende Zeichen usw.; auf das letzte Zeichen einer Zeile folgt das erste Zeichen der nächsten Zeile. Da für ein Bild im Mode 40x25 nur 2000 Byte und im Mode 80x25 nur 4000 Byte Bildwiederholtspeicher genutzt werden, wird die Arbeit mit maximal 8 Seiten (0 bis 7) unterstützt, im Mode 80x25 sind vier Seiten (0 bis 3) verwendbar.

*** BIOS-Interrupts ***

Nachfolgend eine Übersicht über die physischen Anfangsadressen der Seiten im Bildwiederholtspeicher (bei Standardbox und Standardzeilenzahl):

Seite	Format 40 x 25	Format 80 x 25
0	0B8000h	0B8000h
1	0B8800h	0B9000h
2	0B9000h	0BA000h
3	0B9800h	0BB000h
4	0BA000h	nicht nutzbar
5	0BA800h	nicht nutzbar
6	0BB000h	nicht nutzbar
7	0BB800h	nicht nutzbar

7.19.3.2. Grafikmode 320 x 200

7.19.3.2.1. Punktdarstellung

Jeder Punkt wird durch zwei Bits beschrieben. Diese bestimmen die Farbe des Punktes. Dabei ist 00B die Hintergrundfarbe, die aus den eingestellten 16 Farben beliebig wählbar ist. Die anderen drei möglichen Farben gehören zu einem Farbensatz. Es ist einer aus 4 Farbensätzen wählbar. Die Zuordnung der Farben zu dem entsprechenden Zwei-Bit-Kode ist der nachfolgenden Tabelle zu entnehmen. Den Farbensatz bestimmt Bit 5 des Farbauswahlregisters und den Alternativsatz Bit 4.

Farbensatz	alternativer		Zwei-Bit-Kode			
	Farbensatz		00B	01B	10B	11B
0	0		Hintergrund	2	4	6
0	1		Hintergrund	10	12	14
1	0		Hintergrund	3	5	7
1	1		Hintergrund	11	13	15

7.19.3.2.2. Organisation des Bildwiederholtspeichers

Mit einem Byte des Bildwiederholtspeichers werden vier nebeneinanderliegende Punkte beschrieben, wobei der linke Punkt dieser vier von den beiden höchstwertigen und der rechte Punkt von den beiden niederwertigen Bits bestimmt wird.

Für die Darstellung einer Zeile mit 320 Punkten werden 80 aufeinanderfolgende Bytes benötigt. Die Adressierung der Punktzeilen ist der Tabelle unter Punkt 7.19.3. zu entnehmen.

7.19.3.2.3. Zeichendarstellung

Die Darstellung von Alphazeichen erfolgt in einer Box von 8x8 Punkten. Das Muster der Zeichen mit den Codes 0 bis 127 ist im ROM-BIOS enthalten und kann mit Hilfe des INT 10h geändert werden. Das Muster der Zeichen mit den Codes 128 bis 255 wird durch den INT 1Fh bestimmt und führt ohne einen solchen Zei-

chensatz (nach dem Einschalten ist keiner vorhanden) zu einem Leerzeichen für alle diese Codes. Das externe Kommando GRAFTABL.COM lädt den Standardzeichensatz für diese Codes. Für jedes Zeichen ist die Vordergrundfarbe wählbar, die Hintergrundfarbe ist für den gesamten Bildschirm gleich.

7.19.3.3. Grafikmode 640_x_200

7.19.3.3.1. Punktdarstellung

Jeder Punkt wird mit einem Bit beschrieben. Es bestimmt die Farbe des Punktes, wobei 0 die Farbe mit dem Kode 0 und 1 eine aus 16 Farben beliebig wählbare Vordergrundfarbe bedeutet. Ein Byte des Bildwiederholerspeichers beschreibt acht nebeneinanderliegende Punkte, das höchstwertige Bit den linken und das niederwertigste Bit den rechten Randpunkt. Für die Darstellung einer Zeile mit 640 Punkten sind 80 aufeinanderfolgende Bytes des Bildwiederholerspeichers nötig. Die Adressierung der Punktzeilen ist der Tabelle unter Punkt 7.19.3. zu entnehmen.

7.19.3.3.2. Zeichendarstellung

Die Darstellung von Alphazeichen erfolgt mit dem gleichen Zeichensatz wie im Grafikmode 320x200. Die Vordergrundfarbe der Zeichen ist für den ganzen Bildschirm gleich, die Hintergrundfarbe ist die Farbe 0.

7.19.3.4. Grafikmode 640_x_400

7.19.3.4.1. Punktdarstellung

Die Zuordnung der Farben und Punkte zum Bildwiederholerspeicher ist wie beim Grafikmode 640x200. Nur die Adressierung der Punktzeilen unterscheidet sich. Sie ist der Tabelle unter Punkt 7.19.3. zu entnehmen.

7.19.3.4.2. Zeichendarstellung

Die Darstellung von Alphazeichen erfolgt in diesem Mode standardmässig durch Muster von 8 Punkten waagerecht und 16 Punkten senkrecht. Andere Boxgrössen sind auch möglich. Ein solcher Zeichensatz ist der Standardzeichensatz ab 0A000h:0 und gilt für alle 256 Zeichen. Die Videoerweiterung VE89.SYS, mit der dieser Mode nur betreibbar ist, lässt auch andere Zeichensätze zu. Die Vordergrundfarbe ist für den gesamten Bildschirm gleich; die Hintergrundfarbe ist die Farbe 0.

7.19.3.5. Grafikmode_640_x_480

7.19.3.5.1. Punktdarstellung

Der Bildwiederholpeicher von 256 KByte ist in vier Farbebenen zu je 64 KByte aufgeteilt. Ein Byte einer Farbebene beschreibt dabei 8 nebeneinanderliegende Punkte, das niederwertigste Bit den linken und das höchstwertige Bit den rechten Randpunkt. Die Bytes der verschiedenen Farbebenen mit der gleichen Relativadresse zum Farbebenenanfang beschreiben die gleichen Punkte. Damit sind jedem Punkt vier Bits zugeordnet, d.h. in jeder Farbebene 1 Bit. Diese vier Bits bestimmen die Farbe des Punktes. Den Bitwerten in den Farbebenen 0 bis 3 entsprechen die Werte der Bits 0 bis 3 der Farbenkodes 0 bis 15 in binärer Darstellung.

7.19.3.5.2. GDC-Funktionen und Zugriff auf Bildwiederholpeicher

In diesem Mode unterstützt der GDC das Zeichnen von Linien und Rechtecken mit wählbarem Liniennmuster von 16 Bit und wählbarer Darstellungsart sowie das Zeichnen von Grafikzeichen und Füllen von Rechtecken mit Mustern aus 8x8 Bit. Der Zugriff auf den Bildwiederholpeicher ist nur über den GDC möglich (mit Portein- bzw. -ausgabe oder mit DMA).

7.19.3.5.3. Zeichendarstellung

Die Darstellung von Alphazeichen erfolgt mit dem gleichen Zeichensatz wie im Grafikmode 640x400. Dabei entstehen im Unterschied zu den anderen Modi hier im Standardfall 30 Zeilen (25 Zeilen in den anderen Modi) auf dem Bildschirm. Die Vordergrundfarbe ist für jedes Zeichen aus 16 Farben wählbar, die Hintergrundfarbe ist die Farbe 0.

*** BIOS-Interrupts ***

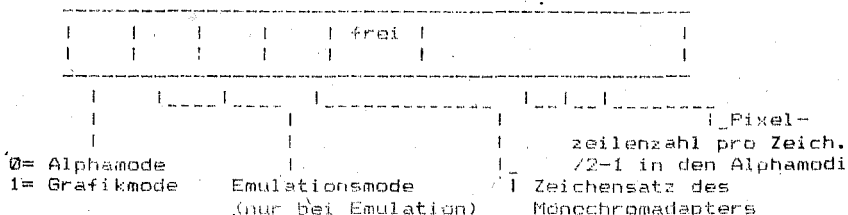
Z.19.4. Reservierte Bytes im Kommunikationsbereich

Physische Adresse	Bytezahl	Name	/ Bedeutung
00449	1	@ CRT_MODE	aktueller Bildschirmcode
0044A	2	@ CRT_COLS	Spaltenanzahl
0044C	2	@ CRT_LEN	Größe einer Seite in Bytes
0044E	2	@ CRT_START	Anfangsadresse der Seite
00450	16	@ CURSOR_POSN	Kursorposition für die einzelnen(8)Seiten
00460	2	@ CURSOR_MODE	Kursorart
00462	1	@ ACTIVE_PAGE	aktuelle Seitennummer
00463	2	@ ADDR_6845	Basisportadresse
00465	1	@ CRT_MODE_SET	Ausgabewert für Modussteuerregister
00466	1	@ CRT_PALETTE	Ausgabewert für Farbauswahlregister
00484	1	@ ROWS	maximale Zeilennummer
00485	2	@ POINTS	Anzahl Pixelzeilen pro Zeichen
00487	1	@ INFO	Status
00488	3	@ INFO_3	Schalterstellung u.a.
004A8	4	@ SAVE_PTR	Zeiger auf eine Tabelle
004E5	1	@ DATM_C	Emulationsmerkmzelle
004E6	1	@ INDEX_C	Emulationsmerkmzelle (Indexregister)
004E7	1	@ INDEX_M	Emulationsmerkmzelle (Indexregister)
004E8	2	@ CURSOR_MODE_C	Emulationsmerkmzelle (Kursorart)
004EA	2	@ CRT_START_C	Emulationsmerkmzelle (Startadresse)
004EC	2	@ CURSOR_MODE_M	Emulationsmerkmzelle (Kursorart)

@ SAVE_PTR zeigt auf eine Tabelle folgenden Aufbaus

DD	?	;reserviert
DD	?	;Zeiger auf Farbpalette
DD	?	;Zeiger auf Alphazeichensatz
DD	?	;Zeiger auf Grafikzeichensatz
DD	3DUP (0)	;reserviert

Aufbau von @ DATM_C



7.19.5. Steuerung des GDC des Farbgrafikadapters

7.19.5.1. Portadressen

Portadr. Bedeutung

3D0h	Parameter (OUT) / Status (IN, s. Punkt 7.19.4.2.) des GDC
3D2h	Kommando (OUT) / Fifo (IN) des GDC
3D4h	löst NMI aus (6845-Indexregister)
3D5h	löst NMI aus (6845-Datenregister)
3D8h	Modussteuerregister (Bitbelegung s. Punkt 7.19.4.3.)
3D9h	Farbauswahlregister (Bitbelegung s. Punkt 7.19.4.4.)
3DAh	Statusregister (Bitbelegung s. Punkt 7.19.4.5.)
3DBh	Intensität des ROT-Strahles der zu programmierenden Farbe (nur Bit 0 bis 3 benutzt)
3DCh	Musterregister zur Zeichengeneratorinitialisierung
3DDh	Intensität des GRÜN-Strahles der zu programmierenden Farbe (nur Bit 0 bis 3 benutzt)
3DEh	Steuerregister (s. Punkt 7.19.4.6.)
3DFh	Intensität des BLAU-Strahles der zu programmierenden Farbe (nur Bit 0 bis 3 benutzt)

7.19.5.2. Bitbelegung des GDC-Status-Ports

Bit Bedeutung für Bit = 1

Bit 0	Daten bereit (zum Lesen, bei CURD, RDAT)
Bit 1	Fifo voll (fehlerhafte Verarbeitung, wenn weitere Kommandos oder Parameter ausgegeben werden)
Bit 2	Fifo leer (es können nun 16 Bytes ausgegeben werden)
Bit 3	Zeichnungsvorgang läuft (bei Ausgabe von Kommandos oder Parametern wird das Zeichnen abgebrochen)
Bit 4	DMA aktiv
Bit 5	Vertikalrücklauf
Bit 6	Horizontaldunkeltastung
Bit 7	(Lichtstift, nicht angeschlossen)

7.19.5.3. Bitbelegung des Modussteuerregisters

In der folgenden Tabelle ist die Bitbelegung des Modussteuerregisters für die verschiedenen Formate angegeben. Das Modussteuerregister ist ein Schreibregister. Ein x in der Tabelle bedeutet, dass beide Bitwerte bei dem Format zulässig sind, wobei die Wirkung der Bitbedeutung entspricht. Ein n in der Tabelle bedeutet, dass beide Bitwerte zulässig sind, aber keine Wirkung haben. Das Bit 5 ist nur in den Alphamodi von Bedeutung, es schaltet zwischen 16 Hintergrundfarben und 8 Hintergrundfarben und Blinkbit um. Das Bit 2 schaltet zu jeder Farbe zwingend den Blauanteil. Bit 3=0 führt zur Dunkeltastung des Bildschirms bis Bit 3 wieder 1 ist.

*** BIOS-Interrupts ***

Bit	Bedeutung für Bit = 1	25	25	200	200	400	480
Bit 0	80 x 25	0	1	0	0	0	1
Bit 1	Grafik	0	0	1	1	1	n
Bit 2	mit Blauanteil	x	x	x	x	x	n
Bit 3	Anzeige erlauben	x	x	x	x	x	x
Bit 4	hochauflösende Grafik	0	0	0	1	1	n
Bit 5	Blinken erlauben	x	x	n	n	n	n
Bit 6	ohne Bedeutung	n	n	n	n	n	n
Bit 7	DMA-Verkehr erlauben	n	n	n	n	n	x

7.19.5.4. Bitbelegung des Farbauswahlregisters

Das Farbauswahlregister ist ein Schreibregister.

Bits	Bedeutung
Bits 0 bis 3	bei Alphamodi und 640x480 keine Wirkung, Hintergrundfarbe bei 320x200, Vordergrundfarbe bei 640x200 und 640x400
Bit 4	= 0: normaler Farbensatz bei 320x200 = 1: alternativer Farbensatz bei 320x200
Bit 5	Auwahl des Vordergrundfarbensatzes bei 320x200 = 0: Farben 2,4,6 (normal) bzw. 10,12,14 (alternativ) = 1: Farben 3,5,7 (normal) bzw. 11,13,15 (altern.)
Bit 6	ohne Bedeutung
Bit 7	ohne Bedeutung

7.19.5.5. Bitbelegung des Statusregisters

Das Statusregister ist ein Leserregister. Das Bit 0 wird 5120 ns vor der Horizontaldunkeltastung auf 1 gesetzt. Der Strahl wird 7600 ns dunkelgetastet (Bit 0 bleibt 1). Während der Displayphase bleibt Bit 0 10240 ns auf 0. Mit der Vertikaldunkeltastung wird Bit 3 auf 1 gesetzt, nach ca. 100000 ns wird Bit 3 wieder 0, die Dunkeltastung aber dauert noch ca. 500000 ns an. Dadurch ist es möglich, Daten in der Horizontal- oder Vertikaldunkeltastphase störungsfrei in den Bildwiederholtspeicher zu schreiben (s. Punkt 7.19.3.).

Bit	Bedeutung für Bit = 1
Bit 0	Strahldunkeltastzeit
Bit 1	statisch auf 0
Bit 2	statisch auf 0
Bit 3	Vertikalrücklauf
Bit 4	statisch auf 0
Bit 5	ohne Bedeutung
Bit 6	ohne Bedeutung
Bit 7	= 0 Farbgrafikbildschirm angeschlossen = 1 monochromatischer Grafikbildschirm angeschlossen

7.19.5.6. Bitbelegung des Steuerregisters

Das Steuerregister ist ein Schreibregister. Die Bits 0 bis 3 haben je nach Betriebsart unterschiedliche Bedeutung. Bei der Programmierung der Farbpalette (Bit 7=1 und zwingend im Mode 640x480, d.h. auch Bit 5=1) wird mit ihnen der Kode der zu programmierenden Farbe ausgegeben. Bei der Programmierung des Zeichengenerators für die Alphamodi (Bit 6=1) wird mit ihnen die Liniennummer des zu programmierenden Zeichens ausgegeben. Im Grafikmode 640x480 werden mit diesen vier Bits die Farbebenen, die beim Datentransfer mit dem Bildwiederholpeicher angesprochen werden, ausgewählt (s. Punkt 7.19.4.7.).

Bits 0 .. 3 Farbenkode für die Programmierung der Farbpalette, Linienadresse für den Zeichengenerator bei Zeichengeneratorprogrammierung, Ebenenauswahl bei 640x480 (s.u.)

Bit 4 = 1: 640x400

Bit 5 = 1: 640x480

Bit 6 = 1: Zeichengeneratorprogrammierung

Bit 7 = 1: Farbpalettenprogrammierung

7.19.5.7. Auswahl der Farbebenen im Grafikmode 640 x 480

Es werden zwei Gruppen von Farbebenen unterschieden, die Gruppe 1 mit den Ebenen 0 (niederwertig) und 1 (höherwertig) und die Gruppe 2 mit den Ebenen 2 (niederwertig) und 3 (höherwertig). Die Bits des Steuerregisters wählen die Ebenen wie folgt aus:

Bit 0 = 1: niederwertige Ebene (0 oder 2 entsprechend Bit 3)

Bit 1 = 1: höherwertige Ebene (1 oder 3 entsprechend Bit 3)

Bit 2 = 1: Beschreiben beider Gruppen

Bit 2 = 0: Beschreiben der durch Bit 3 selektierten Gruppe

Bit 3 = 1: Lesen und Modifikation der Gruppe 2

Bit 3 = 0: Lesen und Modifikation der Gruppe 1

Daraus ergibt sich folgende Zuordnung der Belegungen 0 bis 15 der Bits 0 bis 3 des Steuerregisters zu den Farbenkodes 0 bis 15. RW bedeutet, dass bei dieser Belegung gelesen, modifiziert und geschrieben werden kann. W bedeutet, dass bei dieser Belegung geschrieben werden kann. Lesen und Modifizieren betrifft dann nur eine (die mit Bit 3 ausgewählte) Ebene. Durch maximal zweimaligen Datentransfer (mit jeweils verschiedenem Steuerregisterwert) kann auf jede Farbebenenkombination zugegriffen werden.

*** BIOS-Interrupts ***

Steuerregister (Bits 0 bis 3)	Farbenkode	Lese- zugriff	Schreib- zugriff
0	- = -	Gruppe 1	-
1	1 = 0001B	Gruppe 1	Ebene 0
2	2 = 0010B	Gruppe 1	Ebene 1
3	3 = 0011B	Gruppe 1	Ebenen 0 und 1
4	- = -	Gruppe 1	-
5	- = -	Gruppe 1	-
6	- = -	Gruppe 1	-
7	15 = 1111B	Gruppe 1	alle Ebenen
8	- = -	Gruppe 2	-
9	4 = 0100B	Gruppe 2	Ebene 2
10	8 = 1000B	Gruppe 2	Ebene 3
11	12 = 1100B	Gruppe 2	Ebenen 2 und 3
12	- = -	Gruppe 2	-
13	- = -	Gruppe 2	-
14	- = -	Gruppe 2	-
15	15 = 1111B	Gruppe 2	alle Ebenen

7.20. SCHALTERBELEGUNG

INT11

Funktion:

Diese Routine definiert, welche Geräte zur Maschinenkonfiguration gehören.

Aufruf:

Register	Inhalt/Bedeutung
--	--

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AX	Schalterbelegung (Equipment-Flag)

*** BIOS-Interrupts ***

Die Bits der Schalterbelegung haben folgende Bedeutung:

Bit	15,14:	Anzahl angeschlossener Drucker
Bit	13:	nicht belegt
Bit	12:	zusätzliches Ein-/Ausgabegerät
Bit	11,10,9:	Anzahl RS232-Karten
Bit	8:	nicht belegt
Bit	7,6:	Anzahl Diskettenlaufwerke, wenn Bit 0 = 1
		00 - 1 Laufwerk 01 - 2 Laufwerke
		10 - 3 Laufwerke 11 - 4 Laufwerke
Bit	5,4:	Bildschirmanschluss
		00 - nicht belegt
		01 - 40 * 25 Farbgrafik
		10 - 80 * 25 Farbgrafik
		11 - 80 * 25 Monochrom
Bit	3,2:	RAM-Ausstattung
		00 - 256 KByte 01 - 512 KByte
		10 - 576 KByte 11 - 640 KByte
Bit	1:	Koprozessor
Bit	0:	Umladen von Diskette (Bit zeigt an, dass es im System Diskettenlaufwerke gibt)

Andere Register werden nicht beeinflusst.

7.21. SPEICHERGRÖSSE

INT12

Funktion:

Die Routine ermittelt die Speichergrösse des Systems, wie sie durch die Schalterbelegung definiert ist.

Aufruf:

Register	Inhalt/Bedeutung
--	--

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AX	Speichergrösse

Im Register AX wird die verfügbare Speichergrösse in KByte zurückgegeben.

7.22. DISKETTEN UND HARDDISK

INT13

Der Interrupt 13h (INT13) dient dazu, die Arbeit mit Disketten-Laufwerken und dem HARDDISK zu steuern. Die Übergabe der entsprechenden Parameter erfolgt zum Teil über die CPU-Register und zum Teil über eine Tabelle von Werten; bei der Diskette aber erfolgt sie über die DISK-BASE. Die Unterscheidung zwischen HARDDISK und Diskette erfolgt über die Laufwerkadresse in DL. Die Laufwerkadressen 80h und grösser sprechen den HARDDISK an, die Laufwerkadressen 00-7fh die Disketten. Ist kein HARDDISK installiert, führen auch die Laufwerkadressen 80h und grösser zum Diskettenteil und werden dort abgeblockt.

Begriffe:

HARDDISK	Festplatte, HD
HARDDISK-Adapter	Festplattenadapter, HD-Adapter
DISKBASE	Diskettenparameterblock
Diskettenlaufwerk	FDD
Diskette	Floppy-Disk, FD
L/S-Kopf	Lese-/Schreibkopf
Floppy-Disk-Controller	FDC
Harddisk-Controller	HDC

7.22.1. Disketten-Interrupts

Mit dem INT13 können 40-spurige und 80-spurige 5,25"-Disketten in 80-spurigen 5,25"-Laufwerken bearbeitet werden. Nachfolgend sind die möglichen Funktionen aufgeführt. Sie werden über das Register AH angesprochen.

Die Adresse der DISKBASE wird über den INT1E vermittelt, d.h. über die Speicheradresse 0000:0078h.

Die Daten werden aus dem Speicher zur Diskette und von der Diskette zum Speicher mittels eines speziellen Schaltkreises (DMA) übertragen. Die Adresse des zu verwendenden Speicherbereiches wird dem INT13 vom Aufrufenden in den Registern ES:BX übermittelt. Aus der Sicht des DMA besteht der Speicher des EC1834 aus 16 separaten 64K-Bereichen. Ein Überlauf über eine 64K-Grenze kann der DMA nicht bearbeiten; deshalb ist es in jedem Fall notwendig, die verwendeten Speicher so zu wählen, dass sie in einem geschlossenen 64K-Bereich liegen. Der INT13 prüft in jedem Fall, ob eine solche Grenze überschritten wird. Ist das der Fall, erfolgt eine entsprechende Fehlermeldung; der Befehl wird unbearbeitet abgebrochen. In diesem Fall sollte entweder auf einen anderen Speicherbereich ausgewichen oder die Anzahl der gleichzeitig zu bearbeitenden Sektoren verringert werden.

22.1.1. Funktionen

Bei jedem Befehl muss in DL die Laufwerkadresse übermittelt werden. Es sind dabei die Werte 00h-03h (0-3) und 04h-07h (4-7) zugelassen. Dabei werden mit 04h-07h (4-7) die Laufwerke mit den physischen Adressen 00h-03h (0-3) angesprochen.

Der Unterschied liegt darin, dass bei den Adressen 00h-03h (0-3) eine DISK-BASE mit 11 Byte Länge und bei den Adressen 04h-07h (4-7) eine DISK-BASE mit 21 Byte Länge benötigt wird.

Sollte ein Fehler beim Lesen/Schreiben auftreten, ist es empfehlenswert, bis zu dreimal den entsprechenden Befehl in Verbindung mit einem "Rücksetzen FD" zu wiederholen.

7.22.1.1.1. RÜCKSETZEN FD (RESET)

INT13: 00h

Funktion:

Mit dieser Funktion wird der Floppy-Disk-Schaltkreis (FDC) zurückgesetzt und einige der reservierten Bytes im Speicher in ihre Grundstellung gebracht. Nach dem Rücksetzen wird bei jedem Laufwerk bei der ersten Positionierung eine Synchronisation des FDC mit dem Laufwerk vorgenommen (Rekalibrieren). Der Motor des ausgewählten Laufwerks wird eingeschaltet und die entsprechende Nachlaufzeit eingestellt.

RESET löscht nicht die durch die Funktionen 17h und 18h eingestellten Werte.

Aufruf:

Register	Inhalt/Bedeutung
AH	00h

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

(Fehlerkode siehe Statustabelle - Punkt 7.22.1.2.)

Bemerkung:

Diese Funktion sollte immer nach einem Fehler beim Lesen, Schreiben oder bei anderen Funktionen aufgerufen werden.

7.22.1.1.2. LEBEN DES DISKETTENSTATUS

INT13: 01h

Funktion:

Lesen des Status der vorangegangenen Funktion.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00

Die vorangegangene Funktion wurde mit einem Fehler abgeschlossen.

(Fehlercode siehe Statustabelle - Punkt 7.22.1.2.)

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlercode
AL	Fehlercode

Die vorangegangene Funktion wurde mit einem Fehler abgeschlossen.

(Fehlercode siehe Statustabelle - Punkt 7.22.1.2.)

Bemerkung:

Diese Funktion kann beliebig oft wiederholt werden, ohne den Status zu löschen.

7.22.1.1.3. LESEN SEKTOREN

INT13: 02h

Funktion:

Mit dieser Funktion werden ein oder mehrere Sektoren von der Diskette gelesen.

Aufruf:

Register	Inhalt/Bedeutung
AH	02h
DL	00h-03h, 04h-07h Laufwerkadresse
DH	00h-01h Kopfadresse
CH	00h-27h (4Fh) Spuradresse
CL	01h-08h (09h) Sektoradresse
AL	00h-08h (09h) Anzahl der Sektoren
ES:BX	Speicheradresse (ES=Segment, BX=Offset)

Nur die Laufwerkadresse wird überprüft.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00
AL	Anzahl der gelesenen Sektoren

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode
AL	Anzahl der fehlerfrei gelesenen Sektoren

(Fehlerkode siehe Statustabelle - Punkt 7.22.1.2.)

Bemerkung:

Die zu lesenden Sektoren müssen auf einer Spur stehen.

7.22.1.1.4. SCHREIBEN SEKTOREN

INT13: 03h

Funktion:

Mit dieser Funktion werden ein oder mehrere Sektoren auf die Diskette geschrieben.

Aufruf:

Register	Inhalt/Bedeutung
AH	03h
DL	00h-03h, 04h-07h Laufwerkadresse
DH	00h-01h Kopfadresse
CH	00h-27h (4Fh) Spuradresse
CL	01h-08h (09h) Sektoradresse
AL	00h-08h (09h) Anzahl der Sektoren
ES:BX	Speicheradresse (ES=Segment, BX=Offset)

Nur die Laufwerkadresse wird überprüft.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00
AL	Anzahl der geschriebenen Sektoren

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlercode
AL	Anzahl der fehlerfrei geschriebenen Sektoren

(Fehlercode siehe Statustabelle - Punkt 7.22.1.2.)

Bemerkung:

Die zu schreibenden Sektoren müssen auf einer Spur stehen.

7.22.1.1.5. KONTROLL-LESEN

INT13: 04h

Funktion:

Mit dieser Funktion werden ein oder mehrere Sektoren auf der Diskette kontrolliert.

Aufruf:

Register	Inhalt/Bedeutung
AH	04h
DL	00h-03h, 04h-07h Laufwerkadresse
DH	00h-01h Kopfadresse
CH	00h-27h (4Fh) Spuradresse
CL	01h-08h (09h) Sektoradresse
AL	00h-08h (09h) Anzahl der Sektoren

Nur die Laufwerkadresse wird überprüft.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00
AL	Anzahl der kontrollierten Sektoren

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode
AL	Anzahl der fehlerfrei kontrollierten Sektoren

Fehlerkode siehe Statustabelle - Punkt 7.22.1.2.)

Bemerkung:

Die zu kontrollierenden Sektoren müssen auf einer Spur stehen. Die Kontrolle erfolgt so, dass die entsprechenden Sektoren von der Diskette gelesen werden und der Kontrollmechanismus (CRC-Kontrolle) des FDC für die Fehlerauswertung ausgenutzt wird. Dabei wird nur erkannt, ob der entsprechende Sektor fehlerfrei lesbar ist, er wird aber nicht auf seinen Inhalt untersucht. Der Befehl benötigt keinen Pufferspeicher.

7.22.1.1.6. FORMATIEREN EINER SPUR (FORMAT)

INT13: 05h

Funktion:

Mit dieser Funktion wird eine Spur auf der Diskette formatiert, d.h., es werden die notwendigen Strukturen auf der Spur für einen ordnungsgemässen Datenaustausch eingerichtet (Adressfelder, Zwischenräume und Datenfelder).

Aufruf:

Register	Inhalt/Bedeutung
AH	05h
DL	00h-03h, 04h-07h Laufwerkadresse
DH	00h-01h Kopfadresse
CH	00h-27h (4Fh) Spuraadresse
ES:BX	Tabellenadresse (ES=Segment, BX=Offset)

Nur die Laufwerkadresse wird überprüft.
Die Tabellenadresse weist auf eine Tabelle, die den Inhalt der Adressfelder der Sektoren enthält. Pro Sektor werden vier Bytes benötigt.

Die Bytes enthalten der Reihe nach:

- Spuraadresse,
- Kopfadresse,
- Sektoradresse und
- Sektorgrösse.

Die Werte sind alle hexadezimal.

Bei der Sektorgrösse bedeuten:

- 00 = 128 Byte,
- 01 = 256 Byte,
- 02 = 512 Byte und
- 03 = 1024 Byte.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

(Fehlerkode siehe Statustabelle - Punkt 7.22.1.2.)

Bemerkungen:

1. Die Anzahl der Sektoren pro Spur wird in der DISK-BASE (EOT) übermittelt.

2. Mit der Funktion "Formatieren einer Spur" werden Bytes auf die adressierte Spur geschrieben, aber die Ausführung der Operation nicht kontrolliert. Die Kontrolle muss anschliessend mit einer "Kontroll-Lesen"-Funktion oder "Lesen-Sektoren"-Funktion durchgeführt werden.

In der vorgegebenen Schnittstelle (CFU-Register und normale DISK-BASE) gibt es keine Möglichkeit, dem Modul mitzuteilen, ob es sich um 40- oder 80-spurige Disketten handelt. Aus diesem Grund muss für das Formatieren ein Vorbefehl gesendet werden. Es gibt zwei Vorbefehle "SET-DASD-TYPE" und "SET-MEDIA-TYPE". Eine weitere Möglichkeit besteht in der Verwendung der verlängerten DISK-BASE in Verbindung mit den Laufwerkadressen 04h-07h (4-7).

Die im EC1834 verwendeten Laufwerke haben 80 Spuren. Bei Verwendung von Disketten mit 40 Spuren wird nur jede zweite Spur der Diskette bearbeitet. Die übrigen Spuren (Zwischenspuren) werden nicht benötigt. Beim Formatieren werden sie mit gelöscht, da sie von einer vorhergehenden 80-Spuren-Formatierung noch gültige Adressfelder enthalten können und damit zu Fehlverhalten (falsche Positionierung, Lesen und Schreiben falscher Sektoren) führen können. Laufwerke mit 40 Spuren benötigen die gesamte Spurbreite, so dass keine Zwischenspuren entstehen. Die Löschung der Zwischenspuren einer Diskette mit 40 Spuren im Laufwerk mit 80 Spuren erfolgt so, dass die Adressfelder mit je vier Bytes 0FFh formatiert werden.

Wird in der verlängerten DISK-BASE als Schrittweite 2 oder grösser (n) angegeben, so wird die Spuradresse mit diesem Wert multipliziert. Diese Spur wird formatiert, und die nachfolgende Spur bzw. die nachfolgenden n-1 Spuren werden in der oben beschriebenen Art behandelt.

Beispiele:

Beispiel 1: Tabelle zum Formatieren der Rückseite der Spur 25 mit 9 Sektoren in aufsteigender Reihenfolge zu je 512 Byte.

Spuradresse = 19h, Kopfadresse = 01h, Sektoradresse 01h - 09h, Sektorgrösse = 02h

19 01 01 02 19 01 02 02 19 01 03 02 19 01 04 02 19 01 05 02
19 01 06 02 19 01 07 02 19 01 08 02 19 01 09 02

Beispiel 2: Adressfelder der ersten Sektoren der Vorderseite einer Diskette, die zuvor mit 80 Spuren formatiert war und bei der die Spur 2 als 40-spurig formatiert wurde.

SET DASD TYPE AH=17h, AL=02h, DL=00h
FORMAT AH=05h, DL=00h, DH=00h, CH=02h

*** BIOS-Interrupts ***

Adressfeld Sektor 1 vor der Befehlsausführung	<--->	>--->	Adressfeld Sektor 1 nach der Befehlsausführung
00h 00h 01h 02h			00h 00h 01h 02h
01h 00h 01h 02h			01h 00h 01h 02h
02h 00h 01h 02h			02h 00h 01h 02h
03h 00h 01h 02h			03h 00h 01h 02h
04h 00h 01h 02h			02h 00h 01h 02h
05h 00h 01h 02h			FFh FFh FFh FFh
06h 00h 01h 02h			06h 00h 01h 02h
.....		

Wird der Formatierbefehl ohne Vorbefehl und ohne verlängerte DISK-BASE gebracht, wird automatisch eine 80-spurige Diskette formatiert.

Ein Vorbefehl hat so lange Gültigkeit, bis ein neuer gültiger Vorbefehl mit einem anderen Format gebracht wird (falsche Formate bewirken keine Veränderung).

7.22.1.1.7. LESEN LAUFWERK-PARAMETER

INT13: 08h

Funktion

Mit diesem Befehl werden Informationen über das Laufwerk abgerufen. Es geht dabei darum, die Anzahl der vorhandenen Laufwerke und die maximal mögliche Spur-, Kopf- und Sektoradresse des adressierten Laufwerkes zu erfragen.

Aufruf:

Register	Inhalt/Bedeutung
AH	09h

Rückkehr ohne Fehler:

- für die Laufwerkadressen 00h - 03h

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00
ES	Segment) einer 21 Byte langen DISK-BASE
DI	Offset) im ROM-BIOS
BH	00h
BL	01h,03h DISK-BASE Typ
DH	01h Grösste adressierbare Kopfadresse
DL	00h-04h Anzahl der vorhandenen Laufwerke
CH	27h,4fh Grösste adressierbare Spuradresse
CL	09h Grösste adressierbare Sektoradresse

*** BIOS-Interrupts ***

- für die Laufwerkadressen 04h - 07h

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00
ES	Segment) einer 21 Byte langen DISK-BASE
DI	Offset) (Inhalt von 0000:0078)
BH	00h
BL	00h DISK-BASE Typ
DH	01h Grösste adressierbare Kopfadresse
DL	00h-04h Anzahl der vorhandenen Laufwerke
CH	00h-7Fh Grösste adressierbare Spuradresse
CL	00h-3fh Grösste adressierbare Sektoradresse

Für die Laufwerkadressen zwischen 08h und 7Fh werden die Register ES, DI, CH und CL mit 00 zurückgegeben.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	01h
BH	00h
BL	00h

Bemerkung:

Bei Verwendung der Laufwerkadressen 0-3 erhält man in ES:DI die Adresse einer für das Laufwerk gültigen verlängerten DISK-BASE. Bei Verwendung der Laufwerkadressen 4-7 erhält man in ES:DI den Inhalt aus 0000:0078h als gültige Adresse einer verlängerten DISK-BASE.

Nach dem Einschalten der Maschine wird bei diesem Befehl als maximale Spuranzahl 4Fh (79) zurückgegeben. Sonst wird die durch den letzten gültigen Befehl 17h oder 18h eingestellte Spuranzahl (27h (39) oder 4Fh (79)) zurückgegeben.

Bei allen anderen Funktionen werden die Laufwerkadressen grösser 07 mit einer Fehlermeldung abgewiesen. Mit dieser Funktion soll die Konfiguration erfragt werden, und deshalb erfolgt bei Verwendung der Adressen 08-7Fh die Rückmeldung der vorhandenen Laufwerke ohne Setzen des Fehlerbits in CF.

Für Diskettenarbeit sind nur die Laufwerkadressen 0-3 und 4-7 zugelassen. Die Adressen grösser 7Fh sind für den HARD-DISK bestimmt. Nur wenn kein HARD-DISK installiert ist, wird vom FD-Teil des INT13 auf die Laufwerkadressen grösser 7Fh eine Fehlermeldung abgesetzt.

7.22.1.1.8. ERKENNBARKEIT VON DISKETTENWECHSEL
(READ DASD TYPE)

INT13: 15h

Funktion:

Test auf Erkennbarkeit von Diskettenwechsel.

Aufruf:

Register	Inhalt/Bedeutung
AH	15h
DL	00h-03h, 04h-07h Laufwerkadresse

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h Adressiertes Laufwerk existiert nicht
	01h Laufwerk vorhanden, keine Meldung bei Diskettenwechsel

Diese Antwort teilt mit, dass die vorhandene Hardware (Laufwerk plus Ansteuereinheit) nicht in der Lage ist, Diskettenwechsel zu registrieren und zu melden.

7.22.1.1.9. DISKETTENWECHSEL-INFORMATION

INT13: 16h

Funktion:

Informationen über einen durchgeführten Diskettenwechsel.

Aufruf:

Register	Inhalt/Bedeutung
AH	16h
DL	00h-03h, 04h-07h Laufwerkadresse

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	01h Keine COMBO-Karte vorhanden
	Diskettenwechsel wird nicht erkannt

Da die Hardware nicht in der Lage ist, Diskettenwechsel zu erkennen, wird dieser Befehl mit der Statusmeldung "BAD_CMD" (falscher Befehl) beendet.

**7.22.1.1.10. SETZEN FORMAT-PARAMETER
(SET DABD TYPE)**

INT13: 17h

Funktion:

Setzen der Parameter für das Formatieren einer Diskette.

Aufruf:

Register	Inhalt/Bedeutung
AH	17h
DL	00h-03h, 04h-07h Laufwerkadresse
AL	01h, 02h 40 spur. Diskette in 80 spur. LW *)
	03h reserviert
	04h 80 spur. Diskette in 80 spur. LW
	00h, 05h-FFh ohne Bedeutung

*) Die Funktion 01 sollte möglichst nicht verwendet werden.

Rückkehr ohne Fehler:

- für AL = 01h, 02h, 04h

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00

Rückkehr mit Fehler:

- für AL = 00h, 03h, 05h, ... , FFh

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	01h

Bemerkung:

Dieser Befehl ist als Vorbefehl für den FORMAT-Befehl gedacht. Damit wird der gewünschte Diskettentyp voreingestellt. Dabei kann zwischen 40-spurigen und 80-spurigen Disketten unterschieden werden. Das eigentliche Spurformat muss dann im FORMAT-Befehl selber angegeben werden. Der gewählte Diskettentyp hat so lange Gültigkeit, bis ein neuer Vorbefehl (17h oder 18h) mit einem anderen Format gebracht wird (falsche Formate bewirken keine Änderung der Voreinstellung).

7.22.1.1.11. SETZEN DES DISKETTENTYPS
(SET MEDIA TYPE)

INT13: 18h

Funktion:

Setzen des Diskettentyps für das Formatieren einer Diskette..

Aufruf:

Register	Inhalt/Bedeutung
AH	18h
DL	00h-03h, 04h-07h Laufwerkadresse
DH	ohne Bedeutung
CH	27h, 4Fh Maximale Spur
CL	09, 08 Maximaler Sektor *)
AL	Ohne Bedeutung
ES:BX	Ohne Bedeutung

*) Der Wert 09 wird nur für den Befehl SET MEDIA TYPE benötigt. Beim Formatieren kann dann für den maximalen Sektor ein anderer Wert verwendet werden.
Der Wert 08 gilt nur im Zusammenhang mit CH = 27h.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00
ES:DI	Adresse (SEGMENT:OFFSET) einer 21 Byte langen DISK-BASE

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	0Ch Unbekannte Kombination in CH/CL
ES:DI	Unverändert

Bemerkung:

Dieser Befehl ist als Vorbefehl für den FORMAT-Befehl gedacht. Damit wird der gewünschte Diskettentyp voreingestellt. Dabei kann zwischen 40-spurigen und 80-spurigen Disketten unterschieden werden. Das eigentliche Spurformat muss dann im FORMAT-Befehl selber angegeben werden. Der gewählte Diskettentyp hat so lange Gültigkeit, bis ein neuer Vorbefehl (17h oder 18h) mit einem anderen Format gebracht wird (falsche Formate bewirken keine Änderung der Voreinstellung).

*** BIOS-Interrupts ***

7.22.1.2. Statustabelle

Rückgabeparameter in AH bzw. Inhalt von Byte (0040:0041).

80h	TIME_OUT	Abbruch durch Zeitüberschreitung
40h	BAD_SEEK	Positionierfehler
20h	BAD_NEC	Fehler des FDC
10h	BAD_CRC	CRC-Fehler beim Lesen der Daten
0Ch	MED_NOT_FND	Diskettentyp unbekannt
09h	DMA_BOUNDARY	DMA überschreitet 64K-Grenze
08h	BAD_DMA	Zeit-Fehler beim Datentransport
06h	MEDIA_CHANGE	Diskette wurde gewechselt (nicht verwendet)
04h	RECORD_NOT_FND	Sektor nicht gefunden
03h*	WRITE_PROTECT	Schreibversuch auf schreibgeschützte Diskette
02h	BAD_ADDR_MARK	Adressmarke nicht gefunden
01h	BAD_CMD	Falsches Kommando
00h		Fehlerfrei

7.22.1.3. Reservierte Bytes im Datensegment 40h

Adresse	Name	Bedeutung
40:3Eh	SEEK_STATUS	Eine 0 in Bit 0-3 gibt an, dass das entsprechende Laufwerk A-D bei der ersten Positionierung mit dem FDC zu synchronisieren ist.
40:3Fh	MOTOR_STATUS	Eine 1 in Bit 0-3 gibt das Laufwerk A-D an, dessen Motor eingeschaltet ist. Bit 7 = 1 zeigt an, dass die ablaufende Operation ein Schreib- oder FORMAT-Befehl ist.
40:40h	MOTOR_COUNT	Zähler für die Motornachlaufzeit (Bei Erreichen von 0 wird der Motor abgeschaltet. Die Verwaltung des Zählers erfolgt über das TIMER-Interrupt-Programm.)
40:41h	DISKETTE_STATUS	Statusinformation des INT13 (wird in AH übermittelt)
40:42h	NEC_STATUS	Statusinformationen des FDC
- 40:48h		
40:90h	MEDIA_STATE (A)	Disketten- und Laufwerkinformation für die Laufwerke A-D
40:91h	MEDIA_STATE (B)	
40:92h	MEDIA_STATE (C)	
40:93h	MEDIA_STATE (D)	
40:94h	DSK_TRK (A)	Aktuelle Spur der Laufwerke A und B
40:95h	DSK_TRK (B)	
40:96h	Reserviert	

*** BIOS-Interrupts ***

Aufbau der Bytes 40:90h - 40:93h für die Laufwerke A - D:

Bit:	7	6	5	4	3	2	1	0
					reserviert			
					PRESENT STATE:			
					000 - 101: nicht benutzt			
					110: 40-spurige Diskette in 80-spurigem Laufwerk			
					111: 80-spurige Diskette in 80-spurigem Laufwerk			
					---> Diskettentyp etabliert			
					-----> 1 = Doppel-Spur-Betrieb (40-spurige Diskette in 80-spurigem Laufwerk)			
					-----> Daten-Transfer-Rate:			
					00: 500 KBS			
					01: 300 KBS			
					10: 250 KBS			
					11: RESERVED			

Die Bytes 0040:0090 - 0040:0093 (DSK_STATE) dienen dazu, Informationen über die einzelnen Laufwerke zu speichern. Jedem Laufwerk ist ein Byte zugeordnet. Mit SET DASD TYPE und SET MEDIA TYPE werden die Bits 0 und 4 beeinflusst.

Bit 0 zeigt dabei an, ob mit 40 oder mit 80 Spuren formatiert werden soll. Mit jedem gültigen Befehl SET DASD TYPE oder SET MEDIA TYPE wird Bit 4 gesetzt (ein bestimmter Typ etabliert). Ebenso wird das Bit 4 mit jedem FORMAT-Befehl gesetzt, um den VERIFY-Befehl mit den gleichen Voraussetzungen bzgl. der Schrittweite auszuführen.

Das Bit 4 wird zurückgesetzt, wenn

- ein "TIME-OUT" Fehler erkannt wird,
- ein neuer Vorbefehl mit einem falschen Format gebracht wird,
- ein Lese- oder Schreibfehler beim Lesen (2) oder Schreiben (3) auftritt und durch Kontrolle von Adressfeldern die Vermutung naheliegt, dass der Diskettentyp gewechselt wurde,
- zwischenzeitlich mit einem anderen Format mit einer verlängerten DISK-BASE gearbeitet wurde.

Bit 5 wird für jeden physischen Zugriff eingestellt. Bei Fehlererkennung kann dieses Bit während eines Kommandos mehrfach umgestellt werden.

7.22.1.4. Diskettenparameter-Block (DISK-BASE)

Die DISK-BASE dient dazu, Parameter an den INT13 zu übermitteln. Die Adresse dieser Tabelle wird über die Adresse 0000:0078 (INT1E) vermittelt.

7.22.1.4.1. DISK-BASE_Arten

Normale DISK-BASE: Byte 0 - 10
Anwendung bei den meisten INT13-Aufrufen.

Erweiterte DISK-BASE: Byte 0 - 12
Anwendung bei einigen XT- oder AT-Rechnern

Verlängerte DISK-BASE: Byte 0 - 21
Anwendung beim EC1834 speziell für Sonderlaufwerke und Sonderdisketten. Die Verwendung dieser DISK-BASE macht die Befehle 17h und 18h überflüssig.

7.22.1.4.2. DISK-BASE_Aufbau

Byte Inhalt

-
- | | |
|---|--|
| 0 | 1. Spezifikationsbyte für den FDC (SRT / HUT)
SRT Schrittrate (4 bit)
Zeit von Schritt zu Schritt bei Spurpositionierung
1111b, 1110b ... 0000b
in 2ms Schritten von 2ms, 4ms bis 32ms bei 5,25"
HUT L/S-Kopf-Entlade-Zeit (4 Bit)
Zeit, die der Kopf nach Lese- oder Schreiboperation
noch an der Diskette verbleibt
in 32ms Schritten von 32ms - 480ms bei 5,25" |
| 1 | 2. Spezifikationsbyte für den FDC (HLT/mode)
HLT L/S-Kopf-Beruhigung (7 Bit)
Beruhigungszeit bei L/S-Kopfandruck:
0000001b 1111111b
in 4ms Schritten von 4ms - 508ms bei 5,25"
mode
mode = 0 DMA-Betrieb
mode = 1 kein DMA-Betrieb (nicht verwenden) |
| 2 | Laufzeit des Motors nach einer Operation (1Byte)
01h FFh
in 50ms Schritten von 50ms - 12,75ms bei 5,25" |
| 3 | Bytes pro Sektor
00 bei 128 Byte, 01 bei 256 Byte
02 bei 512 Byte, 03 bei 1024 Byte |
| 4 | Letzter Sektor der Spur (EOT) |
| 5 | Länge des Zwischenraumes zwischen Daten- und Adressfeld
für die Befehle Lesen, Schreiben und Verify |
| 6 | DTL
Wenn Byte 3 den Wert 00 hat, bedeutet DTL die Anzahl der zu
lesenden oder schreibenden Bytes; sonst hat DTL den Wert
FFh. |

Byte Inhalt

- 7 Länge des Zwischenraumes zwischen Daten- und Adressfeld für den Befehl FORMAT
- 8 Füll-Byte für den FORMAT-Befehl
Mit diesem Byte werden alle Sektoren der zu formatierenden Spur gefüllt.
- 9 Beruhigungszeit für den L/S-Kopf nach einer Positionierung in 1ms Schritten von 1ms - 255ms
- 10 Zeit, die das Laufwerk für das Hochlaufen des Motors benötigt (1/8 Sekunden)
in 125ms Schritten von 125ms - 31,875s

----- Ende der NORMAL-DISK-BASE -----

- 11 letzte Spur der Diskette
- 12 Bit 7 & 6 geben die Datentransferrate an
(10 für 250KBS = 5,25")

----- ROBOTRON-ERWEITERUNG -----

- 13 Aufzeichnungsformat
 - 00h-02h = reserviert
 - 03h = 5,25" DD
 - 04h-FFh = reserviert
- 14 Schritte pro Spur (01h - 0Fh); 00 ist nicht erlaubt
- 15 Laufwerktyp
 - 02h = reserviert
 - 06h = reserviert
 - 48h = TEAC-Kompatible
- 16 1. Spur mit Prdkompensation
- 17 Prdkompensationszeit
 - 00h = 125ns, 01h = 250ns, 02h = 375ns, 03h = 500ns
- 18 nicht verwendet (00h)
- 19 nicht verwendet (00h)
- 20 nicht verwendet (00h)

7.22.2.-----Harddisk-Interrupts

7.22.2.1.----Funktionen

7.22.2.1.1. RÜCKSETZEN HARDDISK UND FDC

INT13: 00h

Funktion:

Der HD-Controller und auch der FDC werden in den Grundzustand gesetzt.

Die installierten Festplatten werden auf Zylinder 0 positioniert, und es erfolgt das Initialisieren der LW-Parameter.

Aufruf:

Register	Inhalt/Bedeutung
AH	00h
DL	80h,81h Laufwerkadresse.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlercode

7.22.2.1.2. LESEN DES DISKSTATUS

INT13: 01h

Funktion:

Der Status der zuletzt ausgeführten HD-Funktion wird zurückgegeben.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
DL	80h,81h Laufwerkadresse

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h
AL	Fehlerkode

Bemerkung:

Diese Funktion löscht den Status der vorausgegangenen Funktion, d.h., dass nach dem zweiten Aufruf stets 00h in AL übergeben wird.

7.22.2.1.3. LESEN SEKTOREN

INT13: 02h

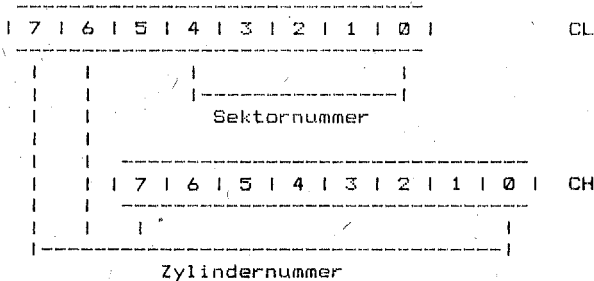
Funktion:

Die adressierten Sektoren werden von der Festplatte gelesen. Eine Datenkontrolle und eventuelle Korrekturversuche werden im Speicher über die ECC-Zeichen vorgenommen.

Aufruf:

Register	Inhalt/Bedeutung
AH	02h
DL	80h, 81h
DH	00h-05h
CH	00h-FFh
CL	01h-11h
	Laufwerkadresse
	Kopfadresse
	Zylinderadresse
	Sektoradresse (Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse)
AL	01h-80h
ES:BX	Speicheradresse

Zuordnung der Bits in CX für Zylinder- und Sektornummer:



*** BIOS-Interrupts ***

Der Sektor wird durch die Bits 4-0 von CL adressiert. Bit 7,6 sind die höchstwertigsten Bits der Zylindernummer.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlercode

7.22.2.1.4. SCHREIBEN SEKTOREN

INT13: 03h

Funktion:

Der Speicherbereich wird auf die adressierten Sektoren der Festplatte geschrieben.

Am Ende eines jeden Sektors werden 4 ECC Zeichen mit aufgezeichnet.

Aufruf:

Register	Inhalt/Bedeutung
AH	03h
DL	80h, 81h
DH	00h-05h
CH	00h-FFh
CL	01h-11h
	Sektoradresse (Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse)
AL	01h-80h
ES:BX	Anzahl der Sektoren Speicheradresse

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

*** BIOS-Interrupts ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

7.22.2.1.5. KONTROLL-LESEN

INT13: 04h

Funktion:

Es wird jeweils 1 Sektor von der Festplatte in den Sektorpuffer eingelesen und geprüft (ECC). Eine Übertragung in den Hauptspeicher erfolgt nicht.

Aufruf:

Register	Inhalt/Bedeutung
AH	04h
DL	80h,81h Laufwerkadresse
DH	00h-05h Kopfadresse
CH	00h-FFh Zylinderadresse
CL	01h-11h Sektoradresse (Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse)
AL	01h-80h Anzahl der Sektoren

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

7.22.2.1.6. FORMATIEREN EINER SPUR

INT13: 05h

Funktion:

Es werden 17 Sektoren pro Spur initialisiert.

Aufruf:

Register	Inhalt/Bedeutung
AH	05h
DL	80h, 81h Laufwerkadresse
DH	00h-05h Kopfadresse
CH	00h-FFh Zylinderadresse
CL	Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

Bemerkung:

Die Formatierung der Spur erfolgt nach einer intern abgespeicherten Tabelle, die von aussen nicht beeinflusst werden kann. Die Sektoren werden mit 0FFh beschrieben. Die Formatierung erfolgt nur für den adressierten Kopf.

7.22.2.1.7. FORMATIEREN EINER SPUR MIT KENNZEICH-
NUNG DER DEFEKTEN SEKTOREN

INT13: 06h

Funktion:

Es werden 17 Sektoren pro Spur initialisiert.

Aufruf:

Register	Inhalt/Bedeutung
AH	06h
DL	80h, 81h Laufwerkadresse
DH	00h-05h Kopfadresse
CH	00h-FFh Zylinderadresse
CL	Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

Bemerkung:

Die Formatierung der Spur erfolgt nach einer intern abgespeicherten Tabelle, die von aussen nicht beeinflusst werden kann. Die Sektoren werden mit 0FFh beschrieben und werden als defekt gekennzeichnet. Die Formatierung erfolgt nur für den adressierten Kopf.

7.22.2.1.8. FORMATIEREN AB ADRESSIERTEM ZYLINDER

INT13: 07h

Funktion:

Der Harddisk wird ab adressiertem Zylinder (Kopf) bis einschliesslich des letzten Zylinders formatiert.

Aufruf:

Register	Inhalt/Bedeutung
AH	07h
DL	80h,81h Laufwerkadresse
DH	00h-05h Kopfadresse
CH	00h-FFh Zylinderadresse
CL	Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

Die Formatierung der Spuren erfolgt nach einer intern abgespeicherten Tabelle, die von aussen nicht beeinflusst werden kann. Die Sektoren werden mit 0FFh beschrieben. Die Formatierung erfolgt über alle Köpfe ab adressierter Spur und adressiertem Kopf bis zum Ende des Harddisk.

7.22.2.1.9. LESEN LAUFWERK-PARAMETER

INT13: 08h

Funktion:

Mit dieser Funktion werden Informationen über den Harddisk abgeholt. Es geht dabei darum, die Anzahl der vorhandenen Laufwerke und die maximal mögliche Zylinder-, Kopf- und Sektoradresse des adressierten Laufwerkes zu erfragen.

*** BIOS-Interrupts ***

Aufruf:

Register	Inhalt/Bedeutung
AH	00h
DL	80h,81h Laufwerkadresse

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h
DH	Max. Kopfnummer
DL	Anzahl der vorhandenen Harddisk
CH	Grösste adressierbare Zylinderadresse
CL	Grösste adressierb. Sektoradresse pro Spur
	(Bit 6 u. 7 sind die 2 höchsten Bit der Zylinderadresse)

Der letzte Zylinder ist für Diagnose reserviert (s. Funktion 13h).

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

7.22.2.1.10. INITIALISIERUNG LAUFWERK-PARAMETER INT13: 09h

Funktion:

Es wird überprüft, ob für die auf dem Harddisk-Adapter eingestellten Schalter und für das adressierte Laufwerk eine Beschreibungstabelle (Harddisk-Parameterblock) vorhanden ist.

Aufruf:

Register	Inhalt/Bedeutung
AH	09h
DL	80h,81h Laufwerkadresse

*** BIOS-Interrupts ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlercode

Es existiert keine Tabelle für das adressierte Laufwerk.

7.22.2.1.11. LESEN LANG

INT13: 0Ah

Funktion:

Es erfolgt die Übertragung der gewählten Sektoren von dem Hard-disk in den Speicher. Dabei werden nach jedem Sektor zusätzlich 4 Bytes eingelesen, die vom Anwender für die eigene Datenfehlererkennung und -korrektur genutzt werden können, wobei keine automatische ECC-Kontrolle und ECC-Korrektur durchgeführt wird.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Ah
DL	80h, 81h Laufwerkadresse
DH	00h-05h Kopfadresse
CH	00h-FFh Zylinderadresse
CL	01h-11h Sektoradresse (Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse)
AL	01h-79h Anzahl der Sektoren
ES:BX	Speicherbereich

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

*** BIOS-Interrupts ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

7.22.2.1.12. SCHREIBEN LANG

INT13: 08h

Funktion:

Es erfolgt die Übertragung der gewählten Sektoren vom Speicher zum Harddisk. Dabei werden nach jedem Sektor zusätzlich 4 Bytes aufgezeichnet, die der Anwender für eine eigene Datenkontrolle nutzen kann. Es werden keine ECC-Zeichen automatisch aufgezeichnet.

Aufruf:

Register	Inhalt/Bedeutung
AH	08h
DL	80h, 81h Laufwerkadresse
DH	00h-05h Kopfadresse
CH	00hh-FFh Zylinderadresse
CL	01h-11h Sektoradresse (Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse)
AL	01h-79h Anzahl der Sektoren
ES:BX	Speicherbereich

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

*** BIOS-Interrupts ***

7.22.2.1.13. POSITIONIEREN

INT13: 0Ch

Funktion:

Der L/S-Kopf des selektierten Harddisk wird auf den adressierten Zylinder positioniert.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Ch
DL	80h,81h Laufwerkadresse
DH	00h-05h Kopfadresse
CH	00h-FFh Zylinderadresse
CL	Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse
ES:BX	Speicherbereich

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlercode

7.22.2.1.14. ALTERNATIVES RÜCKSETZEN

INT13: 0Dh

Funktion:

Mit dieser Funktion wird nur der HD-Controller und nicht der FDC in den Grundzustand gesetzt (siehe auch Funktion 00h).

Die installierten Festplatten werden auf Zylinder 0 positioniert, und es erfolgt das Initialisieren der LW-Parameter.

*** BIOS-Interrupts ***

Aufruf:

Register	Inhalt/Bedeutung
AH	0Eh
DL	80h,81h Laufwerkadresse

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

7.22.2.1.15. LESEN SEKTORPUFFER

INT13: 0Eh

Funktion:

Der Inhalt des Sektorpuffers (512 Byte) wird in den Speicher auf die angegebene Adresse gelesen.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Eh
DL	80h,81h Laufwerkadresse
ES:BX	Speicherbereich

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

*** BIOS-Interrupts ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlercode

7.22.2.1.16. SCHREIBEN SEKTORPUFFER

INT13: 0Fh

Funktion:

Ab der angegebenen Adresse im Speicher wird ein Sektor (512 Byte) in den Sektorpuffer geschrieben.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Fh
DL	80h,81h Laufwerkadresse
ES:BX	Speicherbereich

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlercode

7.22.2.1.17. TEST DER LAUFWERKBEREITSCHAFT

INT13: 10h

Funktion:

Es wird geprüft, ob der Harddisk bereit ist.

Aufruf:

Register	Inhalt/Bedeutung
AH	10h
DL	80h,81h Laufwerkadresse

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Harddisk ist bereit.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlercode

- Harddisk ist nach Zuschalten der Spannung noch nicht betriebsbereit.
- Harddisk ist defekt.

7.22.2.1.18. REKALIBRIEREN

INT13: 11h

Funktion:

Die L/S-Köpfe des gewählten Harddisk werden auf Zylinder 0 positioniert.

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
DL	80h,81h Laufwerkadresse

*** BIOS-Interrupts ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

7.22.2.1.19. RAM-DIAGNOSE

INT13: 12h

Funktion:

Der Speicher auf dem HD-Adapter wird auf Funktionsfähigkeit getestet.

Aufruf:

Register	Inhalt/Bedeutung
AH	12h
DL	80h, 91h Laufwerkadresse

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

Speicher ist defekt.

*** BIOS-Interrupts ***

7.22.2.1.20. LAUFWERK-DIAGNOSE

INT13: 13h

Funktion:

Zum Test auf Funktionsfähigkeit des Harddisk erfolgt eine Diagnose durch Lese- und Schreibversuche auf dem letzten Zylinder.

Aufruf:

Register	Inhalt/Bedeutung
AH	13h
DL	80h,81h Laufwerkadresse

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AH	00h

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AH	Fehlerkode

Harddisk oder Harddisk-Adapter ist defekt.

7.22.2.2. Statustabelle

Rückgabeparameter (Fehlerkode) in AH bzw. Inhalt von Byte (0040:0074).

0FFh	SENSE_FAIL	Kommandoabbruch
0BBh	UNDEF_ERR	Undefinierter Fehler
80h	TIME_OUT	Abbruch durch Zeitüberschreitung
40h	BAD_SEEK	Positionierfehler
20h	BAD_CNTRL	Fehler des HDC
11h	DATA_CORRECTED	ECC-korrigierter Datenfehler.
10h	BAD_ECC	ECC-Fehler (beim Lesen)
0Ah	BAD_SECTOR	Bad-Sector-Kennung gesetzt
09h	DMA-BOUNDARY	DMA überschreitet 64K-Grenze
07h	INIT_FAIL	LW-Parameter nicht aktiviert
05h	BAD_RESET	Reset-Fehler
04h	SECTOR_NOT_FND	Sektor nicht gefunden
02h	BAD_ADDR_MARK	Adressmarke nicht gefunden
01h	BAD_CMD	Falsches Kommando

*** BIOS-Interrupts ***

7.22.2.3. Reservierte Bytes im Datensegment 40h

Adresse	Name	Bedeutung
40:42h	CMD_BLOCK	Kommandoblockeinstellung für Festplatten-Controller
40:42h		Zylinder für Schreibstromreduzierung
40:43h		Sektorenanzahl
40:44h		Sektornummer
40:45h		Zylinder L (niedriger Teil)
40:46h		Zylinder H (höherer Teil)
40:47h		SDH-Register
40:48h		Befehlsregister
40:6Ch	TIMER-LOW	für Initialisierung
40:72h	RESET_FLAG	für Initialisierung
40:74h	DISK_STATUS1	Statusbyte der Funktionen
40:75h	HF_NUM	Merkbyte
	Bit	
	0/1	Anzahl installierte Harddisk
	4 = 1	ECC-korrigierter Datenfehler
	5 = 1	ECC-Fehler
	6 = 1	Schreiben Puffer
	7 = 1	Lesen Puffer
40:76h	CONTROL_BYTE	Steuerbyte für HDC
	Bit	
	0/1/2	L/S-Kopf Auswahl
	3 = 1	HD-LW0
	4 = 1	HD-LW1
	5/6/7	Controller-Steuerung
40:77h	PORT_OFF	Restsektormerkbyte
40:E0h	DMA_MODE	DMA-Mode-Register

7.22.2.4. Festplatten-Architektur

Die logische Festplattenunterteilung erlaubt verschiedenen Betriebssystemen, die Festplatte zu nutzen, ohne dass bei Wechsel des Betriebssystems Daten gerettet bzw. rückgespeichert werden müssen. Ausserdem ist es möglich, das entsprechende Betriebssystem von der Festplatte aus zu starten.

Um die Festplatte für mehrere Betriebssysteme nutzbar zu machen, kann diese logisch in 1-4 primäre Partitions eingeteilt werden. Jedes Betriebssystem kann nur eine primäre Partition besitzen. Die Informationen über die Partition sind in der Partition-Tabelle enthalten. Diese befindet sich im Master-Bootrecord auf dem ersten physischen Sektor der Festplatte.

Jede primäre Partition kann einen Bootrecord auf ihrem ersten Sektor enthalten sowie Programme und Daten, einschliesslich einer Betriebssystemkopie, um das System starten zu können.

DCP 3.30 verwaltet 2 Partitionarten, eine primäre und eine erweiterte.

Die primäre Partition kann bis zu 32 MByte gross sein und muss vorhanden sein, um DCP von der Festplatte aus laden und starten zu können.

Soll eine erweiterte Partition angelegt werden, muss bereits eine primäre Partition existieren. Die erweiterte Partition kann je nach Grösse der Festplatte in mehrere erweiterte logische Einheiten unterteilt werden. Die Grösse der erweiterten logischen Einheiten kann jeweils bis zu 32 MByte betragen. Die Anzahl der erweiterten logischen Einheiten wird nur durch die Anzahl der noch nicht durch das Betriebssystem für logische Einheiten belegten Buchstaben des Alphabets begrenzt. Somit können Festplatten mit hoher Kapazität effektiv genutzt werden. Das Einrichten der primären und der erweiterten Partition und die Unterteilung der erweiterten Partition in erweiterte logische Einheiten erfolgt mit dem DCP-Dienstprogramm FDISK.COM.

Die DCP-Partitions bzw. die erweiterten logischen Einheiten müssen mit dem Programm FORMAT.COM formatiert werden, bevor mit ihnen gearbeitet werden kann.

7.22.2.4.1. Master-Bootrecord

Der erste physische Sektor einer Festplatte muss immer der Master-Bootrecord sein.

Er beinhaltet:

- Laden Bootrecord und Übergeben der Steuerung an eines der vier möglichen Betriebssysteme,
- Partition-Tabelle am Ende des Records.

7.22.2.4.1.1. Funktion

Der Anspung des Master-Bootrecords erfolgt durch den INT19h (Bootstrap-Loader) dann, wenn das Betriebssystem von der Diskette in LWA nicht geladen wurde. Er steuert das Laden und Initialisieren des Systems von der Festplatte.

Über die Partition-Tabelle, die im Master-Bootrecord enthalten ist, kann auf Informationen über die einzelnen Partitions zugegriffen werden.

7.22.2.4.1.2. Partition-Tabelle

Die Partition-Tabelle enthält 4 Tabelleneintragungen mit jeweils 16 Byte (eine Eintragung für jedes der 4 Partitions). Am Ende der Partition-Tabelle ist die Signatur 55AAh zur Kennung eines gültigen Master-Bootrecords eingetragen.

Offset	Inhalt

(bez. Master-Boot-record-Beginn)	
1BEh-1CDh	1. Eintragung
1CEh-1DDh	2. Eintragung
1DEh-1EDh	3. Eintragung
1EEh-1FDh	4. Eintragung
1Fhh-1FFh	Signatur (55AAh)

Aufbau einer Tabelleneintragung

Länge: 16 Byte

Byte	
1	Boot-Indikator 00h Partition beinhaltet kein ladbares Betriebssystem 80h Partition beinhaltet ladbares Betriebssystem
2	Kopfadresse bei Partitionbeginn
3	Sektoradresse bei Partitionbeginn (Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse)
4	Zylinderadresse bei Partitionbeginn (die höchsten 2 Bit befinden sich im Sektoradressenbyte)

*** BIOS-Interrupts ***

- 5 Systemindikator beinhaltet eine Kennung des Betriebssystems einer primären Partition oder verweist auf eine erweiterte Partition.
- Ø0 - unbekannte Partition
 - Ø1 - primäre DCF-Partition mit 12-Bit-FAT
 - Ø4 - primäre DCF-Partition mit 16-Bit-FAT
 - Ø5 - erweiterte DCF-Partition
- 6 Kopfadresse bei Partitionende
- 7 Sektoradresse bei Partitionende (Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse)
- 8 Zylinderadresse bei Partitionende (die höchsten 2 Bit befinden sich im Sektoradressen-byte)
- 9/10 niedriges WORD der Anzahl der Sektoren, die der Partition vorangehen
- 11/12 höheres WORD der Anzahl der Sektoren, die der Partition vorangehen
- 13/14 niedriges WORD der Anzahl der Sektoren, die die Partition belegt
- 15/16 höheres WORD der Anzahl der Sektoren, die die Partition belegt

BOOT-Indikator: Der BOOT-Indikator 80h wird durch FDISK.COM für eine primäre Partition gesetzt, wenn diese als aktiv bzw. ladbar gekennzeichnet werden soll. Er teilt dem Master-Bootrecord mit, dass dieser den Sektor, dessen Lage in den folgenden 3 Bytes eingetragen ist, auf die absolute Adresse 0:7C00h laden und aufrufen soll. Dieser Sektor ist der aktuelle Bootrecord bzw. der Partition-Bootrecord für das gewählte Betriebssystem. Die BOOT-Indikatoren der restlichen 3 Partitions werden gleichzeitig auf 0 gesetzt. Eine erweiterte Partition kann nicht aktiv bzw. ladbar gemacht werden.

7.22.2.4.2. Partitions

Partitions sind Zylindervielfache, die ab Kopf 0, Sektor 1 beginnen. Eine Festplatte kann logisch in 1-4 "primäre" Partitions eingeteilt werden. Jede dieser Partition ist einem bestimmten Betriebssystem zugeordnet; ein Betriebssystem darf jedoch nur eine "primäre" Partition besitzen.

***** BIOS-Interrupts *****

Eine DCP-Partition, die ab Zylinder 0 stehen soll, beginnt erst ab Zylinder 0, Kopf 1. Der Datenbereich Zylinder 0, Kopf 0 ist für den Master-Bootrecord (Sektor 1) und für mögliche Informationen zur Definition des Festplattentyps reserviert. Unter DCP 3.30 werden 2 Partitionarten verwaltet, sowohl eine primäre als auch eine erweiterte DCP-Partition.

Mit dem Programm FDISK.COM ist eine DCP-Partition auf einer Festplatte einrichtbar, und mit FORMAT.COM kann diese formatiert werden.

7.22.2.4.2.1. Primäre DCP-Partition

Die primäre DCP-Partition ist notwendig, um DCP auf der Festplatte zu aktivieren bzw. von der Festplatte laden zu können. Sie kann maximal 32 MByte gross sein und wird in der Partitionstabelle des Master-Bootrecords mit dem Systemindikator 01 (12-Bit-FAT) oder 04 (16-Bit-FAT) gekennzeichnet. Auf dem ersten Sektor der Partition steht der Partition-Bootrecord. Er entspricht einem normalen DCP-Bootrecord und wird durch das Programm FORMAT.COM aufgezeichnet. Wird das Betriebssystem mit auf diese Partition kopiert und die Partition mit FDISK.COM aktiviert, d.h. als ladbar gekennzeichnet, so kann dieses Betriebssystem von dieser Partition aus gestartet werden.

7.22.2.4.2.2. Erweiterte DCP-Partition

Die erweiterte DCP-Partition kann nur eingerichtet werden, wenn bereits eine primäre DCP-Partition auf einer Festplatte existiert. Sie wird in der Partitionstabelle des Master-Bootrecords mit dem Systemindikator 05 gekennzeichnet. Die erweiterte DCP-Partition ist nicht startbar bzw. aktiv und sollte deshalb auch nicht als startbar gekennzeichnet werden.

7.22.2.4.2.2.1. Logischer Aufbau der erweiterten DCP-Partition

Die erweiterte DCP-Partition besteht aus erweiterten logischen Einheiten, die durch einen Zeiger im erweiterten Bootrecord miteinander verkettet werden. Eine erweiterte logische Einheit besteht aus einem erweiterten Bootrecord und einem logischen Laufwerk. Eine erweiterte logische Einheit innerhalb einer erweiterten DCP-Partition kann im DCP 3.30 eine Grösse von minimal 1 Zylinder und maximal 32 MByte einnehmen. Alle erweiterten logischen Einheiten beginnen und enden an einer Zylindergrenze. Der erweiterte Bootrecord ist vergleichbar mit dem Master-Bootrecord, der sich auf dem ersten Sektor der Festplatte befindet, und das logische Laufwerk entspricht der primären DCP-Partition, deren Zeiger im Master-Bootrecord enthalten ist.

*** BIOS-Interrupts ***

Das logische Laufwerk beginnt im DCP (Systemindikator 01 oder 04) mit einem normalen DCP-Bootrecord an einer Spurgrenze (Kopfgrenze) und folgt dem erweiterten Bootrecord der erweiterten logischen Einheit.
Die Begrenzung eines logischen Laufwerkes liegt an der Zylinder-
grenze der erweiterten logischen Einheit.

7.22.2.4.2.2.2. Erweiterter Partition-Bootrecord

Jede erweiterte logische Einheit beinhaltet einen erweiterten Bootrecord auf ihrem ersten Sektor zur Bestimmung ihrer Lage. Dieser Bootrecord besitzt eine logische Laufwerkstabelle, die nur 2 Eintragungstypen enthalten kann. Der Programmcode dieses Bootrecords ist uninteressant, da dieser Bootrecord nicht ladbar bzw. nicht startbar ist. Falls er doch geladen und angesprungen wird, gibt er eine Meldung auf dem Bildschirm aus, die auf eine unladbare Partition hinweist.
Der erweiterte Partition-Bootrecord ist ähnlich dem Master-Bootrecord. Die bedeutendsten Unterschiede sind, dass der erweiterte Bootrecord anstatt einer Partition-Tabelle eine logische Laufwerkstabelle besitzt und dass der Systemindikatorwert 05 anstatt zu einer anderen erweiterten Partition zur nächsten erweiterten logischen Einheit zeigt.

7.22.2.4.2.2.3. Logische Laufwerkstabelle des erweiterten Partition-Bootrecords

Die logische Laufwerkstabelle des erweiterten Bootrecords besitzt dieselbe Struktur wie die Partition-Tabelle im Master-Bootrecord.

Es existieren 4 Eintragungen mit je 16 Bytes. Am Ende der Tabelle ist die Signatur 55AAh zur Kennung eines gültigen Bootrecords für eine erweiterte logische Einheit eingetragen.

Offset	Inhalt

(bez. Bootrecord- Beginn des erweiterten log. Laufwerkes)	
1BEh-1CDh	1. Eintragung
1CEh-1DDh	2. Eintragung
1DEh-1EDh	3. Eintragung
1EEh-1FDh	4. Eintragung
1FEh-1FFh	Signatur (55AAh)

*** BIOS-Interrupts ***

Aufbau einer Tabelleneintragung

Länge: 16 Byte

- Byte
- 1 Boot-Indikator (s. Partition-Tabelle)
→ sollte 0 sein, da das Starten von einer erweiterten Partition aus nicht möglich ist
 - 2 Kopfadresse bei Beginn der nächsten erweiterten logischen Einheit (Systemind. 05) oder des logischen Laufwerkes (Systemind. 01, 04 oder 06)
 - 3 Sektoradresse bei Beginn der nächsten erweiterten logischen Einheit (Systemind. 05) oder des logischen Laufwerkes (Systemind. 01, 04 oder 06) (Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse)
 - 4 Zylinderadresse bei Beginn der nächsten erweiterten logischen Einheit (Systemind. 05) oder des logischen Laufwerkes (Systemind. 01, 04 oder 06) (die höchsten 2 Bit befinden sich im Sektoradressenbyte)
 - 5 Systemindikator
 - 00 - kein Verweis auf ein logisches Laufwerk oder auf nächste erweiterte logische Einheit
 - 01 - logisches Laufwerk mit DCP-12-Bit-FAT
 - 04 - logisches Laufwerk mit DCP-16-Bit-FAT
 - 05 - Verweis auf nächste erweiterte logische Einheit bzw. auf nächsten erweiterten Bootrecord
 - 06 - reserviert für künftige Nutzung
 - 6 Kopfadresse bei Beginn der nächsten erweiterten logischen Einheit (Systemind. 05) oder des logischen Laufwerkes (Systemind. 01, 04 oder 06)
 - 7 Sektoradresse bei Beginn der nächsten erweiterten logischen Einheit (Systemind. 05) oder des logischen Laufwerkes (Systemind. 01, 04 oder 06) (Bit 6 u. 7 sind die höchsten 2 Bit der Zylinderadresse)
 - 8 Zylinderadresse bei Beginn der nächsten erweiterten logischen Einheit (Systemind. 05) oder des logischen Laufwerkes (Systemind. 01, 04 oder 06) (die höchsten 2 Bit befinden sich im Sektoradressenbyte)

*** BIOS-Interrupts ***

9/10	niedriges WORD der Anzahl der Sektoren, die nächster erweiterter logischer Einheit oder logischem Laufwerk vorangehen.
11/12	höheres WORD der Anzahl der Sektoren, die nächster erweiterter logischer Einheit oder logischem Laufwerk vorangehen
13/14	niedriges WORD der Anzahl der Sektoren, die die nächste erweiterte logische Einheit oder das logische Laufwerk belegt
15/16	höheres WORD der Anzahl der Sektoren, die die nächste erweiterte logische Einheit oder das logische Laufwerk belegt

Ist der Systemindikator 0, sind alle Werte in dieser Laufwerkstabelleneintragung ebenfalls 0.

Es kann nur ein logisches Laufwerk pro erweiterter Bootrecord definiert werden. Deshalb sind in jedem erweiterten Bootrecord maximal nur 2 Eintragungen möglich:

- 1 Eintragung mit Systemindikatorbyte 01 od. 04 (logisches Laufwerk)
- 1 Eintragung mit Systemindikatorbyte 05 (Zeiger zur nächsten logischen Einheit)

Start- und Endfelder für die Eintragung, die auf das logische Laufwerk verweist (Systemindikator 01, 04), setzen die physischen Grenzen des Bereiches des logischen Laufwerkes, und sie sind der relative Offset zum Beginn des erweiterten Bootrecords, der die Eintragung beinhaltet.

Start- und Endfelder für die Eintragung, die zur nächsten erweiterten logischen Einheit verweist (Systemindikator 05), setzen die physischen Grenzen der nächsten erweiterten logischen Einheit, und sie sind relativ zu Beginn der gesamten physischen Festplatte.

Das relative Sektoradressen- und das Sektoranzahlfeld werden je nach Systemindikatorbyte ebenfalls unterschiedlich angewandt. Hat das Systemindikatorbyte den Wert 01 oder 04 für eine Eintragung (Zeiger zu einem logischen LW), ist das relative Sektoradressenfeld der Offset ab Beginn des erweiterten Bootrecord der entsprechenden erweiterten logischen Einheit.

Das Sektoranzahlfeld erhält die Grösse des durch das logische Laufwerk belegten Bereiches (Anzahl der Sektoren, die durch Start- und End-Zylinder/Spur/Sektor-Feld begrenzt werden). Die Grösse der erweiterten logischen Einheit kann aus der Addition des relativen Sektoradressenfeldes und des Sektoranzahlfeldes berechnet werden.

*** BIOS-Interrupts ***

Hat das Systemindikatorbyte den Wert 05h für eine Eintragung, ist das relative Sektoradressenfeld der Offset der nächsten erweiterten logischen Einheit vom Beginn der vollständigen erweiterten DCP-Partition. Das Sektoranzahlfeld wird nicht benutzt und ist 00.

Die letzten 2 Bytes (55AAh) des erweiterten Bootrecords werden als Signatur zur Identifikation für einen gültigen Bootrecord benutzt. Sowohl dieser Record als auch die Bootrecords der logischen Laufwerke müssen im Offset 1FEh diese Signatur beinhalten.

7.22.2.4.2.3. Einrichten und Aktivieren der Partitions

Mit dem Programm FDISK.COM wird die primäre und die erweiterte Partition auf der Festplatte eingerichtet. Es können je nach Größe der Festplatte mehrere erweiterte logische Einheiten bzw. logische Laufwerke in der erweiterten Partition angelegt werden. Anschliessend ist die primäre DCP-Partition mit FDISK.COM zu aktivieren. Damit erst wird ein Zugriff auf die DCP-Partition möglich, und das DCP-Betriebssystem übernimmt die Steuerung beim Einschalten des Gerätes oder beim Warmstart.

7.22.2.4.2.4. Formatieren Partition

Mit dem Programm FORMAT.COM wird der Bootrecord auf den ersten Sektor der aktiven primären Partition aufgezeichnet. Es werden FAT und Dateiverzeichnis angelegt.

Wird der Schalter /S mit angegeben, so wird das Betriebssystem mit auf die Partition kopiert, und es wird das Laden des Systems von dieser aktiven Partition bei Systemstart bzw. RESET ermöglicht. Die logischen Laufwerke der erweiterten Partition werden unter Angabe der entsprechenden logischen Laufwerksbezeichnung mit FORMAT ohne /S-Schalter formatiert.

7.22.2.4.3. Neuer INT19h-BOOTSTRAP-LOADER

Der HD-ROMBIOS-Teil enthält einen neuen INT19h-BOOTSTRAP-LOADER. Dieser wird vom Systemplatinen-ROMBIOS als aktueller INT19h angesprungen und realisiert das Laden des Systems von der Festplatte.

- Ablauf:
1. Es wird versucht, das System von Diskette in LW A zu laden und zu starten.
 2. Wenn im LW A keine Diskette ist, wird versucht, den Master-Bootrecord vom ersten Sektor der ersten Festplatte auf Adresse 0:7C00h zu lesen.

*** BIOS-Interrupts ***

- keine Festplatte installiert -|
 - Lesefehler | --> System-Halt
 - keine gültige Signatur 55AAh -|
3. Das Programm des Master-Bootrecords wird aufgerufen. Er durchsucht die Partition-Tabelle nach einem "aktiven" Partition (Boot-Indikator=80h).
- keine Partition "aktiv" --> System-Halt
 - mehrere Partitions "aktiv" --> Meldung:
oder Boot-Indikator un- "INVALID
gültig PARTITION-
TABLE" -->
System-Halt
4. Wenn ein Boot-Indikator als "aktiv" gekennzeichnet ist, wird dieser Partition-Bootrecord auf die Adr. 0:7C00h gelesen und auf Gültigkeit geprüft.
- Lesefehler nach 5 Wieder- --> Meldung:
holungen "ERROR
LOADING OPERA-
TING SYSTEM"
--> System-
Halt
 - Partition-Bootrecord --> Meldung:
enthält keine gültige "MISSING
Signatur 55AAh OPERATING
SYSTEM" -->
System-Halt
5. Das Programm des Partition-Bootrecords wird angesprungen und übernimmt das Laden des Betriebssystems von der "aktiven" Partition. Die Adresse der Partition-Tabelleneintragung der aktiven Partition wird in DS:SI geladen.
- kein gültiges System --> Meldung:
"NO SYSTEM" o.
"ERROR SYSTEM"
--> System-
Halt

*** BIOS-Interrupts ***

Mittels der Kanalnummer kann der Kanal eingestellt werden, über welchen die Datenein- und -ausgabe erfolgen soll.
Es gilt:

Kanalnummer:

20h = COM1: Ein- und Ausgabe über den Adapter für serielle Kommunikation - Kanal A
01h = COM2: Ein- und Ausgabe über den Adapter für serielle Kommunikation - Kanal B

Bei Einsatz einer zweiten Adapterkarte ist die Übertragung über COM3 und COM4 möglich.
Dabei entspricht:

Kanalnummer:

02h = COM3: Ein- und Ausgabe über den Adapter für serielle Kommunikation - Kanal A
03h = COM4: Ein- und Ausgabe über den Adapter für serielle Kommunikation - Kanal B

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AH	Leitungsstatus
AL	Modemstatus

(Erläuterungen zum Status siehe Punkt 7.23.4. - Abfrage Status)

7.23.2. SENDEN EINES ZEICHENS

INT14: 01h

Funktion:

Durch diese Funktion kann ein Zeichen über den ausgewählten COM-Port des Adapters für serielle Kommunikation ausgegeben werden.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
AL	Zeichen
DX	Kanalnummer

(Erläuterungen zur Kanalnummer siehe Punkt 7.23.1. - Initialisierung serieller Anschluss)

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AH	Leistungsstatus
AL	Modemstatus

(Erläuterungen zum Status siehe Punkt 7.23.4. - Abfrage Status)

7.23.3. EMPFANGEN EINES ZEICHENS

INT14: 02h

Funktion:

Durch diese Funktion kann ein Zeichen vom ausgewählten COM-Port des Adapters für serielle Kommunikation eingelesen werden.

Aufruf:

Register	Inhalt/Bedeutung
AH	02h
DX	Kanalnummer

(Erläuterungen zur Kanalnummer siehe Punkt 7.23.1. - Initialisierung serieller Anschluss)

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AH	Leistungsstatus
AL	Zeichen

(Erläuterungen zum Status siehe Punkt 7.23.4. - Abfrage Status)

7.23.4. ABFRAGE STATUS

INT14: 03h

Funktion:

Durch diese Funktion kann der Zustand des ausgewählten COM-Port abgefragt werden.

Aufruf:

Register	Inhalt/Bedeutung
AH	03h
DX	Kanalnummer

(Erläuterungen zur Kanalnummer siehe Punkt 7.23.1. - Initialisierung serieller Anschluss)

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AH	Leitungsstatus
AL	Modemstatus

Der Leitungsstatus liefert Informationen über den SIO-Schaltkreis im Hinblick auf die Datenübertragung.

Der Inhalt wird nachfolgend beschrieben:

Bit	7	6	5	4	3	2	1	0	Bit
									+-- = 1 Empfangsdaten vorhanden
									+-- = 1 Empfängerüberlauf
									+-- = 1 Paritätsfehler
									+-- = 1 BREAK-Folge erkannt
									+-- = 1 Rahmenfehler z.B. kein gültiges Stopp-Bit
									+-- = 1 Sendepuffer leer
									+-- = 1 Sender leer (alle Zeichen gesendet)
									+-- = 1 TIME OUT

Dieses Bit wird gesetzt, wenn ein Zeichen in einer vorgegebenen Zeit nicht gesendet oder empfangen wurde. In diesem Fall sind die übrigen Bits des Leitungs- und Modemstatus ohne Bedeutung.

Der Modemstatus liefert den momentanen Status der Steuerleitungen vom Modem bzw. Peripheriegerät.

Neben diesen Statusinformationen dienen vier Bits (Bit 0-3) zur Ausgabe von Änderungsinformationen. Diese Bits werden auf 1 gesetzt, wenn das entsprechende Steuersignal vom Modem seinen Status seit der letzten Abfrage geändert hat.

*** BIOS-Interrupts ***

Der Inhalt des Modemstatus wird nachfolgend beschrieben:

Bit	7	6	5	4	3	2	1	0	Bit
									+--- Statusänderung
									Signal CTS
									("Bereit zum
									Senden")
									+--- Statusänderung Signal
									DSR ("Modem bereit")
									+--- Statusänderung Signal RI
									("Anrufsignal")
									+--- Statusänderung Signal RLSD
									("Empfangsleitungssignal erkannt")
									+--- Zustand Signal CTS (V.24: Ltg. 106)
									("Bereit zum Senden")
									+--- Zustand Signal DSR (V.24: Ltg. 107)
									("Modem bereit")
									+--- Zustand Signal RI (V.24: Ltg. 125)
									("Anrufsignal")
									+--- Zustand Signal RLSD (V.24: Ltg.109)
									("Empfangsleitungssignal erkannt")

7.24. VERWALTUNGSFUNKTIONEN

INT15

Funktion:

Dieser Interrupt ist vorgesehen zur Steuerung spezieller Geräte und zur Anmeldung von E/A-Ereignissen für Verwaltungsfunktionen.

Bemerkung:

Die meisten Funktionen sind zur Zeit nicht angeschlossen, und es wird deshalb vom ROM-BIOS keine Aktion ausgeführt.

7.25. TASTATUR

INT16

7.25.1. ZEICHEN LESEN

INT16: 00h

Funktion:

Mit dieser Funktion wird ein Zeichen aus dem Tastaturpuffer gelesen. Ist zum Zeitpunkt des Funktionsaufrufs kein Zeichen verfügbar, wird bis zur Eingabe eines Zeichens über die Tastatur gewartet.

*** BIOS-Interrupts ***

Aufruf:

Register	Inhalt/Bedeutung
AH	00h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	ASCII-Kode
AH	Tastatur-Kode

Wenn AL=0 ist, dann enthält AH nicht den SCAN-Kode, sondern einen erweiterten Kode. Die Codes, die bei diesem Funktionsaufruf bereitgestellt werden, sind in Anlage 5 (A5.2.2.) zusammengestellt.

7.25.2. TEST ZEICHEN VERFÜGBAR

INT16: 01h

Funktion:

Test, ob im Tastaturpuffer Zeichen verfügbar sind.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
ZF	Testergebnis

ZF=1 Zeichen verfügbar
ZF=0 kein Zeichen im Tastaturpuffer.

Bemerkung:

Das Zeichen kann mit der Funktion 0 gelesen werden.

7.25.3. TASTATURSTATUS

INT16: 02h

Funktion:

Lesen des Status der Umschalt(Shift)-Tasten der Tastatur.

Aufruf:

Register	Inhalt/Bedeutung
AH	02h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Status der Shift-Tasten

In AL wird der gegenwärtige Zustand der Shift-Tasten gemeldet.

Bit:

7	6	5	4	3	2	1	0	
								+-- = 1 rechte Shift-Taste gedrückt
								+----- = 1 linke Shift-Taste gedrückt
								+----- = 1 CTRL-Taste gedrückt
								+----- = 1 ALT-Taste gedrückt
								+----- = 1 SROLL-LOCK gesetzt
								+----- = 1 NUM-LOCK gesetzt
								+----- = 1 CAPS-LOCK gesetzt
								+----- = 1 INSERT gesetzt

7.25.4. TASTATURPUFFER SCHREIBEN

INT16: 05h

Funktion:

Schreiben von Codes in den Tastaturpuffer.

Aufruf:

Register	Inhalt/Bedeutung
AH	05h
CL	ASCII-Kode
CH	Tastatur-Kode

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	0

Die Codes wurden im Puffer gespeichert.

*** BIOS-Interrupts ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	1

Der Tastaturpuffer ist voll. CX konnte nicht abgespeichert werden.

Bemerkung:

Diese Funktion ist die Umkehr der Funktion 00h.

7.25.5. ZEICHEN LESEN ERWEITERT

INT16: 10h

Funktion:

Mit dieser Funktion wird ein Zeichen aus dem Tastaturpuffer gelesen. Ist zum Zeitpunkt des Funktionsaufrufs kein Zeichen verfügbar, wird bis zur Eingabe eines Zeichens über die Tastatur gewartet.

Diese Funktion erlaubt eine detailliertere Auswertung der betätigten Tasten (z.B. Unterscheidung der numerischen und der Steuertasten).

Aufruf:

Register	Inhalt/Bedeutung
AH	10h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	ASCII-Kode
AH	Tastatur-Kode

Die Codes, die bei diesem Funktionsaufruf bereitgestellt werden, sind in Anlage 5 (A5.2.3.) zusammengestellt.

7.25.6. TEST ZEICHEN VERFÜGBAR ERWEITERT

INT16: 11h

Funktion:

Test, ob im Tastaturpuffer Zeichen verfügbar sind.

Aufruf:

Register	Inhalt/Bedeutung
AH	11h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
ZF	Testergebnis
ZF=1	Zeichen verfügbar
ZF=0	kein Zeichen im Tastaturpuffer.

Bemerkung:

Das Zeichen kann mit der Funktion 10h gelesen werden.

7.25.7. TABTATURSTATUS ERWEITERT

INT16: 12h

Funktion:

Lesen des erweiterten Status der Shift-Tasten der Tastatur.

Aufruf:

Register	Inhalt/Bedeutung
AH	12h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AX	Status der Shift-Tasten

In AL wird der gegenwärtige gemeinsame Zustand der Shift-Tasten gemeldet (analog Funktion 02h).

Bit:

7	6	5	4	3	2	1	0	
								= 1 rechte Shift-Taste gedrückt
							+	= 1 linke Shift-Taste gedrückt
							+	= 1 CTRL-Taste gedrückt
							+	= 1 ALT-Taste gedrückt
							+	= 1 SROLL-LOCK gesetzt
							+	= 1 NUM-LOCK gesetzt
							+	= 1 CAPS-LOCK gesetzt
							+	= 1 INSERT gesetzt

*** BIOS-Interrupts ***

In AH wird der gegenwärtige Zustand folgender Tasten gemeldet:

Bit:

7	6	5	4	3	2	1	0	
								= 1 CTRL-Taste gedrückt
								= 1 ALT-TASTE gedrückt
								nicht benutzt
								= 1 ALTI-Taste gedrückt
								= 1 SROLL LOCK-Taste gedrückt
								= 1 NUM LOCK-Taste gedrückt
								= 1 CAPS LOCK-Taste gedrückt
								= 1 SYS-Taste gedrückt

7.26. AUSGABE ÜBER PARALLELE SCHNITTSTELLE AN DRUCKER

INT17

7.26.1. DRUCKEN EINES ZEICHENS

INT17: 00h

Funktion:

Durch diese Funktion kann ein Zeichen über den CENTRONICS-Adapter an den Drucker ausgegeben werden.

Aufruf:

Register	Inhalt/Bedeutung
AH	00h
AL	Zeichen
DX	00h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AH	Status

(Erläuterungen zum Status siehe Punkt 7.26.3. - Status abfragen)

7.26.2. DRUCKER INITIALISIEREN

INT17: 01h

Funktion:

Durch diese Funktion wird der Drucker über den CENTRONICS-Adapter initialisiert.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
DX	00h

Die Initialisierungsroutine führt ein Drucker-RESET aus und stellt die Steuerleitungen folgendermassen ein:

```

SELECT IN:      AKTIV      (Drucker selektiert)
INIT:           INAKTIV   (kein RESET)
AUTO FEED:     INAKTIV   (kein automatisches LINE FEED
                        (Zeilenvorschub))
IRQ7:          INAKTIV   (Interruptverbot)
    
```

Durch diese Initialisierung ist eine Zeichenausgabe im Handshaking-Betrieb über Polling des BUSY-Signales möglich.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AH	Status

(Erläuterungen zum Status siehe Punkt 7.24.3. - Status abfragen)

7.26.3. DRUCKER-STATUS ABFRAGEN

INT17: 02h

Funktion:

Durch diese Funktion kann der Drucker-Status abgefragt werden.

Aufruf:

Register	Inhalt/Bedeutung
AH	02h
DX	00h

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AH	Status

Das Status-Byte hat folgenden Inhalt:

Bit	Bit
7	0
6	+++ = 1 TIME OUT
5	+++ nicht benutzt
4	+++ nicht benutzt
3	+++ = 1 Druckerfehler, z.B. Papierende, OFFLINE-Zustand
2	+++ = 1 Drucker empfangsbereit (SELECT - Signal)
1	+++ = 1 Papierende
0	++ = 1 Signal ACKNOWLEDGE Impuls vom Drucker, kann programmtechnisch nicht ausgewertet werden
	+++ = 1 Drucker empfangsbereit (BUSY - Signal)

7.27. Reserviert

INT18

Funktion:

Dieser Interrupt ist reserviert.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

7.28. URLADEN

INT19

Funktion:

Diese Funktion liest den Urladesektor (Bootrecord) von Diskette oder Festplatte ein und übergibt ihm die Steuerung. Der Bootrecord enthält ein Programm zum Laden des Betriebssystems.

Bemerkung:

Im Kapitel 4 ist der Vorgang des Urladens detailliert beschrieben.

7.29. TAGESZEIT

INT1A

7.29.1. TAGESZEIT LESEN

INT1A: 00h

Funktion:

Mit dieser Funktion wird die aktuelle Tageszeit bereitgestellt.

Aufruf:

Register	Inhalt/Bedeutung
AH	00h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CX	Höherwertiger Teil des Zählers
DX	Niederwertiger Teil des Zählers
AL	Kennzeichen

Im höherwertigen Teil des Zählers werden die Stunden gespeichert. Werden über 24 Stunden gezählt, dann wird dieser Teil des Zählers auf Null zurückgesetzt und das Kennzeichen ungleich Null gesetzt.

Im niederwertigen Teil des Zählers werden die Interrupts (INT08) gezählt, und bei Nulldurchgang wird der höherwertige Teil des Zählers um 1 erhöht.

Das Kennzeichen hat folgenden Inhalt:

AL = 0, wenn die Anzahl der Stunden kleiner 24
 <>0, wenn über 24 Stunden gezählt wurden.

7.29.2. TAGESZEIT SETZEN

INT1A: 01h

Funktion:

Mit dieser Funktion kann die aktuelle Tageszeit eingestellt werden.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
CX	Höherwertiger Teil des Zählers
DX	Niederwertiger Teil des Zählers

*** BIOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
--	--

Im höherwertigen Teil des Zählers werden die Stunden angegeben.
Im niederwertigen Teil des Zählers werden die Sekunden * 18,75
eingetragen.

7.30. TASTATURUNTERBRECHUNG

INT1B

Funktion:

Diese Interruptroutine wird von der Tastatur-Interruptroutine (INT09-Routine) aufgerufen, wenn die Tastenkombination CTRL/PAUSE erkannt wird.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert. Der Nutzer hat aber die Möglichkeit, eine tastaturgesteuerte Unterbrechungsroutine einzubinden.

Dabei ist zu beachten:

- Mit IRET wird zur Tastatur-Interruptroutine zurückverzweigt.
- Wenn nicht in die Tastatur-Interruptroutine zurückgekehrt werden soll, sind ggf. mehrere End-Of-Interrupt-Kommandos an den Interrupt-Controller zu senden, laufende E/A-Prozesse zu beenden und Stackkorrekturen vorzunehmen.

7.31. ZEITGEBER NUTZER

INT1C

Funktion:

Mit diesem Interrupt besteht die Möglichkeit, eine zeitabhängige nutzerspezifische Routine einzubinden. Diese Routine wird von der INT08-Routine angesprochen.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger auf eine IRET-Anweisung initialisiert.

7.32. ADRESSE BILDSCHIRMPARAMETER

INT1D

Funktion:

Der Interruptvektor 1Dh zeigt auf ein Parameterfeld für die Bildschirminitialisierung. Er wird für interne Zwecke benutzt und darf nicht verändert werden.

7.33. ADRESSE DISKETTENPARAMETER-BLOCK (DISK-BASE)

INT1E

Funktion:

Hier wird die Adresse der DISK-BASE verwaltet. Die DISK-BASE enthält Parameter für die Diskettenarbeit des INT13.

Bemerkung:

Der Aufbau der DISK-BASE ist im INT13 beschrieben.

7.34. ADRESSE GRAFISCHER ZEICHENSATZ

INT1F

Funktion:

Der Interruptvektor 1Fh zeigt auf eine Zeichengeneratortabelle für die Kodierungen zwischen 80h und 0FFh des Farbgrafikadapters.

Bemerkung:

Vom ROM-BIOS wird dieser Interruptvektor als Zeiger mit 0 initialisiert.

Mit dem Kommando GRAFTABL wird eine Zeichensatztablelle mit speziellen Grafikzeichen geladen und der Vektor 1Fh auf diese Tabelle eingestellt.

8. DOS-Interrupts

Im Betriebssystem wird ein Speicherbereich zur Verfügung gestellt, in dem sich die Interruptvektoren 00h bis FFh befinden (absoluter Speicherbereich von 00000h bis 003FFh). Diese dienen der Realisierung bestimmter Systemfunktionen.

DOS reserviert die Interruptvektoren 20h bis 3Fh für den eigenen Gebrauch, d.h., da der absolute Speicherbereich von 00080h bis 000FFh für sie reserviert ist.

Diese Interrupts dienen der Kommunikation zwischen Nutzer und Betriebssystem.

Sollen die im Betriebssystem installierten Inhalte der Interruptvektoren überprüft oder gesetzt werden, so sind die dafür vorgesehenen Funktionen zu verwenden.

Ein direkter Bezug auf die Speicherplätze der Interruptvektoren ist nicht zu empfehlen.

Der Interrupt 21h stellt die Funktionsaufrufe für "Interruptvektor abfragen" (35h) und für "Setzen Interruptvektor" (25h) zur Verfügung.

Das Betriebssystem nutzt die folgenden Register, Zeiger und Flags, wenn Interrupts ausgeführt bzw. DOS-Funktionen aufgerufen werden.

Allgemeine Register	
Register	Definition
AX	Akkumulator (16 Bit)
AH	Akkumulator höherwertiges Byte (8 Bit)
AL	Akkumulator niederwertiges Byte (8 Bit)
BX	Basis (16 Bit)
BH	Basis höherwertiges Byte (8 Bit)
BL	Basis niederwertiges Byte (8 Bit)
CX	Zähler (16 Bit)
CH	Zähler höherwertiges Byte (8 Bit)
CL	Zähler niederwertiges Byte (8 Bit)
DX	Daten (16 Bit)
DH	Daten höherwertig (8 Bit)
DL	Daten niederwertig (8 Bit)
Flags	OF, DF, IF, TF, SF, ZF, AF, FF, CF

*** DOS-Interrupts ***

Zeiger	
Register	Definition
SP	Stapelzeiger (16 Bit)
BP	Basiszeiger (16 Bit)
IP	Befehlszeiger (16 Bit)

Segment-Register	
Register	Definition
CS	Kodesegment (16 Bit)
DS	Datensegment (16 Bit)
SS	Stapelsegment (16 Bit)
ES	Extrasegment (16 Bit)

Index-Register	
Register	Definition
DI	Zielindex (16 Bit)
SI	Stapelindex (16 Bit)

Nachfolgend werden die durch DOS definierten Interrupts 20h bis 27h und 2Fh beschrieben.

8.1. PROGRAMM BEENDEN

INT20

Funktion:

Beenden eines laufenden Programms.

Aufruf:

Register	Inhalt/Bedeutung
CS	Segmentadresse des aktuellen PSP

Vor Aufruf des Interrupts 20h muss das Nutzerprogramm sicherstellen, dass das CS-Register die Segmentadresse des eigenen PSP (Programm-Segment-Präfix) enthält.

Bemerkung:

Wird dieser Interrupt ausgeführt, so werden die Inhalte der Interruptvektoren 22h, 23h und 24h wieder auf die Werte eingestellt, die sie bei Eintritt in das Programm hatten.

*** DOS-Interrupts ***

Alle Dateien, die mit einem FCB bearbeitet werden, müssen vor Ausführung des Interrupts 20h geschlossen werden (siehe Funktionsaufruf 10h des Interrupts 21h). Wird eine geänderte Datei nicht geschlossen, so sind die Werte für Länge, Datum und Zeit nicht richtig im Verzeichnis eingetragen.

Alle Dateien, die über einen Händler bearbeitet werden, werden durch den Interrupt 20h geschlossen.

Ausserdem erfolgt stets die Löschung aller Dateipuffer.

Dieser Interrupt gibt keine Informationen an den übergeordneten Prozess zurück.

Damit bei Austritt ein Rückgabekode bzw. ein Fehlerkode übergeben werden kann, müssen entweder der Funktionsaufruf "Prozess beenden (EXIT)" (4Ch) oder "Prozess beenden und resident bleiben" (31h) des Interrupts 21h verwendet werden.

Der Rückgabe- bzw. Fehlerkode kann in einer Stapelverarbeitung abgefragt werden. Deshalb sind die beiden obigen Funktionsaufrufe der Benutzung des Interrupts 20h vorzuziehen.

8.2. AUFRUF VON DOS-FUNKTIONEN

INT21

Funktion:

Mit dem Interrupt 21h können viele Funktionen, die das Betriebssystem zur Verfügung stellt, aufgerufen werden. Die Auswahl der Funktion erfolgt über das Register AX.

Aufruf:

Register	Inhalt/Bedeutung
AH	Nummer der DOS-Funktion
AL	Unterfunktion
sonstige	Übergabeparameter entsprechend der DOS-Funktion

Bemerkung:

Im weiteren werden alle Funktionsaufrufe des Interrupts 21h als DOS-Funktionsaufrufe bezeichnet.

Eine detaillierte Beschreibung erfolgt im Abschnitt "Die DOS-Funktionsaufrufe" (s. Punkt 9).

8.3. ABSCHLUSSADRESSE

INT22

Funktion:

Wenn ein Programm beendet wird, übergibt das Betriebssystem die Steuerung an den Interrupt 22h.

Bemerkung:

Bei Erstellung eines Programmsegmentes durch das Betriebssystem wird von diesem die Adresse der Interruptroutine 22h in dem entsprechenden PSP (4 Byte ab Offset 0Ah) eingetragen.

Dieser Interrupt sollte nicht direkt angewendet werden. Eine

bessere Methode ist der DOS-Funktionsaufruf "Programm laden und ausführen" (4B00h).

8.4. BEHANDLUNG BEI CTRL-C

INT23

Funktion:

Wenn der Nutzer während einer Standardeingabe, Standardausgabe, Drucker- oder asynchronen Übertragungsoperation CTRL-C bzw. CTRL-PAUSE betätigt, so wird ein Interrupt 23h ausgeführt.

Bemerkung:

Bei Erstellung eines Programmsegmentes durch das Betriebssystem wird von diesem die Adresse der Interruptroutine 23h in dem entsprechenden PSP (4 Byte ab Offset 0Eh) eingetragen.

Wenn das BREAK-Kommando eingeschaltet ist (ON), kann der Interrupt 23h bei jedem DOS-Funktionsaufruf, ausser "Direkte Konsol-Ein-/Ausgabe" (06h) und "Direkte Konsol-Eingabe ohne Echo" (07h), wirksam werden.

Wenn der Nutzer eine eigene Behandlung bei CTRL-C wünscht, so muss er in dieser Routine dafür garantieren, dass sämtliche Register erhalten bleiben, um mit einem IRET-Befehl die Programmausführung fortzusetzen.

Wenn die Interruptroutine des Nutzers mit einem langen Programmrücksprung (RET FAR) zum Betriebssystem zurückkehrt, wird durch das C-Flag dem System mitgeteilt, ob das Programm abgebrochen werden soll oder nicht.

Ist das C-Flag gesetzt (CF=1), wird das Programm abgebrochen. Im anderen Fall wird wie bei einem Rücksprung mit IRET das Nutzerprogramm fortgesetzt.

Werden die DOS-Funktionsaufrufe zur Standardein-/ausgabe aufgerufen und in der Ausführung durch CTRL-C unterbrochen, wird die Bytefolge 03h, 0Dh und 0Ah (dargestellt als ^C, gefolgt von einer Zeilenschaltung) ausgegeben.

Wenn das Programm ein neues Segment anlegt und ein zweites Programm lädt, das die CTRL-PAUSE-Adresse ändert, so führt der Abschluss des zweiten Programms und der Rücksprung in das erste Programm dazu, dass die CTRL-PAUSE-Adresse wieder auf den Wert gestellt wird, den sie vor Ausführung des zweiten Programms hatte.

8.5. VEKTOR DER BEHANDLUNGSRoutine FÜR KRITISCHE FEHLER

INT24

Funktion:

Wenn ein kritischer Fehler während einer E/A-Funktion auftritt, dann übergibt das Betriebssystem die Steuerung an die Behandlungsroutine des Interrupts 24h.

Aufruf:

Bei Aufruf des Interrupts 24h enthalten AX und DI Angaben zum Fehlerkode, und BP:SI (BP=Segmentadresse, SI=Offset) zeigt auf einen Device Header, der Angaben über das Gerät enthält, bei welchem ein Fehler aufgetreten ist.

*** DOS-Interrupts ***

Bemerkung:

Bei Erstellung eines Programmsegmentes durch das Betriebssystem wird von diesem die Adresse der Interruptroutine 24h in dem entsprechenden PSP (4 Byte ab Offset 12h) eingetragen. Der Interrupt 24h wird nicht benutzt, wenn während der Ausführung des Interrupts 25h bzw. 26h ein Fehler auftritt. Diese Fehler werden in der Behandlungsroutine von COMMAND.COM betrachtet. Der Nutzer wird dabei gefragt, ob die Diskettenoperation abgebrochen, wiederholt oder der Fehler ignoriert werden soll.

Allgemeine Grundsätze für Interrupt 24h

1) Bedingungen für eine Interrupt-24h-Behandlungsroutine

Tritt nach mehreren Wiederholungen ein kritischer Fehler auf, wird durch das Betriebssystem eine Behandlungsroutine aufgerufen. In dieser wird der Nutzer aufgefordert, die Handlung, bei der der Fehler aufgetreten ist, abzubrechen, zu wiederholen bzw. ihn zu ignorieren.

Eine nutzergeschriebene Fehlerbehandlungsroutine muss als erstes die Flags im Stapel ablegen und dann die standardmässige Interrupt-24h-Routine durch einen langen Unterprogramm sprung (CALL FAR) aufrufen.

Anschliessend kann die eigene Behandlung durchgeführt werden. Dazu müssen zuerst die Register BX, CX, DX, DS, ES, SS und SP im Stapel abgelegt werden. In der Behandlungsroutine sollten nur die DOS-Funktionen 01h bis 0Ch und 59h verwendet werden, da die Verwendung anderer DOS-Funktionen zum Zerstören des DOS-Stapels führt. Weiterhin sollte der Inhalt des Device Headers nicht verändert werden.

Wenn die Interrupt-24h-Routine zum Nutzerprogramm zurückkehren soll, müssen die Nutzerregister wieder eingestellt werden. Diese werden bei Eintritt in die Routine auf dem Stapel zur Verfügung gestellt. Anschliessend kann durch IRET die Routine beendet werden, und das Nutzerprogramm wird nach dem fehlerverursachenden Interrupt-21h-Aufruf fortgesetzt. Ein solches Vorgehen bewirkt einen instabilen Zustand des Betriebssystems. Dieser wird durch Aufruf einer DOS-Funktion grösser 0Ch aufgehoben.

Im nachfolgenden Beispiel wird eine einfache Verwendung einer Nutzerroutine angegeben:

```
:
:
mov  AX,3524h          ;Interruptvektor 24h abfragen
int  21h
mov  SI,offset INT_ALT ;alten Vektor speichern
mov  [SI],BX          ;in INT_ALT
mov  [SI+2],ES
mov  AX,2524h         ;Interruptvektor 24h setzen
mov  DX,offset ROUT_NEU ;Nutzer-Routine
int  21h
:
```

*** DOS-Interrupts ***

```

INT_ALT dd 0 ;Adresse des alten INT 24h

ROUTINE: ;Fehlerbehandlungsroutine des
: ;Nutzers
:

ROUT_NEU:
pushf ;Flags in Stapel eintragen
call CS:INT_ALT ;Aufruf des Standard-Interrupts
push BX ;Retten der Register
push CX
push DX
push DS
push ES
push SS
push SP
call ROUTINE ;Aufruf Nutzerfehleroutine
pop SP ;Wiederherstellen der Register
pop SS
pop ES
pop DS
pop DX
pop CX
pop BX
iret ;Rückkehr zu DOS

```

Soll direkt in das Nutzerprogramm zurückgekehrt werden, so ist der IRET durch nachfolgende Befehlsfolge zu ersetzen.

```

add SP,6 ;Stapelkorrektur Adresse+Flags (INT 24h)
pop AX ;Einstellen Nutzerregister vom INT 21h
pop BX
pop CX
pop DX
pop SI
pop DI
pop BP
pop DS
pop ES
iret ;Rückkehr ins Nutzerprogramm

```

2. Nutzerstapel

Der Nutzerstapel enthält nachfolgende Angaben von oben nach unten. Das erste beschriebene Element steht am Anfang des Stapels.

IP DOS-Register für Austritt aus dem Interrupt 24h
CS
FLAGS

AX Nutzerregister zum Zeitpunkt des Aufrufes des
BX Interrupts 21h
CX
DX
SI
DI
BP
DS
ES

IP Rückkehradresse des Interrupts 21h zum Nutzer
CS
FLAGS

Bei Rückkehr in das Betriebssystem wird in Register AL übergeben, wie zu reagieren ist. Dabei gilt:

AL = 0 Fehler ignorieren
AL = 1 Wiederholen der Operation
AL = 2 Beenden des Programms durch Interrupt 23h
AL = 3 Verlassen des Systemaufrufs, der abgearbeitet wird
 (Interrupt 20h)

Hinweis: Bei der Antwort AL=0 wurde eine Operation ignoriert. Das Betriebssystem muss daher annehmen, dass die Operationen vollständig und richtig ausgeführt wurden, auch wenn dies nicht der Fall ist.

3. Diskettenfehlerkode in AX

Handelt es sich um einen Hardwarefehler auf der Diskette (Bit 7 von AH=0), so enthält Register AL bei Eintritt in die Routine die Nummer des fehlerhaften Laufwerks (0=Laufwerk A, 1=Laufwerk B usw.).

Weiterhin kennzeichnen die Bits 0 bis 2 von AH den defekten Diskettenbereich und geben ausserdem an, ob es sich um eine Lese- oder Schreiboperation handelt.

*** DOS-Interrupts ***

Es gilt:

Bit 0	Operation
0	bei Leseoperation
1	bei Schreiboperation

Bit 2 1	betroffener Diskettenbereich
0 0	DOS-Bereich (Systemdateien)
0 1	Dateizuordnungstabelle (FAT)
1 0	Verzeichnis
1 1	Datenbereich

Die Bits 3 bis 5 von AH kennzeichnen, welche Antworten in der Standardroutine des Interrupts 24h erlaubt sind.

Es gilt:

Bit 3 = 0,	wenn "Verlassen"	nicht erlaubt ist.
Bit 3 = 1,	wenn "Verlassen"	erlaubt ist.
Bit 4 = 0,	wenn "Wiederholung"	nicht erlaubt ist.
Bit 4 = 1,	wenn "Wiederholung"	erlaubt ist.
Bit 5 = 0,	wenn "Ignorieren"	nicht erlaubt ist.
Bit 5 = 1,	wenn "Ignorieren"	erlaubt ist.

Wenn "Ignorieren" angegeben (AL=0), aber nicht erlaubt (Bit 5=0) ist, so wandelt das Betriebssystem die Antwort in "Verlassen" (AL=3) um.

Wenn "Wiederholung" angegeben (AL=1), aber nicht erlaubt (Bit 4=0) ist, so wandelt das Betriebssystem die Antwort in "Verlassen" (AL=3) um.

Wenn "Verlassen" angegeben (AL=3), aber nicht erlaubt (Bit 3=0) ist, so wandelt das Betriebssystem die Antwort in "Abbruch" (AL=2) um.

Die Antwort "Abbruch" (AL=2) ist stets erlaubt.

4. Anderer Gerätefehlerkode in AX

Wenn Bit 7 von AH gleich 1 ist, so ist der Fehler in einer zeichenorientierten Einheit aufgetreten oder war die Folge eines fehlerhaften Speicherabbildes der Dateizuordnungstabelle (FAT). Der in BP:SI übergebene Device Header enthält ein Attributwort (Offset 04h), das den Gerätetyp und daraus folgend den Fehler kennzeichnet.

*** DOS-Interrupts ***

Der Device Header, auf den BP:SI zeigt, hat folgendes Format:

BP:SI	DWORD	Zeiger zu der nächsten Einheit (FFFFh, wenn letzte Einheit)
BP:SI+04h	WORD	Attribute Bit 15 = 1, wenn Zeicheneinheit Bit 15 = 0, wenn Blockeinheit Wenn Bit 15 = 1, dann ist Bit 0 = 1, wenn aktuelle Standard- eingabeeinheit Bit 1 = 1, wenn aktuelle Standard- ausgabeeinheit Bit 2 = 1, wenn aktuelle NULL-Einheit Bit 3 = 1, wenn aktuelle CLOCK-Einheit Bit 14 = IOCTL-Bit
BP:SI+06h	WORD	Zeiger zu der Eingangsstelle in die Strategieroutine des Einheitentreibers
BP:SI+08h	WORD	Zeiger zu der Eingangsstelle in die Interruptroutine des Einheitentreibers
BP:SI+0Ah	8 Byte	Namensfeld für Zeicheneinheit bei Blockeinheiten. Das erste Byte gibt die Anzahl von Einheiten an.

Um feststellen zu können, ob der Fehler bei einer Block- oder Zeicheneinheit aufgetreten ist, muss Bit 15 im Attribut-WORD überprüft werden.

Wenn Bit 15 gleich 0 ist, so handelt es sich um ein fehlerhaftes Speicherabbild der FAT.

Wenn Bit 15 gleich 1 ist, so handelt es sich um einen Fehler bei einer zeichenorientierten Einheit. Die Bits 0 bis 3 kennzeichnen hierfür die entsprechende Einheit. Der Inhalt von AL ist undefiniert. Der Fehlercode steht in Register DI.

5. Fehlerkode in DI

In DI wird der Fehlerkode übergeben. Das höherwertige Byte von DI ist undefiniert. Das niederwertige Byte von DI enthält nachfolgenden Fehlerkode:

Fehlerkode	Beschreibung
00h	Es wurde versucht, auf eine schreibgeschützte Diskette zu schreiben.
01h	Unbekannte Einheit
02h	Laufwerk nicht bereit
03h	Unbekanntes Kommando
04h	Datenfehler (CRC)
05h	Falsche Länge der Anforderungsstruktur
06h	Positionierfehler
07h	Unbekannter Mediumtyp
08h	Sektor nicht gefunden
09h	Papierende vom Drucker
0Ah	Schreibfehler
0Bh	Lesefehler
0Ch	Allgemeine Störung
0Fh	Ungültiger Diskettenwechsel

Die durch den Nutzer geschriebene Fehlerbehandlungsroutine kann zur weiteren Bearbeitung des Fehlers die DOS-Funktion "Erweiterte Fehlermeldung" (59h) benutzen, um zusätzliche Informationen über den Fehler zu bekommen.

8.6. ABSOLUTES DISKETTENLESEN

INT25

Funktion:

Durch diesen Interrupt werden absolut adressierte Daten von der Platte/Diskette gelesen. Die Steuerung wird direkt an den Gerätetreiber übergeben.

Aufruf:

Register	Inhalt/Bedeutung
AL	Laufwerknummer (0=LW A, 1=LW B, ...)
CX	Anzahl zu lesender Sektoren
DX	Relative Sektornummer
DS:BX	Zeiger auf Puffer

Die angegebene Anzahl von Sektoren wird vom Laufwerk nach der Pufferadresse übertragen.

Die relative Sektornummer CX gibt den Beginn des Lesevorganges an. Die weiteren Sektoren werden fortlaufend gelesen.

Die Numerierung der Sektoren erfolgt entsprechend der logischen Platten-/Diskettenorganisation.

*** DOS-Interrupts ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AL	Fehlerkode
AH	80h Anschluss antwortet fehlerhaft
	40h Positionieroperation fehlerhaft
	08h Fehlerhafte CRC beim Diskettenlesen
	04h Angeforderter Sektor nicht gefunden
	03h Es wurde versucht, auf eine schreibgeschützte Diskette zu schreiben.
	02h Anderer Fehler als die oben angegebenen Fehler.

Der Wert in AL ist identisch mit dem Fehlerkode, der in dem niederwertigen Byte von DI bei Ausführung eines Interrupts 24h ausgegeben wird (siehe 8.5. Pkt.5).

Bemerkung:

Der Nutzer muss bei Anwendung des Interrupts 25h die Kontrolle der Eintrittsparameter selbst übernehmen (besonders die Pufferorganisation am Segmentende).

Sämtliche Register, mit Ausnahme der Segmentregister, werden durch den Aufruf des Interrupts 25h zerstört.

Nach Rückkehr aus dem Interrupt 25h befinden sich im Stapel noch die Flags; wie sie vor dem Aufruf des Interrupts standen. Aus dem Interrupt wird lediglich mit einem RET FAR in die Nutzerebene zurückgekehrt. Dies ist erforderlich, weil die Rückkehrinformation in den aktuellen Flags zurückgegeben wird. Die Stapelverwaltung muss durch den Nutzer erfolgen.

Hinweis:

Das Diskettenlesen über den Interrupt 25h ist nicht unbedingt notwendig, da es entsprechende DOS-Funktionsaufrufe gibt, welche zu weiterführenden Betriebssystemversionen kompatibel sein werden.

8.7. ABSOLUTES DISKETTENSCHREIBEN

INT26

Funktion:

Durch diesen Interrupt werden Daten aus dem Speicher auf die Platte/Diskette geschrieben. Die Steuerung wird direkt an den Gerätetreiber übergeben.

*** DOS-Interrupts ***

Aufruf:

Register	Inhalt/Bedeutung
AL	Laufwerknummer (0=LW A, 1=LW B, ...)
CX	Anzahl zu schreibender Sektoren
DX	Relative Sektornummer
DS:BX	Zeiger auf Puffer

Die angegebene Anzahl von Sektoren wird ab der Pufferadresse zum Laufwerk ab der relativen Sektornummer übertragen.
Die weiteren Sektoren werden fortlaufend geschrieben.
Die Numerierung der Sektoren erfolgt entsprechend der logischen Platten-/Diskettenorganisation.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AL	Fehlercode
AH	80h Anschluss antwortet fehlerhaft
	40h Positionieroperation fehlerhaft
	08h Fehlerhafte CRC beim Diskettenlesen
	04h Angeforderter Sektor nicht gefunden
	03h Es wurde versucht, auf eine schreibgeschützte Diskette zu schreiben.
	02h Anderer Fehler als die oben angegebenen Fehler.

Der Wert in AL ist identisch mit dem Fehlercode, der in dem niederwertigen Byte von DI bei Ausführung eines Interrupts 24h ausgegeben wird. (siehe 8.5. Pkt.5).

Bemerkung:

Für die Benutzung des Interrupts 26h gelten die gleichen Bedingungen wie für den Interrupt 25h (Stapelverwaltung, Fehlercode).

Hinweis:

Das Diskettenschreiben über den Interrupt 26h ist nicht unbedingt notwendig, da es entsprechende DOS-Funktionsaufrufe gibt, welche zu weiterführenden Betriebssystemversionen kompatibel sein werden.

8.8. BEENDEN, ABER RESIDENT BLEIBEN

INT27

Funktion:

Durch diesen Interrupt kann ein Programm bis zu 64K Speichergrösse resident gemacht werden. Er wird meistens dazu genutzt, um gerätespezifische Interruptbehandlungen zu installieren.

Aufruf:

Register	Inhalt/Bedeutung
CS:DX	Zeiger auf erstes Byte nach resident zu machendem Bereich

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Die Ausführung eines Interrupts 27h während eines Programms führt dazu, dass das Programm beendet und der durch CS:DX definierte Speicherbereich resident gemacht wird. CS muss die Segmentadresse des PSP (der Wert von DS und ES vor Programmstart) enthalten. Der resident gemachte Speicherbereich wird durch weitere Programme nicht überladen.

Im Gegensatz zum Interrupt 27h ist der DOS-Funktionsaufruf "Prozess beenden und resident bleiben" (31h) die bessere Methode für resident zu verbleibende Programme, da hier

- eine Rückkehrinformation übergeben werden kann und
- der resident gemachte Speicherbereich grösser als 64 KByte sein kann.

Hinweise:

- Dieser Interrupt muss nicht unbedingt von EXE-Programmen (am oberen Speicherende geladen) benutzt werden.
- Dieser Interrupt stellt die Interruptvektoren 22h, 23h und 24h auf dieselbe Art und Weise wie Interrupt 20h wieder her. Deshalb kann dieser Interrupt nicht zur Installation von dauerhaft residenten CTRL-PAUSE-Behandlungsroutinen oder Behandlungsroutinen für kritische Fehler benutzt werden.
- Die maximale Speichergrösse für Programme, die resident bleiben sollen, beträgt für den Interrupt 27h 64K Byte.
- Der Speicher kann effektiver genutzt werden, wenn der Block, der eine Kopie der Umgebung enthält, vor Beendigung zurückgewiesen wird. Dazu ist ES mit der Segmentadresse, die im PSP (Offset 2Ch) enthalten ist, zu laden und die DOS-Funktion "Zugewiesenen Speicher freigeben" (49h) aufzurufen.
- Der DOS-Funktionsaufruf "Prozess beenden" (4Ch) erlaubt das Beenden eines Programms mit Übermittlung eines Rückgabekodes an DOS. Dieser kann innerhalb der Stapelverarbeitung verwendet werden.

6. Bei Anwendung dieses Interrupts werden keine Dateien, die eröffnet sind, geschlossen.

8.9. MULTIPLEX-INTERRUPT

INT2F

Funktion:

Es wird hiermit eine allgemeine Schnittstelle zwischen 2 Prozessen definiert. Der Interrupt 2Fh ist ein für eine spezifische Anwendung benutzter Interrupt, um spezielle Funktionen und Parameter zu definieren.

Aufruf:

Register	Inhalt/Bedeutung
AH	Multiplex-Nummer
AL	Funktionsnummer

Jeder Multiplex-Interruptbehandlungsroutine wird eine spezielle Multiplexnummer zugewiesen. Die Multiplexnummer ist in Register AH festgelegt. Die genaue Funktion, die die Behandlungsroutine ausführen soll, wird in Register AL festgelegt. Weitere benötigte Parameter werden in anderen Registern übergeben.

Bemerkung:

Die Behandlungsroutinen werden an den Interrupt 2Fh angekettet. Es gibt keine festgelegte Richtlinie für die Zuweisung einer Multiplexnummer an eine Behandlungsroutine. Es muss genau eine Nummer festgelegt werden. Um Konflikte bei der Verwendung gleicher Multiplexnummern zu vermeiden, sollten diese in den Nutzerprogrammen austauschbar sein.

Die Multiplexnummern (AH) 00h bis BFh sind für das Betriebssystem reserviert.

Der Nutzer sollte die Multiplexnummern C0h bis FFh benutzen.

Die Funktionen mit AL=F0h bis AL=FFh sind bei allen Multiplexnummern für das Betriebssystem reserviert.

8.9.1. MULTIPLEX-INTERRUPT PRINT

INT2F: 01h

Die folgende Tabelle enthält die Funktionscodes, die in AL spezifiziert werden, um den residenten Teil von PRINT aufzufordern, eine spezielle Funktion auszuführen:

Funktionskode	Beschreibung
00	Ermittlung des installierten Status
01	Submit-Datei
02	Dateiabbruch
03	Alle Dateien abbrechen
04	Status
05	Statusende

*** DOS-Interrupts ***

Der Funktionsaufruf "Ermittlung des installierten Status" (AL=00h) muss bei allen Multiplex-Interruptbehandlungsroutinen definiert sein. Er wird durch den Nutzer verwendet, um festzustellen, ob die Behandlungsroutine bereits vorhanden ist. Bei Rückkehr wird in AL der installierte Status angezeigt.

8.9.1.1. Ermittlung des installierten Status INT2F: 01h 00h

Funktion:
Ermittlung des installierten Status.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
AL	00h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	00h Ist nicht installiert, kann installiert werden
	01h Ist nicht installiert, kann nicht installiert werden
	FFh Ist bereits installiert

8.9.1.2. Submit-Datei INT2F: 01h 01h

Funktion:
Festlegen der zu druckenden Datei.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
AL	01h
DS:DX	Zeiger auf Submit-Paket

Ein Submit-Paket enthält ein Kennzeichen (1 Byte=00h) und einen Zeiger auf die ASCII-Zeichenfolge (1 DWORD). Das Kennzeichen ist abhängig vom Betriebssystem. Die ASCII-Zeichenfolge muss Laufwerk, Pfad und Dateibezeichnung der zu druckenden Datei enthalten. Die Dateibezeichnung kann keine globalen Dateibezeichner enthalten.

*** DOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

8.9.1.3. Dateiabbruch

INT2F: 01h 02h

Funktion:

Entfernen einer Datei aus der Druckwarteschlange.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
AL	02h
DS:DX	Zeiger auf ASCII-Zeichenfolge

Die ASCII-Zeichenfolge enthält die abzubrechenden Dateien. Globale Dateibezeichner sind in der Dateibezeichnung erlaubt.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

8.9.1.4. Alle Dateien abbrechen

INT2F: 01h 03h

Funktion:

Entfernen aller Dateien aus der Druckwarteschlange.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
AL	03h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

8.9.1.5. Status

INT2F: 01h 04h

Funktion:

Wiedergabe des Zustandes der Druckwarteschlange.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
AL	04h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
DS:DI	Zeiger auf Druckwarteschlange
DX	Fehlerwert

Bei Rückkehr enthält DX einen Fehlerwert. Der Fehlerwert ist eine Anzahl von aufeinanderfolgenden Störungen von PRINT, die während der Ausgabe des letzten Zeichens aufgetreten sind. DS:SI zeigt auf die Druckwarteschlange. Die Druckwarteschlange enthält eine Folge von Dateinameeintritten, die jeweils 64 Byte lang ist. Der erste Eintrittspunkt in der Warteschlange ist die Datei, die gerade gedruckt wird. Das Ende der Warteschlange ist durch 00h gekennzeichnet.

Bemerkung:

Dieser Aufruf verändert die gesamte Druckwarteschlange nicht, so dass sie geprüft werden kann. Durch Auswahl einer anderen Funktion können die Aufträge aus der Warteschlange entfernt werden.

8.9.1.6. Statusende

INT2F: 01h 05h

Funktion:

Verlassen der Druckwarteschlange.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h
AL	05h

*** DOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AX	01h Ungültige Funktionsnummer
	02h Datei nicht gefunden
	03h Pfad nicht gefunden
	04h Zu viele eröffnete Dateien
	05h Zugriff abgelehnt
	08h Warteschlange voll
	09h Besetzt
	0Ch Name zu lang
	0Fh Ungültiges Laufwerk

8.9.2. MULTIPLEX-INTERRUPT ASSIGN

INT2F: 02h 00h

Funktion:

Ermittlung des installierten Status.

Aufruf:

Register	Inhalt/Bedeutung
AH	02h
AL	00h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	00h Ist nicht installiert, kann installiert werden
	01h Ist nicht installiert, kann nicht installiert werden
	FFh Ist bereits installiert

8.9.3. MULTIPLEX-INTERRUPT SHARE

INT2F: 10h 00h

Funktion:

Ermittlung des installierten Status.

Aufruf:

Register	Inhalt/Bedeutung
AH	10h
AL	00h

*** DOS-Interrupts ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	00h Ist nicht installiert, kann installiert werden
	01h Ist nicht installiert, kann nicht installiert werden
	FFh Ist bereits installiert

8.9.4. MULTIPLEX-INTERRUPT APPEND

INT2F: 0B7h

Die folgende Tabelle enthält die Funktionscodes, die in AL spezifiziert werden, um den residenten Teil von APPEND aufzufordern, eine spezielle Funktion auszuführen:

Funktionskode	Beschreibung
00	Ermittlung des installierten Status
01	Ausgabe Fehlertext und Programm beenden
02	AX auf 0FFFFh setzen
03	Ermitteln der Adresse der APPEND-INT21-Routine
04	Zeiger auf APPEND-Liste
16	Ermittlung der APPEND-Version

8.9.4.1. Ermittlung des installierten Status

INT2F: 0B7h 00h

Funktion:

Ermittlung des installierten Status.

Aufruf:

Register	Inhalt/Bedeutung
AH	0B7h
AL	00h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	00h Ist nicht installiert, kann installiert werden
	01h Ist nicht installiert, kann nicht installiert werden
	FFh Ist bereits installiert

8.9.4.2. Ausgabe Fehlertext und Programm beenden INT2F: 0B7h 01h

Funktion:

Ausgabe eines Fehlertextes und Abbruch des Nutzerprogrammes mit Rückgabekode 01.

Aufruf:

Register	Inhalt/Bedeutung
AH	0B7h
AL	01h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Der ausgegebene Fehlertext weist auf eine falsche APPEND-Version hin.

8.9.4.3. AX auf 0FFFFh setzen

INT2F: 0B7h 02h

Funktion:

Register AX wird auf 0FFFFh gesetzt.

Aufruf:

Register	Inhalt/Bedeutung
AH	0B7h
AL	02h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AX	0FFFFh

8.9.4.4. Ermitteln der Adresse der APPEND-INT21-Routine

INT2F: 0B7h 03h

Funktion:

Es wird die Adresse der von APPEND unterstützten INT21-Routine zurückgeliefert.

Aufruf:

Register	Inhalt/Bedeutung
AH	0B7h
AL	03h
ES:DI	Zeiger auf Routine des aktuellen INT21

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
ES:DI	Zeiger auf APPEND-INT21-Routine

8.9.4.5. Zeiger auf APPEND-Liste

INT2F: 0B7h 04h

Funktion:

Es wird ein Zeiger auf die aktuelle APPEND-Liste geliefert.

Aufruf:

Register	Inhalt/Bedeutung
AH	0B7h
AL	04h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
ES:DI	Zeiger auf APPEND-Liste

8.9.4.6. Ermittlung der APPEND-Version

INT2F: 0B7h 10h

Funktion:

Es können die aktuelle APPEND-Version sowie die installierten Schalter ermittelt werden.

Aufruf:

Register	Inhalt/Bedeutung
AH	0B7h
AL	10h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AX	installierte Schalter
BX	0000h
CX	0000h
DX	APPEND-Version x.y (DH=x, DL=y)

In AX sind die installierten Schalter eingetragen. Dabei bedeutet

- Bit 15 von AX : Schalter X
- Bit 14 von AX : Schalter E

8.9.5. Beispiel für eine 2Fh-Behandlungsroutine

```

mult_nr  db    x           ;spezielle Multiplexnummer
int_alt  dd    0           ;Speichern Interrupt-2Fh-Adresse alt

int_2f_1:
  cmp    ah,mult_nr       ;Vergleich Behandlungsroutine mit
                          ;eingestellter Multiplexnummer
  je     routine          ;Übergang zur nächsten
  jmp    int_alt          ;Behandlungsroutine des INT 2Fh

routine:
  cmp    al,0F8h         ;reservierte Funktion ?
  jb     funktion        ;nein
  iret                    ;bei reservierter Funktion
    
```

*** DOS-Interrupts ***

```
funktion:
  or  al,al
  jne sonst_fkt      ;weitere Funktionen ausser 00h
  mov al,0FFh       ;ist bereits installiert
  iret
```

```
sonst_fkt:
:
:
```

Installation der Behandlungsroutine

```
mov  ah,mult_nr
xor  al,al          ;Ermittlung des installierten Status
int  2Fh
or   al,al
jz   inst          ;Installieren ist möglich

inst_n:            ;Installieren ist nicht möglich
:                 ;entsprechende Behandlungsroutine
:
inst:
mov  ax,352Fh
int  21h          ;Abfrage Interruptvektor 2Fh
mov  word ptr int_alt+2,es ;Eintragen Adresse alter INT 2Fh
mov  word ptr int_alt,bx
mov  ax,252Fh
mov  dx,offset int_2f_1 ;Adresse Interrupt 2Fh
int  21h          ;Setzen Interruptvektor 2Fh
:
:
```

8.10. Reservierte Interrupts

Die Interrupts 28h bis 2Eh und 30h bis 3Fh sind für DOS reserviert.

9. DOS-Funktionsaufrufe

Das Betriebssystem bietet umfangreiche Funktionsaufrufe für Tastatureingaben (mit und ohne Echo sowie CTRL-C-Erkennung), für Konsol- und Drückerausgaben, für das Erstellen von Dateisteuerblättern, für die Speicherverwaltung, Datums- und Zeitfunktionen sowie für eine Vielzahl von Disketten-, Inhaltsverzeichnis- und Dateibearbeitungsfunktionen.

9.1. DOS-Funktionen nach Funktionsgruppen sortiert

9.1.1. Ein- und Ausgabe für zeichenorientierte Standardgeräte

Durch diese DOS-Funktionen werden alle Ein- bzw. Ausgaben von bzw. zu zeichenorientierten Standardgeräten (Konsole, Drucker, serielle Geräte) behandelt.

Im allgemeinen handelt es sich bei den Standardgeräten um folgende Zuordnung:

Standardeingabegerät	-	Tastatur	(CON)
Standardausgabegerät	-	Bildschirm	(CON)
Standardhilfsgerät	-	serielle Schnittstelle	(AUX)
Standarddrucker	-	Drucker	(PRN)

Wenn ein Programm diese DOS-Funktionen anfordert, können dessen Ein- und Ausgaben umgeleitet sein.

Nachfolgend sind sämtliche DOS-Funktionen dieser Funktionsgruppe aufgeführt:

01h	Konsol-Eingabe mit Echo	Einlesen eines Zeichens vom Standardeingabegerät und Ausgabe auf Standardausgabegerät
02h	Konsol-Ausgabe	Ausgabe eines Zeichens auf Standardausgabegerät
03h	Hilfeingabe	Einlesen eines Zeichens vom Standardhilfsgerät
04h	Hilfsausgabe	Ausgabe eines Zeichens auf Standardhilfsgerät
05h	Ausgabe auf Listgerät	Ausgabe eines Zeichens auf Standarddrucker
06h	Direkte Konsol-Ein-/Ausgabe	Einlesen eines Zeichens vom Standardeingabegerät und Ausgabe auf Standardausgabegerät
07h	Direkte Konsol-Eingabe ohne Echo	Einlesen eines Zeichens vom Standardeingabegerät ohne Ausgabe auf Standardausgabegerät und CTRL-C-Erkennung.
08h	Konsol-Eingabe ohne Echo	Einlesen eines Zeichens vom Standardeingabegerät mit CTRL-C-Erkennung
09h	Zeichenkettenausgabe	Ausgabe einer mit "Ø" abgeschlossenen Zeichenkette auf Standardausgabegerät
0Ah	Eingabe Konsol-Puffer	Eingabe einer Zeichenkette in den Konsol-Puffer

*** DOS-Funktionsaufrufe ***

0Bh	Konsol-Status	Prüfung des Standardeingabegerätes, ob ein Zeichen vorhanden ist
0Ch	Löschen des Konsol-Puffers und Aufruf einer Standardeingabe	Löschen des Konsolpuffers mit anschliessendem Aufruf einer DOS-Funktion (01h, 06h, 07h, 08h, 0Ah)

7.1.2. Speicherbehandlung

Das Betriebssystem verwaltet einen Steuerblock über den zugewiesenen Speicher und beinhaltet Angaben über die Grösse des Speicherbereiches, den Namen des Prozesses, an den der Bereich vergeben wurde und einen Zeiger auf den nächsten Speicherbereich. Wenn der Speicherbereich nicht vergeben ist, so ist er für den Nutzer verfügbar.

Nachfolgend sind sämtliche DOS-Funktionen dieser Funktionsgruppe aufgeführt:

48h	Speicher zuweisen	Reservierung von Systemspeicher
49h	Zugewiesenen Speicher freigeben	Freigeben von reservierten Systemspeicher
4Ah	Zugewiesenen Speicherblock verändern	Verändern der Grösse eines reservierten Systemspeicherbereiches

7.1.3. Prozessbehandlung

Im Betriebssystem werden verschiedene Funktionen zum Laden, Ausführen und Beenden von Programmen realisiert. Durch Nutzerprogramme können diese Funktionen verwendet werden.

Nachfolgend sind sämtliche DOS-Funktionen dieser Funktionsgruppe aufgeführt:

00h	Programm beenden	Beenden eines laufenden Programmes
31h	Prozess beenden und resident bleiben	Beenden eines laufenden Programmes, das jedoch im Speicher verbleibt
4B00h	Programm laden und ausführen	Laden und Ausführen eines Programmes
4B03h	Programm laden	Laden eines Programmes
4Ch	Prozess beenden	Beenden eines Programmes mit Übergabe eines Rückgabekodes
4Dh	Beendigungskode liefern	Ermitteln des Beendigungskodes, den ein aufgerufenes Programm an den Nutzer zurückgibt
62h	Ermitteln aktueller PSP	Ermitteln der aktuellen PSP-Adresse

7.1.4. Datei- und Verzeichnisbehandlung

7.1.4.1. Handler

Zum Anlegen oder Eroffnen einer Datei werden dem Betriebssystem eine vollstandige ASCII-Z-Dateispezifikation und die gewunschten Dateiattribute ubergeben. Durch das System wird eine 16-bit-Zahl zuruckgegeben, die Handler genannt wird.

Fur viele nachfolgende DOS-Funktionen ist dann nur noch diese Handlernummer erforderlich, wenn mit der entsprechenden Datei gearbeitet werden soll.

Ein Handler kann aber auch fur Gerate stehen. Im Betriebssystem sind 5 Handler fest vergeben. Diese sind immer eroffnet, d.h., sie brauchen vor Benutzung nicht extra eroffnet werden.

Nachfolgend sind diese 5 Handler aufgefuhrt:

Hex-Wert	Beschreibung
0000h	Standardeingabeeinheit; die Eingabe kann umgeleitet werden.
0001h	Standardausgabeeinheit; die Ausgabe kann umgeleitet werden.
0002h	Standardfehlerausgabeeinheit; die Ausgabe kann nicht umgeleitet werden.
0003h	Standardhilfeeinheit
0004h	Standarddruckeinheit

Bei der Vergabe eines Handlers durch das Betriebssystem wird die erste freie zur Verfugung stehende Handlernummer genommen.

Einem Programm steht die definierte Anzahl (FILES-Kommando) von gleichzeitig eroffneten Handler zur Verfugung. Dabei ist zu beachten, dass die 5 vordefinierten Handler darin eingeschlossen sind.

Durch die DOS-Funktion 46h kann jeder der 5 vordefinierten Handler auf einen anderen Handler gesetzt werden.

7.1.4.2. Dateiabhangige Funktionsaufrufe

7.1.4.2.1. Aufruf uber Handler

Das Betriebssystem behandelt eine Datei wie eine Folge von Bytes. Bei der Bearbeitung einer Datei ist keine Satzstruktur oder Zugriffstechnik erforderlich. Das Lesen bzw. Schreiben erfordert lediglich die Handlernummer, einen Zeiger auf einen Datenpuffer und eine Anzahl uber die zu lesenden bzw. zu schreibenden Bytes.

Nachfolgend sind samtliche DOS-Funktionen dieser Funktionsgruppe aufgefuhrt:

3Ch	Handler erstellen	Anlegen einer neuen Datei/ Handler
3Dh	Handler eroffnen	Eroffnen einer Datei/Handler
3Eh	Handler schliessen	Schliessen einer Datei/Handler

*** DOS-Funktionsaufrufe ***

3Fh	Lesen von Datei/Einheit	Lesen von Datei oder Gerät
40h	Schreiben auf Datei/Einheit	Schreiben auf Datei oder Ausgabe auf ein Gerät
42h	Verschieben des Lese-/Schreibzeigers	Verschieben des Lese-/Schreibzeigers innerhalb einer Datei
45h	Händler duplizieren	Anlegen eines neuen Händlers, der sich auf die gleiche Datei wie der alte Händler bezieht
46h	Händlerduplikat erstellen	Neuer Händler bezieht sich auf die gleiche Datei wie der bereits existierende Händler
5Ah	Einzigartige Datei erstellen	Anlegen einer Datei mit einem willkürlichen Namen
5Bh	Neuen Händler erstellen	Anlegen einer neuen Datei, die noch nicht vorhanden ist
67h	Setzen der Händleranzahl	Unterstützung von mehr als 20 Händlern pro Prozess
68h	Übergabe an Händler	Sicherung der Daten für einen Händler

2.1.4.2.2. Aufruf über FCB

Im Gegensatz zur Arbeit mit Händlern benötigen nachfolgende DOS-Funktionen zur Abarbeitung einen Block von Daten zur Steuerung der Dateiarbeit. Dieser Block wird FCB genannt und muss durch den Nutzer aufbereitet werden.

Der Aufruf einer DOS-Funktion über FCB lässt sowohl einen standardmässigen als auch einen erweiterten FCB zu.

Im einzelnen handelt es sich um die DOS-Funktionen:

0Fh	Datei eröffnen
10h	Datei schliessen
11h	Suche nach dem ersten Eintrag
12h	Suche nach dem nächsten Eintrag
13h	Datei löschen
14h	Sequentielles Lesen
15h	Sequentielles Schreiben
16h	Datei erstellen
17h	Datei umbenennen
21h	Wahlfreies Lesen
22h	Wahlfreies Schreiben
23h	Dateigrösse
24h	Aktualisieren relative Nummer des Datensatzes
27h	Wahlfreies Blocklesen
28h	Wahlfreies Blockschreiben

2.1.4.3. Funktionsaufrufe für gemeinsamen Dateizugriff (Datei-Sharing)

Durch Datei-Sharing kann mehr als ein Prozess auf eine Datei zugreifen.

Datei-Sharing funktioniert nur dann, wenn das Programm SHARE geladen und abgearbeitet wurde und somit resident den gemeinsamen Dateizugriff unterstützt.

*** DOS-Funktionsaufrufe ***

Nachfolgend sind sämtliche DOS-Funktionen dieser Funktionsgruppe aufgeführt:

3Dh	Händler eröffnen	Eröffnen einer Datei mit Modus für Datei-Sharing
440Bh	Verändern des Zählers für Wiederholungsversuche bei gemeinsamer Dateinutzung	Festlegen der Anzahl von Wiederholungen, bevor die Fehlerbehandlung eintritt
5Ch	Dateizugriff verbieten/erlauben	Unterbinden bzw. Freigabe des Zugriffes auf einen Bereich einer Datei

9.1.4.4. Geräteabhängige Funktionsaufrufe

Die E/A-Steuerung für Geräte wird durch die DOS-Funktion 44h vorgenommen. Sie enthält Spezifikationen für die verschiedensten Geräteforderungen.

Nachfolgend sind sämtliche DOS-Funktionen dieser Funktionsgruppe aufgeführt:

4400h	Abfrage von Händlerinformationen
4401h	Setzen von Händlerinformationen
4402h	Lesen vom Zeichengerät
4403h	Schreiben auf Zeichengerät
4404h	Lesen vom Blockgerät
4405h	Schreiben auf Blockgerät
4406h	Eingabestatus abfragen
4407h	Ausgabestatus abfragen
4408h	Abfrage, ob ein einzelnes Blockgerät auswechselbar ist
440Ch	Art und Weise der E/A-Steuerung für Händler
440Dh	Art und Weise der E/A-Steuerung für Blockgeräte
440Eh	Abfrage des logischen Laufwerkes
440Fh	Setzen des logischen Laufwerkes

9.1.4.5. Verzeichnisabhängige Funktionsaufrufe

Das Wurzelverzeichnis hat auf der Diskette eine feste Anzahl von Eintragungen. Für Festplatten ist die Anzahl von der Größe der DOS-Partition abhängig.

Ein Unterverzeichnis wird wie eine Datei, jedoch mit einem bestimmten Kennzeichen eingetragen. Die Tiefe der Verzeichnisstruktur ist durch die Anzahl der Eintragungen und durch Länge der Pfadbezeichnung (max. 64 Zeichen) begrenzt.

Nachfolgend sind sämtliche DOS-Funktionen dieser Funktionsgruppe aufgeführt:

39h	Verzeichnis erstellen
3Ah	Verzeichnis löschen
3Bh	Aktuelles Verzeichnis wechseln
41h	Löschen einer Datei aus dem angegebenen Verzeichnis
43h	Dateiattribut abfragen/verändern

*** DOS-Funktionsaufrufe ***

47h Aktuelles Verzeichnis holen
4Eh Erste Dateieintragung finden
4Fh Nächste Dateieintragung finden
56h Datei umbenennen
5700h Datum und Uhrzeit einer Datei abfragen
5701h Datum und Uhrzeit einer Datei setzen

7.1.5. Netzwerkaufrufe

Für eine Netzwerkbearbeitung stellt das Betriebssystem einige DOS-Funktionen zur Verfügung.

4409h Abfrage, ob ein logisches Gerät lokal oder entfernt ist
440Ah Abfrage, ein Händler lokal oder entfernt ist
5E00h Maschinename abfragen
5E01h Maschinename setzen
5E02h Drucker-Setup setzen
5E03h Drucker-Setup abfragen
5F02h Eintragungen der Zuweisungstabelle abfragen
5F03h Zuweisung eines Gerätes
5F04h Zuweisung aufheben

7.1.6. Sonstige Systemaufrufe

Die übrigen Funktionsaufrufe beinhalten Funktionen für die Arbeit mit Laufwerken, mit Datum und Zeit, mit Fehlern, mit Zeichensätzen, mit Tabellen u.a.m.

Nachfolgend sind sämtliche DOS-Funktionen dieser Funktionsgruppe aufgeführt:

0Dh Diskette rücksetzen
0Eh Auswahl Bezugs-Laufwerk
19h Abfrage Bezugs-Laufwerk
1Ah Setzen DTA
1Bh Informationen der Zuordnungstabelle
1Ch Informationen der Zuordnungstabelle für bestimmtes Laufwerk
25h Setzen Interruptvektor
26h Erstellen neuer PSP
29h Dateibezeichnung analysieren
2Ah Datum abfragen
2Bh Datum setzen
2Ch Zeit abfragen
2Dh Zeit setzen
2Eh Prüfllesen setzen/rücksetzen
2Fh DTA holen
30h DCP-Versionsnummer abfragen
33h Prüfung auf CTRL-C
35h Interruptvektor abfragen
36h Organisation des Datenträgers abfragen
38h Landesspezifische Informationen abfragen/setzen

54h Status Prüfllesen abfragen
59h Erweiterte Fehlermeldung

*** DOS-Funktionsaufrufe ***

65h Ermittlung einer erweiterten Landesinformation
66h Ermitteln bzw. Setzen der Zeichensatztabelle

9.1.7. Reservierte DOS-Funktionsaufrufe

Im Betriebssystem sind einige Funktionen für dessen internen Gebrauch reserviert. Die Schnittstellen dieser Funktionen werden nicht weiter beschrieben.

Solche Funktionen sind z.B.:

1Fh DPB für Standard-Laufwerk holen
32h DPB für bestimmtes Laufwerk holen
34h Adresse des kritischen Flags holen
37h Parametertrennzeichen abfragen/setzen
50h Aktuellen PSP setzen
51h Aktuellen PSP holen
52h Adresse der DOS-Informationstabelle holen
53h Füllen DPB vom BPB
55h Neues Programmsegment erstellen
58h Selektor für Speicherzuweisen abfragen/setzen

9.2. DOS-Funktionen

9.2.1. Allgemeine Grundsätze

9.2.1.1. Aufruf der DOS-Funktionen

Die Funktionsaufrufe erfordern bestimmte Parameter, die ihnen in Registern übergeben werden.

Nach dem Setzen aller notwendigen Registerwerte kann die DOS-Funktion auf eine der nachfolgenden Arten aufgerufen werden:

1. Die Funktionsnummer der auszuführenden DOS-Funktion wird in Register AH eingestellt und ein Interrupt 21h ausgeführt. Diese Methode ist im allgemeinen für die Nutzung der DOS-Funktionen zu verwenden.
2. Die Funktionsnummer der auszuführenden DOS-Funktion wird in Register AH eingestellt. Durch einen langen Unterprogrammaufruf (CALL FAR) an Offset 50h innerhalb des aktuellen PSP wird die DOS-Funktion aufgerufen.

9.2.1.2. Interner Stapel

Das Betriebssystem verwendet für die Ausführung der Interrupt-routinen und somit auch der DOS-Funktionen einen internen Stapel.

Die Register des Nutzerprogramms bleiben erhalten, wenn sie nicht gerade der Parameterübergabe bzw. -rückgabe dienen. Der Nutzerstapel muss gross genug sein, um das Interruptsystem anzupassen. Es ist ausreichend, wenn der Nutzerstapel zusätzlich zu seinem benötigten Bereich 200h Byte zur Verfügung stellt.

9.2.1.3. Dateispezifikation

Das Betriebssystem identifiziert jede Datei durch einen eindeutigen Namen. Diese Dateispezifikation setzt sich aus den Teilen

- Laufwerkangabe,
 - Pfad und
 - Dateibezeichnung
- zusammen.

Die Dateibezeichnung steht für den Dateinamen und den Dateityp.

Angabe des Laufwerkes:

Die Definition eines Laufwerkes erfolgt durch einen Laufwerkbuchstaben mit anschliessendem Doppelpunkt. Als Laufwerkangabe sind die Buchstaben A bis Z zulässig, wobei kein Unterschied zwischen Gross- und Kleinbuchstaben besteht.

Hinweis:

Der letzte für das Betriebssystem gültige Laufwerkbuchstabe kann durch das Kommando LASTDRIVE bestimmt werden.

Wenn der Laufwerkbuchstabe nicht angegeben wird, so wird das Standardlaufwerk verwendet.

Angabe des Pfades:

Die Länge einer Pfadangabe darf 63 Zeichen nicht überschreiten.

Die folgenden Zeichen sind ebenso ungültig wie die ASCII-Zeichen mit der Kodierung 00h bis 1Fh:

" / [] : | \ < > + = ; ,

Angabe des Dateinamens:

Die Länge des Dateinamens kann 1 bis 8 Zeichen betragen.

Die folgenden Zeichen sind ebenso ungültig wie die ASCII-Zeichen mit der Kodierung 00h bis 1Fh:

" / [] : | \ < > + = ; , .

Angabe des Dateityps:

Die Länge des Dateityps kann 1 bis 3 Zeichen betragen. Der Dateityp ist durch "." vom Dateinamen getrennt.

Die folgenden Zeichen sind ebenso ungültig wie die ASCII-Zeichen mit der Kodierung 00h bis 1Fh:

" / [] : | \ < > + = ; , .

Im Betriebssystem besitzen bestimmte Bezeichnungen eine besondere Bedeutung und können daher als Name für eine neuerstellte Datei nicht verwendet werden.

Für Systemeinheiten sind folgende Bezeichnungen reserviert:

CON, AUX, COM1, COM2, LPT1, PRN, LPT2, LPT3 sowie NUL.

Diese Bezeichnungen können anstelle der Dateispezifikation verwendet werden.

***** DOS-Funktionsaufrufe *****

In der Angabe des Dateinamens und Dateityps können die mehrdeutigen Zeichen "?" und "*", die globale Dateibezeichner genannt werden, enthalten sein.

9.2.1.4. ASCIIZ-Zeichenfolgen

Mehrere DOS-Funktionen benötigen zu ihrer Ausführung eine ASCIIZ-Zeichenfolge als Eingabeparameter. Eine ASCIIZ-Zeichenfolge besteht aus einer ASCII-Zeichenfolge mit dem Endekennzeichen 00h.

So enthält z.B. die ASCIIZ-Dateispezifikation die ASCII-Zeichenfolge mit

- Laufwerkangabe,
- Pfad und
- Dateibezeichnung.

Z.B. C:\PFAD1\PFAD2\DATEI.TYP gefolgt von 00h.

Die maximale Grösse einer ASCIIZ-Zeichenfolge beträgt 128 Byte (einschliesslich Laufwerk, Doppelpunkt und 00h).

Für die Trennung der Verzeichnispfade ist der inverse Schrägstrich ("\") zugelassen.

9.2.1.5. Netzwerkpfade

Im Betriebssystem akzeptieren verschiedene DOS-Funktionsaufrufe als Eingabe einen Netzwerkpfad. Die Netzwerk-Funktionen werden aber nur realisiert, wenn das Netzwerkprogramm geladen wurde.

Ein Netzwerkpfad besteht aus einer ASCII-Zeichenfolge mit dem Endekennzeichen 00h.

Die ASCII-Zeichenfolge enthält

- einen Gerätenamen,
- einen Pfad und
- eine Dateibezeichnung.

Sie kann keine Laufwerkangabe enthalten.

Z.B. \\GERAET1\PFAD1\PFAD2\DATEI1.TYP gefolgt von 00h.

Alle DOS-Funktionsaufrufe, die einen ASCIIZ-Pfad als Eingabeparameter zulassen, akzeptieren auch einen Netzwerkpfad.

Ausgenommen davon sind die DOS-Funktionen "Aktuelles Verzeichnis wechseln" (3Bh) und "Erste Dateieintragung finden" (4Eh).

9.2.1.6. Netzwerk-Zugriffsrechte

Die Beschreibung von einigen DOS-Funktionen enthält eine Bemerkung zu Netzwerk-Zugriffsrechten.

Die dort angeführte Information gibt die notwendigen Bedingungen für den Zugriff auf eine Datei an.

*** DOS-Funktionsaufrufe ***

Wenn zum Beispiel die DOS-Funktion "Neue Händler erstellen" (5Bh) ausgeführt werden soll, muss der Zugriff für Lesen/Schreiben/ Erstellen oder für Schreiben/Erstellen auf das Verzeichnis erlaubt sein, um die neue Datei zu erstellen. Ist dagegen nur Lesen oder Schreiben und kein Erstellen erlaubt, so kann keine neue Datei in dem Verzeichnis erstellt werden.

9.2.1.7. Fehlerrückgabeinformation

Viele DOS-Funktionen geben dem aufrufenden Programm eine Meldung darüber zurück, ob die Funktion erfolgreich oder nicht erfolgreich ausgeführt wurde.

Die FCB-Funktionsaufrufe melden einen Fehler mit AL=FFh zurück. Eine grössere Anzahl nutzt das C-Flag zur Fehlerkennzeichnung. Bei fehlerfrei durchgeführter Funktion wird das C-Flag gelöscht (CF=0).

Wenn die aufgerufene DOS-Funktion nicht erfolgreich war, so wird das C-Flag gesetzt (CF=1). Zusätzlich steht in AX eine Meldung über den spezifischen Fehler.

Diese Meldung wird im folgenden als Fehlerkode bezeichnet.

9.2.1.7.1. Der Fehlerkode

In der folgenden Tabelle sind die Fehlerkodes und ihre Bedeutung angeführt:

Fehlercode	Bedeutung
1	01h Ungültige Funktionsnummer
2	02h Datei nicht gefunden
3	03h Pfad nicht gefunden
4	04h Zu viele eröffnete Dateien (keine Dateinummern mehr frei)
5	05h Zugriff abgelehnt
6	06h Ungültige Händlernummer
7	07h Speichersteuerblöcke zerstört
8	08h Nicht genügend Speicherplatz
9	09h Ungültige Speicherblockadresse
10	0Ah Ungültige Umgebung
11	0Bh Ungültiges Format
12	0Ch Ungültiger Zugriffskode
13	0Dh Ungültige Daten
14	0Eh Reserviert
15	0Fh Angabe eines ungültigen Laufwerks
16	10h Versuch, das aktuelle Verzeichnis zu löschen
17	11h Nicht dieselbe Einheit
18	12h Keine weiteren Dateien mehr
19	13h Es wurde versucht, auf schreibgeschützte Diskette zu schreiben
20	14h Unbekannte Einheit
21	15h Laufwerk nicht bereit
22	16h Unbekanntes Kommando

*** DOS-Funktionsaufrufe ***

Fehlercode	Bedeutung
23	17h Datenfehler (CRC-Fehler)
24	18h Falsch geforderte Strukturlänge
25	19h Positionierfehler
26	1Ah Unbekannter Typ des Mediums
27	1Bh Sektor nicht gefunden
28	1Ch Papierende Drucker
29	1Dh Schreibfehler
30	1Eh Lesefehler
31	1Fh Allgemeine Störung
32	20h Sharing-Verletzung
33	21h Verriegelungsverletzung
34	22h Ungültiger Diskettentausch
35	23h FCB nicht verfügbar
36	24h Sharing-Pufferüberlauf
37-49	25h-31h Reserviert
50	32h Netzwerkanforderung ist nicht unterstützt
51	33h Entfernter Computer hört nicht
52	34h Duplikatname im Netzwerk
53	35h Netzwerkname nicht gefunden
54	36h Netzwerk ist besetzt
55	37h Netzwerkeinheit existiert nicht länger
56	38h Netzwerk-BIOS-Kommando übersteigt das Limit
57	39h Hardwarefehler im Netzwerk-Adapter
58	3Ah Unkorrekte Antwort vom Netzwerk
59	3Bh Unerwarteter Netzwerkfehler
60	3Ch Unverträglicher entfernter Adapter
61	3Dh Druckwarteschlange voll
62	3Eh Nicht genügend Platz für Druckdateien
63	3Fh Druckdatei ist gelöscht
64	40h Netzwerkname ist gelöscht
65	41h Zugriff verweigert
66	42h Netzwerkeinheitentyp unkorrekt
67	43h Netzwerkname nicht gefunden
68	44h Netzwerkname übersteigt das Limit
69	45h Netzwerk-BIOS-Abschnitt übersteigt das Limit
70	46h Zeitlich unterbrochen
71	47h Netzwerkanforderung nicht akzeptiert
72	48h Drucker- oder Diskettenumlenkung unterbrochen
73-79	49h-4Fh Reserviert
80	50h Datei existiert
81	51h Reserviert
82	52h Keine Verzeichniseintragung möglich
83	53h Verlassen durch Interrupt 24h
84	54h Zu viele Umlenkungen
85	55h Duplikat der Umlenkung
86	56h Ungültiges Passwort
87	57h Ungültiger Parameter
88	58h Datenfehler im Netzwerk

Will der Nutzer mehr Informationen über den aufgetretenen Fehler haben, so erhält er diese durch Aufruf der DOS-Funktion "Erweiterte Fehlermeldung" (59h).

9.2.1.7.2. Der erweiterte Fehler

Die DOS-Funktion "Erweiterte Fehlermeldung" (59h) liefert den Fehlercode im Register AX und zusätzlich Informationen über den aufgetretenen Fehler an das aufrufende Programm zurück. Diese beinhaltet Angaben über die Fehlerklasse, Nutzerhinweise sowie Angaben zum Fehlerort.

9.2.1.7.2.1. Fehlerklasse

Die Fehlerklasse liefert Informationen über den Fehlertyp (z.B. Hardware, intern, System), die in der nachfolgenden Tabelle beschrieben sind:

Wert	Beschreibung
01h	Ausserhalb des Bereiches (Mediums), z.B. nicht genügend Speicherplatz, zu viele Händler oder keine Verzeichniseintragen mehr möglich
02h	Zeitweilige Situation (z.B. eine Dateiverriegelung). Es handelt sich hierbei nicht um einen "harten" Fehler, da nach einer gewissen Zeit die Fehlerbedingung wieder weg ist.
03h	Nicht erlaubter Zugriff (z.B. beim Versuch, das aktuel- le Verzeichnis zu löschen)
04h	Ein interner Fehler in der Systemsoftware (z.B. bei An- gabe eines unbekanntes Kommandos)
05h	Fehler in der Hardware (z.B. bei Diskettenoperationen)
06h	Störung in der Systemsoftware (z.B. bei fehlender oder falscher Konfigurationsdatei)
07h	Fehlerhafte Anforderungen im Nutzerprogramm (z.B. bei falschem FCB, ungültiger DOS-Funktionsnummer u.a.m.)
08h	Datei oder Einheit nicht gefunden
09h	Datei oder Einheit besitzen ungültiges Format
0Ah	Datei oder Einheit sind verriegelt
0Bh	Störung des Datenträgers (z.B. bei falscher Diskette, CRC-Fehler u.a.m.)
0Ch	Konflikt mit einer bereits existierenden Einheit (z.B. bei einer Deklaration eines bereits existierenden Ma- schinennamens).
0Dh	Fehler kann in anderer Fehlerklasse nicht eingeordnet werden.

9.2.1.7.2.2. Nutzerhinweise

Nutzerhinweise beziehen sich auf solche Fehler, die nicht dem spezifischen Fehlerkode unterstehen.

In der nachfolgenden Tabelle sind diese beschrieben.

Wert	Beschreibung
01h	DOS-Funktionen einige Male wiederholen. Anschliessend den Nutzer veranlassen, das Programm fortzusetzen oder abzubrechen.
02h	DOS-Funktionen einige Male nach einer Pause wiederholen. Anschliessend den Nutzer veranlassen, das Programm fortzusetzen oder abzubrechen.
03h	Der Nutzer soll die Eingabe wiederholen. Ein typisches Beispiel dafür ist ein falsch eingegebener Laufwerkbuchstabe oder Dateiname.
04h	Abbruch des Nutzerprogramms mit Löschen der Puffer. Das Nutzerprogramm kann nicht fortgesetzt werden; das System ist jedoch noch in der Lage, das Nutzerprogramm definiert abzubrechen.
05h	Das Nutzerprogramm ist sofort ohne Säuberung der Puffer zu beenden. Es wird nicht empfohlen, dass das Nutzerprogramm noch versucht, Dateien zu schliessen oder Indizes zu aktualisieren.
06h	Fehler ist informativ.
07h	Der Nutzer muss eine Handlung (z.B. Diskettenwechsel) ausführen und anschliessend die gleiche Funktion wiederholen.

9.2.1.7.2.3. Fehlerort

Dieser Wert liefert zusätzliche Informationen über den Bereich (serielles Gerät, Blockgerät, Netzwerk oder Speicher), in dem die Störung aufgetreten ist.

In der nachfolgenden Tabelle ist der Fehlerort beschrieben:

Wert	Beschreibung
01h	Fehler lässt sich nicht lokalisieren.
02h	Fehler in der Blockeinheit Steht in Beziehung mit direktem Zugriff auf Platte/Diskette.
03h	Steht in Beziehung mit Netzwerken.
04h	Steht in Beziehung mit seriellem, zeichenorientiertem Gerät (z.B. Drucker).
05h	Steht in Beziehung mit direktem Speicherzugriff.

9.2.2. Beschreibung der DOS-Funktionen

9.2.2.1. PROGRAMM BEENDEN

INT21: 00h

Funktion:

Es wird das laufende Programm beendet.

Aufruf:

Register	Inhalt/Bedeutung
AH	00h
CS	Zeiger auf PSP

Durch das Programm ist sicherzustellen, dass das Register CS die Segmentadresse des aktuellen PSP enthält, bevor diese Funktion aufgerufen wird.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Das Betriebssystem holt aus dem PSP die Originaladressen der Interruptvektoren 22h, 23h sowie 24h und trägt diese Werte für die entsprechenden Interruptvektoren ein. Damit soll nach Verlassen eines Programmes erreicht werden, dass durch den Nutzer veränderte Interruptvektoren wieder den Originalwert erhalten. Sämtliche Dateipuffer werden gelöscht und die in dem Prozess eröffneten Händler werden geschlossen. Dateien, die mit einem FCB eröffnet und bearbeitet wurden, müssen geschlossen werden. Es erfolgt bei ihnen keine automatische Aktualisierung der Verzeichniseintragungen.

Die weitere Steuerung wird an den INT 22h übergeben.

Bemerkung:

Dieser Aufruf führt genau dieselbe Funktion wie Interrupt 20h aus.

9.2.2.2. KONSOL-EINGABE MIT ECHO

INT21: 01h

Funktion:

Einlesen eines Zeichens von der Konsole (i.allg. Tastatur) mit Echo auf dem Bildschirm.

Aufruf:

Register	Inhalt/Bedeutung
AH	01h

*** DOS-Funktionsaufrufe ***

Durch diese DOS-Funktion wird auf die Eingabe eines Zeichens von dem Standardeingabegerät gewartet, falls dieses noch nicht bereitsteht.

Befinden sich im Konsol-Puffer zum Zeitpunkt des Aufrufs noch Zeichen, so werden diese zuerst übergeben.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	ASCII-Zeichen

Das Zeichen wird zusätzlich als Echo auf dem Standardausgabegerät ausgegeben.

Bemerkung:

Das Zeichen wird auf CTRL-C überprüft, und wenn dies der Fall ist, so wird ein Interrupt 23h ausgeführt.

Bei dem Funktionsaufruf 01h erfordert der erweiterte ASCII-Kode 2 Funktionsaufrufe. Der erste Aufruf gibt 00h zurück, womit angezeigt wird, dass der nächste Aufruf den erweiterten Kode zurückgibt.

9.2.2.3. KONSOL-AUSGABE

INT21: 02h

Funktion:

Ausgabe eines Zeichens auf die Konsole (i.allg. Bildschirm).

Aufruf:

Register	Inhalt/Bedeutung
AH	02h
DL	ASCII-Zeichen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Wenn das Zeichen ein Rückschritt (08h) ist, wird der Cursor um eine Position nach links bewegt, ohne dabei das dort stehende Zeichen zu löschen.

Ist das Zeichen ein CTRL-C, so wird nach der Ausgabe ein Interrupt 23h ausgeführt.

9.2.2.4. HILFSEINGABE

INT21: 03h

Funktion:

Einlesen eines Zeichens vom Gerät AUX (i.allg. serielle Schnittstelle).

Aufruf:

Register	Inhalt/Bedeutung
AH	03h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	ASCII-Zeichen

Bemerkung:

Es wird solange gewartet, bis ein Zeichen vom Gerät übergeben werden kann.

Dieser Funktionsaufruf gibt keinen Status- oder Fehlercode zurück.

Durch Eingabe von CTRL-C über die Konsole kann diese DOS-Funktion unterbrochen werden. Es wird der Interrupt 23h ausgeführt.

9.2.2.5. HILFSAUSGABE

INT21: 04h

Funktion:

Ausgabe eines Zeichens zum Gerät AUX (i.allg. serielle Schnittstelle).

Aufruf:

Register	Inhalt/Bedeutung
AH	04h
DL	ASCII-Zeichen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Dieser Funktionsaufruf gibt keinen Status- oder Fehlercode zurück.

*** DOS-Funktionsaufrufe ***

Durch Eingabe von CTRL-C über die Konsole kann diese DOS-Funktion unterbrochen werden. Es wird der Interrupt 23h ausgeführt.

9.2.2.6. AUSGABE AUF LISTGERAT

INT21: 05h

Funktion:

Ausgabe eines Zeichens auf den Drucker.

Aufruf:

Register	Inhalt/Bedeutung
AH	05h
DL	ASCII-Zeichen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Dieser Funktionsaufruf gibt keinen Status- oder Fehlercode zurück.

Durch Eingabe von CTRL-C über die Konsole kann diese DOS-Funktion unterbrochen werden. Es wird der Interrupt 23h ausgeführt.

9.2.2.7. DIREKTE KONSOL-EIN-/AUSGABE

INT21: 06h

Funktion:

Ein-/Ausgabe von/nach Konsole (i.allg. von Tastatur auf Bildschirm).

Aufruf:

Register	Inhalt/Bedeutung
AH	06h
DL	FFh für Konsol-Eingabe
DL	00h - FEh für Konsol-Ausgabe

Ist DL ungleich FFh, dann wird der in diesem Register stehende Kode ausgegeben.

*** DOS-Funktionsaufrufe ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
ZF	0
AL	ASCII-Zeichen
ZF	1
AL	00h es liegt kein Zeichen vor

Bemerkung:

Durch diese DOS-Funktion wird ein Zeichen, falls vorhanden, vom Standardeingabegerät übergeben. Durch Eingabe von CTRL-C über die Konsole kann diese DOS-Funktion nicht unterbrochen werden. Bei dem Funktionsaufruf 06h erfordert der erweiterte ASCII-Kode 2 Funktionsaufrufe. Der erste Aufruf gibt 00h zurück, womit angezeigt wird, dass der nächste Aufruf den erweiterten Kode zurückgibt.

9.2.2.8. DIREKTE KONSOL-EINGABE OHNE ECHO

INT21: 07h

Funktion:

Eingabe eines Zeichens von der Konsole (i.allg. Tastatur).

Aufruf:

Register	Inhalt/Bedeutung
AH	07h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	ASCII-Zeichen

Bemerkung:

Durch diese DOS-Funktion wird auf ein Zeichen vom Standardeingabegerät gewartet, ausser wenn schon ein Zeichen vorhanden ist. Es erfolgt kein Echo des Zeichens. Durch Eingabe von CTRL-C über die Konsole kann diese DOS-Funktion nicht unterbrochen werden.

9.2.2.9. KONSOLE-EINGABE OHNE ECHO

INT21: 08h

Funktion:

Eingabe eines Zeichens von der Konsole (i.allg. Tastatur).

Aufruf:

Register	Inhalt/Bedeutung
AH	08h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	ASCII-Zeichen

Bemerkung:

Durch diese DOS-Funktion wird auf ein Zeichen vom Standardeingabegerät gewartet, ausser wenn schon ein Zeichen vorhanden ist. Es erfolgt kein Echo des Zeichens.

Durch Eingabe von CTRL-C über die Konsole kann diese DOS-Funktion unterbrochen werden.

Bei dem Funktionsaufruf 08h erfordert der erweiterte ASCII-Kode 2 Funktionsaufrufe. Der erste Aufruf gibt 00h zurück, womit angezeigt wird, dass der nächste Aufruf den erweiterten Kode zurückgibt.

9.2.2.10. ZEICHENKETTENAUSGABE

INT21: 09h

Funktion:

Ausgabe einer mit "Ø" abgeschlossenen Zeichenkette auf die Konsole.

Aufruf:

Register	Inhalt/Bedeutung
AH	09h
DS:DX	Zeiger auf die Zeichenkette

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

*** DOS-Funktionsaufrufe ***

Bemerkung:

Die Zeichenkette muss durch "␣" (24h) abgeschlossen sein. Soll das Zeichen "␣" ausgegeben werden, so muss eine Sonderbehandlung erfolgen.

Jedes Zeichen der Zeichenkette wird wie bei der DOS-Funktion 02h an die Standardausgabeeinheit ausgegeben.

9.2.2.11. EINGABE KONSOL-PUFFER

INT21: 0Ah

Funktion:

Eingabe einer Zeichenkette von Konsole in den Konsol-Puffer mit Echo.

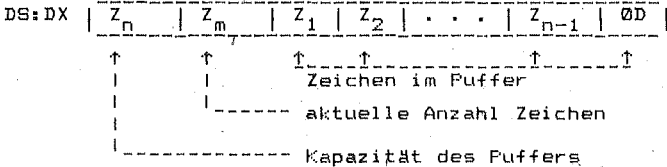
Aufruf:

Register	Inhalt/Bedeutung
AH	0Ah
DS:DX	Zeiger auf Konsol-Puffer

Bei Eintritt zeigt DS:DX auf den Konsol-Puffer. Das erste Zeichen innerhalb dieses Puffers steht auf dem dritten Byte.

Das erste Byte des Konsol-Puffers gibt die Anzahl von Zeichen an, die in den Puffer eingetragen werden können. Die Anzahl muss um 3 ($Z_n, Z_m, 0Dh$) grösser sein als die maximal einzutragende Zeichenzahl.

Das zweite Byte des Konsol-Puffers gibt den aktuellen Füllungsstand an. Dabei ist das Abschlusszeichen nicht mitgezählt. Dieses wird mit Abschluss als letztes Zeichen eingetragen.



Rückkehr:

Reg./Flag	Inhalt/Bedeutung
DS:DX	ASCII-Zeichen

Bemerkung:

Die von der Tastatur gelieferten Zeichen werden in den Puffer eingetragen.

Während der Eingabe können Rückschritt (08h) und Löschen (7Fh) zur Korrektur im Puffer verwendet werden.

Wenn der Puffer bis auf ein Zeichen unterhalb seiner Kapazität

*** DOS-Funktionsaufrufe ***

gefüllt ist, wird jedes weitere Zeichen von der Tastatur ignoriert.
Die DOS-Funktion wird mit Betätigen der Abschlusstaste beendet.

9.2.2.12. KONSOL-STATUS

INT21: 0Bh

Funktion:

Prüfung der Konsole (i.allg. Tastatur), ob ein Zeichen verfügbar ist.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Bh

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Zeichen vorhanden
AL	00h kein Zeichen vorhanden

Bemerkung:

Durch Eingabe von CTRL-C über die Konsole kann diese DOS-Funktion unterbrochen werden.

9.2.2.13. LÖSCHEN DES KONSOL-PUFFERS UND AUFRUF
EINER STANDARDEINGABE

INT21: 0Ch

Funktion:

Löschen aller Zeichen im Tastaturpuffer mit anschließendem Aufruf einer DOS-Funktion (01h, 06h, 07h, 08h, 0Ah).

Aufruf:

Register	Inhalt/Bedeutung
AH	0Ch
AL	DOS-Funktionsnummer (01h, 06h, 07h, 08h, 0Ah)

Bei Aufruf dieser DOS-Funktion wird zuerst der Tastaturpuffer gelöscht und ggf. die in AL enthaltene Funktionsnummer aufgerufen.

*** DOS-Funktionsaufrufe ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
	ohne

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h

Es wurde lediglich der Tastaturpuffer gelöscht. In AL steht eine ungültige Funktionsnummer.

9.2.2.14. DISKETTE RÜCKSETZEN

INT21: 0Dh

Funktion:

Alle für die Dateiarbeit benutzten Puffer werden auf den Datenträger geschrieben.

Aufruf:

Register	Inhalt/Bedeutung
AH	0Dh

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Durch diese DOS-Funktion werden sämtliche Datenpuffer auf den Datenträger zurückgeschrieben und anschließend freigegeben, d.h., alle Puffer, die Eintragungen auf geöffnete Dateien besitzen, werden geleert. Dabei ist zu beachten, dass Verzeichniseintragungen nicht aktualisiert werden. Geänderte Dateien müssen deshalb durch eine andere DOS-Funktion gesondert geschlossen werden.

Die DOS-Funktion 0Ch wird benutzt, damit das Dateisystem in einen definierten Zustand gelangt. So sollte z.B. eine CTRL-C-Behandlungsroutine diese DOS-Funktion benutzen.

9.2.2.15. AUSWAHL BEZUGS-LAUFWERK

INT21: 0Eh

Funktion:

Bestimmen eines Laufwerkes zum Bezugs-Laufwerk (Standardlaufwerk).

Aufruf:

Register	Inhalt/Bedeutung
AH	0Eh
DL	Laufwerksnummer (0=LW A, 1=LW B,...)

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Anzahl der Laufwerke

Bemerkung:

Wenn die Laufwerksangabe gültig ist (d.h., dass das Laufwerk im Betriebssystem definiert ist), kann auf das selektierte Laufwerk ohne jeweilige Laufwerksangabe zugegriffen werden. Verfügt das System nur über ein Diskettenlaufwerk, so wird dennoch von 2 Laufwerken ausgegangen, um die Philosophie der logischen Laufwerke A und B aufrecht erhalten zu können.

9.2.2.16. DATEI ERÖFFNEN

INT21: 0Fh

Funktion:

Eröffnen einer Datei über einen Dateikontrollblock (FCB).

Aufruf:

Register	Inhalt/Bedeutung
AH	0Fh
DS:DX	Zeiger auf nicht eröffneten FCB

Folgende Werte müssen im nicht eröffneten FCB eingetragen sein:

- Offset 00h Laufwerksnummer mit 0=Standard-LW, 1=LW A, 2=LW B, usw.
- Offset 01h Dateiname (max. 8 Zeichen)
- Offset 09h Dateityp (max. 3 Zeichen)

Im aktuellen Verzeichnis wird nach der im nicht eröffneten FCB angegebenen Dateibezeichnung gesucht.

*** DOS-Funktionsaufrufe ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Datei eröffnet Eintragungen im FCB

Wenn die Datei vorhanden ist, werden in dem nun eröffneten FCB folgende Daten ergänzt bzw. modifiziert:

Offset 00h	Aktuelles Laufwerk
Offset 0Ch/0Dh	Aktuelle Blocknummer = 0000h
Offset 0Eh/0Fh	Grösse eines Datensatzes = 0080h
Offset 10h bis 13h	Grösse der Datei
Offset 14h/15h	Datum
Offset 16h/17h	Zeit

Wenn als Laufwerknummer 0 eingetragen wurde, so handelt es sich um das Standardlaufwerk. Im FCB wird dafür das entsprechende aktuelle Laufwerk eingesetzt (1=LW A, 2=LW B, ...). Die Datei ist im Kompatibilitätsmodus eröffnet. Für Informationen über diesen Modus ist in der Beschreibung der DOS-Funktion "Händler eröffnen" (3Dh) nachzulesen.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Datei nicht eröffnet

Bemerkung:

Der Standardwert für die Grösse des Datensatzes (0080h) ist voreingestellt und kann vor Lese- oder Schreiboperationen verändert werden.

Falls die zu eröffnende Datei mit den Attributen für eine Systemdatei bzw. für eine verborgene Datei versehen ist, so wird die Datei nicht gefunden (AL=FFh). Zur Bearbeitung dieser muss ein erweiterter FCB verwendet werden.

9.2.2.17. DATEI SCHLIESSEN

INT21: 10h

Funktion:

Schliessen einer Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	10h
DS:DX	Zeiger auf eröffneten FCB

Das aktuelle Verzeichnis der Diskette wird abgesucht.

Wird die Datei gefunden, so wird ihre Position mit der in dem FCB angegebenen Position verglichen.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Datei geschlossen

Das Verzeichnis der Datei und die letzten Daten im Puffer werden auf dem Datenträger aktualisiert.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Datei nicht geschlossen

Wird die Datei nicht an der entsprechenden Position im aktuellen Verzeichnis gefunden, so wird davon ausgegangen, dass der Datenträger ausgetauscht wurde. Das Verzeichnis wird nicht aktualisiert.

Bemerkung:

Diese DOS-Funktion muss nach Schreibvorgängen in einer Datei aufgerufen werden, um sicherzustellen, dass alle Verzeichniseinträge aktualisiert werden.

9.2.2.18. SUCHE NACH DEM ERSTEN EINTRAG

INT21h: 11h

Funktion:

Suche nach der ersten Dateibezeichnung, die einem vorgegebenen Suchmuster entspricht.

Aufruf:

Register	Inhalt/Bedeutung
AH	11h
DS:DX	Zeiger auf nicht eröffneten FCB

Folgende Werte müssen im nicht eröffneten FCB eingetragen sein:

- Offset 00h Laufwerknummer mit 0=Standard-LW, 1=LW A, 2=LW B, usw.
- Offset 01h Dateiname (max. 8 Zeichen)
- Offset 09h Dateityp (max. 3 Zeichen)

Bei Verwendung eines erweiterten FCB muss zusätzlich eingetragen sein:

- Offset -01h Attribut
- Offset -07h FFh

Im aktuellen Verzeichnis wird nach der im nicht eröffneten FCB angegebenen Dateibezeichnung gesucht. Bei der Angabe von Dateiname und -typ sind globale Dateibezeichner zulässig.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Dateibezeichnung gefunden

Wenn eine entsprechende Datei-Bezeichnung gefunden wurde, so wird ein nicht eröffneter FCB im aktuellen DTA aufgebaut.

Normaler FCB:

Handelt es sich bei dem für die Suche zur Verfügung gestellten FCB um einen standardmässigen FCB, so wird das erste Byte auf die benutzte Laufwerknummer gesetzt (1=LW A, 2=LW B, ...). Die nächsten 32 Byte enthalten den übereinstimmenden Verzeichniseintrag.

Es entsteht somit ein gültiger standardmässiger, aber nicht eröffneter FCB. Er besitzt die gleichen Attribute wie der für die Suche benutzte FCB.

Erweiterter FCB:

Handelt es sich bei dem für die Suche zur Verfügung gestellten FCB um einen erweiterten FCB, so wird das erste Byte der DTA auf FFh gesetzt. Die folgenden 5 Bytes sind 00h.

Danach werden das Attributbyte des für die Suche benutzten FCB, die benutzte Laufwerknummer (1=LW A, 2=LW B, ...) und schliesslich die 32 Bytes des Verzeichniseintrages gesetzt.

Es entsteht somit ein gültiger erweiterter, aber nicht eröffneter FCB. Er besitzt die gleichen Attribute wie der für die Suche benutzte FCB.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Datei-Bezeichnung nicht gefunden

Bemerkung:

Wenn ein erweiterter FCB benutzt wird, so gilt entsprechend dem verwendeten Attribut folgendes Suchmuster:

1. Attributbyte = 00h

Es werden nur normale Dateieintragen gefunden. Eintragungen für Datenträgerkennsatz, für Verzeichnisse, für verborgene Dateien und für Systemdateien werden nicht gefunden.

2. Attributbyte = 02h (verborgene Dateien)

" = 04h (Systemdateien)

" = 10h (Verzeichniseintragen)

Es erfolgt eine inklusive Suche, d.h., sämtliche normalen Dateieintragen und alle Eintragungen, die mit den angegebenen Attributen übereinstimmen, werden gefunden.

Zur Überprüfung sämtlicher Verzeichniseintragen, mit Ausnahme des Datenträgerkennsatzes, kann das Attribut auf "Verborgen + System + Verzeichnis" (=16h) gesetzt werden.

*** DOS-Funktionsaufrufe ***

3. Attributbyte = 08h (Datenträgerkennsatz)
Es erfolgt eine exklusive Suche, d.h., es wird nur der Eintrag für den Datenträgerkennsatz gefunden.

9.2.2.19. SUCHE NACH DEM NÄCHSTEN EINTRAG

INT21: 12h

Funktion:

Suchen nach der nächsten Dateibezeichnung.

Aufruf:

Register	Inhalt/Bedeutung
AH	12h
DS:DX	Zeiger auf nicht eröffneten FCB

Nachdem bei Aufruf der DOS-Funktion 12h eine Übereinstimmung in der Dateibezeichnung gefunden wurde, kann diese DOS-Funktion aufgerufen werden, um die nächste Übereinstimmung bei einer mehrdeutigen Anforderung zu finden. Die globalen Dateibezeichner sind erlaubt.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Dateibezeichnung gefunden

Wenn eine entsprechende Dateibezeichnung gefunden wurde, so wird ein nicht eröffneter FCB im aktuellen DTA aufgebaut.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Dateibezeichnung nicht gefunden

Bemerkung:

Für diese DOS-Funktionen gelten die gleichen Anmerkungen wie für die Funktion 11h.

9.2.2.20. DATEI LÖSCHEN

INT21: 13h

Funktion:
Löschen einer Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	13h
DS:DX	Zeiger auf nicht eröffneten FCB

In der Dateibezeichnung ist der Dateibezeichner "?" erlaubt.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Datei gelöscht

Alle übereinstimmenden Dateien im aktuellen Verzeichnis sind gelöscht.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Datei nicht gelöscht

Bemerkung:

Besitzt die zu löschende Datei das Attribut "Nur Lesen" bzw. "Verborgen", so wird diese nicht gelöscht. Eröffnete Dateien sind erst zu schliessen, bevor sie gelöscht werden.

Netzwerk-Zugriffsrechte:

Es sind Anlege-Zugriffsrechte erforderlich.

9.2.2.21. SEQUENTIELLES LESEN

INT21: 14h

Funktion:
Lesen des nächsten Datensatzes einer Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	14h
DS:DX	Zeiger auf eröffneten FCB

Der von der aktuellen Blocknummer (Offset 0Ch und 0Dh des FCB) und dem aktuellen Datensatz (Offset 20h des FCB) adressierte

*** DOS-Funktionsaufrufe ***

Datensatz wird an die DTA übertragen. Die Länge des Datensatzes wird durch die Angabe im Feld für die Grösse des Datensatzes (Offset 0Eh und 0Fh des FCB) bestimmt.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h

Die aktuelle Satznummer wird um 1 erhöht (Aktualisierung von den Feldern Offset 0Ch/0Dh und 20h).

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	01h Dateiende erreicht, aber keine Datensätze gelesen
AL	02h Lesevorgang nicht ausgeführt, weil Datenpuffer zu klein
AL	03h Dateiende beim Lesen eines Datensatzes erreicht

Bei dem Fehler AL=03h wurde nur ein Teil des Satzes gelesen und der Rest mit 00h aufgefüllt.

Netzwerk-Zugriffsrechte:

Es sind Lese-Zugriffsrechte erforderlich.

9.2.2.22. SEQUENTIELLES SCHREIBEN

INT21: 15h

Funktion:

Schreiben des aktuellen Datensatzes auf eine Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	15h
DS:DX	Zeiger auf eröffneten FCB

Die Daten vom aktuellen DTA werden an die von der aktuellen Blocknummer (Offset 0Ch und 0Dh des FCB) und dem aktuellen Datensatz (Offset 20h des FCB) adressierte Datensatz übertragen. Die Länge des Datensatzes wird durch die Angabe im Feld für die Grösse des Datensatzes (Offset 0Eh und 0Fh des FCB) bestimmt.

*** DOS-Funktionsaufrufe ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Schreiben erfolgreich

Die aktuelle Satznummer wird um 1 erhöht (Aktualisierung von den Feldern Offset 0Ch/0Dh und 20h).

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	01h Schreibvorgang wurde nicht ausgeführt, weil der Datenträger voll ist.
AL	02h Schreibvorgang wird nicht ausgeführt, weil der Datenpuffer zu klein ist.

Bemerkung:

Falls die Grösse des Datensatzes nicht mit der physischen Sektorgrösse übereinstimmt, so erfolgt im Betriebssystem eine Pufferung der Daten. Mit der DOS-Funktion "Datei schliessen" (10h) werden die gepufferten Daten auf den Datenträger übertragen. Besitzt die Datei das Attribut "Nur Lesen", wird das sequentielle Schreiben nicht ausgeführt.

Netzwerk-Zugriffsrechte:

Es sind Schreib-Zugriffsrechte erforderlich.

9.2.2.23. DATEI ERSTELLEN

INT21: 16h

Funktion:

Erstellen einer neuen Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	16h
DS:DX	Zeiger auf nicht eröffneten FCB

Das aktuelle Verzeichnis der Diskette wird nach einer übereinstimmenden Eintragung abgesucht.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Datei erstellt

*** DOS-Funktionsaufrufe ***

Es wird eine Datei mit der Länge 0 erzeugt und anschliessend über die DOS-Funktion 0Fh eröffnet.
Falls die zu erstellende Datei bereits existiert, wird diese gelöscht und eine leere Datei mit der gleichen Dateibezeichnung angelegt.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Datei nicht erstellt, da Verzeichnis oder Datenträger voll

Bemerkung:

Bei Verwendung eines erweiterten FCB erhält die erstellte Datei das in Offset -1 angegebene Attribut.

Netzwerk-Zugriffsrechte:

Es sind Anlege-Zugriffsrechte erforderlich.

9.2.2.24. DATEI UMBENENNEN

INT21: 17h

Funktion:

Umbenennen einer Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	17h
DS:DX	Zeiger auf modifizierten FCB

Der durch DS:DX modifizierte FCB muss folgenden Aufbau haben:

Offset 00h Laufwerknummer mit 0=Standard-LW, 1=LW A, 2=LW B, usw.
Offset 01h Quell-Dateiname (max. 8 Zeichen)
Offset 09h Quell-Dateityp (max. 3 Zeichen)
Offset 11h Ziel-Dateiname (max. 8 Zeichen)
Offset 19h Ziel-Dateityp (max. 3 Zeichen)

Bei Angabe der Quell- und Ziel-Dateibezeichnungen kann der globale Dateibezeichner "?" verwendet werden.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Datei umbenannt

*** DOS-Funktionsaufrufe ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Datei nicht umbenannt

Die Datei wird nicht umbenannt, wenn

- die Ziel-Dateibezeichnung bereits vorhanden ist,
- die Quell-Dateibezeichnung nicht vorhanden ist,
- eine unzulässige bzw. falsche Laufwerknummer vorliegt,
- die Quell-Datei das Attribut "Verborgene" besitzt.

Bemerkung:

Bei jedem übereinstimmenden Auftreten der Quell-Dateibezeichnung wird diese gegen die Ziel-Dateibezeichnung ausgetauscht.

Die Quell- und die Ziel-Dateibezeichnungen dürfen nicht gleich sein.

Enthält die Ziel-Dateibezeichnung den globalen Dateibezeichner "?", so bleiben die entsprechenden Positionen in der Quell-Dateibezeichnung unverändert.

Zum Umbenennen von Dateien mit Attributen ist ein erweiterter FCB zu verwenden.

Netzwerk-Zugriffrechte:

Es sind Anlege-Zugriffsrechte erforderlich.

9.2.2.25. ABFRAGE BEZUGS-LAUFWERK

INT21: 19h

Funktion:

Abfrage des selektierten Laufwerkes.

Aufruf:

Register	Inhalt/Bedeutung
AH	19h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Laufwerknummer (0=LW A, 1=LW B, ...)

Bemerkung:

Durch diese DOS-Funktion wird das aktuelle Standardlaufwerk bestimmt, welches in AL zurückgegeben wird.

9.2.2.26. SETZEN DTA

INT21: 1Ah

Funktion:

Adresse des Puffers (DTA) für die Ein-/Ausgabe auf Diskette/
Platte ändern.

Aufruf:

Register	Inhalt/Bedeutung
AH	1Ah
DS:DX	Zeiger auf DTA

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Der Nutzer muss darauf achten, dass der DTA innerhalb eines Segmentes die Segmentgrenze nicht überschreitet. Wird der DTA nicht durch diese DOS-Funktion gesetzt, wird standardmässig die Adresse 0080h (Grösse 80h Byte) innerhalb des PSP verwendet.

9.2.2.27. INFORMATIONEN DER ZUORDNUNGSTABELLE

INT21: 1Bh

Funktion:

Abfrage der Informationen des aktuellen Laufwerkes.

Aufruf:

Register	Inhalt/Bedeutung
AH	1Bh

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Anzahl Sektoren pro Cluster
CX	Anzahl Byte pro Sektor
DS:BX	Zeiger auf Kennzeichenbyte der FAT
DX	Anzahl Cluster

Bemerkung:

Mit Rückkehr aus dieser DOS-Funktion erhält der Nutzer einen Überblick über die Gesamtorganisation des selektierten Datenträgers.

**9.2.2.28. INFORMATIONEN DER ZUORDNUNGSTABELLE
FÜR BELIEBIGES LAUFWERK**

INT21: 1Ch

Funktion:

Abfrage der Informationen eines beliebigen Laufwerkes.

Aufruf:

Register	Inhalt/Bedeutung
AH	1Ch
DL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Anzahl Sektoren pro Cluster
CX	Anzahl Byte pro Sektor
DS:BX	Zeiger auf Kennzeichenbyte der FAT
DX	Anzahl Cluster

Bemerkung:

Mit Rückkehr aus dieser DOS-Funktion erhält der Nutzer einen Überblick über die Gesamtorganisation des angegebenen Datenträgers.

9.2.2.29. WAHLFREIES LESEN

INT21: 21h

Funktion:

Wahlfreier (direkter) Lese-Zugriff auf einen einzelnen Datensatz innerhalb einer Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	21h
DS:DX	Zeiger auf eröffneten FCB

Die relative Nummer des Datensatzes von Beginn der Datei (Offset 21h bis 24h) muss im FCB gesetzt werden.

Davon ausgehend werden entsprechend der Größe des Datensatzes (Offset 0Eh und 0Fh) die aktuelle Blocknummer (Offset 0Ch und 0Dh) und der aktuelle Datensatz (Offset 20h) aktualisiert.

Der hierdurch adressierte Datensatz wird an die DTA übertragen.

*** DOS-Funktionsaufrufe ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Lesen erfolgreich

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	01h Dateiende erreicht, aber keine Datensätze gelesen
AL	02h Lesevorgang nicht ausgeführt, weil Datenpuffer zu klein
AL	03h Dateiende beim Lesen eines Datensatzes erreicht

Bei dem Fehler AL=03h wurde nur ein Teil des Satzes gelesen und der Rest mit 00h aufgefüllt.

Netzwerk-Zugriffsrechte:

Es sind Lese-Zugriffsrechte erforderlich.

9.2.2.30. WAHLFREIES SCHREIBEN

INT21: 22h

Funktion:

Wahlfreier (direkter) Schreib-Zugriff auf einen einzelnen Datensatz innerhalb einer Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	22h
DS:DX	Zeiger auf eröffneten FCB

Die relative Nummer des Datensatzes von Beginn der Datei (Offset 21h bis 24h) muss im FCB gesetzt werden. Davon ausgehend werden entsprechend der Grösse des Datensatzes (Offset 0Eh und 0Fh) die aktuelle Blocknummer (Offset 0Ch und 0Dh) und der aktuelle Datensatz (Offset 20h) aktualisiert. Ab dem DTA werden die Daten in den hierdurch adressierten Datensatz übertragen.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Schreiben erfolgreich

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	01h Schreibvorgang nicht ausgeführt, weil Datenträger voll
AL	02h Schreibvorgang nicht ausgeführt, weil Datenpuffer zu klein

Bemerkung:

Falls die Grösse des Datensatzes nicht mit der physischen Sektorgrösse übereinstimmt, erfolgt im Betriebssystem eine Pufferung der Daten. Mit der DOS-Funktion "Datei schliessen" (10h) werden die gepufferten Daten auf den Datenträger übertragen. Besitzt die Datei das Attribut "Nur Lesen", wird das sequentielle Schreiben nicht ausgeführt.

Netzwerk-Zugriffsrechte:

Es sind Schreib-Zugriffsrechte erforderlich.

9.2.2.31. DATEIGROSSE

INT21: 23h

Funktions:

Ermitteln der Grösse einer Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	23h
DS:DX	Zeiger auf nicht eröffneten FCB

Vor Ausführung dieser DOS-Funktion muss die Grösse des Datensatzes (Offset 0Eh und 0Fh des FCB) gesetzt werden. Die Verwendung des globalen Dateibezeichners "?" in der Dateibezeichnung ist erlaubt. Bei Dateien mit gesetzten Attributen ist ein erweiterter FCB zu verwenden.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Datei vorhanden

Wenn die erste dem Suchmuster entsprechende Verzeichniseintragung gefunden wird, so enthält das Feld für die relative Nummer des Datensatzes (Offset 21h bis 24h des FCB) die Dateigrösse in einem Vielfachen von der Grösse des Datensatzes (Offset 0Eh und 0Fh).

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Datei nicht vorhanden

Bemerkung:

Es kann vorkommen, dass das Produkt aus der Blockgrösse und der durch diese DOS-Funktion zurückgemeldeten Blockanzahl grösser als die tatsächliche Dateigrösse ist. Das ist genau dann der Fall, wenn bei der Division der Dateigrösse durch die Blockgrösse ein Rest entsteht.

9.2.2.32. AKTUALISIEREN RELATIVE NUMMER DES DATENSATZES

INT21: 24h

Funktion:

Aktualisieren der relativen Nummer eines Datensatzes für wahl- freien Zugriff.

Aufruf:

Register	Inhalt/Bedeutung
AH	24h
DS:DX	Zeiger auf eröffneten FCB

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Die Nummer des relativen Datensatzes (Offset 21h bis 24h des FCB) wird entsprechend der Grösse des Datensatzes (Offset 0Eh und 0Fh) und des aktuellen Datensatzes (Offset 20h) aktuali- siert. Somit wird der gleiche Datensatz bei direkten bzw. se- quentiellen Lese- oder Schreiboperationen adressiert.

9.2.2.33. SETZEN INTERRUPTVEKTOR

INT21: 25h

Funktion:

Interruptvektor setzen.

Aufruf:

Register	Inhalt/Bedeutung
AH	25h
AL	Nummer des Interruptvektors
DS:DX	Zeiger auf Interruptroutine

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Die Interruptvektoren sollten nie direkt im Speicher verändert werden.

9.2.2.34. ERSTELLEN NEUER PSP

INT21: 26h

Funktion:

Erzeugen eines neuen PSP (Programm-Segment-Präfix) aus dem aktuellen Codesegment.

Aufruf:

Register	Inhalt/Bedeutung
AH	26h
DX	Segmentadresse

Der gesamte Bereich von 00h bis FFh (PSP) des aktuellen Codesegmentes wird in den gleichen Bereich des neuen Codesegmentes (DX) kopiert.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Die Information über den im neuen Segment zur Verfügung stehenden Speicherplatz in Offset 06h des neuen PSP wird aktualisiert.

9.2.2.35. WAHLFREIES BLOCKLESEN

INT21: 27h

Funktion:

Wahlfreier (direkter) Lese-Zugriff auf mehrere Datensätze innerhalb einer Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	27h
CX	Anzahl zu lesender Datensätze
DS:DX	Zeiger auf eröffneten FCB

Ab der Nummer des relativen Datensatzes (Offset 21h bis 24h des FCB) werden CX-Datensätze in den aktuellen DTA gelesen. Bei CX=0 wird die Leseoperation nicht ausgeführt.

*** DOS-Funktionsaufrufe ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Lesen erfolgreich
CX	Anzahl der tatsächlich gelesenen Datensätze

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	01h Dateiende erreicht, aber keine Datensätze gelesen
AL	02h Lesevorgang abgebrochen, weil Datenpuffer zu klein ist
AL	03h Dateiende beim Lesen eines Datensatzes erreicht
CX	Anzahl der tatsächlich gelesenen Datensätze

Bei dem Fehler AL=03h wurde nur ein Teil des Satzes gelesen und der Rest mit 00h aufgefüllt.

Kommt es beim Füllen des DTA zu einem Überlauf über die Segmentgrenze, so werden keine weiteren Datensätze gelesen. Das Lesen wird abgebrochen und in AL wird 02h übergeben.

Bemerkung:

In jedem Fall wird in CX die tatsächliche Anzahl von gelesenen Datensätzen zurückgegeben.

Die relative Nummer des Datensatzes von Beginn der Datei (Offset 21h bis 24h des FCB), die aktuelle Blocknummer (Offset 0Ch und 0Dh) und der aktuelle Datensatz (Offset 20h) werden so gesetzt, dass der nächste Datensatz adressiert wird (der erste nicht gelesene Datensatz).

Netzwerk-Zugriffsrechte:

Es sind Lese-Zugriffsrechte erforderlich.

9.2.2.36. WAHLFREIES BLOCKSCHREIBEN

INT21: 28h

Funktion:
Wahlfreier (direkter) Schreib-Zugriff auf mehrere Datensätze innerhalb einer Datei über einen FCB.

Aufruf:

Register	Inhalt/Bedeutung
AH	28h
CX	Anzahl zu schreibender Datensätze
DS:DX	Zeiger auf eröffneten FCB

Vom DTA werden CX Datensätze, beginnend mit der Nummer des relativen Datensatzes (Offset 21h bis 24h des FCB), geschrieben. Bei CX=0 werden keine Datensätze geschrieben.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Schreiben erfolgreich
CX	Anzahl der tatsächlich geschriebenen Datensätze

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	01h Schreibvorgang wurde abgebrochen, weil der Datenträger voll ist.
AL	02h Schreibvorgang wurde abgebrochen, weil der Datenpuffer zu klein.
CX	Anzahl der tatsächlich geschriebenen Datensätze

Kommt es beim Schreiben aus dem DTA zu einem Überlauf über die Segmentgrenze, so werden keine weiteren Datensätze geschrieben. Das Schreiben wird abgebrochen, und in AL wird 02h übergeben.

Bemerkung:

In jedem Fall wird in CX die tatsächliche Anzahl von geschriebenen Datensätzen eingetragen. Die relative Nummer des Datensatzes von Beginn der Datei (Offset 21h bis 24h des FCB), die aktuelle Blocknummer (Offset 0Ch und 0Dh) und der aktuelle Datensatz (Offset 20h) werden so gesetzt, dass der nächste Datensatz adressiert wird (der erste nicht geschriebene Datensatz). Ist bei Aufruf der DOS-Funktion CX=0, so wird die Dateigröße auf die von der relativen Nummer des Datensatzes (Offset 21h bis 24h) angegebenen Länge gesetzt.

Netzwerk-Zugriffsrechte:

Es sind Schreib-Zugriffsrechte erforderlich.

9.2.2.37. DATEIBEZEICHNUNG ANALYSIEREN

INT21: 29h

Funktion:

Durchsuchen einer ASCII-Zeichenkette nach einer Dateibezeichnung und Separieren der Dateibezeichnung.

Aufruf:

Register	Inhalt/Bedeutung
AH	29h
AL	Analysemodus
DS:SI	Zeiger auf zu analysierende Befehlszeile
ES:DI	Zeiger auf nicht eröffneten FCB

Durch diese DOS-Funktion wird eine durch DS:SI angegebene Befehlszeile analysiert.

Die globalen Dateibezeichner "*" und "?" sind in der Dateibezeichnung erlaubt.

Für "*" werden alle restlichen Zeichen auf "?" gesetzt.

Die Befehlszeile wird auf eine Dateibezeichnung, einschliesslich der Laufwerkspezifikation, analysiert, und wenn vorhanden, wird in ES:DI ein entsprechender FCB erstellt, jedoch nicht eröffnet.

In AL wird der Analysemodus angegeben. Dabei gilt:

- Bit 0 = 1: Ein führendes Trennzeichen in der Befehlszeile wird ignoriert.
- Bit 0 = 0: Ein führendes Trennzeichen wird nicht ignoriert; wirkt als Beendigungszeichen.
- Bit 1 = 1: Laufwerknummer im Ziel-FCB wird nur verändert, wenn eine Laufwerkbezeichnung in der Befehlszeile angegeben wurde.
- Bit 1 = 0: Laufwerknummer im Ziel-FCB wird auf 00h (Standardlaufwerk) gesetzt, wenn keine Laufwerkbezeichnung in der Befehlszeile angegeben wurde.
- Bit 2 = 1: Dateiname im FCB wird nicht geändert, wenn die Befehlszeile keinen Dateinamen enthält.
- Bit 2 = 0: Dateiname im FCB wird mit 8 Leerzeichen (20h) gefüllt, wenn die Befehlszeile keinen Dateinamen enthält.
- Bit 3 = 1: Dateityp im FCB wird nicht geändert, wenn die Befehlszeile keinen Dateityp enthält.
- Bit 3 = 0: Dateityp im FCB wird mit 3 Leerzeichen (20h) gefüllt, wenn die Befehlszeile keinen Dateityp enthält.
- Bit 4 wird nicht benutzt
- Bit 5 wird nicht benutzt
- Bit 6 wird nicht benutzt
- Bit 7 wird nicht benutzt

*** DOS-Funktionsaufrufe ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Es sind keine globalen Dateibezeichner in der Dateispezifikation.
AL	01h Es sind globale Dateibezeichner in der Dateispezifikation.
DS:SI	Zeiger auf erstes Zeichen nach letztem analysierten Teil der Befehlszeile
ES:DI	Zeiger auf generierten, nicht eröffneten FCB

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Laufwerkangabe ist ungültig

Bemerkung:

Für die Verwendung einzelner Zeichen als Trenn- bzw. Beendigungszeichen gilt:

Leerzeichen wird als führendes Leerzeichen übergangen, wirkt am Ende als Trennzeichen.
 " / [\] wirken als Beendigungszeichen.
 Zeichen 00h-08h wirken als Beendigungszeichen.
 Zeichen 0Ah-1Fh wirken als Beendigungszeichen.
 + , : < = > ; | \ sind führende Trennzeichen.
 . wirkt als Trennzeichen und wird immer übergangen.
 Zeichen 09h wird als führendes Zeichen übergangen, wirkt am Ende als Trennzeichen.

9.2.2.38. DATUM ABFRAGEN

INT21: 2Ah

Funktion:

Ermitteln des System-Datums.

Aufruf:

Register	Inhalt/Bedeutung
AH	2Ah

*** DOS-Funktionsaufrufe ***

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Wochentag (0=Sonntag, ..., 6=Samstag)
CX	Jahr (1980 - 2099)
DH	Monat (1 - 12)
DL	Tag (1 - 31)

Die Werte werden in binärer Form übergeben.

Bemerkung:

Läuft die System-Uhr auf den nächsten Tag über, so wird das Datum entsprechend angepasst, wobei die Anzahl von Tagen eines jeden Monats und die Schaltjahre berücksichtigt werden.

9.2.2.39. DATUM SETZEN

INT21: 2Bh

Funktion:

Setzen des System-Datums.

Aufruf:

Register	Inhalt/Bedeutung
AH	2Bh
CX	Jahr (1980 - 2099)
DH	Monat (1 - 12)
DL	Tag (1 - 31)

Die Werte sind in binärer Form anzugeben.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Datum ist gültig

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Datum ist ungültig

9.2.2.40. ZEIT ABFRAGEN

INT21: 2Ch

Funktion:
Ermitteln der System-Zeit.

Aufruf:

Register	Inhalt/Bedeutung
AH	2Ch

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CH	Stunden (0 - 23)
CL	Minuten (0 - 59)
DH	Sekunden (0 - 59)
DL	Hundertstelsekunden (0 - 99)

Die Werte werden in binärer Form übergeben.

9.2.2.41. ZEIT SETZEN

INT21: 2Dh

Funktion:
Setzen der System-Zeit.

Aufruf:

Register	Inhalt/Bedeutung
AH	2Dh
CH	Stunden (0 - 23)
CL	Minuten (0 - 59)
DH	Sekunden (0 - 59)
DL	Hundertstelsekunden (0 - 99)

Die Werte sind in binärer Form anzugeben.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	00h Zeit ist gültig

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Zeit ist ungültig

9.2.2.42. PRÜFLESEN SETZEN/RUECKSETZEN

INT21: 2Eh

Funktion:

Verändern des Schalters für das Prüfllesen.

Aufruf:

Register	Inhalt/Bedeutung
AH	2Eh
AL	00h Prüfllesen AUS
AL	01h Prüfllesen EIN

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Bemerkung:

Ist das Prüfllesen EIN, so führt das Betriebssystem eine Prüfung bei jeder Schreiboperation auf den Datenträger aus, um eine korrekte Aufzeichnung der Daten zu gewährleisten. Durch das zusätzliche Kontrolllesen verlangsamt sich der Schreibvorgang. Bei Netzwerkdatenträgern wird die Kontrolle der geschriebenen Daten nicht unterstützt.

9.2.2.43. DTA HOLEN

INT21: 2Fh

Funktion:

Ermitteln der Adresse des Puffers (DTA) für die Ein-/Ausgabe auf Diskette/Platte.

Aufruf:

Register	Inhalt/Bedeutung
AH	2Fh

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
ES:BX	Zeiger auf DTA

9.2.2.44. DCP-VERSIONSNUMMER ABFRAGEN

INT21: 30h

Funktion:

Ermitteln der Versionsnummer des Betriebssystems DCP.

Aufruf:

Register	Inhalt/Bedeutung
AH	30h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Oberer Teil der Versionsnummer
AH	Unterer Teil der Versionsnummer
BX	Unbestimmt
CX	Unbestimmt

In AL steht der höherwertige und in AH der niederwertige Teil der verwendeten Version.

Beispiel:

Bei Verwendung der DCP-Version 3.30 werden in AL der Wert 03h und in AH der Wert 1Eh dem aufrufenden Programm übergeben.

9.2.2.45. PROZESS BEENDEN UND RESIDENT BLEIBEN

INT21: 31h

Funktion:

Laufendes Programm beenden, wobei es jedoch im Speicher verbleibt.

Aufruf

Register	Inhalt/Bedeutung
AH	31h
AL	Rückgabekode
DX	Speichergrösse in Paragraphen

Ein Paragraph entspricht einer Grösse von 16 Byte.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

*** DOS-Funktionsaufrufe ***

Bemerkung:

Dieser Funktionsaufruf beendet das aktuelle Programm und macht den in DX übergebenen Speicherbereich resident. Eröffnete Dateien werden nicht geschlossen. Im Unterschied zum Interrupt 27h kann bei dieser DOS-Funktion ein Programm, das grösser als 64K ist, resident gemacht werden.

9.2.2.46. PRÜFUNG AUF CTRL-C

INT21: 33h

Funktion:

Abfragen, Setzen und Rücksetzen der Reaktionen auf Unterbrechung (CTRL-C).

Aufruf:

Register	Inhalt/Bedeutung
AH	33h
AL	00h Abfrage des aktuellen Status
AL	01h Verändern des aktuellen Status
DL	00h CTRL-C-Erkennung AUS
DL	01h CTRL-C-Erkennung EIN

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
DL	00h CTRL-C-Erkennung ist AUS.
DL	01h CTRL-C-Erkennung ist EIN.

Bei Rückkehr aus der DOS-Funktion mit AL=00h enthält DL den gegenwärtigen Status.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AL	FFh Ungültiger Funktionskode in DL

Bei Rückkehr aus der DOS-Funktion mit AL=01h kennzeichnet AL=FFh einen in DL ungültig verwendeten Funktionskode.

Bemerkung:

Durch diese DOS-Funktion kann gesteuert werden, ob ein laufendes Programm durch CTRL-C abgebrochen werden kann. Normalerweise unterbricht CTRL-C nur die DOS-Funktionsaufrufe der Ein- und Ausgabe für Standardgeräte.

Durch diese DOS-Funktion jedoch lässt sich die Unterbrechungsmöglichkeit auf alle DOS-Funktionen ausdehnen.

Soll mit den DOS-Funktionen 06h und 07h die ASCII-Darstellung von CTRL-C (03h) eingelesen werden, so ist durch die DOS-Funktion 33h die CTRL-C-Erkennung auf AUS zu schalten.

9.2.2.47. INTERRUPTVEKTOR ABFRAGEN

INT21: 35h

Funktion:
Ermitteln des Interruptvektors.

Aufruf:

Register	Inhalt/Bedeutung
AH	35h
AL	Nummer des Interruptvektors (00h - FFh)

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
ES:BX	Zeiger auf die Interruptroutine

Bemerkung:

Die Interruptvektoren sollten nur durch die DOS-Funktion abgefragt werden.

9.2.2.48. ORGANISATION DES DATENTRIGERS ABFRAGEN

INT21: 36h

Funktion:

Liefert Informationen über die Organisation des Datenträgers.

Aufruf:

Register	Inhalt/Bedeutung
AH	36h
DL	Laufwerknummer (0=Standard-LW, 1=LW A,..)

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
AX	Anzahl von Sektoren pro Cluster
BX	Anzahl der verfügbaren freien Cluster
CX	Anzahl der Bytes pro Sektor
DX	Gesamtzahl der Cluster

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
AX	FFFFh Ungültige Laufwerknummer in DL

9.2.2.49. LANDESSPEZIFISCHE INFORMATIONEN
ABFRAGEN/SETZEN

INT21: 38h

Funktion:

Ermitteln bzw. Setzen von landesspezifischen Informationen (Aufbau, Schreibweise und Interpunktion der Landeswährung und des Datum- und Zeitformates).

9.2.2.49.1. ABFRAGE DER AKTUELLEN LANDESSPEZIFISCHEN INFORMATION

INT21: 38h 00h

Aufruf:

Register	Inhalt/Bedeutung
AH	38h
AL	00h
DS:DX	Zeiger auf 34 Byte grosses Feld

Die zurückgegebenen Informationen werden nach DS:DX eingetragen.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
DS:DX	Landesspezifische Informationen
BX	Landeskennzahl

*** DOS-Funktionsaufrufe ***

Die folgende Tabelle zeigt den Aufbau des Speicherbereiches, der die landesspezifischen Informationen enthält:

Offset hex/dez	Anzahl Byte	Bedeutung								
00h 00	2	Format von Datum Es gilt: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Kode</th> <th>Datum</th> </tr> </thead> <tbody> <tr> <td>0 = USA</td> <td> m t j</td> </tr> <tr> <td>1 = Europa</td> <td> t m j</td> </tr> <tr> <td>2 = Japan</td> <td> j m t</td> </tr> </tbody> </table>	Kode	Datum	0 = USA	m t j	1 = Europa	t m j	2 = Japan	j m t
Kode	Datum									
0 = USA	m t j									
1 = Europa	t m j									
2 = Japan	j m t									
02h 02	5	Währungssymbol (mit 00h begrenzt)								
07h 07	2	Tausenderinterpunktionszeichen (mit 00h begrenzt)								
09h 09	2	Dezimalinterpunktionszeichen (mit 00h begrenzt)								
0Bh 11	2	Trennzeichen im Datum (mit 00h begrenzt)								
0Dh 13	2	Trennzeichen in der Zeitangabe (mit 00h begrenzt)								
0Fh 15	1	Währungsformat Bit 0 = 0 Währungssymbol vor dem Wert Bit 0 = 1 Währungssymbol nach dem Wert Bit 1 = 0 kein Leerzeichen zwischen Wert und Währungssymbol Bit 1 = 1 ein Leerzeichen zwischen Wert und Währungssymbol Bit 2 = 1 Dezimaltrennzeichen wird durch das Währungssymbol ersetzt.								
10h 16	1	Anzahl von signifikanten Dezimalstellen in der Währungsangabe								
11h 17	1	Zeitformat Bit 0 = 0 für eine 12-Stundenanzeige Bit 0 = 1 für eine 24-Stundenanzeige								
12h 18	4	Adresse für Aufruf eines Programmes für die Behandlung von ASCII-Zeichen > 7Fh								
16h 22	2	Trennzeichen in Datenliste (mit 00h begrenzt)								
18h 24	10	Reserviert								

9.2.2.49.2. ABFRAGE EINER BELIEBIGEN LANDESSPEZIFISCHEN INFORMATION

INT21: 38h xxh

Aufruf:

Register	Inhalt/Bedeutung
AH	38h
DS:DX	Zeiger auf 34 Byte grosses Feld
AL	Landeskennzahl xx (000 - 254)
AL	FFh, wenn Landeskennzahl > 255
BX	Landeskennzahl

Sollen die landesspezifischen Informationen abgefragt werden, so ist in AL die entsprechende Landeskennzahl zu übergeben. Ist die Landeskennzahl 255 oder grösser, so muss zusätzlich zu AL=FFh das Register BX auf die entsprechende Landeskennzahl eingestellt sein. Die zurückgegebenen Informationen werden nach DS:DX eingetragen.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
DS:DX	Landesspezifische Informationen

Für die landesspezifische Information gilt die obige Tabelle.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	02h Landeskennzahl nicht vorhanden

9.2.2.49.3. SETZEN DER LANDESSPEZIFISCHEN INFORMATION

INT21: 38h xxh

Aufruf:

Register	Inhalt/Bedeutung
AH	38h
DX	FFFFh
AL	Landeskennzahl (000 - 254)
AL	FFh, wenn Landeskennzahl > 255
BX	Landeskennzahl

*** DOS-Funktionsaufrufe ***

Für das Setzen der landesspezifischen Information ist DX=FFFFh und in AL die gewünschte Landeskennzahl einzustellen.

Ist die Landeskennzahl 255 oder grösser, so muss zusätzlich zu AL=FFh das Register BX auf die entsprechende Landeskennzahl eingestellt sein.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	02h Landeskennzahl nicht vorhanden

Bemerkung:

Die Adresse für Aufruf eines Programmes (Offset 12h in der Tabelle) ist 4 Byte (Segment:Offset) lang. An dieser Adresse steht ggf. eine Behandlungsroutine zur landesspezifischen Umwandlung von Klein- in Grossbuchstaben für Zeichen, deren Kodierung grösser als 7Fh ist. Dazu ist das Zeichen in AL einzutragen. Das umgewandelte Zeichen wird nach Abarbeitung der Behandlungsroutine in AL zurückgegeben.

9.2.2.50. VERZEICHNIS ERSTELLEN (MKDIR)

INT21: 39h

Funktion:

Anlegen eines neuen Verzeichnisses.

Aufruf:

Register	Inhalt/Bedeutung
AH	39h
DS:DX	Zeiger auf ASCII-Zeichenfolge

DS:DX enthält die Adresse einer ASCII-Zeichenfolge mit Laufwerk und Verzeichnispfad.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0 Verzeichnis erstellt

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1 Verzeichnis nicht erstellt
AX	03h Pfad nicht gefunden
AX	05h Zugriff abgelehnt

Der Zugriff wird abgelehnt, wenn

- bereits eine Datei mit dem für das Verzeichnis vorgesehenen Namen existiert,
- der Name ungültig ist (nicht zulässige Sonderzeichen bzw. Bezeichnungen eines Gerätes),
- im übergeordneten Verzeichnis kein Platz für das zu erstellende Verzeichnis vorhanden ist..

Bemerkung:

Diese DOS-Funktion ermöglicht den Aufbau eines hierarchischen Dateisystems, indem zu jedem Verzeichnis ein, untergeordnetes Verzeichnis erstellt werden kann.

Netzwerk-Zugriffsrechte:

Es sind Anlage-Zugriffsrechte erforderlich.

9.2.2.51. VERZEICHNIS LÖSCHEN (RMDIR)

INT21: 3Ah

Funktion:

Löschen eines leeren Verzeichnisses.

Aufruf:

Register	Inhalt/Bedeutung
AH	3Ah
DS:DX	Zeiger auf ASCII-Zeichenfolge

DS:DX enthält die Adresse einer ASCII-Zeichenfolge mit Laufwerk und Verzeichnispfad.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0 Verzeichnis gelöscht

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1 Verzeichnis nicht gelöscht
AX	03h Pfad nicht gefunden
AX	05h Zugriff abgelehnt
AX	10h Versuch, das aktuelle Verzeichnis zu löschen

Der Zugriff wird abgelehnt, wenn

- das angegebene Verzeichnis nicht existiert,
- das angegebene Verzeichnis nicht leer ist,
- der Name ungültig ist (nicht zulässige Sonderzeichen bzw. Bezeichnungen eines Gerätes),
- der angegebene Pfad kein Verzeichnis darstellt,
- das hierarchisch am höchsten priorisierte Verzeichnis (Wurzelverzeichnis) angegeben ist.

Bemerkung:

Das aktuelle Verzeichnis kann nicht gelöscht werden.

Netzwerk-Zugriffsrechte:

Es sind Anlage-Zugriffsrechte erforderlich.

9.2.2.52. AKTUELLES VERZEICHNIS WECHSELN (CHDIR) INT21: 3Bh

Funktion:

Wechsel des aktuellen Verzeichnisses.

Aufruf:

Register	Inhalt/Bedeutung
AH	3Bh
DS:DX	Zeiger auf ASCII-Zeichenfolge

DS:DX enthält die Adresse einer ASCII-Zeichenfolge mit Laufwerk und Verzeichnispfad. Die Folge (Gesamtpfad) ist auf 64 Zeichen begrenzt und kann keinen Netzwerkpfad enthalten.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0 Aktuelles Verzeichnis gewechselt

*** DOS-Funktionsaufrufe ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1 Aktuelles Verzeichnis nicht gewechselt
AX	03h Pfad nicht gefunden

9.2.2.53. HANDLER ERSTELLEN (CREAT)

INT21: 3Ch

Funktion:
Anlegen eines neuen Händlers.

Aufruf:

Register	Inhalt/Bedeutung
AH	3Ch
CX	Dateiattribut
DS:DX	Zeiger auf ASCII-Z-Dateispezifikation

CX enthält das Attribut der Datei.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0 Neuer Händler angelegt
AX	1 Händlernummer

Es wurde entweder eine neue Datei erstellt, oder in einer eventuell bereits existierenden Datei wurden die Daten "gelöscht" (Länge = 0).

Der Datei wurde die erste zur Verfügung stehende Händlernummer zugewiesen. Der angelegten Datei wird das Attribut zugeordnet. Die Datei wird für Lesen/Schreiben eröffnet.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1 Kein neuer Händler angelegt
AX	03h Pfad nicht gefunden
AX	04h Zu viele eröffnete Dateien (keine Dateinummern mehr frei)
AX	05h Zugriff abgelehnt

Wenn zu viele eröffnete Dateien gemeldet werden, so reicht die durch das Kommando FILES installierte Zahl für gleichzeitig eröffnete Dateien nicht aus.

*** DOS-Funktionsaufrufe ***

Der Zugriff wird abgelehnt, wenn

- das angegebene Verzeichnis keine weiteren Einträge aufnehmen kann,
- bereits eine Datei existiert, deren Attribute höhere Priorität besitzen (z.B. "Verborgen").

Netzwerk-Zugriffsrechte:

Es sind Anlege-Zugriffsrechte erforderlich.

9.2.2.54. HANDLER ERÖFFNEN

INT21: 3Dh

Funktion:

Eröffnen einer Datei über Händler.

Aufruf:

Register	Inhalt/Bedeutung
AH	3Dh
AL	Eröffnungs-Modus
DS:DX	Zeiger auf ASCII-Z-Dateispezifikation

Durch diese DOS-Funktion kann jede Datei (auch die mit dem Attribut "Verborgen" bzw. "System") eröffnet werden.

Der Eröffnungs-Modus, der in AL übergeben wird, besteht aus folgenden 4 Bit-Feldern:

- Bit 7 : Übernahmeflag (Inheritance Flag)
- Bit 6 - 4 : Nutzungsartenfeld (Sharing Mode Field)
- Bit 3 : Reservefeld (Reserved Field)
- Bit 2 - 0 : Zugriffsfeld (Access Field)

Das Übernahmeflag kennzeichnet, dass die zu eröffnende Datei von einem untergeordneten Prozess übernommen wird. Das Zugriffsfeld definiert die Operationen, die dieser Prozess mit der Datei ausführen kann und das Nutzungsartenfeld definiert die Operationen, die andere Prozesse mit der Datei ausführen können. Eine detaillierte Beschreibung der einzelnen Felder des Eröffnungs-Modus erfolgt unter dem Punkt "Bemerkungen".

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0 / Händler eröffnet
AX	Händlernummer

Der Les-/Schreibzeiger wird auf das erste Byte der Datei gesetzt. Die Rekordgröße der Datei beträgt 1 Byte. Die zurückgegebene Händlernummer muss für die nachfolgenden Ein- und Ausgaben von bzw. zu der Datei benutzt werden.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1 Kein Händler eröffnet
AX	02h Datei nicht gefunden
AX	03h Pfad nicht gefunden
AX	04h Zu viele eröffnete Dateien (keine Dateinummern mehr)
AX	05h Zugriff abgelehnt
AX	0Ch Ungültiger Zugriffskode

*** DOS-Funktionen ***

Wenn zu viele eröffnete Dateien (AL=04h) gemeldet werden, reicht die durch das Kommando FILES installierte Zahl für gleichzeitig eröffnete Dateien nicht aus.

Der Zugriff wird abgelehnt (AL=05h), wenn

- ein Eintrag für einen Datenträger bzw. ein Verzeichnis eröffnet wird,
- eine Datei mit dem Attribut "Nur Lesen" für einen Schreibvorgang eröffnet wird.

Bemerkung:

Wenn eine Datei geschlossen wird, werden alle Einschränkungen für eine gemeinsame Dateinutzung, die bei der Eröffnung für sie festgelegt wurden, beendet.

Für das Festlegen von Nutzungsarten für eine Datei muss das SHARE-Kommando ausgeführt werden.

Das Dateiattribut "Nur Lesen" kann durch

- Erzeugung einer Datei über erweiterten FCB,
- Festlegung des Attributs in CX für den Händler und Aufruf der DOS-Funktion "Dateiattribut abfragen/verändern" (CHMOD) (43h) oder
- das ATTRIB-Kommando gesetzt werden.

Wenn eine Datei von einem untergeordneten Prozess übernommen wird, werden auch alle Nutzungs- und Zugriffseinschränkungen übernommen.

Bei Verwendung der DOS-Funktion "Händler duplizieren (DUP)" (45h) werden alle Nutzungs- und Zugriffseinschränkungen mit dupliziert.

Beschreibung der einzelnen Felder des Eröffnungs-Modus:

Bit 7: Übernahmeflag

- = 0 Datei ist vom untergeordneten Prozess übernommen
- = 1 Datei nur für den augenblicklichen Prozess

Bit 6-4: Nutzungsartenfeld

- = 000 Kompatibilitätsmodus
- = 001 Lesen/Schreiben unterbinden (exklusiv)
- = 010 Nur Schreiben unterbinden
- = 011 Nur Lesen unterbinden
- = 100 Nichts unterbinden

Wenn eine Datei eröffnet wird, muss das Betriebssystem darüber informiert werden, welche Operationen andere Prozesse mit dieser Datei ausführen dürfen (Nutzungsart). Der Standardfall (Kompatibilitätsmodus) versagt allen anderen Prozessen den Zugriff auf diese Datei.

Es ist möglich, dass ein anderer Prozess fortlaufend die Datei lesen will, mit der der aktuelle Prozess gerade arbeitet. In diesem Fall sollte man die Nutzungsart "Nur Schreiben unterbinden" festlegen. Dies verhindert das Schreiben, aber ermöglicht das Lesen durch den anderen Prozess.

- Bit 3: Reservefeld**
Dieses Bit ist auf 0 zu setzen.
- Bit 2-0: Zugriffsfeld**
= 000 Nur Lesen
= 001 Nur Schreiben
= 010 Lesen/Schreiben

Alle anderen Kombinationen werden nicht berücksichtigt.

Die Zugriffsart gibt eine Aussage darüber, welche Operationen der eigene Prozess mit der Datei ausführen kann.

Im Standardfall "Lese-/Schreib-Zugriff" wird die Eröffnung einer Datei untersagt, wenn bereits ein anderer Prozess diese Datei mit einer anderen Nutzungsart ausser "Nichts unterbinden" eröffnet hat.

Das Eröffnen einer Datei wird immer erfolgreich sein, wenn man nur von dieser Datei zu lesen beabsichtigt. Dazu müssen andere Prozesse jedoch für diese Datei die Nutzungsarten "Nichts unterbinden" bzw. "Nur Schreiben unterbinden" festgelegt haben. Hierdurch können mehrere Prozesse auf diese Datei zugreifen. Eine gemeinsame Dateinutzung erfordert ein bestimmtes Zusammenwirken von allen beteiligten Prozessen. Dieses wird durch die Nutzungs- und Zugriffsart festgelegt.

Beschreibung der Nutzungsarten

a) Kompatibilitätsmodus

Eine Datei befindet sich im Kompatibilitätsmodus, wenn sie durch

- die DOS-Funktionsaufrufe "Händler erstellen (CREAT)" (3Ch), "Einzigartige Datei erstellen" (5Ah) bzw. "Neuen Händler erstellen" (5Bh),
- die DOS-Funktionsaufrufe "Datei eröffnen" (0Fh) bzw. "Datei erstellen" (16h) oder
- den DOS-Funktionsaufruf "Händler erstellen (CREAT)" (3Ch) mit spezifiziertem Kompatibilitätsmodus eröffnet wurde.

Eine Datei kann mehrmals im Kompatibilitätsmodus durch einen einfachen Prozess eröffnet werden. Das setzt voraus, dass diese Datei momentan nicht durch eine der anderen vier Nutzungsarten geöffnet ist.

Wenn die Datei als "Nur Lesen" markiert und augenblicklich in der Nutzungsart "Nur Schreiben unterbinden" mit der Zugriffsart "Lesen" offen ist, kann die Datei im Kompatibilitätsmodus mit Zugriffsart "Lesen" eröffnet werden.

Wurde die Datei erfolgreich in einer der anderen Nutzungsarten eröffnet und ein Versuch unternommen, die Datei nochmals im Kompatibilitätsmodus zu eröffnen, signalisiert ein Interrupt 24h den Fehler 16h (Laufwerk nicht bereit). Der Aufruf des erweiterten Fehlers zeigt dann an, dass es sich um eine Verletzung der gemeinsamen Dateinutzung handelte.

Die Nutzungsarten einer im Kompatibilitätsmodus eröffneten Datei werden vom Betriebssystem geändert, wenn diese Datei das Attribut "Nur Lesen" besitzt. Damit wird die gemeinsame Nutzung von Dateien mit diesem Attribut ermöglicht.

*** DOS-Funktionen ***

Die folgenden Tabellen zeigen die vom Betriebssystem eingestellten Nutzungsarten bei verschiedenen Dateieröffnungsverfahren und Dateiattributen.

Dateiattribut "Nur Lesen"		
Dateieröffnung durch	Zugriff	Nutzungsart
FCB	Nur Lesen	Nur Schreiben unterbinden
Händler (Lesen)	Nur Lesen	Nur Schreiben unterbinden
Händler (Schreiben)	Fehler	entfällt
Händler (Lesen/Schreiben)	Fehler	entfällt

Dateiattribut nicht "Nur Lesen"		
Dateieröffnung durch	Zugriff	Nutzungsart
FCB	Lesen/Schreiben	Kompatibilität
Händler (Lesen)	Nur Lesen	Kompatibilität
Händler (Schreiben)	Nur Schreiben	Kompatibilität
Händler (Lesen/Schreiben)	Lesen/Schreiben	Kompatibilität

b) Modus "Lesen/Schreiben unterbinden"

Wurde eine Datei erfolgreich in diesem Modus eröffnet, ist der Zugriff auf diese exklusiv, d. h., es kann sowohl gelesen als auch geschrieben werden. Ein anderer Prozess kann auf diese Datei nicht zugreifen. Eine augenblicklich in diesem Modus eröffnete Datei kann nicht nochmals in irgendeiner Nutzungsart durch irgendeinen Prozess (einschliesslich dem aktuellen Prozess) eröffnet werden, bevor sie nicht geschlossen wurde.

c) Modus "Nur Schreiben unterbinden"

Eine Datei, die erfolgreich in diesem Modus eröffnet wurde, ist vor jeglichen anderen Eröffnungen mit definierten Zugriffsarten für gemeinsame Schreibvorgänge (Bit 2-0 = 001 oder Bit 2-0 = 010) geschützt. Dies wird erst mit dem Schliessen der Datei aufgehoben. Der Versuch, eine Datei im Modus "Nur Schreiben unterbinden" zu eröffnen, ist nicht erfolgreich, wenn die Datei augenblicklich in der Zugriffsart "Nur Schreiben" offen ist.

d) Modus "Nur Lesen unterbinden"

Eine Datei, die erfolgreich in diesem Modus eröffnet wurde, ist vor jeglichen anderen Eröffnungen mit definierten Zugriffsarten für gemeinsame Lesevorgänge (Bit 2-0 = 000 oder Bit 2-0 = 010) geschützt. Dies wird erst mit dem Schliessen der Datei aufgehoben.

*** DOS-Funktionen ***

Der Versuch, eine Datei im Modus "Nur Lesen unterbinden" zu eröffnen, ist nicht erfolgreich, wenn die Datei augenblicklich in der Zugriffsart "Nur Lesen" offen ist.

e) Modus "Nichts unterbinden"

Eine erfolgreich in diesem Modus eröffnete Datei besitzt keine Einschränkungen an die Zugriffsmöglichkeiten für Lese-/ Schreibvorgänge.

Der Versuch, eine Datei in der Nutzungsart "Nichts unterbinden" zu eröffnen, ist erfolglos, wenn die Datei augenblicklich im Kompatibilitätsmodus eröffnet ist.

Die folgende Matrix zeigt die Ergebnisse von Eröffnungen und folgenden Wiedereröffnungsversuchen einer Datei unter Benutzung aller Kombinationen der Nutzungs- und Zugriffsarten:

2., 3., ... Eröffnung

		DRW		DW		DR		alles	
		I	O	I	O	I	O	I	O
1.	D	I	N	N	N	N	N	N	N
	R	O	N	N	N	N	N	N	N
	W	O	N	N	N	N	N	N	N
E r ö f f n u n g	D	I	N	N	N	J	N	N	N
	W	O	N	N	N	N	N	N	N
	R	O	N	N	N	N	N	J	N
a l l e s	D	I	N	N	N	N	J	N	N
	W	O	N	N	N	N	N	J	N
	R	O	N	N	N	N	N	N	J

- J : 2. und 3. Eröffnung ist erlaubt
- N : 2. und 3. Eröffnung ist nicht erlaubt
- DRW : Nutzungsart "Lesen/Schreiben unterbinden" (exklusiv)
- DW : Nutzungsart "Nur Schreiben unterbinden"
- DR : Nutzungsart "Nur Lesen unterbinden"
- I : Zugriffsart "Nur Lesen"
- O : Zugriffsart "Nur Schreiben"
- IO : Zugriffsart "Lesen/Schreiben"

*** DOS-Funktionen ***

Wenn auf Dateien zugegriffen wird, die sich auf einem Datenträger aus dem Netzwerk befinden, erfolgt keine lokale Pufferung der Daten, wenn diese Datei in den Nutzungsarten:

- "Nur Lesen unterbinden" oder
- "Nichts unterbinden" eröffnet wurde.

Somit werden in einer Netzwerkumgebung die Nutzungsarten "Lesen/Schreiben unterbinden", "Kompatibilitätsmodus" und "Nur Schreiben unterbinden" lokal gepuffert.

Netzwerk-Zugriffsrechte:

Diese sind erforderlich, wenn im Eröffnungs-Modus das Zugriffsfeld (Bit 2-0 von AL) folgendermassen gesetzt ist:

- 000 : Erfordert Lese-Zugriffsrecht
- 001 : Erfordert Schreib-Zugriffsrecht
- 010 : Erfordert Lese-/Schreib-Zugriffsrecht.

9.2.2.55. HANDLER SCHLIESSEN

INT21: 3EH

Funktion:

Beenden des Zugriffs auf eine Datei oder Gerät über einen Händler.

Aufruf:

Register	Inhalt/Bedeutung
AH	3EH
BX	Händlernummer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0 Händler geschlossen

Nach Funktionsausführung ist die Datei geschlossen und das Inhaltsverzeichnis aktualisiert. Ausserdem werden alle internen Dateipuffer abgespeichert und anschliessend gelöscht.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	06h Ungültige Händlernummer

Der Fehler tritt auf, wenn der angegebene Händler nicht eröffnet oder ungültig ist.

9.2.2.56. LESEN VON DATEI/EINHEIT

INT21: 3Fh

Funktion:

Lesen einer bestimmten Anzahl Zeichen von einer Datei oder einem Gerät über Händler.

Aufruf:

Register	Inhalt/Bedeutung
AH	3Fh
BX	Händlernummer
CX	Anzahl zu lesender Bytes
DS:DX	Zeiger auf Puffer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Anzahl gelesener Bytes

Wird vor Erreichen der in CX übergebenen Anzahl beim Lesen auf einen gültigen Begrenzer (z.B. 0Dh beim Lesen von CON oder AUX bzw. 1Ah beim Lesen von einer Datei) getroffen, so wird in AX die Anzahl der tatsächlich gelesenen Bytes übergeben.

Werden mehr als CX Zeichen angeboten, so wird nur die geforderte Anzahl gelesen, und die restlichen werden ohne Meldung ignoriert.

Wenn in AX der Wert 00h zurückgegeben wird, dann wurde versucht, nach dem Dateiende zu lesen.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	05h Zugriff abgelehnt
AX	06h Ungültige Händlernummer

Der Zugriff (AL=05h) wird abgelehnt, wenn der Händler nicht für Lesezugriffe eröffnet ist.

Der Fehler 06h tritt auf, wenn der angegebene Händler nicht eröffnet oder ungültig ist.

Bemerkung:

Wird der Lesevorgang über die Standardeingabeeinheit durchgeführt, so kann die Eingabe umgeleitet werden.

Netzwerk-Zugriffsrechte:

Es sind Lese-Zugriffsrechte erforderlich.

9.2.2.57. SCHREIBEN AUF DATEI/EINHEIT

INT21: 40h

Funktion:

Schreiben einer bestimmten Anzahl Zeichen auf eine Datei oder ein Gerät über Händler.

Aufruf:

Register	Inhalt/Bedeutung
AH	40h
BX	Händlernummer
CX	Anzahl zu schreibender Bytes
DS:DX	Zeiger auf Puffer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Anzahl geschriebener Bytes

AX gibt die Anzahl der tatsächlich geschriebenen Bytes zurück. Entspricht dieser Wert nicht der übergebenen Anzahl von Bytes, muss dies als Fehler angesehen werden (es wird in diesem Fall kein Fehlercode zurückgegeben, das Benutzerprogramm kann diese Werte jedoch vergleichen).

Wenn in AX der Wert 00h zurückgegeben wird, dann wurden keine Daten geschrieben, da der Datenträger bereits voll ist.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	05h Zugriff abgelehnt
AX	06h Ungültige Händlernummer

Der Zugriff (AL=05h) wird abgelehnt, wenn der Händler nicht für Schreibzugriffe eröffnet ist.

Der Fehler 06h tritt auf, wenn der angegebene Händler nicht eröffnet oder ungültig ist.

Bemerkung:

Geht dieser Schreibvorgang an die Standardausgabeeinheit, kann die Ausgabe umgeleitet werden.

Netzwerk-Zugriffsrechte:

Es sind Schreib-Zugriffsrechte erforderlich.

9.2.2.58. LÖSCHEN EINER DATEI AUS ANGEGEBENEM
VERZEICHNIS (UNLINK)

INT21: 41h

Funktion:

Löschen einer Datei aus einem Verzeichnis.

Aufruf:

Register	Inhalt/Bedeutung
AH	41h
DS:DX	Zeiger auf ASCII-Z-Dateispezifikation

Dieser Funktionsaufruf löscht einen mit einer Dateibezeichnung verknüpften Verzeichniseintrag. Globale Dateibezeichner sind in der Dateispezifikation nicht zulässig. Dateien mit dem Attribut "Nur Lesen" oder "Verborgnen" können durch diesen Aufruf nicht gelöscht werden.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0 Datei gelöscht

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1 Datei nicht gelöscht
AX	02h Datei nicht gefunden
AX	03h Pfad nicht gefunden
AX	05h Zugriff abgelehnt

Der Fehler 03h tritt z.B. auf, wenn globale Dateibezeichner verwendet werden.

Der Zugriff wird abgelehnt (AL=05h), wenn die angegebene Datei das Attribut "Nur Lesen" besitzt bzw. ein Verzeichnis gelöscht werden soll.

Netzwerk-Zugriffsrechte:

Es sind Anlege-Zugriffsrechte erforderlich.

Funktion:

Verschieben des Lese-/Schreibzeigers innerhalb einer Datei.

Aufruf:

Register	Inhalt/Bedeutung
AH	42h
BX	Händlernummer
CX:DX	Offset der Verschiebung in Bytes (CX enthält höherwertiges WORD)
AL	00h Der Offset wird auf den Dateibeginn addiert (absolutes Positionieren innerhalb der Datei).
AL	01h Der Offset wird zum momentanen Stand des Schreib-/Lesezeigers addiert (relative Positionierung in Richtung Dateifeinde).
AL	02h Der Offset wird zum Dateifeinde addiert (Erweiterung der Datei).

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0 Verschiebung erfolgreich
DX:AX	Neuer Stand des Zeigers in Bytes (DX enthält höherwertiges WORD)

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1 Verschiebung nicht erfolgreich
AX	01h Ungültige Funktionsnummer
AX	06h Ungültige Händlernummer

Der Fehler 01h tritt auf, wenn diese DOS-Funktion aufgerufen wird, wobei in AL nicht die Werte 00h, 01h oder 02h angegeben wurden.

Wurde bei Aufruf der DOS-Funktion eine ungültige oder nicht eröffnete Händlernummer in AX angegeben, wird AL=06h zurückgegeben.

Bemerkung:

Wenn diese DOS-Funktion auf eine Datei eines Netzwerkdatenträgers ausgeführt wird, ist folgendes zu beachten:

*** DOS-Funktionen ***

Ist die Datei in den Nutzungsarten "Nur Lesen unterbinden" oder "Nur Schreiben unterbinden" eröffnet, wird der Lese-/Schreibzeiger auf dem Computer verändert, auf dem die Datei tatsächlich existiert. Ist die Datei in einer anderen Nutzungsart eröffnet, wird der Lese-/Schreibzeiger auf dem entfernten Computer beibehalten.

9.2.2.60. DATEIATTRIBUT ABFRAGEN/VERÄNDERN (CHMOD)

INT21: 43h

Funktion:

Abfragen bzw. Verändern eines Dateiattributes.

Aufruf:

Register	Inhalt/Bedeutung
AH	43h
DS:DX	Zeiger auf ASCIIZ-Dateispezifikation
AL	00h Dateiattribut abfragen
AL	01h Dateiattribut setzen
CX	Dateiattribut

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
CX	Dateiattribut (bei AL=00h)

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	02h Datei nicht gefunden
AX	03h Pfad nicht gefunden
AX	05h Zugriff abgelehnt

Der Fehler 01h tritt auf, wenn diese DOS-Funktion aufgerufen wird, wobei in AL nicht die Werte 00h oder 01h angegeben wurden. Der Zugriff wird abgelehnt, wenn die Attribute "Verzeichnis" oder "Datenträgerkennsatz" mit gesetzt werden sollen. In diesem Fall wird auch kein anderes mit angegebenes Attribut gesetzt.

Netzwerk-Zugriffsrechte:

Um das Attribut "Archiv" (AL = 20h) zu ändern, werden keine Zugriffsrechte gefordert. Um ein anderes Attribut zu ändern, sind Anlege-Zugriffsrechte erforderlich.

9.2.2.61. E/A-STEUERUNG FÜR EINHEITEN (IOCTL)

INT21: 44h

Funktion:

Abfragen und Verändern von Geräteinformationen sowie geräte-spezifische Ein- und Ausgaben.

Diese DOS-Funktion ist sehr umfangreich. In AL werden die jeweiligen Unterfunktionen definiert. Die folgenden Werte sind in AL erlaubt:

AL = 00h	Abfrage von Händlerinformationen
AL = 01h	Setzen von Händlerinformationen
AL = 02h	Lesen vom Zeichengerät
AL = 03h	Schreiben auf Zeichengerät
AL = 04h	Lesen vom Blockgerät
AL = 05h	Schreiben auf Blockgerät
AL = 06h	Eingabestatus abfragen
AL = 07h	Ausgabestatus abfragen
AL = 08h	Abfrage, ob ein einzelnes Blockgerät auswechselbar ist
AL = 09h	Abfrage, ob ein logisches Gerät lokal oder entfernt ist
AL = 0Ah	Abfrage, ob ein Händler lokal oder entfernt ist
AL = 0Bh	Verändern des Zählers für Wiederholungsversuche bei gemeinsamer Dateinutzung
AL = 0Ch	Art und Weise der E/A-Steuerung für Händler
AL = 0Dh	Art und Weise der E/A-Steuerung für Blockgeräte
AL = 0Eh	Abfrage des logischen Laufwerkes
AL = 0Fh	Setzen des logischen Laufwerkes

Die E/A-Steuerung kann zur Abfrage von Informationen über die Geräte genutzt werden. Handelt es sich um normale Dateien, so sind nur die Unterfunktionen 00h, 06h und 07h definiert. Alle anderen Unterfunktionen liefern in diesem Fall einen Fehler.

Die Unterfunktionen 00h bis 08h werden nicht auf Netzwerkgeräten unterstützt. Vor Aufruf der Unterfunktion 0Bh muss das Kommando SHARE ausgeführt worden sein.

9.2.2.61.1. Abfrage von Händlerinformationen

INT21: 44h 00h

Funktion:

Abfrage von Informationen über den Händler.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	00h
BX	Händlernummer

*** DOS-Funktionen ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
DX	Händlerinformation

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	06h Ungültige Händlernummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.

Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

Bemerkung:

Beziehen sich die Händlerinformationen auf ein Gerät und nicht auf eine Datei, so gilt folgende Tabelle:

Bit	Wert	Bezeichnung	Bedeutung
0Fh 15			Gerät ist Netzwerkgerät
0Eh 14	1	CTRL	Gerät kann Steuerdaten empfangen oder abgeben (AL=2,AL=3); kann nur abgefragt werden
	0		Gerät kann keine Steuerdaten empfangen oder abgeben (AL=2,AL=3, AL=4,AL=5)
0Dh 13			Reserviert
0Ch 12			Reserviert
0Bh 11			Reserviert
0Ah 10			Reserviert
09h 9			Reserviert
08h 8			Reserviert
07h 7	1	ISDEV	Händler spezifiziert ein Gerät
06h 6	0	EOF	Beim Lesen Dateiende erreicht
05h 5	1	BINARY	Binär-Modus
	0		Keine Reaktion auf CTRL-Zeichen
			ASCII-Modus
			Reaktion auf CTRL-Zeichen
04h 4			Reserviert
03h 3	1	ISCLK	Gerät ist die Systemuhr
02h 2	1	ISNUL	Gerät ist Null-Gerät
01h 1	1	ISCOT	Gerät ist Standardausgabegerät
00h 0	1	ISCIN	Gerät ist Standardeingabegerät

*** DOS-Funktionen ***

Bei BINARY=1 werden die Steuerzeichen CTRL-C, CTRL-P, CTRL-S und CTRL-Z als Daten (03h, 10h, 13h, 1Ah) interpretiert. Zusätzlich muss durch die DOS-Funktion "Prüfung auf CTRL-C" (33h) oder durch das Kommando BREAK die CTRL-C-Unterbrechung ausgeschaltet sein.

Beziehen sich die Händlerinformationen auf eine Datei und nicht auf ein Gerät, so gilt folgende Tabelle:

Bit	Wert	Bezeichnung	Bedeutung
0Fh	15		Reserviert
0Eh	14		Reserviert
0Dh	13		Reserviert
0Ch	12		Reserviert
0Bh	11		Reserviert
0Ah	10		Reserviert
09h	9		Reserviert
08h	8		Reserviert
07h	7	0	ISDEV Händler spezifiziert eine Datei
06h	6	0	EOF Auf Händler wurde geschrieben
05h	5		**
04h	4		**
03h	3		**
02h	2		**
01h	1		**
00h	0		**

**

Die Bits 5 bis 0 geben die Laufwerknummer (0=LW A, 1=LW B, ...) der Datei, die durch den Händler bestimmt wird, an.

9.2.2.61.2. Setzen von Händlerinformationen INT21: 44h 01h

Funktion:
Setzen von Informationen über den Händler.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	01h
BX	Händlernummer
DH	00h
DL	Händlerinformation

Die reservierten Bits und das Bit CTRL können nicht gesetzt werden.

*** DOS-Funktionen ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	06h Ungültige Händlernummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.
 Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

9.2.2.61.3. Lesen vom Zeichengerät

INT21: 44h 02h

Funktion:

Lesen von beliebigen Steuerzeichen vom zeichenorientierten Gerät in einen Puffer.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	02h
BX	Händlernummer
CX	Anzahl zu lesender Bytes
DS:DX	Zeiger auf Puffer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Anzahl der tatsächlich gelesenen Bytes

*** DOS-Funktionen ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	06h Ungültige Händlernummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder wenn das Bit CTRL=0 ist. Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

9.2.2.61.4. Schreiben auf Zeichengerät INT21: 44h 03h

Funktion:

Schreiben von beliebigen Steuerzeichen aus dem Puffer zum zeichenorientierten Gerät.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	03h
BX	Händlernummer
CX	Anzahl zu schreibender Bytes
DS:DX	Zeiger auf Puffer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Anzahl der tatsächlich geschriebenen Bytes

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	06h Ungültige Händlernummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder wenn das Bit CTRL=0 ist. Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

9.2.2.61.5. Lesen vom Blockgerät

INT21: 44h 04h

Funktion:

Lesen von beliebigen Steuerzeichen vom Blockgerät in einen Puffer.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	04h
BL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)
CX	Anzahl zu lesender Bytes
DS:DX	Zeiger auf Puffer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Anzahl der tatsächlich gelesenen Bytes

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	05h Zugriff abgelehnt

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder wenn das Bit CTRL=0 ist. Der Zugriff wird abgelehnt, wenn das angegebene Laufwerk ungültig ist.

*** DOS-Funktionen ***

9.2.2.61.6. Schreiben auf Blockgerät

INT21: 44h 05h

Funktion:

Schreiben von beliebigen Steuerzeichen aus dem Puffer zum Blockgerät.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	05h
BL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)
CX	Anzahl zu schreibender Bytes
DS:DX	Zeiger auf Puffer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Anzahl der tatsächlich geschriebenen Bytes

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	05h Zugriff abgelehnt

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder wenn das Bit CTRL=0 ist. Der Zugriff wird abgelehnt, wenn das angegebene Laufwerk ungültig ist.

9.2.2.61.7. Eingabestatus abfragen

INT21: 44h 06h

Funktion:

Prüfen, ob ein Gerät oder eine Datei für Eingaben bereit ist.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	06h
BX	Händlernummer

*** DOS-Funktionen ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AL	00h für Gerät: Gerät ist nicht empfangsbereit.
AL	FFh Gerät ist empfangsbereit.
	für Datei:
AL	00h Dateiende erreicht
AL	FFh Dateiende noch nicht erreicht

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	05h Zugriff abgelehnt
AX	06h Ungültige Händlernummer
AX	0Dh Ungültige Daten

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.

Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

9.2.2.61.8. Ausgabestatus abfragen

INT21: 44h 07h

Funktion:

Prüfen, ob ein Gerät oder eine Datei für Ausgaben bereit ist.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	07h
BX	Händlernummer

*** DOS-Funktionen ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AL	00h für Gerät: Gerät ist nicht sendebereit.
AL	FFh Gerät ist sendebereit.
AL	00h für Datei: Dateiende erreicht
AL	FFh Dateiende noch nicht erreicht

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	05h Zugriff abgelehnt
AX	06h Ungültige Händlernummer
AX	0Dh Ungültige Daten

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.

Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

9.2.2.61.9. Abfrage, ob ein einzelnes Blockgerät austauschbar ist

INT21: 44h 08h

Funktion:

Information darüber, ob ein Gerät austauschbare Datenträger (Diskette) unterstützen kann.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	08h
BL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	00h Austauschbar
AX	01h Nicht austauschbar

*** DOS-Funktionen ***

Ein austauschbarer Datenträger ist z.B. eine Diskette, während ein nicht austauschbarer Datenträger z.B. die Festplatte ist.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	0Fh Angabe eines ungültigen Laufwerks

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder wenn das Gerät diesen Aufruf nicht unterstützt (z.B. Netzwerkgeräte).

9.2.2.61.10. Abfrage, ob ein logisches Gerät lokal oder entfernt ist

INT21: 44h 09h

Funktion:

Information darüber, ob ein logisches Gerät mit einem Netzwerkverzeichnis verbunden ist oder nicht.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	09h
BL	Laufwerksnummer (0=Standard-LW, 1=LW A, ...)

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
DX	Bit 12=0 Gerät ist lokal. Bit 12=1 Gerät ist entfernt.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	0Fh Angabe eines ungültigen Laufwerkes

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder wenn das Kommando SHARE nicht geladen wurde.

9.2.2.61.11. Abfrage, ob ein Händler lokal
oder entfernt ist

INT21: 44h 0Ah

Funktion:

Information darüber, ob ein Händler auf ein lokales oder entferntes Gerät im Netzwerk weist.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Ah
BX	Händlernummer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
DX	Bit 15=0 Gerät ist lokal. Bit 15=1 Gerät ist entfernt.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	06h Ungültige Händlernummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder wenn das Netzwerkprogramm nicht geladen wurde.

Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

**9.2.2.61.12. Verändern des Zählers für Wiederholungsversuche
bei gemeinsamer Dateieinutzung** INT21: 44h 0Bh

Funktion:

Einstellen der Anzahl der Wiederholungsversuche und der Pause zur nächsten Wiederholung bei einer gemeinsamen Dateieinutzung.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Bh
CX	Dauer der Pause
DX	Anzahl der Wiederholungen

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder wenn das Kommando SHARE nicht geladen wurde.

Bemerkung:

Alle Operationen, die zu Dateieinutzungs- und Verriegelungskonflikten führen, werden automatisch eine gewisse Anzahl oft wiederholt, bevor sie als Fehler zurückgegeben werden. Man kann die Anzahl der Wiederholungen (DX) und die Pause (CX) dazwischen wählen.

Die Pause besteht aus folgender Sequenz:

```
XOR  CX, CX
LOOP R      ; Schleifenanzahl 64 KByte.
```

9.2.2.61.13. Art und Weise der E/A-Steuerung für
Händler

INT21: 44h 0Ch

Funktion:

Dieser Aufruf ermöglicht die Überleitung von Unterfunktionsaufrufe zur Umstellung von Zeichensätzen an einen Gerätetreiber.

Folgende Funktionen sind möglich:

- Beginn der Vorbereitung von Zeichensatztabellen
- Ende der Vorbereitung von Zeichensatztabellen
- Auswahl bzw. Aktivierung einer Zeichensatztabelle
- Anzeige der aktuellen Zeichensatztabelle
- Anzeige der vorbereiteten Zeichensatztabellen

Eine Refresh-Operation fordert den Gerätetreiber auf, die zuvor aktiven Zeichensatztabellen zu reaktivieren. Durch Ausführung von "Beginn der Vorbereitung von Zeichensatztabellen" mit 0FFFFh für alle Zeichensatztabellen (ab Offset 06h des Parameterblockes) wird die Refresh-Operation erzwungen.

9.2.2.61.13.1. Beginn der Vorbereitung von
Zeichensatztabellen

INT21: 44h 0Ch 4Ch

Funktion:

Dieser Aufruf dient der Vorbereitung zur Umstellung von Zeichensatztabellen an einen Gerätetreiber.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Ch
BX	Händlernummer des eröffneten Gerätes
CH	Gerätetreiberkennzeichen
CL	4Ch
DS:DX	Zeiger auf Parameterblock

Für das Gerätetreiberkennzeichen gilt:

- CH = 00h unbekanntes Gerät
- CH = 01h COM
- CH = 03h CON
- CH = 05h LPT

*** DOS-Funktionen ***

Der Parameterblock hat folgenden Aufbau:

Offset hex/dez	Anzahl Bytes	Bedeutung
00h 00	2	Kennzeichen
02h 02	2	(n+1)*2 = Länge der nachfolgenden Tabelle
04h 04	2	n = Anzahl der nachfolgenden Zeichensätze
06h 06	2	Zeichensatztabelle 1
08h 08	2	Zeichensatztabelle 2
0Ah 10	:	:
:	:	:
:	2	Zeichensatztabelle n

Wenn in dem Offset für eine Zeichensatztabelle 0FFFFh steht, so kann die Zeichensatztabelle für den Gerätetreiber nicht gegen eine andere ausgetauscht werden. Jeder andere Wert entspricht der vorzubereitenden Zeichensatztabelle.

Der Wert n entspricht der Anzahl der durch die Konfigurationsdatei CONFIG.SYS installierten Gerätetreiber (DEVICE= Befehl für den Treiber), wobei n Werte bis zu 12 annehmen kann.

Für Steckmodule (Hardware-Zeichensatztabelle) ist Offset 00h auf 1 zu setzen.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	16h Unbekanntes Kommando
AX	1Bh Fehlerhafte Zeichensatztabelle
AX	1Dh Gerätefehler

Um mehr Informationen über den Fehler zu erhalten, ist die DOS-Funktion 59h (Erweiterte Fehlermeldung) zu nutzen.

9.2.2.61.13.2. Ende der Vorbereitung von Zeichensatztabellen

INT21: 44h 0Ch 4Dh

Funktion:

Dieser Aufruf ist bei Abschluss zur Umstellung von Zeichensatztabellen eines Gerätetreibers zu verwenden.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Ch
BX	Händlernummer des eröffneten Gerätes
CH	Gerätetreiberkennzeichen
CL	4Dh
DS:DX	Zeiger auf Parameterblock

Für das Gerätetreiberkennzeichen gilt das bereits unter "Beginn der Vorbereitung von Zeichensatztabellen" geschriebene. Der Parameterblock hat hier folgenden Aufbau:

Offset	Anzahl	Bedeutung
hex/dez	Bytes	
00h 00	2	= 0002 (Länge)
02h 02	2	Zeichensatztabelle

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	13h Daten von der FNT-Datei fehlerhaft
AX	1Fh kein "Beginn der Vorbereitung von Zeichensatztabellen" ausgeführt

Um mehr Informationen über den Fehler zu erhalten, ist die DOS-Funktion 59h (Erweiterte Fehlermeldung) zu nutzen.

9.2.2.61.13.3. Auswahl bzw. Aktivierung einer Zeichensatzta-
belle

INT21: 44h 0Ch 4Ah

Funktion:

Dieser Aufruf dient der Auswahl einer Zeichensatzta-
belle für einen Gerätetreiber.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Ch
BX	Händlernummer des eröffneten Gerätes
CH	Gerätetreiberkennzeichen
CL	4Ah
DS:DX	Zeiger auf Parameterblock

Für das Gerätetreiberkennzeichen gilt das bereits unter "Beginn der Vorbereitung von Zeichensatztabellen" geschriebene. Der Parameterblock hat den gleichen Aufbau wie bei "Ende der Vorbereitung von Zeichensatztabellen".

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	1Ah Zeichensatzta- belle nicht vorbereitet
AX	1Bh aktuelles KEYB-Programm unterstüzt nicht diese Zeichensatzta- belle
AX	1Dh Gerätefehler

Um mehr Informationen über den Fehler zu erhalten, ist die DOS-
Funktion 59h (Erweiterte Fehlermeldung) zu nutzen.

9.2.2.61.13.4. Anzeige der aktuellen Zeichensatztabelle

INT21: 44h 0Ch 6Ah

Funktion:

Dieser Aufruf dient der Anzeige der aktuellen Zeichensatztabelle für einen Gerätetreiber.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Ch
BX	Händlernummer des eröffneten Gerätes
CH	Gerätetreiberkennzeichen
CL	6Ah
DS:DX	Zeiger auf Parameterblock

Für das Gerätetreiberkennzeichen gilt das bereits unter "Beginn der Vorbereitung von Zeichensatztabellen" geschriebene. Der Parameterblock hat den gleichen Aufbau wie bei "Ende der Vorbereitung von Zeichensatztabellen".

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	1Ah keine Zeichensatztabelle selektiert
AX	1Bh Gerätefehler

Um mehr Informationen über den Fehler zu erhalten, ist die DOS-Funktion 59h (Erweiterte Fehlermeldung) zu nutzen.

9.2.2.61.13.5. Anzeige der vorbereiteten Zeichensatztabellen

INT21: 44h 0Ch_68h

Funktion:

Dieser Aufruf dient der Anzeige der Liste der für einen aktiven Gerätetreiber vorbereiteten Zeichensatztabellen.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Ch
BX	Händlernummer des eröffneten Gerätes
CH	Gerätetreiberkennzeichen
CL	68h
DS:DX	Zeiger auf Parameterblock

Für das Gerätetreiberkennzeichen gilt das bereits unter "Beginn der Vorbereitung von Zeichensatztabellen" geschriebene. Der Parameterblock hat folgenden Aufbau:

Offset hex/dez	Anzahl Bytes	Bedeutung
00h 00	2	$((n+1)+(m+1))*2$ Länge der nachfolgenden Tabelle
02h 02	2	n = Anzahl der nachfolgenden Hardware-Zeichensätze
04h 04	2	Hardware-Zeichensatztable 1
06h 06	2	Hardware-Zeichensatztable 2
0Ah 10	:	:
:	:	:
:	2	Hardware-Zeichensatztable n
:	2	m = Anzahl der nachfolgenden vorbereiteten Zeichensätze
:	2	vorbereitete Zeichensatztable 1
:	2	vorbereitete Zeichensatztable 2
:	:	:
:	2	vorbereitete Zeichensatztable m

Die Gerätetreiber können bis zu 12 Zeichensatztabellen für jeden Typ (Hardware bzw. vorbereitete) enthalten.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

*** DOS-Funktionen ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	1Ah keine Zeichensatztabelle selektiert
AX	1Dh Gerätefehler

Um mehr Informationen über den Fehler zu erhalten, ist die DOS-Funktion 59h (Erweiterte Fehlermeldung) zu nutzen.

9.2.2.61.14. Art und Weise der E/A-Steuerung
für Blockgeräte

INT21: 44h 0Dh

Funktion:

Folgende E/A-Steuerung ist für die Blockgeräte möglich:

- Abfrage der Geräteparameter
- Setzen der Geräteparameter
- Lesen einer Spur vom logischen Gerät
- Schreiben einer Spur auf logisches Gerät
- Formatieren und Überprüfen einer Spur auf logischem Gerät
- Überprüfen einer Spur auf logischem Gerät.

9.2.2.61.14.1. Abfrage der Geräteparameter

INT21: 44h 0Dh 60h

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Dh
BL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)
CH	08h
CL	60h Abfrage der Geräteparameter
DS:DX	Zeiger auf Parameterblock

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	0
DS:DX	Gefüllter Parameterblock

*** DOS-Funktionen ***

Der Parameterblock hat folgenden Aufbau:

Offset hex/dez	Anzahl Bytes	Bezeichnung
00h 00	1	SpecialFunctions
01h 01	1	DeviceType
02h 02	2	DeviceAttributes
04h 04	2	NumberOfCylinders
06h 06	1	MediaType
		DeviceBPB
07h 07	2	BytesPerSector
09h 09	1	SectorsPerCluster
0Ah 10	2	ReservedSectors
0Ch 12	1	NumberOfFATs
0Dh 13	2	RootEntries
0Fh 15	2	TotalSectors
11h 17	1	MediaDescriptor
12h 18	2	SectorsPerFAT
14h 20	2	SectorsPerTrack
16h 22	2	Heads
18h 24	4	HiddenSectors
1Ch 28	4	Reserved_1
20h 32	6	Reserved_2
		TrackLayout
26h 38	2	SectorCount
28h 40	2	SectorNumber_1
2Ah 42	2	SectorSize_1
2Ch 44	2	SectorNumber_2
2Eh 46	2	SectorSize_2
:	:	:
:	2	SectorNumber_n
:	2	SectorSize_n

Für die einzelnen Felder gilt:

- Offset 00h (SpecialFunctions)

- Bit 0 = 1 Rückgabe der Adresse des BPB, die BUILD BPB zurückgeben würde
- Bit 0 = 0 Rückgabe der Adresse des Standard-BPB für dieses Gerät
- Bit 1-7=0

- Offset 01h (DeviceType)

Dieses Feld definiert den physischen Gerätetyp und kann nicht durch IOCTL gesetzt werden.

00h =	320/360 K Byte	5,25 Zoll
01h =	1,2 M Byte	5,25 Zoll
02h =	720 K Byte	3,5 Zoll
03h =	einfache Dichte	8 Zoll
04h =	doppelte Dichte	8 Zoll
05h =	Festplatte	
06h =	Magnetband	
07h =	Sonstiges	

Bei DCP-Disketten liefert das Betriebssystem den Wert 02h.

*** DOS-Funktionen ***

- Offset 02h (DeviceAttributes)
Dieses Feld definiert die physischen Attribute des Gerätes und kann nicht durch IOCTL gesetzt werden.
Bit 0 = 1 Datenträger ist nicht auswechselbar.
Bit 0 = 0 Datenträger ist auswechselbar.
Bit 1 = 1 Diskettenwechsellerkennung wird unterstützt.
Bit 1 = 0 Diskettenwechsellerkennung wird nicht unterstützt.
Bit 2-7 reserviert
- Offset 04h (NumberOfCylinders)
Dieses Feld gibt die maximale Anzahl von Zylindern an, die auf dem physischen Gerät unterstützt werden können (unabhängig vom Feld Offset 06h (MediaType)). Es kann nicht gesetzt werden.
- Offset 06h (MediaType)
Dieses Feld zeigt den momentanen Typ des im Laufwerk befindlichen Datenträgers an. Es wird nur genutzt, wenn der aktuelle Datenträger im Laufwerk nicht anders bestimmt werden kann (ist abhängig vom Gerätetyp).
Unabhängig vom Gerätetyp stellt der Wert 0 immer den Standard dar.
- DeviceBPB
Der Bereich von 07h bis 25h enthält die Struktur des BPB. Der BPB wird in Abhängigkeit des Bit 0 von Offset 00h (SpecialFunctions) gesetzt.
Bit 0 von Offset 00h = 1 Rückgabe der Adresse des BPB, die BUILD BPB zurückgeben würde;
Bit 0 von Offset 00h = 0 Rückgabe der Adresse des Standard-BPB für dieses Gerät.

9.2.2.61.14.2. Setzen der Geräteparameter INT21: 44h 0Dh 40h

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Dh
BL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)
CH	08h
CL	40h Setzen der Geräteparameter
DS:DX	Zeiger auf Parameterblock

Der Parameterblock ist analog der vorhergehenden Funktion aufgebaut.

Für die einzelnen Felder gilt:

- Offset 00h (SpecialFunctions)
Bit 0 = 1 Alle nachfolgenden BUILD BPB-Anforderungen geben in Offset 07h (DeviceBPB) zurück.

*** DOS-Funktionen ***

- Bit 0 = 0 DeviceBPB enthält den neuen Standard-BPB für dieses Gerät.
 Wenn ein vorhergehender Aufruf der Funktion "Setzen der Geräteparameter" mit Bit 0=1 ausgeführt worden ist, dann wird mit Bit 0=0 der aktuelle Datenträger-BPB zurückgegeben. Sonst wird der Standard-BPB des Gerätes von BUILD BPB zurückgegeben.
- Bit 1 = 1 Alle Felder des Parameterblockes, ausser Offset 0Bh (TrackLayout), werden ignoriert.
- Bit 1 = 0 Alle Felder des Parameterblockes werden gelesen.
- Bit 2 = 1 Alle Sektoren einer Spur haben die gleiche Grösse.
- Bit 2 = 0 Alle Sektoren einer Spur haben nicht die gleiche Grösse.
- Bit 3-7 müssen 0 sein.

- DeviceBPB

Der Bereich von 07h bis 25h enthält die Struktur des BPB. Der BPB wird in Abhängigkeit des Bit 0 von Offset 00h (Special Functions) gesetzt.

Bit 0 von Offset 00h = 1 Rückgabe der Adresse des BPB für alle nachfolgenden BUILD BPB (erfolgt solange, bis dieses Bit zurückgesetzt wird);

Bit 0 von Offset 00h = 0 Setzen der Adresse des Standard-BPB für dieses Gerät.

- TrackLayout

Der Bereich ab 26h stellt eine Tabelle mit variabler Länge dar, die die Datenträgerstruktur (Aufbau der Sektoren pro Spur) wiedergibt.

Die Übergabe dieser Tabelle ist notwendig.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	0

9.2.2.61.14.3. Lesen einer Spur vom logischen Gerät

INT21: 44h 0Dh 61h

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Dh
BL	Laufwerksnummer (0=Standard-LW, 1=LW A, ...)
CH	08h
CL	61h Lesen einer Spur vom logischen Gerät
DS:DX	Zeiger auf Parameterblock

*** DOS-Funktionen ***

Vor Aufruf dieser Unterfunktion ist zunächst die Funktion "Setzen der Geräteparameter" (440Dh mit CL=40h) aufzurufen, damit die Daten für die Diskettenorganisation (TrackLayout) dem Betriebssystem mitgeteilt sind.

Der Parameterblock hat folgenden Aufbau:

Offset hex/dez	Anzahl Bytes	Bezeichnung
00h 00	1	SpecialFunctions
01h 01	2	Head
03h 03	2	Cylinder
05h 05	2	FirstSector
07h 07	2	NumberOfSectors
09h 09	4	TransferAddress

Offset 00h (SpecialFunctions) muss 00h sein. Die Werte in Offset 05h (FirstSector) und 07h (NumberOfSectors) beginnen mit 0.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	0

9.2.2.61.14.4. Schreiben einer Spur auf logisches Gerät

INT21: 44h 0Dh 41h

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Dh
BL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)
CH	08h
CL	41h Schreiben einer Spur auf logisches Gerät
DS:DX	Zeiger auf Parameterblock

Vor Aufruf dieser Unterfunktion ist zunächst die Funktion "Setzen der Geräteparameter" (440Dh mit CL=40h) aufzurufen, damit die Daten für die Diskettenorganisation (TrackLayout) dem Betriebssystem mitgeteilt sind.

Für den Parameterblock gilt das oben Beschriebene.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	0

*** DOS-Funktionen ***

9.2.2.61.14.5. Formatieren und Überprüfen einer Spur auf logischem Gerät INT21: 44h 0Dh 42h

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Dh
BL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)
CH	08h
CL	42h Formatieren und Überprüfen einer Spur auf logischem Gerät
DS:DX	Zeiger auf Parameterblock mit Bit 0 von Offset 00h = 1 oder 0

Der Parameterblock hat folgenden Aufbau:

Offset	Anzahl	Bezeichnung
hex/dez	Bytes	
00h 00	1	SpecialFunctions
01h 01	2	Head
03h 03	2	Cylinder

Für Offset 00h gilt:

- Bit 0 = 1: Überprüfung des Formatierens, ob die Kombination von Anzahl Spuren und Sektoren pro Spur hardware-mässig unterstützt wird;
- Bit 0 = 0: Aufruf von Formatieren.

Um die Überprüfung des Formatierens, in bezug auf hardware-mässige Unterstützung der Kombination von Anzahl Spuren und Sektoren pro Spur ausführen zu können, muss zunächst die Funktion "Setzen der Geräteparameter" (440Dh mit CL=40h) aufgerufen werden. Es wird dadurch dem System die gewünschte Formatierungsart in einem korrekten BPB übergeben. Der Gerätetreiber liefert einen Kode, der die Art der Unterstützung anzeigt.

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	0

Bei Bit 0 = 1 von Offset 00h (Specialfunctions) wird in Offset 00h ein Kode zurückgegeben, der die Art der Unterstützung wie folgt anzeigt:

- = 00h : Funktion wird durch ROM BIOS unterstützt, Kombination ist für Diskettentreiber möglich;
- = 01h : Funktion wird nicht durch ROM BIOS unterstützt;
- = 02h : Funktion wird durch ROM BIOS unterstützt, Kombination ist für Diskettentreiber nicht möglich;

*** DOS-Funktionen ***

= 03h : Funktion wird durch ROM BIOS unterstützt, aber es kann nicht festgestellt werden, ob die Kombination möglich ist, da der Diskettentreiber nicht vorhanden ist.

9.2.2.61.14.6. Überprüfen einer Spur auf logischem Gerät

INT21: 44h 0Dh 62h

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Dh
BL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)
CH	08h
CL	62h Überprüfen einer Spur auf logischem Gerät
DS:DX	Zeiger auf Parameterblock

Vor Aufruf dieser Unterfunktion ist zunächst die Funktion "Setzen der Geräteparameter" (440Dh mit CL=40h) aufzurufen, damit die Daten für die Diskettenorganisation (TrackLayout) dem Betriebssystem mitgeteilt sind.

Der Parameterblock hat folgenden Aufbau:

Offset	Anzahl Byte	Bezeichnung
00h	1	Specialfunctions
01h	2	Head
03h	2	Cylinder

Für Offset 00h gilt:

Bit 0 = 0 : Aufruf von Überprüfen

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	0

9.2.2.61.15. Abfrage des logischen Laufwerkes INT21: 44h 0Eh

Funktion:

Information darüber, ob einem Blockgerät mehr als ein logisches Laufwerk zugewiesen ist.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Eh
BL	Laufwerknummer (0=Standard-LW, 1=LW A, ...)

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AL	00h Nur ein logisches Laufwerk
AL	Laufwerknummer (1=LW A, 2=LW B, ...)

Die zurückgegebene Laufwerknummer stellt das zuletzt zugewiesene logische Laufwerk dar.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	0Fh Angabe eines ungültigen Laufwerkes

9.2.2.61.16. Setzen des logischen Laufwerkes INT21: 44h 0Fh

Funktion:

Setzen eines logischen Laufwerkes auf ein Blockgerät.

Aufruf:

Register	Inhalt/Bedeutung
AH	44h
AL	0Fh
BL	Laufwerknummer (1=LW A, 2=LW B, ...)

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AL	00h Nur ein logisches Laufwerk
AL	Laufwerknummer (1=LW A, 2=LW B, ...)

Die zurückgegebene Laufwerknummer stellt das zuletzt zugewiesene logische Laufwerk dar.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	0Fh Angabe eines ungültigen Laufwerkes

9.2.2.62. HANDLER DUPLIZIEREN (DUP)

INT21: 45h

Funktion:

Erzeugen eines zweiten Händlers für den Zugriff auf eine bereits eröffnete Datei.

Aufruf:

Register	Inhalt/Bedeutung
AH	45h
BX	Händlernummer

Der Händler muss eröffnet sein.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Händlernummer

Bezog sich der ursprünglich eröffnete Händler auf eine Datei, so zeigt nach der Funktionsausführung der jedem der beiden Händler zugeordnete Schreib-/Lesezeiger stets auf die gleiche Position innerhalb der Datei.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	04h Zu viele eröffnete Dateien (keine Dateinummern mehr)
AX	06h Ungültige Händlernummer

Wenn zu viele eröffnete Dateien (AL=04h) gemeldet werden, so reicht die durch das Kommando FILES installierte Zahl für gleichzeitig eröffnete Dateien nicht aus.

Der Fehler tritt auf, wenn der angegebene Händler nicht eröffnet oder ungültig ist.

9.2.2.63. HANDLERDUPLIKAT ERSTELLEN (FORCDUP)

INT21: 46h

Funktion:

Einen Händler zwangsweise auf einen zweiten Händler umsetzen.

Aufruf:

Register	Inhalt/Bedeutung
AH	46h
BX	Existierende Händlernummer
CX	Zweite Händlernummer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Der Händler in CX bezieht sich nach der Funktionsausführung auf denselben Datenstrom wie der Händler in BX. Bezog sich der Händler in CX auf eine bereits existierende Datei, so wird diese geschlossen.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	04h Zu viele eröffnete Dateien (keine Dateinummern mehr)
AX	06h Ungültige Händlernummer

Wenn zu viele eröffnete Dateien (AL=04h) gemeldet werden, so reicht die durch das Kommando FILES installierte Zahl für gleichzeitig eröffnete Dateien nicht aus. Der Fehler tritt auf, wenn der angegebene Händler nicht eröffnet oder ungültig ist.

Bemerkung:

Wird der Lese-/Schreibzeiger eines Händlers verändert, so wird der Zeiger für den zweiten Händler ebenfalls geändert.

9.2.2.64. AKTUELLES VERZEICHNIS HOLEN

INT21: 47h

Funktion:

Feststellen des durch CHDIR angewählten Verzeichnisses.

Aufruf:

Register	Inhalt/Bedeutung
AH	47h
DL	Laufwerksnummer (0=Standard-LW, 1=LW A, ...)
DS:SI	Zeiger auf 64-Byte-Feld

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
DS:SI	vollständiger ASCII-Z-Pfad

Der vollständige Pfad (ab dem Stammverzeichnis) des aktuellen Verzeichnisses wird in den Bereich geschrieben, auf den DS:SI zeigt. Hier muss beachtet werden, dass der Laufwerksbuchstabe nicht in der zurückgegebenen Zeichenfolge enthalten ist. Die Zeichenfolge beginnt nicht mit einem umgekehrten Schrägstrich (\). Sie wird durch das Byte 00h abgeschlossen.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	0Fh Angabe eines ungültigen Laufwerks

9.2.2.65. SPEICHER ZUWEISEN

INT21: 48h

Funktion:

Reservieren von Systemspeicher.

Aufruf:

Register	Inhalt/Bedeutung
AH	48h
BX	Anzahl von Paragraphen

*** DOS-Funktionen ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX:0	Segmentadresse des reservierten System- speichers

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
BX	Anzahl des max. zur Verfügung stehenden zusammenhängenden Speicherbereiches (in Paragraphen)
AX	07h Speichersteuerblöcke zerstört
AX	08h Nicht genügend Speicherplatz

Der Fehler 07h wird gemeldet, wenn z.B. ein Programm einen Speicherbereich beschrieben hat, der gar nicht diesem zugeordnet ist.

Der Fehler 08h wird gemeldet, wenn ein zusammenhängender Speicherbereich in der geforderten Grösse nicht zur Verfügung steht.

9.2.2.66. ZUGEWIESENEN SPEICHER FREIGEBEN

INT21: 49h

Funktion:

Freigabe von reserviertem Systemspeicher.

Aufruf:

Register	Inhalt/Bedeutung
AH	49h
ES:0	Segmentadresse des freizugebenden Speichers

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	07h Speichersteuerblöcke zerstört
AX	09h Ungültige Speicherblockadresse

*** DOS-Funktionen ***

Der Fehler 07h wird gemeldet, wenn z.B. ein Programm einen Speicherbereich beschrieben hat, der gar nicht diesem zugeordnet ist.

Der Fehler 09h wird gemeldet, wenn der freizugebende Speicherbereich nicht mit der DOS-Funktion "Speicher zuweisen" (48h) reserviert wurde.

9.2.2.67. ZUGEWIESENEN SPEICHERBLOCK VERANDERN
(SETBLOCK)

INT21: 4Ah

Funktion:

Verändern der Grösse eines reservierten Systemspeicherbereiches.

Aufruf:

Register	Inhalt/Bedeutung
AH	4Ah
BX	Anzahl zu reservierender Paragraphen
ES:0	Segmentadresse des zu modifizierenden Speicherbereichs

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
BX	Anzahl der max. verfügbaren Paragraphen im betreffenden Speicherbereich
AX	07h Speichersteuerblöcke zerstört
AX	08h Nicht genügend Speicherplatz
AX	09h Ungültige Speicherblockadresse

Der Fehler 07h wird gemeldet, wenn z.B. ein Programm einen Speicherbereich beschrieben hat, der gar nicht diesem zugeordnet ist.

Der Fehler 08h wird gemeldet, wenn ein zusammenhängender Speicherbereich in der geforderten Grösse nicht zur Verfügung steht.

Der Fehler 09h wird gemeldet, wenn der freizugebende Speicherbereich nicht mit der DOS-Funktion "Speicher zuweisen" (48h) reserviert wurde.

*** DOS-Funktionen ***

9.2.2.68. PROGRAMM LADEN MIT UND OHNE AUSFÜHREN
(EXEC)

INT21: 4Bh

Durch diese DOS-Funktion werden die Unterfunktionen
- Laden und Ausführen eines Programmes
- Laden eines Programmes
realisiert.

9.2.2.68.1. Programm laden und ausführen (EXEC) INT21: 4Bh 00h

Funktion:
Laden und Ausführen eines Programmes.

Aufruf:

Register	Inhalt/Bedeutung
AH	4Bh
AL	00h
ES:BX	Zeiger auf einen Parameterblock
DS:DX	Zeiger auf ASCIIZ-Dateispezifikation

Der Parameterblock hat folgenden Aufbau:

Offset hex/dez	Anzahl Bytes	Bedeutung
00h 00	2	Segmentadresse der zu übergebenden Programm- umgebung
02h 02	4	Zeiger (Segment:Offset) auf Kommandozeile, die ab PSP + 80h übergeben werden soll
06h 06	4	Zeiger (Segment:Offset) auf ungeöffneten Standard-FCB, der ab PSP + 5Ch übergeben werden soll
0Ah 10	4	Zeiger (Segment:Offset) auf ungeöffneten Standard-FCB, der ab PSP + 6Ch übergeben werden soll

Enthält Offset 00h den Inhalt 0000h, so wird die aktuelle Pro-
grammumgebung an das aufzurufende Programm übergeben.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	02h Datei nicht gefunden
AX	03h Pfad nicht gefunden
AX	04h Zu viele eröffnete Dateien (keine Dateinummern mehr)
AX	05h Zugriff abgelehnt
AX	08h Nicht genügend Speicherplatz
AX	0Ah Ungültige Umgebung
AX	0Bh Ungültiges Format

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.

Wenn zu viele eröffnete Dateien (AL=04h) gemeldet werden, so reicht die durch das Kommando FILES installierte Zahl für gleichzeitig eröffnete Dateien nicht aus.

Der Zugriff wird abgelehnt (AL=05h), wenn ein Eintrag für einen Datenträger bzw. ein Verzeichnis geladen werden soll.

Der Fehler 0Ah tritt auf, wenn die übergebene Programmumgebung inkonsistent ist.

Wenn das aufgerufene Programm eine EXE-Datei ist, die einen inkonsistenten Inhalt hat, so wird der Fehler 0Bh gemeldet.

Bemerkung:

Diese DOS-Funktion benutzt das Ladeprogramm in COMMAND.COM (am oberen Ende des Speichers), um den Ladevorgang durchzuführen. Hat das Nutzerprogramm das Ladeprogramm überlagert, so wird versucht, COMMAND.COM nachzuladen.

Da beim Ladeprozess eines Nutzerprogramms das Betriebssystem i. allg. den gesamten Speicher für dieses reserviert, sollte vor Aufruf der DOS-Funktion 4Bh die DOS-Funktion "Zugewiesenen Speicherblock verändern (SETBLOCK)" (4Ah) dazu genutzt werden, um genügend freien Speicher zur Verfügung zu stellen.

Für das zu ladende Programm wird zunächst ein PSP eingerichtet. Die Interruptvektoren 22h und 23h werden auf die Adresse des Befehls gesetzt, der im aufrufenden Programm diesem DOS-Funktionsaufruf unmittelbar folgt. Damit wird die Fortführung des aufrufenden Programmes gesichert.

Wird die Steuerung an den aufrufenden Prozess zurückgegeben, so sind sämtliche Register, einschliesslich des Stapelregisters, geändert.

Da dem geladenen Programm alle bereits eröffneten Dateien zur Verfügung stehen, kann dieser durch eine Umdefinierung der durch das Betriebssystem vorbestimmten Ausgabekanäle (CON, AUX, PRN) den Zugriff auf diese Kanäle beschränken. Das bedeutet aber z.B. auch, dass ein ablaufendes Programm, das die Ein- und Ausgabe über das Standardgerät CON durchführt, durch Umdefinieren der Ein- und Ausgabe (z.B. auf eine Datei) dazu benutzt werden kann, die Daten eines aufrufenden Programmes zu verarbeiten bzw. zu filtern.

*** DOS-Funktionen ***

Für jedes Programm kann eine besondere Umgebung erstellt werden. In dieser kann z.B. enthalten sein, wie der Kommandoprozessor (COMMAND.COM) heisst und wo er sich befindet. Die Einträge für die Umgebung bestehen aus ASCII-Zeichenfolgen. Die letzte ASCII-Zeichenfolge ist zusätzlich durch 00h abgeschlossen. Der gesamte Bereich für die Umgebung ist auf 32 KByte begrenzt. Die Einträge für die Umgebung weisen folgende typische Form auf:
Parameter = Wert

So könnte z.B. die Zeichenfolge VERIFY = 0N übergeben werden. Enthält der Offset 00h des Parameterblockes 0000h, so übernimmt der neu erstellte Prozess die Umgebung des übergeordneten Prozesses ohne Änderungen. Die Segmentadresse der Umgebung wird im PSP + 2Ch für das aufgerufene Programm durch das Betriebssystem eingetragen.

9.2.2.68.2. Programm laden

INT21: 4Bh 03h

Funktion:
Laden eines Programmes (Treiber oder Overlay).

Aufruf:

Register	Inhalt/Bedeutung
AH	4Bh
AL	03h
ES:BX	Zeiger auf einen Parameterblock
DS:DX	Zeiger auf ASCII-Zeichenspezifikation

Der Parameterblock hat folgenden Aufbau:

Offset hex/dez	Anzahl Bytes	Bedeutung
00h 00	2	Segmentadresse, an deren Offset 00h das Programm geladen werden soll
02h 02	2	Relokationsfaktor

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	02h Datei nicht gefunden
AX	03h Pfad nicht gefunden
AX	04h Zu viele eröffnete Dateien (keine Dateinummern mehr)
AX	05h Zugriff abgelehnt
AX	08h Nicht genügend Speicherplatz
AX	0Ah Ungültige Umgebung

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.
 Wenn zu viele eröffnete Dateien (AL=04h) gemeldet werden, so reicht die durch das Kommando FILES installierte Zahl für gleichzeitig eröffnete Dateien nicht aus.
 Der Zugriff wird abgelehnt (AL=05h), wenn ein Eintrag für einen Datenträger bzw. ein Verzeichnis geladen werden soll.
 Der Fehler 0Ah tritt auf, wenn die übergebene Programmumgebung inkonsistent ist.

Bemerkung:

Diese DOS-Funktion benutzt das Ladeprogramm in COMMAND.COM (am oberen Ende des Speichers), um den Ladevorgang durchzuführen. Hat das Nutzerprogramm das Ladeprogramm überlagert, so wird versucht, COMMAND.COM nachzuladen.

Es wird kein PSP angelegt. Die Programmausführung wird nicht begonnen (nützlich für Laden von Overlays).
 Mit dem Laden eines Programmes werden sämtliche Register, einschliesslich des Stapelregisters, geändert.

9.2.2.69. PROZESS BEENDEN (EXIT)

INT21: 4Ch

Funktion:

Beenden eines Programmes mit Übergabe eines Rückgabekodes an das aufrufende Programm.

Aufruf:

Register	Inhalt/Bedeutung
AH	4Ch
AL	Rückgabekode

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
	ohne

Alle eröffneten Dateien werden geschlossen. Die Interruptvektoren 22h, 23h und 24h werden wieder auf den Stand gebracht, den sie vor Aufruf des gerade beendeten Programms hatten.

Bemerkung:

Dieser Funktionsaufruf beendet den momentanen Prozess, wobei die Steuerung an den aufrufenden Prozess übergeben wird. Der Rückgabekode kann von den Stapelverarbeitungsbefehlen IF bzw. ERRORLEVEL und von dem DOS-Funktionsaufruf "Beendigungskode liefern (WAIT)" (4Dh) abgefragt werden.

9.2.2.70. BEENDIGUNGSKODE LIEFERN (WAIT)

INT21: 4Dh

Funktion:

Ermitteln des Beendigungskodes, den ein aufgerufenes Programm an den aufrufenden Prozess zurückgibt.

Aufruf:

Register	Inhalt/Bedeutung
AH	4Dh

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	Rückgabekode des Nutzers
AH	00h Normale Beendigung
AH	01h Beendigung durch Abbruch mit CTRL-C
AH	02h Beendigung als Ergebnis eines kritischen Gerätefehlers
AH	03h Beendigung durch "Prozess beenden und resident bleiben" (31h)

Bemerkung:

Diese DOS-Funktion kann nur einmal aufgerufen werden, um den richtigen Rückgabekode zu erhalten. Bei einem nochmaligen Aufruf wird 0000h übergeben.

**9.2.2.71. ERSTE DATEIEINTRAGUNG FINDEN
(FIND FIRST)**

INT21: 4Eh

Funktion:

Suchen nach dem ersten Dateinamen, der zu einem vorgegebenen Suchmuster passt.

Aufruf:

Register	Inhalt/Bedeutung
AH	4Eh
CX	Dateiattribut
DS:DX	Zeiger auf ASCIIZ-Dateispezifikation

In der Datei-Bezeichnung der zu suchenden Datei können die globalen Datei-Bezeichner "*" und "?" enthalten sein.

Das Dateiattribut dient als Suchkriterium. Dabei gilt:

1. Dateiattribut = 00h

Es werden nur normale Dateieintragungen gefunden. Eintragungen für Datenträgerkennsatz, Verzeichnisse, verborgene Dateien und Systemdateien werden nicht gefunden.

2. Dateiattribut = 02h (verborgene Dateien)

Dateiattribut = 04h (Systemdateien)

Dateiattribut = 10h (Verzeichniseintragungen)

Es erfolgt eine inklusive Suche, d.h., sämtliche normalen Eintragungen und alle Eintragungen, die mit den angegebenen Dateiattributen übereinstimmen, werden gefunden.

Zur Überprüfung sämtlicher Verzeichniseintragungen, mit Ausnahme des Datenträgerkennsatzes, kann das Dateiattribut auf "Verborgen + System + Verzeichnis" (=16h) gesetzt werden.

3. Dateiattribut = 08h (Datenträgerkennsatz)

Es erfolgt eine exklusive Suche, d.h., es wird nur der Eintrag für den Datenträgerkennsatz gefunden.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
	Aktueller DTA gefüllt

*** DOS-Funktionen ***

Der aktuelle DTA wird wie folgt gefüllt:

Offset hex/dez	Anzahl Bytes	Bedeutung
00h 00	21	enthält die für eine nachfolgende Suche notwendigen Informationen (Bereich sollte durch den Nutzer nicht verändert werden)
15h 21	1	Attribut der gefundenen Datei
16h 22	2	Zeit, zu der die Datei zuletzt geschrieben wurde
18h 24	2	Datum, an dem die Datei zuletzt geschrieben wurde
1Ah 26	4	Dateigrösse (Offset 1Ah/1Bh enthält niederwertigen Teil)
1Eh 30	13	Name und Erweiterung der gefundenen Datei (getrennt durch ".", beendet durch 00h)

Alle Leerzeichen werden aus dem Namen und der Erweiterung entfernt.

Die zurückgegebene Dateibezeichnung sieht genauso aus, wie sie als Befehlsparameter eingegeben wird.
Zum Beispiel LABEL.COM gefolgt von einem Byte mit 00h.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	02h Datei nicht gefunden
AX	03h Pfad nicht gefunden
AX	12h Keine weiteren Dateien mehr

Diese Fehler werden gemeldet, wenn die Datei bzw. der Pfad ungültig ist oder nicht gefunden wurde.

9.2.2.72. NÄCHSTE DATEIEINTRAGUNG FINDEN
(FIND NEXT)

INT21: 4Fh

Funktion:

Suchen nach dem nächsten Dateinamen, wobei die zuletzt mit der DOS-Funktion 4Eh bzw. 4Fh gefundene Datei als Referenz auf den Beginn des Suchvorgangs benutzt wird.

Aufruf:

Register	Inhalt/Bedeutung
AH	4Fh Informationen im aktuellen DTA

Bei Aufruf muss der aktuelle DTA die Informationen enthalten, die durch einen vorhergehenden Aufruf der DOS-Funktion "Erste Datei-

*** DOS-Funktionen ***

eintragung finden (FIND FIRST)" (4Eh) bzw. durch diese Funktion selber aufgefüllt wurde.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
	Aktueller DTA gefüllt

Die nächste Dateieintragung wurde gefunden und der aktuelle DTA wird mit Informationen gefüllt, wie bei der DOS-Funktion "Erste Dateieintragung finden (FIND FIRST)" (4Eh) beschrieben.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	12h Keine weiteren Dateien mehr

Der Fehler wird gemeldet, wenn keine weiteren Dateien mehr vorhanden sind.

9.2.2.73. STATUS PRÜFLESEN ABFRAGEN

INT21: 54h

Funktion:

Abfrage des Status, ob nach einem Schreibversuch ein Prüfllesen ausgeführt werden soll.

Aufruf:

Register	Inhalt/Bedeutung
AH	54h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
AL	00h Ohne Prüfllesen
AL	01h Mit Prüfllesen

9.2.2.74. DATEI UMBENENNEN

INT21: 56h

Funktion:

Umbenennen einer Datei über eine ASCII-Zeichenfolge.

Aufruf:

Register	Inhalt/Bedeutung
AH	56h
DS:DX	Zeiger auf Quell-ASCII-Zeichenspezifikation
ES:DI	Zeiger auf Ziel-ASCII-Zeichenspezifikation

Globale Dateibezeichner sind in der ASCII-Zeichenspezifikation nicht erlaubt.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	02h Datei nicht gefunden
AX	03h Pfad nicht gefunden
AX	05h Zugriff abgelehnt
AX	11h Nicht dieselbe Einheit

Der Fehler 05h wird gemeldet, wenn

- die Quell-ASCII-Zeichenspezifikation keine gültige Datei, eine Datei mit den Attributen "System" oder "Verborgenen" bzw. eine Verzeichniseintragung kennzeichnet oder
- die Ziel-ASCII-Zeichenspezifikation bereits existiert.

Bei Auftreten des Fehlers 11h wurde versucht, eine Datei auf ein anderes Laufwerk umzubenennen.

Bemerkung:

Die Laufwerkangaben in der Quell- und Ziel-ASCII-Zeichenspezifikation müssen übereinstimmen.
Die Pfade müssen nicht identisch sein. Dadurch kann eine Datei in ein anderes Verzeichnis verschoben bzw. umbenannt werden.

Netzwerk-Zugriffsrechte:

Es sind Anlage-Zugriffsrechte erforderlich.

**9.2.2.75. DATUM UND UHRZEIT EINER DATEI
ABFRAGEN/SETZEN**

INT21: 57h

Diese DOS-Funktion ist in 2 Unterfunktionen gegliedert.

**9.2.2.75.1. Datum und Uhrzeit einer Datei
abfragen**

INT21: 57h 00h

Funktion:
Abfrage des Datums des letzten Schreibzugriffs auf eine Datei.

Aufruf:

Register	Inhalt/Bedeutung
AH	57h
AL	00h
BX	Händlernummer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
CX	Zeit
DX	Datum

Die Zeit wird in den Bits wie folgt abgebildet:
 Bit 15-11 binäre Zahl für Stunden (0 bis 23).
 Bit 10- 5 binäre Zahl für Minuten (0 bis 59).
 Bit 4- 0 binäre Zahl für 2-Sek.-Intervalle.

Das Datum wird in den Bits wie folgt abgebildet:
 Bit 15-9 binäre Zahl für Monat (1 bis 12)
 Bit 8-5 binäre Zahl für Tag (1 bis 31)
 Bit 4-0 binäre Zahl für Jahr (0 bis 119)
 entspricht den Jahren 1980 bis 2099

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	06h Ungültige Händlernummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.
 Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

9.2.2.75.2. Datum und Uhrzeit einer Datei setzen

INT21: 57h 01h

Funktion:

Setzen des Datums des letzten Schreibzugriffs auf eine Datei.

Aufruf:

Register	Inhalt/Bedeutung
AH	57h
AL	01h
BX	Händlernummer
CX	Zeit
DX	Datum

Das Format von Zeit und Datum ist wie oben beschrieben.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	06h Ungültige Händlernummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.

Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

9.2.2.76. ERWEITERTE FEHLERMELDUNG

INT21: 59h

Funktion:

Übergabe von zusätzlichen Informationen nach Auftreten eines Fehlers.

Aufruf:

Register	Inhalt/Bedeutung
AH	59h
BX	0000h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Fehlerkode
BH	Fehlerklasse
BL	Nutzerhinweise
CH	Fehlerort

Nach Austritt sind die Inhalte der Register DX, SI, DI, ES, CL und DS zerstört.

Eine detaillierte Beschreibung der zurückgegebenen Werte ist im Abschnitt 9.2.1.7. (Fehlerrückgabeinformation) vorhanden.

Bemerkung:

Man benutzt diese Funktion als Reaktion auf folgende Betriebssystemfunktionen:

- Interrupt 24h
- Interrupt 21h
 - . Alle Funktionsaufrufe, die einen Fehler im C-Flag (CF=1) zurückliefern
 - . FCB-Funktionsaufrufe, die AL=FFh zurückgeben.

Für Funktionen, die einen Fehler durch CF=1 anzeigen, empfiehlt sich als Anwendung dieser DOS-Funktion folgende Verfahrensweise:

1. Register für beliebige DOS-Funktion laden
2. Interrupt 21h aufrufen
3. Wenn CF=0, dann Fortsetzung des Programmes
4. Wenn CF=1, dann Fehlerkode vernachlässigen und DOS-Funktion 59h aufrufen.

Für Funktionen, die einen Fehler durch AL=FFh anzeigen, empfiehlt sich als Anwendung dieser DOS-Funktion folgende Verfahrensweise:

1. Register für beliebige DOS-Funktion laden
2. Interrupt 21h aufrufen
3. Wenn AL=00h, dann Fortsetzung des Programmes
4. Wenn AL=FFh, dann Fehlerkode vernachlässigen und DOS-Funktion 59h aufrufen.

9.2.2.77. EINZIGARTIGE DATEI ERSTELLEN

INT21: 5Ah

Funktion:

Anlegen und Eröffnen einer Datei mit einem willkürlichen durch das Betriebssystem festgelegten Dateinamen.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Ah
CX	Dateiattribut
DS:DX	Zeiger auf ASCIIZ-Pfadbezeichnung

Die ASCIIZ-Pfadbezeichnung hat folgenden Aufbau:

- Jeder Pfadname wird durch "\" abgeschlossen.
- Nach dem letzten Pfadnamen muss 00h stehen.

Zusätzlich muss nach dem Endekennzeichen 00h noch ein Bereich von 12 Byte reserviert sein, da dort die generierte Dateibezeichnung angefügt wird.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Händlernummer
DS:DX	Zeiger auf ASCIIZ-Dateispezifikation

Die generierte Dateibezeichnung wird dem in DS:DX festgelegten Pfad angefügt. Die Datei ist in der Nutzungsart "Kompatibilitätsmodus" und der Zugriffsart "Lesen/Schreiben" eröffnet.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	03h Pfad nicht gefunden
AX	04h Zu viele eröffnete Dateien (keine Dateinummern mehr)
AX	05h Zugriff abgelehnt

Der Fehler 03h wird gemeldet, wenn die angegebene Pfadbezeichnung falsch ist oder der Pfad nicht existiert.

Wenn zu viele eröffnete Dateien (AL=04h) gemeldet werden, so reicht die durch das Kommando FILES installierte Zahl für gleichzeitig eröffnete Dateien nicht aus.

Bemerkung:

Programme, die temporäre Dateien benötigen, sollten diese Funktion zur Generierung der zufälligen, bisher unbenutzten Dateibezeichnungen, verwenden.

Eine mit Hilfe dieser Funktion erzeugte Datei wird vom System nicht automatisch gelöscht, wenn das Programm beendet wird.

Netzwerk-Zugriffsrechte:
Es sind Anlege-Zugriffsrechte erforderlich.

9.2.2.78. NEUEN HANDLER ERSTELLEN

INT21: 5Bh

Funktion:

Erstellen einer vollkommen neuen (d.h. noch nicht existierenden) Datei.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Bh
CX	Dateiattribut
DS:DX	Zeiger auf ASCII-Z-Dateispezifikation

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
AX	Händlernummer

Die Datei ist in der Nutzungsart "Kompatibilitätsmodus" und der Zugriffsart "Lesen/Schreiben" eröffnet.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	03h Pfad nicht gefunden
AX	04h Zu viele eröffnete Dateien (keine Dateinummern mehr)
AX	05h Zugriff abgelehnt
AX	50h Datei existiert

Der Fehler 03h wird gemeldet, wenn die angegebene Pfadbezeichnung falsch ist oder der Pfad nicht existiert.
Wenn zu viele eröffnete Dateien (AL=04h) gemeldet werden, so reicht die durch das Kommando FILES installierte Zahl für gleichzeitig eröffnete Dateien nicht aus.
Wenn die zu erstellende Datei bereits existiert, so wird AL=50h gemeldet.

Netzwerk-Zugriffsrechte:

Es sind Anlege-Zugriffsrechte erforderlich.

9.2.2.79. DATEIZUGRIFF VERBIETEN/ERLAUBEN

INT21: 5Ch

Funktion:

Zugriff eines anderen Prozesses auf einen Bereich einer eigenen Datei verbieten bzw. erlauben.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Ch
BX	Händlernummer
CX	Offset des Bereichs der Datei (höherwertig)
DX	Offset des Bereichs der Datei (niederwertig)
SI	Länge des Bereichs der Datei (höherwertig)
DI	Länge des Bereichs der Datei (niederwertig)
AL	00h Zugriff verbieten
AL	01h Zugriff erlauben

Dieser Funktionsaufruf sollte nur benutzt werden, wenn die Datei in den Nutzungsarten "Nur Lesen unterbinden" bzw. "Nichts unterbinden" eröffnet ist. Bei diesen Nutzungsarten erfolgt keine lokale Pufferung der Daten, wenn auf Dateien von Netzwerkdisketten zugegriffen wird.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	06h Ungültige Händlernummer
AX	21h Verriegelungsverletzung
AX	24h Überlauf des Nutzungspuffers

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.

Wurde bei Aufruf der Unterfunktion eine ungültige oder nicht eröffnete Händlernummer in BX angegeben, so wird AL=06h zurückgegeben.

Wenn der Fehler 21h gemeldet wird, so ist der Zugriff auf den Bereich bereits vollständig oder teilweise verboten.

*** DOS-Funktionen ***

Bemerkung:

Dieser Funktionsaufruf sollte nur auf solche Dateien angewendet werden, die unter folgenden Bedingungen eröffnet wurden:

- Benutzung der Nutzungsart "Nur Lesen unterbinden"
- Benutzung der Nutzungsart "Nichts unterbinden"
- Benutzung der Zugriffsart "Lesen/Schreiben", verbunden mit der Nutzungsart "Nur Schreiben unterbinden"
- Benutzung der Zugriffsart "Nur Schreiben", verbunden mit der Nutzungsart "Nur Schreiben unterbinden".

Wenn ein anderer Prozess in einem Bereich zu lesen oder zu schreiben versucht, so wird der Aufruf entsprechend der durch die DOS-Funktion "Verändern des Zählers für Wiederholungsversuche bei gemeinsamer Dateinutzung" (440Bh) festgelegten Anzahl wiederholt.

Wenn nach diesen Wiederholungen kein Erfolg erzielt wurde, wird eine Fehlermeldung ausgegeben.

Die verbotenen Bereiche können an beliebigen Stellen in der Datei sein. Wird der Dateizugriff hinter dem Dateiende verboten, so bewirkt dies keinen Fehler.

Der Zugriff auf einen Dateibereich sollte möglichst nur für kurze Zeit verboten werden.

Die DOS-Funktionen "Händler duplizieren" (45h) und "Händlerduplikat erstellen" (46h) bewirken, dass der Zugriff auf die verbotenen Bereiche auf das Duplikat übertragen wird.

Durch die DOS-Funktion "Programm laden und ausführen" (4B00h) werden die verbotenen Bereiche bei der Übergabe von eröffneten Dateien nicht mit übergeben.

Wenn ein Programm

- eine Datei schliesst, die verbotene Bereiche enthält oder
 - mit einer noch eröffneten Datei, die einen verbotenen Bereich enthält, beendet wird oder
 - durch INT 23h bzw. INT 24h abgebrochen wird,
- so kommt es zu undefinierten Ergebnissen. Damit es nicht zu undefinierten Ergebnissen kommt, sollten alle verbotenen Bereiche wieder erlaubt werden.

Mit einem Nutzerprogramm kann der aktuelle Status (Zugriff verboten bzw. erlaubt) eines Bereiches einer Datei bestimmt werden, indem versucht wird, den Zugriff auf den Bereich zu verbieten. Anschliessend muss der Fehlercode ausgewertet werden.

9.2.2.80. MASCHINENNAME ABFRAGEN

INT21: 5Eh 00h

Funktion:

Abfrage des augenblicklichen lokalen Maschinennamens.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Eh
AL	00h
DS:DX	Zeiger auf 16 Byte grosses Feld

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
DS:DX	ASCIIZ-Maschinenname
CX	Identifikationsnummer der lokalen Maschine

Der ASCIIZ-Maschinenname kann 15 Zeichen einschliesslich dem Leerzeichen als Trennzeichen, lang sein. Er wird durch 00h abgeschlossen.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.

Bemerkung:

Wurde niemals ein Maschinenname festgelegt, so wird CH=00h zurückgegeben. Der Wert von CL ist unbestimmt.

9.2.2.81. MASCHINENNAME SETZEN

INT21: 5Eh 01h

Funktion:

Setzen des lokalen Maschinennamens.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Eh
AL	01h
CX	Identifikationsnummer der lokalen Maschine
DS:DX	Zeiger auf ASCIIZ-Maschinennamen

Für den ASCIIZ-Maschinennamen gilt das bereits bei der vorhergehenden Funktion Beschriebene.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde.

9.2.2.82. DRUCKER-SETUP SETZEN

INT21: 5Eh 02h

Funktion:

Festlegen einer Zeichenfolge zur Initialisierung des Netzwerk-Druckers für den Druck einer Datei.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Eh
AL	02h
BX	Index der Zuweisungstabelle
CX	Länge der Initialisierungs-Zeichenfolge
DS:SI	Zeiger auf Initialisierungs-Zeichenfolge

Die angegebene Zeichenfolge kann max. 64 Byte gross sein.

*** DOS-Funktionen ***

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder das Netzwerkprogramm nicht aktiviert ist.

Bemerkung:

Vor jeder auf dem durch den Listenindex festgelegten Netzwerk-Drucker zu druckenden Datei wird die angegebene Zeichenfolge dem Drucker übertragen. Dadurch ist den verschiedenen Nutzern eines gemeinsamen Druckers die Möglichkeit gegeben, ihre eigene Initialisierung einzustellen.

9.2.2.83. DRUCKER-SETUP ABFRAGEN

INT21: 5Eh 03h

Funktion:

Übergabe der Initialisierungs-Zeichenfolge eines Netzwerk-Druckers für den Druck einer Datei.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Eh
AL	03h
BX	Index der Zuweisungstabelle
ES:DI	Zeiger auf 64 Byte grosses Feld

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
CX	Länge der Initialisierungs-Zeichenfolge
ES:DI	Gefüllt mit Initialisierungs-Zeichenfolge

*** DOS-Funktionen ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder das Netzwerkprogramm nicht aktiviert ist.

9.2.2.84. EINTRAGUNGEN DER ZUWEISUNGSTABELLE
ABFRAGEN

INT21: 5Fh 02h

Funktions:

Holen der Eintragungen aus der Zuweisungstabelle für das Netzwerk.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Fh
AL	02h
BX	Index der Zuweisungstabelle
DS:SI	Zeiger auf 128 Byte grosses Feld
ES:DI	Zeiger auf 128 Byte grosses Feld

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
BH	Status-Kennzeichen des Gerätes
BL	Typ des Gerätes
CX	Nutzerkennzeichen
DS:SI	Gefüllt mit ASCII-Zeichenfolge des lokalen Gerätenamens
ES:DI	Gefüllt mit ASCII-Zeichenfolge des Netzwerknamens

Für BH gilt:

Bit 0 = 0 Gerät ist gültig
 Bit 0 = 1 Gerät ist nicht gültig
 Bit 1-7 reserviert

Für BL gilt:

BL = 03h Drucker
 BL = 04h Laufwerk

Die Register DX und BP sind undefiniert.

Das Nutzerkennzeichen ist ein spezifisches Kennzeichen für das verwendete Netzwerkprogramm.

*** DOS-Funktionen ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	12h Keine weiteren Dateien mehr

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder das Netzwerkprogramm nicht aktiviert ist.

Wenn der Index der Zuweisungstabelle (BX) grösser als die Anzahl der Eintragung in der Zuweisungstabelle ist, so wird der Fehler 12h gemeldet.

Bemerkung:

Jeder Aufruf dieser DOS-Funktion liefert eine Eintragung der Zuweisungstabelle. Durch schrittweise Erhöhung (um 1) des Index der Zuweisungstabelle kann die gesamte Tabelle abgearbeitet werden. Das Ende der Tabelle wird durch den Fehler 12h angezeigt.

Der Inhalt der Tabelle kann zwischen den Aufrufen dieser DOS-Funktion durch die DOS-Funktion "Zuweisung eines Gerätes" (5F03h) verändert werden.

9.2.2.85. ZUWEISUNG EINES GERÄTES

INT21: 5Fh 03h

Funktion:

Zuweisen eines Gerätes (Drucker oder Laufwerk) an ein Netzwerkgerät.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Fh
AL	03h
BL	Typ des Gerätes
CX	Nutzerkennzeichen
DS:SI	Zeiger auf ASCIIIZ-Quellgerät
ES:DI	Zeiger auf ASCIIIZ-Zielgerät

Für BL gilt:

BL = 03h Drucker

BL = 04h Laufwerk

Das ASCIIIZ-Quellgerät enthält die Bezeichnung für einen Drucker oder ein Laufwerk.

Die Angabe für ASCIIIZ-Zielgerät stellt einen Namen für ein Netzwerkverzeichnis dar und kann max. 128 Byte gross sein.

Das Nutzerkennzeichen ist ein spezifisches Kennzeichen für das verwendete Netzwerkprogramm.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	03h Pfad nicht gefunden
AX	05h Zugriff abgelehnt
AX	08h Nicht genügend Speicherplatz

Der Fehler 01h tritt auf, wenn

- in AL eine nicht zulässige Funktionsnummer angegeben wurde,
- in BL ein nicht zulässiger Typ angegeben wurde,
- das Netzwerkprogramm nicht aktiviert ist,
- die Quell- bzw. Ziel-ASCII-Zeichenfolge ein falsches Format besitzt oder
- das Quell-Gerät bereits zugewiesen ist.

Der Fehler 03h tritt auf, wenn der Netzwerkverzeichnispfad ungültig ist oder nicht existiert.
 Der Zugriff wird abgelehnt, wenn die Kombination von Netzwerkverzeichnis und Passwort ungültig ist bzw. wenn das Netzwerkverzeichnis nicht existiert.
 Wenn die ASCII-Zeichenfolgen zu lang sind, wird der Fehler 08h gemeldet.

Bemerkung:

Wenn BL = 03h ist, so ist das Quellgerät ein Drucker und das Zielgerät ein Netzwerkverzeichnis.
 Alle Ausgaben an den definierten Drucker werden gepuffert und zum entfernten Drucker-Zwischenspeicher (Spooler), der als Zielgerät definiert wurde, gesendet.
 Die Quell-ASCII-Zeichenfolge muss PRN, LPT1, LPT2 oder LPT3 sein (einschliesslich 00h).
 Die Ziel-ASCII-Zeichenfolge muss ein Netzwerkverzeichnis in der folgenden Form enthalten:

\\maschinename\pfad 00 passwort 00

maschinename ist der Netzwerkname für den Server, der das Netzwerkverzeichnis enthält.
 pfad ist der angenommene Name des Netzwerkverzeichnisses (nicht der Verzeichnispfad), welchem das Quellgerät zugewiesen wurde.
 00 ist das Byte 00h.
 passwort ist das Passwort (0 bis 8 Zeichen) für den Zugriff auf das Netzwerkverzeichnis.

Wird kein Passwort (0 Zeichen) festgelegt, so steht am Ende der Angabe für das Netzwerkverzeichnis zweimal 00h.

*** DOS-Funktionen ***

Wenn BL = 04h ist, so ist das Quellgerät ein Laufwerk und das Zielgerät ein Netzwerkverzeichnis. Das Laufwerk wird gekennzeichnet durch einen Laufwerksbuchstaben mit Doppelpunkt und ist durch 00h abgeschlossen.

Wenn die Quelle ein Laufwerk war, wird eine Verbindung zwischen Laufwerk und Netzwerkpfad hergestellt. Alle weiteren Bezugnahmen auf dieses Laufwerk werden als Bezugnahme auf das Netzwerkverzeichnis interpretiert. Falls die Quelle eine leere Zeichenfolge war, versucht das System den Zugriff zu dem Netzwerkverzeichnis mit dem festgelegten Passwort abzusichern. Dies erfolgt ohne Neuzuweisung eines Gerätes.

9.2.2.86. ZUWEISUNG AUFHEBEN

INT21: 5Fh 04h

Funktion:

Aufheben der Verbindung zwischen Drucker bzw. Laufwerk und dem zugewiesenen Netzwerkverzeichnis.

Aufruf:

Register	Inhalt/Bedeutung
AH	5Fh
AL	04h
DS:SI	Zeiger auf ASCII-Z-Quellgerät

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	0Fh Angabe eines ungültigen Laufwerks

Der Fehler 01h tritt auf, wenn

- in AL eine nicht zulässige Funktionsnummer angegeben wurde,
- das Netzwerkprogramm nicht aktiviert ist,
- die Quell-ASCII-Zeichenfolge ein falsches Format besitzt oder
- das Quell-Gerät nicht existiert.

Der Fehler 0Fh tritt auf, wenn die Verbindung zwischen den zugewiesenen Geräten unterbrochen ist.

*** DOS-Funktionen ***

Bemerkung:

Wird als ASCIIZ-Quellgerät

- eine Laufwerkbezeichnung (Laufwerkbuchstabe mit Doppelpunkt), die mit einem Netzwerkverzeichnis verbunden ist oder
 - ein Drucker (PRN, LPT1, LPT2 bzw. LPT3), der mit einem Netzwerkpfad verbunden ist, angegeben, wird die bestehende Verbindung aufgelöst.
- Die Geräte erhalten ihre ursprüngliche physische Bedeutung zurück.

Wenn als ASCIIZ-Quellgerät ein Netzwerkverzeichnis, verbunden mit einem Passwort, angegeben wurde, wird die Verbindung zwischen der lokalen Maschine und dem Netzwerk aufgelöst.

9.2.2.87. ERMITTELN AKTUELLER PSP

INT21: 62h

Funktion:

Rückgabe der Segmentadresse des aktuellen Prozesses.

Aufruf:

Register	Inhalt/Bedeutung
AH	62h

Rückkehr:

Reg./Flag	Inhalt/Bedeutung
BX:0	Segmentadresse des aktuellen Prozesses

**9.2.2.88. ERMITTLUNG EINER ERWEITERTEN LANDES-
INFORMATION**

INT21: 65h

Funktion:

Es wird eine erweiterte Landesinformation zurückgeliefert.

Aufruf:

Register	Inhalt/Bedeutung
AH	65h
AL	Informationskennzeichen (1,2,4,5,6)
BX	Zeichensatztabelle
CX	Anzahl der zurückzugebenden Bytes
DX	Landeskod
ES:DI	Zeiger auf Feld

*** DOS-Funktionen ***

Für CX müssen Werte grösser gleich 5 angegeben sein. Für AL gilt:

- AL=1: Ermittlung der Landesinformation
- AL=2: Ermittlung einer allgemeinen Konvertierungstabelle
- AL=4: Ermittlung einer Konvertierungstabelle für Dateinamen
- AL=5: Ermittlung einer Tabelle mit ungültigen Zeichen
- AL=6: Ermittlung einer allgemeinen Vergleichsfolge

Wenn für BX und DX der Wert 0FFFFh angegeben ist, so werden die jeweiligen aktuellen Werte genommen.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
ES:DI	erweiterte Landesinformation

Entsprechend des Informationskennzeichens wird das Feld für die erweiterte Landesinformation (ES:DI) unterschiedlich gefüllt.

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h Ungültige Funktionsnummer
AX	02h Pfad nicht gefunden

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde oder wenn CX falsch angegeben wurde. Der Fehler 02h wird zurückgegeben, wenn der Landeskode oder die Zeichensatztabelle nicht gefunden wurden.

Bemerkung:

Entsprechend des Informationskennzeichens in AL wird das Feld für die erweiterte Landesinformation (ES:DI) unterschiedlich gefüllt.

*** DOS-Funktionen ***

Für AL=1 (Ermittlung der Landesinformation) enthält der Speicherbereich ES:DI folgende Angaben:

Offset hex/dezi	Anzahl Byte	Bedeutung
00h 00	1	Informationskennzeichen (=01)
01h 01	2	Grösse der Tabelle (max. 38)
03h 03	2	Landeskod
05h 05	2	Zeichensatztabelle
07h 07	2	Format von Datum
09h 09	5	Währungssymbol (mit 00h begrenzt)
0Eh 14	2	Tausenderinterpunktionszeichen
10h 16	2	Dezimalinterpunktionszeichen
12h 18	2	Trennzeichen im Datum
14h 20	2	Trennzeichen in der Zeitangabe
16h 22	1	Währungsformat
17h 23	1	Anzahl von signifikanten Dezimalstellen in der Währungsangabe
18h 24	1	Zeitformat
19h 25	4	Adresse für Aufruf eines Programmes für die Behandlung von ASCII-Zeichen > 7FH
1Dh 29	2	Trennzeichen in Datenliste
1Fh 31	10	Reserviert

Diese Tabelle ist ab Offset 07h identisch mit der Tabelle der landesspezifischen Information, die durch die DOS-Funktion 38h geliefert wird.

*** DOS-Funktionen ***

Für AL=2 (Ermittlung einer allgemeinen Konvertierungstabelle) enthält der Speicherbereich ES:DI folgende Angaben:

Offset	Anzahl	Bedeutung
hex/dez	Byte	
00h 00	1	Informationskennzeichen (=02)
01h 01	4	Zeiger auf Konvertierungstabelle (Offset, Segment)

Die Konvertierungstabelle ist 130 Byte lang (2 Byte Längenangabe und anschliessend 128 Byte Konvertierungswerte für ASCII-Zeichen mit der Kodierung grösser gleich 128 bzw. 80h).

Für AL=4 (Ermittlung einer Konvertierungstabelle für Dateinamen) enthält der Speicherbereich ES:DI folgende Angaben:

Offset	Anzahl	Bedeutung
hex/dez	Byte	
00h 00	1	Informationskennzeichen (=04)
01h 01	4	Zeiger auf Konvertierungstabelle für Dateinamen (Offset, Segment)

Diese Konvertierungstabelle ist ebenfalls 130 Byte lang (2 Byte Längenangabe und anschliessend 128 Byte Konvertierungswerte für ASCII-Zeichen des Dateinamens mit der Kodierung grösser gleich 128 bzw. 80h).

Für AL=5 (Ermittlung einer Tabelle mit ungültigen Zeichen) enthält der Speicherbereich ES:DI folgende Angaben:

Offset	Anzahl	Bedeutung
hex/dez	Byte	
00h 00	1	Informationskennzeichen (=05)
01h 01	4	Zeiger auf Tabelle, die die ungültigen Zeichen für den Dateinamen enthält (Offset, Segment)

Die Tabelle enthält auf Offset 0 (2 Byte) die Anzahl der ungültigen Zeichen und anschliessend die Zeichen selbst.

*** DOS-Funktionen ***

Für AL=6 (Ermittlung einer allgemeinen Vergleichsfolge) enthält der Speicherbereich ES:DI folgende Angaben:

Offset	Anzahl	Bedeutung
hex/dez	Byte	
00h 00	1	Informationskennzeichen (=06)
01h 01	4	Zeiger auf Tabelle mit Vergleichsfolge (Offset, Segment)

Diese Tabelle ist 258 Byte lang (2 Byte Längenangabe und anschließend 256 Byte ASCII-Zeichen in einer bestimmten Anordnung).

9.2.2.89. ERMITTELN/SETZEN DER GLOBALEN ZEICHENSATZTABELLE INT21: 66h

Funktions:
Die Zeichensatztabelle wird ermittelt bzw. gesetzt.

9.2.2.89.1. Ermitteln der Zeichensatztabelle INT21: 66h 01h

Funktions:
Die aktive und die System-Zeichensatztabelle wird ermittelt.

Aufruf:

Register	Inhalt/Bedeutung
AH	66h
AL	01h

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0
BX	aktive Zeichensatztabelle
DX	System-Zeichensatztabelle

*** DOS-Funktionen ***

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h ungültige Funktionsnummer
AX	02h Pfad nicht gefunden

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde. Der Fehler 02h wird zurückgegeben, wenn der Landeskodex oder die Zeichensatztabelle nicht gefunden wurden.

9.2.2.89.2. Setzen der Zeichensatztabelle

INT21: 66h 02h

Funktion:

Es wird eine neue aktive Zeichensatztabelle definiert.

Aufruf:

Register	Inhalt/Bedeutung
AH	66h
AL	02h
BX	aktive Zeichensatztabelle

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h ungültige Funktionsnummer
AX	02h Pfad nicht gefunden
AX	41h Zugriff verweigert

Der Fehler 01h tritt auf, wenn in AL eine nicht zulässige Funktionsnummer angegeben wurde. Der Fehler 02h wird zurückgegeben, wenn die Zeichensatztabelle nicht gefunden wurde bzw. nicht mit dem aktuellen Landeskodex korrespondiert. Wenn bei einem zu selektierenden Gerät ein Fehler auftritt, wird der Fehler 41h gemeldet.

Bemerkung:

- Um diese DOS-Funktion überhaupt nutzen zu können, muss der Befehl NLSFUNC und alle Geräte fehlerfrei installiert sein.

2. Vom Betriebssystem wird die neue Zeichensatztafel von der Datei COUNTRY.SYS in einen residenten Speicherbereich übertragen. Das Betriebssystem benutzt die neue Zeichensatztafel, um alle angeschlossenen Geräte, die die Zeichensatztafel umschalten können, zu selektieren. Diese Geräte müssen in der CONFIG.SYS angegeben werden. Sollten die Zeichensatztafel und der aktuelle Landeskod nicht korrespondieren, so ist das System mit einer entsprechenden CONFIG.SYS neu zu starten (Reset).

9.2.2.90. SETZEN DER HANDLERANZAHL

INT21: 67h

Funktion:

Die Funktion ermöglicht die Unterstützung von mehr als 20 eröffneten Händlern pro Prozess.

Aufruf:

Register	Inhalt/Bedeutung
AH	67h
BX	Anzahl der gleichzeitig eröffneten Händler

Der Standardwert für BX ist 20. Bei Angabe einer kleinere Anzahl wird der Standardwert angenommen. Die maximale Anzahl der gleichzeitig eröffneten Händler darf nicht den Wert 0FFFFh annehmen.

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	01h ungültige Funktionsnummer
AX	04h zu viel eröffnete Dateien (keine Dateinummern mehr frei)
AX	08h nicht genügend Speicherplatz

Der Fehler 01h tritt auf, wenn in BX der Wert 0FFFFh angegeben wurde. Der Fehler 04h wird zurückgegeben, wenn im System bereits mehr Händler eröffnet sind, als durch den Funktionsaufruf festgelegt werden sollen. Existiert im Betriebssystem nicht genügend freier Speicherplatz, so wird Fehler 08h zurückgegeben.

Bemerkung:

1. Es erfolgt eine Prüfung, ob die augenblickliche Anzahl von gleichzeitig zu eröffneten Händlern grösser als die gewünschte Anzahl ist. Bevor der neue Wert als aktuelle Anzahl eingetragen werden kann, müssen erst alle Händler, die über die gewünschte Anzahl hinaus existieren, geschlossen werden.
2. Die Festlegung gleichzeitig eröffneter Händler kann auch in der CONFIG.SYS mit dem Kommando FILES erfolgen, wenn die gewünschte Anzahl kleiner gleich 255 sein soll.
3. Benötigt ein Anwendungsprogramm viele gleichzeitig eröffnete Händler, kann diese Funktion die Notwendigkeit des Händlerwechsels verringern.
4. Wenn die zu definierende Anzahl gleichzeitig eröffneter Händler grösser als der Standardwert ist, muss vor Aufruf dieser DOS-Funktion dem Betriebssystem genügend Speicher zur Verfügung gestellt werden, um die erweiterte Händlerliste anzulegen. Dies ist mit dem DOS-Funktionsaufruf "Zugewiesenen Speicher freigeben" (4Ah) möglich.

9.2.2.91. UBERGABE AN HANDLEH

INT21: 68h

Funktion:

Diese Funktion bewirkt, dass alle gepufferten Daten einer Datei auf ein Gerät geschrieben werden (anstelle der normalen Folge von Dateischliessen und -eröffnen, um die Daten zu sichern).

Aufruf:

Register	Inhalt/Bedeutung
AH	68h
BX	Händlernummer

Rückkehr ohne Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	0

Rückkehr mit Fehler:

Reg./Flag	Inhalt/Bedeutung
CF	1
AX	06h, ungültige Händlernummer

Bemerkung:

Diese Funktion wirkt nur in Verbindung mit einem Netzwerkprogramm.

10. Strukturen von COM- und EXE-Dateien

In dem Betriebssystem existieren 2 Typen von ausführbaren Programmen, deren Aufbau und Behandlung im System unterschiedlich sind.

10.1. PSP (Program-Segment-Präfix)

Ein PSP dient als allgemeine Schnittstelle zwischen Betriebssystem und allen Typen von Nutzerprogrammen (COM- und EXE-Dateien).

Der PSP ist wie folgt aufgebaut:

Offset hex/dez	Anzahl Bytes	Inhalt/Bedeutung
00h 00	2	INT 20h Ein Sprung aus dem Nutzerprogramm an diesen Offset bewirkt die Beendigung des Programmes.
02h 02	2	Zeiger auf die erste freie Segmentadresse, die für das Nutzerprogramm nicht mehr verfügbar ist. Der Speicher kann jedoch ab dieser Adresse durch die DOS-Funktion "Speicherzuweisen" (48h) für das Nutzerprogramm reserviert werden.
04h 04	6	Reserviert
0Ah 10	4	Adresse des Interruptvektors 22h
0Eh 14	4	Adresse des Interruptvektors 23h
12h 18	4	Adresse des Interruptvektors 24h
16h 22	2	Segmentadresse des aufrufenden Programms; meist die Adresse von COMMAND.COM.
18h 24	20	Reserviert Beinhaltet eine Liste für die standardmässig 20 möglichen gleichzeitig eröffneten Händler
2Ch 44	2	Segmentadresse des die Programmumgebung beschreibenden Bereichs. In der Programmumgebung stehen u.a. Angaben: - über welchen Pfad der Kommando-Prozessor aktiviert werden kann, - über alle mit SET definierten Variablen sowie - über den durch PATH definierten Suchpfad für das Auffinden von Programmen.
2Eh 46	4	Reserviert Beinhaltet die Adresse des Stapels während der DOS-Funktionsaufrufe.
32h 50	2	Reserviert Beinhaltet die aktuelle Anzahl der möglichen gleichzeitig eröffneten Händler
34h 52	28	Übergabebereich für betriebssysteminterne Werte

*** Strukturen von COM- und EXE-Dateien ***

Offset hex/dez	Anzahl Bytes	Inhalt/Bedeutung
50h 80	3	Kodierung des INT 21h und RET FAR Zur Ausführung der DOS-Funktionen über den Aufruf durch CALL FAR.
53h 83	9	Reserviert
5Ch 92	16	FCB1 enthält die erste im aufrufenden Kommando übergebene Dateibezeichnung im Format eines nicht eröffneten FCB.
6Ch 108	16	FCB2 enthält die zweite im aufrufenden Kommando übergebene Dateibezeichnung im Format eines nicht eröffneten FCB.
7Ch 124	4	Reserviert
80h 128	128	DTA Dieser Bereich wird nach dem Programmstart als Puffer für die Datenträgerarbeit benutzt. Durch die DOS-Funktion "Setzen DTA" (1Ah) kann ein anderer Bereich zugewiesen werden. Nach dem Programmaufruf ist in diesem Bereich die dem Aufruf folgende Kommandozeile enthal- ten. Dabei steht in 80h die Anzahl und ab 81h die Kommandozeile selbst.

Bemerkungen:

Die Einträge von Offset 00h bis 50h im PSP sollten nicht durch den Nutzer verändert werden.

Enthielt einer der Parameter in der Kommandozeile ab Offset 81h eine Pfadbezeichnung, so wird in den entsprechenden FCB nur die gültige Laufwerknummer eingetragen.

Wurden in der Kommandozeile die Zeichen "<", ">" oder "|" eingegeben, so werden sie (einschliesslich der mit ihnen verknüpften Dateibezeichnung) nicht in den Bereich ab Offset 80h eingetragen. Es erfolgt eine Umleitung von Standardeingabe und Standardausgabe, die für den Nutzer nicht transparent ist.

10.2. Struktur einer COM-Datei

Eine COM-Datei hat eine maximale Grösse von 64 KByte und ist auf dem Datenträger folgendermassen gespeichert:

- Kode und Daten der COM-Datei mit variabler Länge.

Der Ladevorgang geschieht folgendermassen:

1. Entsprechend der Speicherverwaltung des Systems wird ein freies Segment vom Betriebssystem gesucht und dort ein PSP angelegt.
2. Die Register CS, DS, ES und SS werden auf den Anfang des PSP dieses Segmentes gesetzt.

*** Strukturen von COM- und EXE-Dateien ***

3. Das Register SP wird auf FFFh gesetzt. Die Adressen FFFh und FFFFh werden auf 00h gesetzt.

4. Das Register IP wird auf 100h eingestellt.

Nach dem Ladevorgang zeigt AX die Gültigkeit der Laufwerkangabe an, die mit den ersten beiden Parametern in der Kommandozeile eingegeben wurden. Es gilt:

- AL=FFh Der erste Parameter enthält eine ungültige Laufwerkangabe.
- AL=00h sonst
- AH=FFh Der zweite Parameter enthält eine ungültige Laufwerkangabe.
- AH=00h sonst

Das Betriebssystem reserviert nach dem Laden von COM-Dateien den gesamten zur Verfügung stehenden Speicher (entsprechend der Konfiguration der Maschine). Wenn das Nutzerprogramm ein weiteres Programm laden soll (DOS-Funktion "Programm laden mit und ohne ausführen" (4Bh)), so muss es vorher einen entsprechenden Speicherbereich durch Aufruf der DOS-Funktion "Zugewiesenen Speicherblock verändern" (4Ah) freigeben.

10.3. Struktur einer EXE-Datei

Im Gegensatz zu COM-Dateien sind EXE-Dateien in ihrer Grösse nur durch die Systemparameter (z.B. maximaler Speicherbereich oder Kapazität der verwendeten Speichermedien) begrenzt.

EXE-Dateien unterliegen keinen Segmentbeschränkungen. Sie können an jede beliebige Stelle im Systemspeicher geladen werden. Dazu bedarf es jedoch besonderer Massnahmen. Diese stehen in einem Vorsatz zu dem eigentlichen Programm.

Eine EXE-Datei besteht aus zwei Teilen und ist auf dem Datenträger folgendermassen gespeichert:

- Header und Relokationstabelle (Vorsatz)
- Lademodul (Programm).

Der Header und die Relokationstabelle befinden sich am Beginn der Datei.

Der Header enthält Informationen darüber, wohin das Programm in den Speicher geladen wird und wie bestimmte Anfangsbedingungen für das Programm stehen.

*** Strukturen von COM- und EXE-Dateien ***

Er hat folgenden Aufbau:

Offset	Anzahl	Inhalt/Bedeutung
hex/dez	Bytes	
00h 00	2	Enthält "MZ" (4Dh,5Ah) als Kennzeichen für eine EXE-Datei
02h 02	2	Anzahl der Bytes im letzten Datensatz
04h 04	2	Grösse der EXE-Datei einschliesslich des Headers in Anzahl Datensätze (512 Byte)
06h 06	2	Anzahl der Eintragungen in der Relokationstabelle
08h 08	2	Grösse des Headers in Paragraphen (16 Byte)
0Ah 10	2	Mindestanzahl an Paragraphen, die nach dem Laden über das Ende des Programms hinaus benötigt werden (MINALLOC)
0Ch 12	2	Maximale Anzahl an Paragraphen, die nach dem Laden über das Ende des Programms hinaus benötigt werden (MAXALLOC)
0Eh 14	2	Wert des Registers SS nach dem Laden
10h 16	2	Wert des Registers SP nach dem Laden
12h 18	2	Negative Prüfsumme über die EXE-Datei (nur Programm)
14h 20	2	Wert des Registers IP nach dem Laden
16h 22	2	Wert des Registers CS nach dem Laden
18h 24	2	Offset der Relokationstabelle vom Beginn der Datei
1Ah 26	2	Nummer des durch den Linker erzeugten Overlay (0 ist das Kennzeichen für den residenten Teil des Programms)
1Ch 28	2	Tabelle der Relokationseinträge (variable Länge)

Bemerkung zu den Offsets 0Ah und 0Ch:

MAXALLOC (Offset 0Ch) kann durch einen Schalter beim Linken gesetzt werden. Wird der Wert nicht explizit angegeben, so wird MAXALLOC auf FFFFh festgelegt. Bei FFFFh wird für das Programm der gesamte verfügbare Speicher reserviert. Sind MINALLOC (Offset 0Ah) und MAXALLOC gleich 0000h, so wird das Nutzerprogramm an die höchstmögliche Speicheradresse geladen.

Der Ladevorgang geschieht folgendermassen:

1. Entsprechend der Speicherverwaltung des Systems wird ein freies Segment vom Betriebssystem gesucht und dort ein PSP angelegt.
2. Einlesen des Headers von Offset 00h bis 18h
Aus der Grösse des Programms SIZE (Offset 04h/05h) sowie aus MINALLOC (Offset 0Ah) und MAXALLOC (Offset 0Ch) wird die für das Laden notwendige Speichergrösse ermittelt.

*** Strukturen von COM- und EXE-Dateien ***

- Es können 3 Fälle auftreten:
- SIZE + MINALLOC > verfügbarer Speicher
Der Ladeprozess wird wegen unzureichendem Speicher abgebrochen.
 - SIZE + MAXALLOC < verfügbarer Speicher
Das Betriebssystem reserviert den entsprechenden Speicherbereich.
 - SIZE + MINALLOC < verfügbarer Speicher < SIZE + MAXALLOC
Das Betriebssystem reserviert den gesamten restlichen Speicher.
3. Das Betriebssystem bestimmt aus der aktuellen Systembelegung und den Angaben im Header das erste für den Ladevorgang verwendbare Segment. Dort wird ein PSP angelegt, und das Programm wird in den Speicher geladen.
4. Einlesen der Relokationstabelle
Anhand dieser Einträge werden die notwendigen Modifikationen des Programms vorgenommen.
5. Die Register CS und SS werden unter Beachtung des "Segmentoffsets" und der Angabe im Header (Offset 16h/17h und 0Eh/0Fh) gesetzt.
Die Register SP und IP werden direkt dem Header (Offset 10h/11h und 14h/15h) entnommen.
Die Register DS und ES werden mit dem "Segmentoffset" geladen. Sie adressieren somit den PSP.
6. Nach dem Ladevorgang zeigt AX die Gültigkeit der Laufwerkangabe an, die mit den ersten beiden Parametern in der Kommandozeile eingegeben wurden. Es gilt:
- AL=FFh Der erste Parameter enthält eine ungültige Laufwerkangabe.
 - AL=00h sonst
 - AH=FFh Der zweite Parameter enthält eine ungültige Laufwerkangabe.
 - AH=00h sonst

10.4. Relokationstabelle

Die Relokationstabelle enthält die Adressen der im Programm zu modifizierenden Segment-Informationen.
Beim Verschieben eines Programms sind diejenigen Befehle zu verändern, die eine Segmentadresse enthalten. Nach dem Linken ist das Programm an der Speicheradresse 0000h:0000h lauffähig. Um es jedoch an anderer Speicheradresse lauffähig zu machen, werden alle Befehle, die Segmentadressen enthalten, durch Addieren des Offsets des Segmentes (Segmentteil der Adresse des für dieses Programm erzeugten PSP) bzgl. 0000h modifiziert.
Die Einträge in der Relokationstabelle bestehen immer aus jeweils 4 Bytes, d.h. 2 Bytes Offset und 2 Bytes Segment.
Durch den Offset 18h/19h des Headers wird der erste Eintrag der Relokationstabelle gekennzeichnet.

*** Strukturen von COM- und EXE-Dateien ***

In der Relokationstabelle kann eine variable Anzahl (Offset 06h/07h) von Relokationseinträgen stehen.

Beispiel:

Das folgende Beispiel zeigt ein kurzes Programm, welches das Betriebssystem prüft, ob es die DCP-Version 3.30 besitzt. Es ist so aufgebaut, dass 3 Befehle Segmentadressen enthalten. Die endgültigen Segmentadressen sind dem Assembler noch unbekannt. Er markiert diese Adressen durch "---- R".

```

                                NAME      VERSION
0000                          STAPEL     SEGMENT STACK
0000 0032[                      dw      50 dup(0)
                                ]
0064                          STAPEL     ENDS

;-----
0000                          PROG      SEGMENT BYTE  'CODE'
                                ASSUME    CS:PROG, DS:PROG
                                ASSUME    ES:NOTHING, SS:STAPEL

0000                          BEGIN:
0000 08 ---- R                 mov    ax,PROG    ;DS setzen
0003 8E 08                     mov    ds,ax
0005 B4 30                     mov    ah,30h
0007 CD 21                     int    21h        ;DCP-Version abfragen
0009 86 E0                     xchg  ah,al
000B 3D 031E                   cmp    ax,31Eh   ;Vergleich auf 3.30
000E 75 05                     jnz   ENDE       ;Version ist nicht 3.30
0010 9A 0017 ---- R           call  far ptr RICHTIG ;Aufruf Ausgabeprozedur
0015                          ENDE:
0015 B4 4C                     mov    ah,4Ch
0017 CD 21                     int    21h        ;Programm beenden

0019                          PROG      ENDS

;-----
0000                          AUSGABE   SEGMENT BYTE  'CODE'
                                ASSUME    CS:AUSGABE, DS:AUSGABE

0000                          TEXT
0000 44-43 50 2D 56 65 72      db    'DCP-Version ist 3.30R'
                                73 69 6F 6E 20 69 73
                                74 20 33 2E 33 30 24
0015 0D 0A                      db    0Dh,0Ah

```

*** Strukturen von COM- und EXE-Dateien ***

```

0017                                RICHTIG    PROC    FAR

0017 B8 ---- R                    mov    ax,AUSGABE ;DS setzen
001A 8E D8                        mov    ds,ax
001C BA 0000 R                    mov    dx,OFFSET TEXT ;Offset Text einstellen
001F B4 07                        mov    ah,07h
0021 CD 21                        int    21h          ;Ausgabe der Zeichenkette
0023 CB                            ret

                                RICHTIG    ENDP

0024                                AUSGABE    ENDS

                                END BEGIN
    
```

Die nachfolgende Tabelle zeigt den Inhalt (Hex-Werte) des EXE-Headers und des übersetzten Programms für das obige Beispiel:

EXE-Header:

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0000	4D	5A	A1	00	02	00	03	00-20	00	00	00	FF	FF	00	00	00
0010	64	00	FD	F0	04	00	06	00-1E	00	00	00	01	00	05	00	00
0020	06	00	17	00	06	00	25	00-07	00	00	00	00	00	00	00	00
0030	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0040	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0050	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0060	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0070	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
:																
:																
01D0	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
01E0	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
01F0	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00

Ende des EXE-Headers

Beginn des Programms:

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0200	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0210	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0220	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0230	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0240	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0250	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00
0260	00	00	00	00	B8	06	00	8E-D8	B4	30	CD	21	86	E0	3D	00
0270	1E	03	75	05	9A	24	00	07-00	B4	4C	CD	21	44	43	50	00
0280	2D	56	65	72	73	69	6F	6E-20	69	73	74	20	33	2E	33	00
0290	30	24	0D	0A	B8	07	00	8E-D8	BA	0D	00	B4	09	CD	21	00
02A0	CB															

Ende des Programms

*** Strukturen von COM- und EXE-Dateien ***

In einzelnen bedeutet dabei:

Offset hex	Inhalt	Erklärung
00h	5A4Dh	Kennzeichen "MZ"
		Die Dateigrösse lässt sich aus den Offsets 02h bis 05h nach folgender Formel bestimmen:
02h	00A1h	((Offset04h/05h)-1)*512+(Offset02h/03h)
04h	0002h	
06h	0003h	Anzahl der Einträge
08h	0020h	Grösse des Headers (200h)
0Ah	0000h	MINALLOC
0Ch	FFFFh	MAXALLOC
0Eh	0000h	Anfangswert SS
10h	0064h	Anfangswert SP
12h	F0FDh	Prüfsumme
14h	0004h	Anfangswert IP
16h	0006h	Anfangswert CS
18h	001Eh	Offset der Relokationstabelle
1Ah	0000h	Durch Linker erzeugt
1Ch	0001h	Reserviert
		Relokationstabelle (bzgl. 0000h:0000h)
1Eh	0006h:0005h	Entspricht 6*10h+05h = 00065h
21h	0006h:0017h	Entspricht 6*10h+17h = 00077h
25h	0007h:0025h	Entspricht 7*10h+25h = 00095h

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

Anlage_1: Die BIOS-Interrupts im Überblick

INT Nr.	Aufruf	Rückkehr
00h	Division durch Null	---
01h	Einzel-schrittverarbeitung	---
02h	NMI-Unterbrechung	---
03h	Prüfpunkt	---
04h	Oberlauf-Unterbrechung	---
05h	Bildschirminhalt drucken	---
06h	Reservierter Interrupt	---
07h	Reservierter Interrupt	---
08h	Zeitgeber	---
09h	Tastatur	---
0Ah	Reservierter Interrupt	---
0Bh	Kommunikation	---
0Ch	Kommunikation	---
0Dh	Harddisk	---
0Eh	Disketten	---
0Fh	Drucker	---

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
10h	Videosteuerung AH=Funktionsnummer	
10h 00h	Bildschirmmode setzen AH=00h AL=einzustellender Mode	----
10h 01h	Kursorart setzen AH=01h CX=Kursorart	----
10h 02h	Kursorposition setzen AH=02h DH=Zeile des Cursors DL=Spalte des Cursors BH=Seite, auf der der Cursor gesetzt werden soll	----
10h 03h	Lesen der Kursorposition AH=03h BH=Seite	DH=Zeile der Kursorposition DL=Spalte der Kursorposition CX=Aktuelle Kursorart
10h 04h	Lichtstiftposition ermitteln AH=04h	AH=0
10h 05h	Aktuelle Seite setzen AH=05h AL=Neue aktive Seite	----
10h 06h	Fenster nach oben rollen AH=06h AL=Anzahl Zeilen CH=Oberste Zeile des Rechtecks CL=Linke Spalte des Rechtecks DH=Unterste Zeile des Rechtecks DL=Rechte Spalte des Rechtecks BH=Attribut für zu löschende Zeilen	----
10h 07h	Fenster nach unten rollen AH=07h AL=Anzahl Zeilen CH=Oberste Zeile des Rechtecks CL=Linke Spalte des Rechtecks DH=Unterste Zeile des Rechtecks DL=Rechte Spalte des Rechtecks BH=Attribut für zu löschende Zeilen	----
10h 08h	Zeichen und Attribut lesen AH=08h BH=Seite	AL=Gelesenes Zeichen AH=Gelesenes Attribut

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
10h 09h	Zeichen und Attribut schreiben AH=09h BH=Seite AL=Auszugebendes Zeichen BL=Auszugebendes Attribut CX=Wiederholungsfaktor	----
10h 0Ah	Zeichen schreiben AH=0Ah BH=Seite AL=Auszugebendes Zeichen BL=Zeichenfarbe	----
10h 0Bh	Farbe auswählen AH=0Bh BX=Farbauswahl	----
10h 0Ch	Punkt zeichnen AH=0Ch DX=Punktzeile CX=Punktspalte AL=Farbe des Punktes	----
10h 0Dh	Punkt lesen AH=0Dh DX=Punktzeile CX=Punktspalte	AL=Farbe des Punktes
10h 0Eh	Zeichenausgabe AH=0Eh AL=Auszugebendes Zeichen BL=Zeichenfarbe	----
10h 0Fh	Bildschirmstatus ermitteln AH=0Fh	AH=Zeichenzahl pro Zeile AL=Aktueller Bildschirmode BH=Aktuelle Seite
10h 10h	Farbpalette programmieren AH=10h AL=Unterfunktion BX=Wert ES:DX=Adresse der Palette	----
10h 10h/00h	Eine Farbe der Palette ändern AH=10h AL=00h BL=Farbnummer (0-15) BH=Farbwert (0-63)	----

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
10h	10h/02h Palette ändern AH=10h AH=02h ES:DX=Adresse eines Puffers mit 17 Farbwerten	----
10h	10h/03h Blinken erlauben/verbieten AH=10h AL=03h BL=00 Blinken verboten BL=01 Blinken erlaubt	----
10h	11h Zeichengenerator programmieren AH=11h AL=Unterfunktion CX=Anzahl zu progr. Zeichen DX=Code des ersten zu progr. Zeichens/Zeilenzahl BH=Anzahl Pixelzeilen pro Zeichen BL=Zeilenspezifikation	Status bei Unterfunktion 30h
10h	11h/01h Standardzeichensatz 14x8 aktiv setzen (ohne Veränderung Systemeinstellung) AH=11h AL=01h	----
10h	11h/02h Standardzeichensatz 8x8 aktiv setzen (ohne Veränderung Systemeinstellung) AH=11h AL=02h	----
10h	11h/04h Standardzeichensatz 16x8 aktiv setzen (ohne Veränderung Systemeinstellung) AH=11h AL=04h	----
10h	11h/10h Nutzerzeichensatz laden (mit Veränderung Systemeinstellung) AH=11h AL=10h ES:BP=Adresse der Zeichenauster CX =Auswahl des programmierten Zeichens DX =Kode des ersten zu programmierenden Zeichens BH =Anzahl Byte/Zeichen	----
10h	11h/11h Standardzeichensatz 14x8 aktiv setzen (mit Veränderung Systemeinstellung) AH=11h AL=11h	----
10h	11h/12h Standardzeichensatz 8x8 aktiv setzen (mit Veränderung Systemeinstellung) AH=11h AL=12h	----

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
10h	11h/14h Standardzeichensatz 16x8 aktiv setzen (mit Veränderung Systemeinstellung) AH=11h AL=14h	----
10h	11h/20h INT 1Fh ändern AH=11h AL=20h ES:PB=neue INT 1Fh-Adresse	----
10h	11h/21h Zeichengenerator für Grafikaodes ändern AH=11h AL=21h ES:PB=Adresse des Zeichengenerators CX =Byte/Zeichen BL =00 Zeichenzahl wird in DL übergeben =01 14 Zeilen =02 25 Zeilen =03 43 Zeilen DL =Zeilenzahl	----
10h	11h/22h Standardzeichensatz 14x8 für Grafikaodus setzen AH=11h AL=22h BL/DL=wie in Funktion AL=21h	----
10h	11h/23h Standardzeichensatz 8x8 für Grafikaodus aktiv setzen AH=11h AL=23h BL/DL=wie in Funktion AL=21h	----
10h	11h/24h Standardzeichensatz 16x8 für Grafikaodus aktiv setzen AH=11h AL=24h BL/DL=wie in Funktion AL=21h	----
10h	11h/30h Status ermitteln AH=11h AL=30h BH=Auswahl Zeichengenerator- adresse	CX=Byte/Zeichen DL=Zeilenzahl ES:BP=Adressen-Rückgabe (ent- sprechend der Einstel- lung von BH)
10h	12h Adapterstatus ermitteln AH=12h BL=10h	AX,BX,CX = Adapterstatus

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
10h	13h Zeichenkettenausgabe AH=13h ES:BP=Adresse der Zeichenkette CX=Zeichenzahl der Zeichenkette DX=Kursorposition für die Zeichenkette BH=Seite AL=Modifikationswert BL=Attribut	----
11h	Schalterbelegung ----	AX=Schalterbelegung
12h	Speichergröße ----	AX=verfügbare Speicher in KByte
13h	Disketten-Interrupts AH=Funktionsnummer	----
13h	00h Rücksetzen FD AH=00h	CF=0 ohne Fehler CF=1 mit Fehler AH=00h bzw. Fehlerkode
13h	01h Lesen des Diskettenstatus AH=01h	CF=0 ohne Fehler CF=1 mit Fehler AH=00h bzw. Fehlerkode AL=Fehlerkode
13h	02h Lesen Sektoren (FD) AH=02h DL=Laufwerkadresse DH=Kopfadresse CH=Spuradresse CL=Sektoradresse AL=Anzahl Sektoren ES:BX=Speicheradresse	CF=0 ohne Fehler CF=1 mit Fehler AH=00h bzw. Fehlerkode AL=Anzahl gelesener Sektoren
13h	03h Schreiben Sektoren (FD) AH=03h DL=Laufwerkadresse DH=Kopfadresse CH=Spuradresse CL=Sektoradresse AL=Anzahl Sektoren ES:BX=Speicheradresse	CF=0 ohne Fehler CF=1 mit Fehler AH=00h bzw. Fehlerkode AL=Anzahl geschr. Sektoren

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
13h 04h	Kontroll-Lesen (FD) AH=04h DL=Laufwerkadresse DH=Kopfadresse CH=Spuradresse CL=Sektoradresse AL=Anzahl Sektoren	CF=0 ohne Fehler AH=00h bzw. Fehlerkode AH=00h bzw. Fehlerkode AL=Anzahl kontroll. Sektoren
13h 05h	Formatieren einer Spur (FORMAT) AH=05h DL=Laufwerkadresse DH=Kopfadresse CH=Spuradresse ES:BX=Tabellenadresse	CF=0 ohne Fehler CF=1 mit Fehler AH=00h bzw. Fehlerkode
13h 08h	Lesen Laufwerk-Parameter (FD) AH=08h	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=01h ES:DI=DISK-BASE BH=00h DL=DISK-BASE Typ DH=grösste Kopfadresse DL=Anzahl vorh. Laufwerke CH=grösste Spuradresse CL=grösste Sektoradresse
13h 15h	Erkennbarkeit von Diskettenwechsel (READ DASD TYPE) AH=15h DL=Laufwerkadresse	CF=0 AH=Testergebnis (00h 01h)
13h 16h	Diskettenwechsel-Information AH=16h DL=Laufwerkadresse	CF=1 AH=01h (wird nicht erkannt)
13h 17h	Setzen Format-Parameter (SET DASD TYPE) AH=17h DL=Laufwerkadresse AL=Diskettenformat	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=01h
13h 18h	Setzen des Diskettentyps (SET MEDIA TYPE) AH=18h DL=Laufwerkadresse CH=maximale Spur CL=maximaler Sektor	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=0Ch ES:DI=Adresse DISK-BASE
13h	Harddisk-Interrupts AH=Funktionsnummer	

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
13h 00h	Rücksetzen Harddisk und FDC AH=00h DL=Laufwerkadresse	CF=0 ohne Fehler CF=1 mit Fehler AH=Fehlerkode
13h 01h	Lesen des Diskstatus AH=01h DL=Laufwerkadresse	CF=0 AH=00h AL=Fehlerkode
13h 02h	Lesen Sektoren (HD) AH=02h DL=Laufwerkadresse DH=Kopfadresse CH=Zylinderadresse CL=Zylinder-/Sektoradresse 1) AL=Anzahl Sektoren ES:BX=Speicheradresse	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
13h 03h	Schreiben Sektoren (HD) AH=03h DL=Laufwerkadresse DH=Kopfadresse CH=Zylinderadresse CL=Zylinder-/Sektoradresse 1) AL=Anzahl Sektoren ES:BX=Speicheradresse	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
13h 04h	Kontroll-Lesen (HD) AH=04h DL=Laufwerkadresse DH=Kopfadresse CH=Zylinderadresse CL=Zylinder-/Sektoradresse 1) AL=Anzahl Sektoren	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
13h 05h	Formatieren einer Spur (HD) AH=05h DL=Laufwerkadresse DH=Kopfadresse CH=Zylinderadresse CL=die höchsten 2 Bit der Zylinderadresse (6,7)	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
13h 06h	Formatieren einer Spur mit Kennzeichnung defakter Sektoren (HD) AH=06h DL=Laufwerkadresse DH=Kopfadresse CH=Zylinderadresse CL=die höchsten 2 Bit der Zylinderadresse (6,7)	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
13h 07h	Formatieren ab adressiertem Zylinder (HD) AH=07h DL=Laufwerkadresse DH=Kopfadresse CH=Zylinderadresse CL=die höchsten 2 Bit der Zylinderadresse (6,7)	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlercode
13h 08h	Lesen Laufwerk-Parameter (HD) AH=08h DL=Laufwerkadresse	CF=0 ohne Fehler AH=00h DH=Max.Kopfnummer DL=Anzahl vorh.Harddisk CH=Grösste Zylinderadresse CL=Grösste Sektoradresse CF=1 mit Fehler AH=Fehlercode
13h 09h	Initialisierung Laufwerk-Parameter (HD) AH=09h DL=Laufwerkadresse	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlercode
13h 0Ah	Lesen lang (HD) AH=0Ah DL=Laufwerkadresse DH=Kopfadresse CH=Zylinderadresse CL=Zylinder-/Sektoradresse 1) AL=Anzahl Sektoren ES:BX=Speicherbereich	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlercode
13h 0Bh	Schreiben lang (HD) AH=0Bh DL=Laufwerkadresse DH=Kopfadresse CH=Zylinderadresse CL=Zylinder-/Sektoradresse 1) AL=Anzahl Sektoren ES:BX=Speicherbereich	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlercode
13h 0Ch	Positionieren (HD) AH=0Ch DL=Laufwerkadresse DH=Kopfadresse CH=Zylinderadresse CL=die höchsten 2 Bit der Zylinderadresse (Bit 6 und 7) ES:BX=Speicherbereich	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlercode

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
13h 0Dh	Alternatives Rücksetzen (HD) AH=0Dh DL=Laufwerkadresse	CF=0 AH=00h CF=1 AH=Fehlerkode
13h 0Eh	Lesen Sektorpuffer (HD) AH=0Eh DL=Laufwerkadresse ES:BX=Speicherbereich	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
13h 0Fh	Schreiben Sektorpuffer (HD) AH=0Fh DL=Laufwerkadresse ES:BX=Speicherbereich	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
13h 10h	Test der Laufwerkbereitschaft (HD) AH=10h DL=Laufwerkadresse	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
13h 11h	Rekalibrieren (HD) AH=11h DL=Laufwerkadresse	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
13h 12h	RAM-Diagnose (HD) AH=12h DL=Laufwerkadresse	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
13h 13h	Laufwerk-Diagnose (HD) AH=13h DL=Laufwerkadresse	CF=0 ohne Fehler AH=00h CF=1 mit Fehler AH=Fehlerkode
14h	Ein- und Ausgabe über einen seriellen Anschluss AH=Funktionsnummer	
14h 00h	Initialisieren serieller Anschluss AH=00h AL=Parameter Initialisierung DX=Kanalnummer	AH=Leitungsstatus AL=Modemstatus

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
14h 01h	Senden eines Zeichens AH=01h AL=Zeichen DX=Kanalnummer	AH=Leitungsstatus AL=Modemstatus
14h 02h	Empfangen eines Zeichens AH=02h DX=Kanalnummer	AH=Leitungsstatus AL=Zeichen
14h 03h	Abfrage Status AH=03h DX=Kanalnummer	AH=Leitungsstatus AL=Modemstatus
15h	Verwaltungsfunktionen	
16h	Tastatur AH=Funktionsnummer	
16h 00h	Zeichen lesen AH=00h	AL=ASCII-Kode AH=Tastaturkode
16h 01h	Test Zeichen verfügbar AH=01h	ZF=Testergebnis
16h 02h	Tastaturstatus AH=02h	AL=Status der Shift-Tasten
16h 03h	Tastaturpuffer schreiben AH=03h CL=ASCII-Kode CH=Tastaturkode	AL=0 (ohne Fehler) AL=1 (mit Fehler)
16h 10h	Zeichen lesen erweitert AH=10h	AL=ASCII-Kode AH=Tastaturkode
16h 11h	Test Zeichen verfügbar erweitert AH=11h	ZF=Testergebnis
16h 12h	Tastaturstatus erweitert AH=12h	AX=Status der Shift-Tasten
17h	Ausgabe über parallele Schnittstelle an Drucker AH=Funktionsnummer	
17h 00h	Drucken eines Zeichens AH=00h AL=Zeichen DX=00h	AH=Status

*** Anlage 1: Die BIOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
17h 01h	Drucker initialisieren AH=01h DX=00h	AH=Status
17h 02h	Drucker-Status abfragen AH=02h DX=00h	AH=Status
18h	Reservierter Interrupt ----	----
19h	Urladen ----	----
1Ah	Tageszeit AH=Funktionsnummer	
1Ah 00h	Tageszeit lesen AH=00h	CX=höherwertiger Teil des Zählers DX=niederwertiger Teil des Zählers AL=Kennzeichen
1Ah 01h	Tageszeit setzen AH=01h CX=höherwertiger Teil des Zählers DX=niederwertiger Teil des Zählers	----
1Bh	Tastaturunterbrechung ----	----
1Ch	Zeitgeber Nutzer ----	----
1Dh	Adresse Bildschiraparameter ----	----
1Eh	Adresse Diskettenparameter-Block (DISK-BASE) ----	----
1Fh	Adresse grafischer Zeichensatz ----	----

- 1) Das Register CL enthält die Sektoradresse (Bit 0-5) und die werthöchsten 2 Bit der Zylinderadresse (Bit 6 und 7).

*** Anlage 2: Die DOS-Interrupts im Überblick ***

Anlage 2: Die DOS-Interrupts im Überblick

INT Nr.	Aufruf	Rückkehr
20h	Programm beenden CS = --> PSP	----
21h	Aufruf von DOS-Funktionen AH = Nummer der DOS-Funktion AL = Unterfunktion sonstige = Übergabeparameter entsprechend DOS-Funktion	----
22h	Abschlussadresse ----	----
23h	Behandlung bei CTRL-C ----	----
24h	Vektor der Behandlungsroutine für kritische Fehler AX,DI = Fehlerkode BP:SI = --> Device Header	----
25h	Absolutes Diskettenlesen AL = Laufwerknummer CX = Anzahl zu lesender Sektoren DX = Relative Sektor-Nummer DS:BX = --> Puffer	CF = 00h erfolgreich CF = 01h fehlerhaft AL = Fehlerkode AH = Fehlerkode
26h	Absolutes Diskettenschreiben AL = Laufwerknummer CX = Anzahl zu schreibender Sektoren DX = Relative Sektor-Nummer DS:BX = --> Puffer	CF = 00h erfolgreich CF = 01h fehlerhaft AL = Fehlerkode AH = Fehlerkode
27h	Beenden, aber resident bleiben CS:DX = --> 1. Byte nach resident zu machendem Bereich	----

*** Anlage 2: Die DOS-Interrupts im Überblick ***

Nr.	Aufruf	Rückkehr
2Fh	Multiplex-Interrupt AH = Multiplexnummer AL = Funktionsnummer	
2Fh 01h	Multiplex-Interrupt PRINT AH = 01h AL = Funktionsnummer	----
2Fh 01h/00h	Ermittlung des installierten Status AH = 01h AL = 00h	AL = installierter Status
2Fh 01h/01h	Submit-Datei AH = 01h AL = 01h DS:DX = --> Submit-Paket	----
2Fh 01h/02h	Dateiabbruch AH = 01h AL = 02h DS:DX = --> ASCII-Zeichenfolge	----
2Fh 01h/03h	Alle Dateien abbrechen AH = 01h AL = 03h	----
2Fh 01h/04h	Status AH = 01h AL = 04h	DS:DI = --> Druckwar- teschlange DX = Fehlerwert
2Fh 01h/05h	Statusende AH = 01h AL = 00h	AX = Fehlerkode
2Fh 02h/00h	Multiplex-Interrupt ASSIGN AH = 02h AL = 00h	AL = installierter Status
2Fh 10h/00h	Multiplex-Interrupt SHARE AH = 10h AL = 00h	AL = installierter Status

*** Anlage 2: Die DOS-Interrupts im Überblick ***

INT Nr.	Aufruf	Rückkehr
2Fh 07h	Multiplex-Interrupt APPEND AH = 07h AL = Funktionsnummer	----
2Fh 07h/00h	Ermittlung des installierten Status AH = 07h AL = 00h	AL = installierter Status
2Fh 07h/01h	Ausgabe Fehlertext und Programm beenden AH = 07h AL = 01h	----
2Fh 07h/02h	AX auf 0FFFFh setzen AH = 07h AL = 02h	AX = 0FFFFh
2Fh 07h/03h	Ermitteln der Adresse der APPEND-INT21-Routine AH = 07h AL = 03h ES:DI=--> Routine des aktuellen INT21	ES:DI = --> APPEND-INT21-Routine
2Fh 07h/04h	Zeiger auf APPEND-Liste AH = 07h AL = 04h	ES:DI = --> APPEND-Liste
2Fh 07h/10h	Ermittlung der APPEND-Version AH = 07h AL = 10h	AX = installierter Schalter BX = CX = 0000h DX = APPEND-Version

--> bedeutet: Zeiger auf

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Anlage 3: Die DOS-Funktionen im Überblick

Bei Aufruf der DOS-Funktion ist in Register AH bzw. AX die jeweilige Funktionsnummer einzustellen. Dazu sind noch zusätzlich die als Eintrittsparameter gekennzeichneten Werte zu übergeben.

Die Funktionsnummern sind hexadezimal angegeben.

Nr.	Aufruf	Rückkehr
00h	Programm beenden CS--> PSP	----
01h	Konsol-Eingabe mit Echo ----	AL=Zeichen
02h	Konsol-Ausgabe DL=Zeichen	----
03h	Hilfseingabe ----	AL=Zeichen
04h	Hilfsausgabe DL=Zeichen	----
05h	Ausgabe auf Listgerät DL=Zeichen	----
06h	Direkte Konsol-Ein-/Ausgabe DL=FFh (Eingabe) DL=00h-FEh (Ausgabe)	ZF=0 AL=Zeichen ZF=1 AL=00h, kein Zeichen
07h	Direkte Konsol-Eingabe ohne Echo ----	AL=Zeichen
08h	Konsol-Eingabe ohne Echo ----	AL=Zeichen
09h	Zeichenkettenausgabe DS:DX--> Zeichenkette mit EKZ="X"	----
0Ah	Eingabe Konsol-Puffer DS:DX--> Konsol-Puffer (Kapaz: Z1: ... : Zn: 0Dh)	Zeichen im Puffer
0Bh	Konsol-Status ----	AL=FFh, Zeichen vorhanden AL=00h, kein Zeichen vorhanden

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
0Ch	Löschen des Konsol-Puffers und Aufruf einer Standardeingabe AL=DOS-Funktionsnummer (01h,06h,07h,08h,0Ah)	AL=00h, ungültige Fkts.-nummer ----
0Dh	Diskette rücksetzen ----	----
0Eh	Auswahl Bezugs-Laufwerk DL=Laufwerknummer (0=LW A,...)	AL=Anzahl der Laufwerke
0Fh	Datei eröffnen DS:DX=--> nicht eröffneten FCB Offset 00h Laufwerknummer (0=Std-LW, 1=LW A, ...) Offset 01h Dateiname Offset 09h Dateityp	AL=00h erfolgreich AL=FFh nicht eröffnet
10h	Datei schliessen DS:DX=--> eröffneten FCB	AL=00h erfolgreich AL=FFh nicht geschlossen
11h	Suche nach dem ersten Eintrag DS:DX=--> nicht eröffneten FCB Offset 00h Laufwerknummer (0=Std-LW, 1=LW A, ...) Offset 01h Dateiname Offset 09h Dateityp	AL=00h erfolgreich AL=FFh nicht gefunden
12h	Suche nach dem nächsten Eintrag DS:DX=--> nicht eröffneten FCB	AL=00h erfolgreich AL=FFh nicht gefunden
13h	Datei löschen DS:DX=--> nicht eröffneten FCB	AL=00h erfolgreich AL=FFh nicht gelöscht
14h	Sequentielles Lesen DS:DX=--> eröffneten FCB	AL=00h erfolgreich AL=01h EOF, keine Daten gelesen =02h DTA zu klein =03h EOF, Teilsatz gelesen
15h	Sequentielles Schreiben DS:DX=--> eröffneten FCB	AL=00h erfolgreich AL=01h Datenträger voll =02h DTA zu klein
16h	Datei erstellen DS:DX=--> nicht eröffneten FCB	AL=00h erfolgreich AL=FFh nicht erstellt

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
17h	Datei umbenennen DS:DX--> modifizierten FCB Offset 00h Laufwerknummer (0=Std-LW, 1=LW A, ...) Offset 01h Quell-Dateiname Offset 09h Quell-Dateityp Offset 11h Ziel-Dateiname Offset 19h Ziel-Dateityp	AL=00h erfolgreich AL=FFh nicht umbenannt
19h	Abfrage Bezugs-Laufwerk ----	AL=Laufwerknummer (0=LW A,...)
1Ah	Setzen DTA DS:DX--> DTA	----
1Bh	Informationen der Zuordnungstabelle ----	AL=Anzahl Sektoren/Cluster CX=Anzahl Bytes/Sektor DS:BX--> Kennzeichenbyte der FAT DX=Anzahl Cluster
1Ch	Informationen der Zuordnungstabelle für bestimmtes Laufwerk DL=Laufwerknummer (0=Std-LW, 1=LW A,...)	AL=Anzahl Sektoren/Cluster CX=Anzahl Bytes/Sektor DS:BX--> Kennzeichenbyte der FAT DX=Anzahl Cluster
21h	Wahlfreies Lesen DS:DX--> eröffneten FCB Offset 21h bis 24h setzen	AL=00h erfolgreich AL=01h EOF, keine Daten gelesen =02h DTA zu klein =03h EOF, Teilsatz gelesen
22h	Wahlfreies Schreiben DS:DX--> eröffneten FCB Offset 21h bis 24h setzen	AL=00h erfolgreich AL=01h Datenträger voll =02h DTA zu klein
23h	Dateigröße DS:DX--> nicht eröffneten FCB Offset 0Eh und 0Fh setzen	AL=00h erfolgreich Offset 21h - 24h gesetzt AL=FFh nicht vorhanden
24h	Aktualisieren relative Nummer des Datensatzes DS:DX--> eröffneten FCB	Offset 21h - 24h gesetzt
25h	Setzen Interruptvektor AL=Nummer des Interruptvektors DS:DX--> Interruptroutine	----

*** Anlage 3: Die DCS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
24h	Erstellen neuer PSP DX=Segmentadresse	----
27h	Wahlfreies Blocklesen CX=Anzahl zu lesender Datensätze DS:DX---> eröffneten FCB	AL=00h erfolgreich AL=01h EOF, keine Daten gelesen =02h DTA zu klein =03h EOF, Teilsatz gelesen CX=Anzahl gelesener Datensätze
28h	Wahlfreies Blockschreiben CX=Anzahl zu schreibender Datensätze DS:DX---> eröffneten FCB	AL=00h erfolgreich AL=01h Datenträger voll =02h DTA zu klein CX=Anzahl geschriebener Daten- sätze
29h	Dateibezeichnung analysieren AL=Analysenodus DS:SI---> Befehlszeile ES:DI---> nicht eröffneten FCB	AL=00h keine globalen Datei- bezeichner AL=01h globale Dateibezeichner AL=FFh ungültiges Laufwerk DS:DI---> 1. Zeichen nach analy- sierter Dateibezeichnung ES:DI---> generierten nicht eröffneten FCB
2Ah	Datum abfragen ----	AL=Wochentag (0=So,...,6=Sa) CX=Jahr (1980-2099) DH=Monat (1-12) DL=Tag (1-31)
2Bh	Datum setzen CX=Jahr (1980-2099) DH=Monat (1-12) DL=Tag (1-31)	AL=00h erfolgreich AL=FFh ungültiges Datum
2Ch	Zeit abfragen ----	CH=Stunden (0-23) CL=Minuten (0-59) DH=Sekunden (0-59) DL=Hundertstelsekunden (0-99)
2Dh	Zeit setzen CH=Stunden (0-23) CL=Minuten (0-59) DH=Sekunden (0-59) DL=Hundertstelsekunden (0-99)	AL=00h erfolgreich AL=FFh ungültige Zeit
2Eh	Prüfllesen setzen/rücksetzen AL=00h Prüfllesen AUS =01h Prüfllesen EIN	----

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
2Fh	DTA holen -----	ES:BX=--> DTA
30h	DCP-Versionsnummer abfragen -----	AL=oberer Teil Versionsnummer AH=unterer Teil Versionsnummer BX=unbestimmt CX=unbestimmt
31h	Prozess beenden und resident bleiben AL=Rückgabekode DX=Speichergrösse in Paragraphen -----	-----
33h	Prüfung auf CTRL-C AL=00h Abfrage aktueller Status AL=01h Verändern Status DL=00h Verändern auf AUS =01h Verändern auf EIN -----	DL=aktueller Status mit 00h=AUS und 01h=EIN AL=FFh ungdltiger Funktions- kode in DL
35h	Interruptvektor abfragen AL=Nummer des Interruptvektors -----	ES:BX=--> Interruptroutine
36h	Organisation des Datenträgers abfragen DL=Laufwerknummer (0=Std-LW, 1=LW A, ...) -----	AX=FFFFh ungdltige Laufwerk- nummer in DL AX=sonst Anzahl Sektoren/Cluster BX=verfügbare freie Cluster CX=Bytes/Sektor DX=Gesamtzahl Cluster
3000h	Abfrage der aktuellen landesspezifischen Information DS:DX=--> Feld (34 Byte gross) -----	DS:DX=landesspezif. Information DX=Landeskennzahl
38h	Abfrage einer beliebigen landesspezifischen Information AL=Landeskennzahl (00h-FEh) AL=FFh, wenn Landeskennzahl > FFh BX=Landeskennzahl DS:DX=--> Feld (34 Byte gross) -----	CF=0 DS:DX=landesspezifische Information DX=Landeskennzahl CF=1 AX=Fehlerkode
38h	Setzen der landesspezifischen Information AL=Landeskennzahl (00h-FEh) AL=FFh, wenn Landeskennzahl > FFh BX=Landeskennzahl DX=FFFFh -----	CF=0 erfolgreich CF=1 AX=Fehlerkode
39h	Verzeichnis erstellen (MKDIR) DS:DX=--> ASCII-Zeichenfolge mit Laufwerk und Pfad -----	CF=0 erfolgreich CF=1 AX=Fehlerkode

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
3Ah	Verzeichnis löschen (RMDIR) DS:DX--> ASCIIIZ-Zeichenfolge mit Laufwerk und Pfad	CF=0 erfolgreich CF=1 AX=Fehlerkode
3Bh	Aktuelles Verzeichnis wechseln (CHDIR) DS:DX--> ASCIIIZ-Zeichenfolge mit Laufwerk und Pfad	CF=0 erfolgreich CF=1 AX=Fehlerkode
3Ch	Händler erstellen (CREAT) CX=Dateiattribut DS:DX--> ASCIIIZ-Dateispezifikation	CF=0 AX=Händlernummer CF=1 AX=Fehlerkode
3Dh	Händler eröffnen DS:DX--> ASCIIIZ-Dateispezifikation AL=Eröffnungsmodus	CF=0 AX=Händlernummer CF=1 AX=Fehlerkode
3Eh	Händler schliessen BX=Händlernummer	CF=0 erfolgreich CF=1 AX=Fehlerkode
3Fh	Lesen von Datei/Einheit BX=Händlernummer CX=Anzahl zu lesender Bytes DS:DX--> Puffer	CF=0 AX=Anzahl gelesener Bytes CF=1 AX=Fehlerkode
40h	Schreiben auf Datei/Einheit BX=Händlernummer CX=Anzahl zu schreibender Bytes DS:DX--> Puffer	CF=0 AX=Anzahl geschriebener Bytes CF=1 AX=Fehlerkode
41h	Löschen einer Datei aus angegebenem Verzeichnis (UNLINK) DS:DX--> ASCIIIZ-Dateispezifikation	CF=0 erfolgreich CF=1 AX=Fehlerkode
42h	Verschieben des Lese-/Schreibzeigers (LSEEK) AL=Modus (00h,01h,02h) BX=Händlernummer CX:DX=Offset der Verschiebung	CF=0 DX:AX=neue Zeigerstellung CF=1 AX=Fehlerkode
43h	Dateiattribut abfragen/verändern (CHMOD) AL=00h (Abfrage) =01h (Setzen) CX=Dateiattribut DS:DX--> ASCIIIZ-Dateispezifikation	CF=0 CX=neues Dateiattribut CF=1 AX=Fehlerkode
4400h	Abfrage von Händlerinformationen BX=Händlernummer	CF=0 DX=Händlerinformation CF=1 AX=Fehlerkode
4401h	Setzen von Händlerinformationen BX=Händlernummer DH=00h DL=Händlerinformation	CF=0 erfolgreich CF=1 AX=Fehlerkode

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
4402h	Lesen vom Zeichengerät BX=Händlernummer CX=Anzahl zu lesender Bytes DS:DX=--> Puffer	CF=0 AX=Anzahl gelesener Bytes CF=1 AX=Fehlerkode
4403h	Schreiben auf Zeichengerät BX=Händlernummer CX=Anzahl zu schreibender Bytes DS:DX=--> Puffer	CF=0 AX=Anzahl geschriebener Bytes CF=1 AX=Fehlerkode
4404h	Lesen vom Blockgerät BL=Laufwerknummer (0=Std-LW, 1=LW A, ...) CX= Anzahl zu lesender Bytes DS:DX=--> Puffer	CF=0 AX=Anzahl gelesener Bytes CF=1 AX=Fehlerkode
4405h	Schreiben auf Blockgerät BL=Laufwerknummer (0=Std-LW, 1=LW A, ...) CX= Anzahl zu schreibender Bytes DS:DX=--> Puffer	CF=0 AX=Anzahl geschriebener Bytes CF=1 AX=Fehlerkode
4406h	Eingabestatus abfragen BX=Händlernummer	CF=0 für Gerät: AL=00h nicht empfangsbereit AL=FFh empfangsbereit CF=0 für Datei: AL=00h EOF erreicht AL=FFh EOF nicht erreicht CF=1 AX=Fehlerkode
4407h	Ausgabestatus abfragen BX=Händlernummer	CF=0 für Gerät: AL=00h nicht sendebereit AL=FFh sendebereit CF=0 für Datei: AL=00h EOF erreicht AL=FFh EOF nicht erreicht CF=1 AX=Fehlerkode
4408h	Abfrage, ob ein einzelnes Blockgerät auswechselbar ist BL=Laufwerknummer (0=Std-LW, 1=LW A, ...)	CF=0 AX=00h austauschbar AX=01h nicht austauschbar CF=1 AX=Fehlerkode
4409h	Abfrage, ob ein logisches Gerät lokal oder entfernt ist BL=Laufwerknummer (0=Std-LW, 1=LW A, ...)	CF=0 Bit 12 von DX=0 lokal Bit 12 von DX=1 entfernt CF=1 AX=Fehlerkode

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
440Ah	Ist ein Händler lokal oder entfernt? BX=Händlernummer	CF=0 Bit 15 von DX=0 lokal Bit 15 von DX=1 entfernt CF=1 AX=Fehlercode
440Bh	Verändern des Zählers für Wiederholungsversuche bei gemeinsamer Dateinutzung CX=Dauer der Pause DX=Anzahl der Wiederholungen	CF=0 erfolgreich CF=1 AX=Fehlercode
440Ch/4Ah	Auswahl bzw. Aktivierung einer Zeichensatztabelle CL=4Ah CH=Gerätetreiberkennzeichen BX=Händlernummer des eröffneten Gerätes DS:DX=--> Parameterblock	CF=0 erfolgreich CF=1 AX=Fehlercode
440Ch/4Ch	Beginn der Vorbereitung von Zeichensatztabellen CL=4Ch CH=Gerätetreiberkennzeichen BX=Händlernummer des eröffneten Gerätes DS:DX=--> Parameterblock	CF=0 erfolgreich CF=1 AX=Fehlercode
440Ch/4Dh	Ende der Vorbereitung von Zeichensatztabellen CL=4Dh CH=Gerätetreiberkennzeichen BX=Händlernummer des eröffneten Gerätes DS:DX=--> Parameterblock	CF=0 erfolgreich CF=1 AX=Fehlercode
440Ch/6Ah	Anzeige der aktuellen Zeichensatztabelle CL=6Ah CH=Gerätetreiberkennzeichen BX=Händlernummer des eröffneten Gerätes DS:DX=--> Parameterblock	CF=0 erfolgreich CF=1 AX=Fehlercode
440Ch/6Bh	Anzeige der vorbereiteten Zeichensatztabellen CL=6Bh CH=Gerätetreiberkennzeichen BX=Händlernummer des eröffneten Gerätes DS:DX=--> Parameterblock	CF=0 erfolgreich CF=1 AX=Fehlercode
440Dh/40h	Setzen der Geräteparameter CL=40h CH=08h BL=Laufwerksnummer (0=Std-LW, 1=LW A, ...) DS:DX=--> Parameterblock	CF=0 erfolgreich

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
440Dh/41h	Schreiben einer Spur auf logisches Gerät CL=41h CH=08h BL=Laufwerknummer (0=Std-LW, 1=LW A, ...) DS:DX=--> Parameterblock	CF=0 erfolgreich
440Dh/42h	Formatieren und Überprüfen einer Spur auf logischem Gerät CL=42h CH=08h BL=Laufwerknummer (0=Std-LW, 1=LW A, ...) DS:DX=--> Parameterblock	CF=0 Offset 00h von DS:DX gesetzt
440Dh/60h	Abfrage der Geräteparameter CL=60h CH=08h BL=Laufwerknummer (0=Std-LW, 1=LW A, ...) DS:DX=--> Parameterblock	CF=0 DS:DX ist gefüllt
440Dh/61h	Lesen einer Spur vom logischen Gerät CL=61h CH=08h BL=Laufwerknummer (0=Std-LW, 1=LW A, ...) DS:DX=--> Parameterblock	CF=0 erfolgreich
440Dh/62h	Überprüfen einer Spur auf logischem Gerät CL=62h CH=08h BL=Laufwerknummer (0=Std-LW, 1=LW A, ...) DS:DX=--> Parameterblock	CF=0 erfolgreich
440Eh	Abfrage des logischen Laufwerkes BL=Laufwerknummer (0=Std-LW, 1=LW A, ...)	CF=0 AL=00h nur 1 logisches LW AL=Laufwerknummer (1=LW A, 2=LW B, ...) CF=1 AX=Fehlerkode
440Fh	Setzen des logischen Laufwerkes BL=Laufwerknummer (1=LW A, 2=LW B, ...)	CF=0 AL=00h nur 1 logisches LW AL=Laufwerknummer (1=LW A, 2=LW B, ...) CF=1 AX=Fehlerkode
45h	Händler duplizieren (DUP) BX=Händlernummer	CF=0 AX=neue Händlernummer (C=0) CF=1 AX=Fehlerkode

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
46h	Händlerduplikat erstellen (FORCDUP) BX=existierende Händlernummer CX=zweite Händlernummer	CF=0 erfolgreich CF=1 AX=Fehlercode
47h	Aktuelles Verzeichnis holen DL=Laufwerknummer (0=Std-LW, 1=LW A, ...) DS:SI=--> Feld (64 Byte gross)	CF=0 DS:SI=vollständiger ASCIIIZ-Pfad CF=1 AX=Fehlercode
48h	Speicher zuweisen BX=Anzahl Paragraphen	CF=0 AX:0=--> reservierten Systemspeicher CF=1 AX=Fehlercode BX=grösster verfügbarer Speicherbereich (in Paragraphen)
49h	Zugewiesenen Speicher freigeben ES=Segment des freizugebenden Speichers	CF=0 erfolgreich CF=1 AX=Fehlercode
4Ah	Zugewiesenen Speicherblock verändern (SETBLOCK) BX=Anzahl Paragraphen ES=Segmentadresse des zu modi- fizierenden Speicherbereiches	CF=0 erfolgreich CF=1 AX=Fehlercode BX=max. Anzahl verfügbarer Paragraphen
4B00h	Programme laden und ausführen (EXEC) DS:DX=--> ASCIIIZ-Dateispezifikation ES:BX=--> Parameterblock	CF=0 erfolgreich CF=1 AX=Fehlercode
4B03h	Programme laden DS:DX=--> ASCIIIZ-Dateispezifikation ES:BX=--> Parameterblock	CF=0 erfolgreich CF=1 AX=Fehlercode
4Ch	Prozess beenden (EXIT) AL=Rückgabekode	----
4Dh	Beendigungskode liefern (WAIT) ----	AL=Rückgabekode AH=Art und Weise der Beendigung
4Eh	Erste Dateieintragung finden (FIND FIRST) CX=Dateiattribut DS:DX=--> ASCIIIZ-Dateispezifikation	CF=0 aktueller DTA gefüllt CF=1 AX=Fehlercode
4Fh	Nächste Dateieintragung finden (FIND NEXT) DTA=Information von vorherigen Finden (4Eh bzw. 4Fh)	CF=0 aktueller DTA gefüllt CF=1 AX=Fehlercode
54h	Status Prüfllesen abfragen ----	AL=00h Prüfllesen AUS =01h Prüfllesen EIN

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
56h	Datei umbenennen DS:DX--> Quell-ASCIIIZ-Dateispez. ES:DI--> Ziel-ASCIIIZ-Dateispez.	CF=0 erfolgreich CF=1 AX=Fehlerkode
5700h	Datum und Uhrzeit einer Datei abfragen BX=Händlernummer	CF=0 CX=Zeit DX=Datum CF=1 AX=Fehlerkode
5701h	Datum und Uhrzeit einer Datei setzen BX=Händlernummer CX=Zeit DX=Datum	CF=0 erfolgreich CF=1 AX=Fehlerkode
59h	Erweiterte Fehlermeldung BX=0000h	AX=Fehlerkode BH=Fehlerklasse BL=Nutzerhinweise CH=Fehlerort
5Ah	Einzigartige Datei erstellen CX=Dateiattribut DS:DX--> ASCIIIZ-Pfad	CF=0 AX=Händlernummer DS:DX=ASCIIIZ-Dateispez. CF=1 AX=Fehlerkode
5Bh	Neuen Händler erstellen CX=Dateiattribut DS:DX--> ASCIIIZ-Dateispezifikation	CF=0 AX=Händlernummer CF=1 AX=Fehlerkode
5Ch	Dateizugriff verbieten/erlauben AL=00h (verbieten) =01h (erlauben) BX=Händlernummer CX=Offset des Bereiches (höherwertig) DX=Offset des Bereiches (niederwertig) SI=L(nge des Bereiches (höherwertig) DI=L(nge des Bereiches (niederwertig)	CF=0 erfolgreich CF=1 AX=Fehlerkode
5E00h	Maschinenname abfragen DS:DX--> Feld (16 Byte gross)	CF=0 CX=Identifikationsnummer der lokalen Maschine DS:DX=ASCIIIZ-Maschinenname CH=00h kein Maschinenname CL=unbestimmt CF=1 AX=Fehlerkode
5E01h	Maschinenname setzen CX=Identifikationsnummer der lokalen Maschine DS:DX--> ASCIIIZ-Maschinennamen	CF=0 erfolgreich CF=1 AX=Fehlerkode

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
5E02h	Drucker-Setup setzen BX=Index der Zuweisungstabelle CX=Länge der Initialisierungs- zeichenfolge (max. 64 Byte) DS:SI=--> Initialisierungszeichen- folge	CF=0 erfolgreich CF=1 AX=Fehlercode
5E03h	Drucker-Setup abfragen BX=Index der Zuweisungstabelle ES:DI=--> Feld (64 Byte gross)	CF=0 CX=Länge der Initiali- sierungszeichenfolge ES:DI=gefüllt CF=1 AX=Fehlercode
5F02h	Eintragungen der Zuweisungstabelle abfragen BX=Index der Zuweisungstabelle DS:SI=--> Feld (128 Byte gross) ES:DI=--> Feld (128 Byte gross)	CF=0 BH=Status-Kennzeichen des Gerätes BL=Typ des Gerätes CX=Nutzerkennzeichen DS:SI=ASCIIIZ-Gerätename ES:DI=ASCIIIZ-Netzwerkname CF=1 AX=Fehlercode
5F03h	Zuweisung eines Gerätes BL=Typ des Gerätes CX=Nutzerkennzeichen DS:SI=--> ASCIIIZ-Quellgerät ES:DI=--> ASCIIIZ-Zielgerät	CF=0 erfolgreich CF=1 AX=Fehlercode
5F04h	Zuweisung aufheben DS:SI=--> ASCIIIZ-Quellgerät	CF=0 erfolgreich CF=1 AX=Fehlercode
62h	Ermitteln aktueller PSP ----	BX=Segmentadresse des aktuellen Prozesses
65h	Ermittlung einer erweiterten Landesinformation AL=Informationskennzeichen BX=Zeichensatztablelle CX=Anzahl der zurückzugebenden Bytes DX=Landeskod ES:DI=-->Feld	CF=0 ES:DI ist gefüllt CF=1 AX=Fehlercode
6601h	Ermitteln der Zeichensatztablelle ----	CF=0 BX=aktive Zeichensatztablelle DX=System-Zeichensatztablelle CF=1 AX=Fehlercode
6602h	Setzen der Zeichensatztablelle BX=aktive Zeichensatztablelle	CF=0 erfolgreich CF=1 AX=Fehlercode

*** Anlage 3: Die DOS-Funktionen im Überblick ***

Nr.	Aufruf	Rückkehr
67h	Setzen der Händleranzahl BX=Anzahl der gleichzeitig eröffneten Händler	CF=0 erfolgreich CF=1 AX=Fehlerkode
68h	Übergabe an Händler BX=Händlernummer	CF=0 erfolgreich CF=1 AX=Fehlerkode

--> bedeutet: Zeiger auf

EKZ: Endkennzeichen

*** Anlage 4: Speicherbelegung ***

Anlage 4: Speicherbelegung

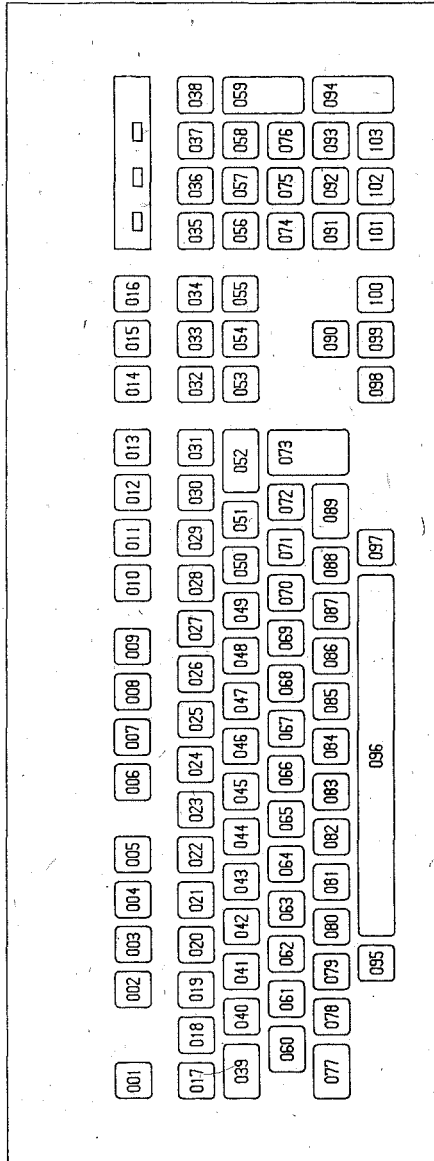
Adresse (hexad.)		INT- Nr.	Bedeutung
00000	Software-Interrupts:	00h	Division durch Null
00004		01h	Einzelsschritt
00008		02h	NMI
0000C		03h	Prüfpunkt
00010		04h	Überlaufunterbrechung
00014		05h	Bildschirminhalt drucken
00018		06h	reserviert
0001D		07h	reserviert
00020		08h	Zeitgeber
00024		09h	Tastatur
00028		0Ah	reserviert
0002C		0Bh	Kommunikation
00030		0Ch	Kommunikation
00034		0Dh	Harddisk
00038		0Eh	Diskette
0003C		0Fh	Drucker
00040		10h	Bildschirm(Videosteuerung)
00044		11h	Schalterbelegung
00048		12h	Speichergrösse
0004C		13h	Diskette/Harddisk
00050		14h	Kommunikation (serieller Anschluss)
00054		15h	Verwaltungsfunktionen
00058		16h	Tastatur
0005C		17h	Drucker(parallele Schnittstelle)
00060		18h	reserviert
00064		19h	Urladen
00068		1Ah	Tageszeit
0006C		1Bh	Tastatur-Interrupt
00070		1Ch	Zeitgeber Nutzer
00074		1Dh	Adresse Bildschirmparameter
00078		1Eh	Adresse Diskettenparameter-Block
0007C		1Fh	Adresse grafischer Zeichensatz
00080	DOS-Interrupts:	20h	Programm beenden
00084		21h	Aufruf DOS-Funktionen
00088		22h	Abschluss-Adresse
0008C		23h	Aussprung-Adresse bei CTRL-C
00090		24h	Vektor der Behandlungsroutine für kritische Fehler
00094		25h	Absolutes Diskettenlesen
00098		26h	Absolutes Diskettenschreiben
0009C		27h	Beendigung festspeichern
000A0			für DOS reserviert
000FF			
00100			reserviert
0017F			

*** Anlage 4: Speicherbelegung ***

Adresse (hexad.)	Bedeutung	
00180	Benutzersoftware-Interrupts	
0019F		
001A0	nicht verwendet	
001FF		
00200	reserviert	
00217		
00218	reserviert	
003C3		
003C4	Stapel bei Einschaltinitialisierung und Umladen	
003FF		
00400	BIOS-Datenbereich	Basisadressen für RS232
00407		
0048F	reserviert	
00490		
004EF	reserviert	
004F0		
004FF	anwendungsinterner Übertragungsbereich	
00500	Statuskennzeichen Bildschirm drücken	
00504	DOS-Statusbyte für Einzel-Laufwerk	
00510	reserviert	
00511	reserviert	
00512	reserviert	
00515	Takt-Interruptvektorsegment:	
00516	Relativzeiger speichern	
00519	reserviert	
0051A	reserviert	
0051D	reserviert	
005FF	reserviert	
00700	BIOS, DOS, Treiber, Puffer, ...	
00700	COMMAND	
RAM-Ende	Anwenderprogramm	
C8000	Harddisk-Adapter	
F0000	ROM-Bereich	
FE000	ROM-BIOS-Programmbereich	

Anlage 5: Tastatur-Übersicht

05.1.1. Tastaturpositionen



*** Anlage 5: Tastatur-Übersicht ***

05.2.2.1_Kodes_(INI16_00h)

T-Nr	SCAN	Gravur deutsch	shift		CAPS LOCK		ALT		CTRL		ALT +	
			AH AL	AH AL	AH AL	AH AL	AH AL	AH AL	AH AL	AH AL		
17 01 ESC 01 1B 01 1B 01 1B 01 1B -- -- 01 1B -- --												
18 02 1 ! 02 31 02 21 02 31 02 21 78 00 -- -- 78 00												
19 03 2 " 03 32 03 22 03 32 03 22 79 00 03 00 03 FD												
20 04 3 @ 04 33 04 15 04 33 04 15 7A 00 -- -- 7A 00												
21 05 4 # 05 34 05 24 05 34 05 24 7B 00 -- -- 7B 00												
22 06 5 % 06 35 06 25 06 35 06 25 7C 00 -- -- 7C 00												
23 07 6 & 07 36 07 26 07 36 07 26 7D 00 07 1E 7D 00												
24 08 7 / 08 37 08 2F 08 37 08 2F 7E 00 -- -- 08 7B												
25 09 8 (09 38 09 28 09 38 09 28 7F 00 -- -- 09 5B												
26 0A 9) 0A 39 0A 29 0A 39 0A 29 80 00 -- -- 0A 5D												
27 0B 0 = 0B 30 0B 3D 0B 30 0B 3D 81 00 -- -- 0B 7D												
28 0C - ? 0C E1 0C 3F 0C E1 0C 3F 82 00 -- -- 0C 5C												
29 0D _ ` 0D 27 0D 60 0D 27 0D 60 83 00 -- -- 83 00												
30 29 ^ 29 5E 29 F8 29 5E 29 F8 -- -- -- -- -- --												
31 0E < --- 0E 08 0E 08 0E 08 0E 08 -- -- 0E 7F -- --												
52 0F > --- 0F 09 0F 00 0F 09 0F 00 -- -- -- -- -- --												
40 10 0 10 71 10 51 10 51 10 71 10 00 10 11 10 40												
41 11 W 11 77 11 57 11 57 11 77 11 00 11 17 11 00												
42 12 E 12 65 12 45 12 45 12 65 12 00 12 05 12 00												
43 13 R 13 72 13 52 13 52 13 72 13 00 13 12 13 00												
44 14 T 14 74 14 54 14 54 14 74 14 00 14 14 14 00												
45 15 Z 15 7A 15 5A 15 5A 15 7A 1C 00 15 1A 2C 00												
46 16 U 16 75 16 55 16 55 16 75 16 00 16 15 16 00												
47 17 I 17 69 17 49 17 49 17 69 17 00 17 09 17 00												
48 18 O 18 6F 18 4F 18 4F 18 6F 18 00 18 0F 18 00												
49 19 P 19 70 19 50 19 50 19 70 19 00 19 10 19 00												
50 1A J 1A 81 1A 9A 1A 9A 1A 81 -- -- 1A 1B -- --												
51 1B + * 1B 2B 1B 2A 1B 2B 1B 2A -- -- 1B 1D 1B 7E												
73 1C enter 1C 0D 1C 0D 1C 0D 1C 0D -- -- 1C 0A -- --												
39 1D CTRL -- -- -- -- -- -- -- -- -- -- -- -- -- --												
61 1E A 1E 61 1E 41 1E 41 1E 61 1E 00 1E 01 1E 00												
62 1F S 1F 73 1F 53 1F 53 1F 73 1F 00 1F 13 1F 00												
63 20 D 20 64 20 44 20 44 20 64 20 00 20 04 20 00												
64 21 F 21 66 21 46 21 46 21 66 21 00 21 06 21 00												
65 22 G 22 67 22 47 22 47 22 67 22 00 22 07 22 00												
66 23 H 23 68 23 48 23 48 23 68 23 00 23 08 23 00												
67 24 J 24 6A 24 4A 24 4A 24 6A 24 00 24 0A 24 00												
68 25 K 25 6B 25 4B 25 4B 25 6B 25 00 25 0B 25 00												
69 26 L 26 6C 26 4C 26 4C 26 6C 26 00 26 0C 26 00												
70 27 \ 27 94 27 99 27 99 27 94 -- -- -- -- -- --												
71 28 I 28 84 28 8E 28 8E 28 84 -- -- -- -- -- --												
72 2B # 2B 23 2B 27 2B 23 2B 27 -- -- 2B 1C -- --												
77 2A sh. left -- -- -- -- -- -- -- -- -- -- -- -- -- --												
78 56 < > 56 3C 56 3E 56 3C 56 3E -- -- -- -- 56 7C												
79 2C Y 2C 79 2C 59 2C 59 2C 79 15 00 2C 19 15 00												

Die Umschalttaste NUM LOCK hat keinen Einfluss auf die obigen Tasten.

*** Anlage 5: Tastatur-Übersicht ***

T-Nr	SCAN	Gravur deutsch	shift		CAPS LOCK		ALT		CTRL		ALT +
			AH	AL	AH	AL	AH	AL	AH	AL	AH
80	2D	X	2D 78	2D 58	2D 58	2D 78	2D 00	2D 18	2D 00		
81	2E	C	2E 63	2E 43	2E 43	2E 63	2E 00	2E 03	2E 00		
82	2F	V	2F 76	2F 56	2F 56	2F 76	2F 00	2F 16	2F 00		
83	30	B	30 62	30 42	30 42	30 62	30 00	30 02	30 00		
84	31	N	31 6E	31 4E	31 4E	31 6E	31 00	31 0E	31 00		
85	32	M	32 6D	32 4D	32 4D	32 6D	32 00	32 0D	32 E6		
86	33	,	33 2C	33 3B	33 2C	33 3B	---	---	---		
87	34	;	34 2E	34 3A	34 2E	34 3A	---	---	---		
88	35	-	35 2D	35 5F	35 2D	35 5F	---	35 1F	---		
89	36	sh, right	---	---	---	---	---	---	---		
95	38	ALT	---	---	---	---	---	---	---		
96	39	Space	39 20	39 20	39 20	39 20	39 20	39 20	---		
60	3A	CAPS LOCK	---	---	---	---	---	---	---		
97	E0 38	ALT 1	---	---	---	---	---	---	---		

Die Umschalttaste NUM LOCK hat keinen Einfluss auf die obigen Tasten.

T-Nr	SCAN	Gravur deutsch	shift		ALT		CTRL	
			AH	AL	AH	AL	AH	AL
2	3B	F1	3B 00	54 00	68 00	5E 00		
3	3C	F2	3C 00	55 00	69 00	5F 00		
4	3D	F3	3D 00	56 00	6A 00	60 00		
5	3E	F4	3E 00	57 00	6B 00	61 00		
6	3F	F5	3F 00	58 00	6C 00	62 00		
7	40	F6	40 00	59 00	6D 00	63 00		
8	41	F7	41 00	5A 00	6E 00	64 00		
9	42	F8	42 00	5B 00	6F 00	65 00		
10	43	F9	43 00	5C 00	70 00	66 00		
11	44	F10	44 00	5D 00	71 00	67 00		
12	57	F11	---	---	---	---		
13	58	F12	---	---	---	---		

Die Umschalttasten CAPS LOCK und NUM LOCK haben keinen Einfluss auf die obigen Tasten.

*** Anlage 5: Tastatur-Übersicht ***

AS.2.3. Codes (INT16_10h)

IT-Nr	SCAN	Bravur deutsch	shift		CAPS LOCK		ALT		CTRL		ALT +	
			AH	AL	AH	AL	AH	AL	AH	AL	AH	AL
			AL	AL	AL	AL	AL	AL	AL	AL	AL	AL
17	01	ESC	01 1B	01 1B	01 1B	01 1B	01 00	01 1B	01 00	01 00	01 00	01 00
18	02	1	02 31	02 21	02 31	02 21	78 00	-- --	-- --	78 00	-- --	78 00
19	03	2	03 32	03 22	03 32	03 22	79 00	03 00	03 00	03 FD	-- --	03 FD
20	04	3	04 33	04 15	04 33	04 15	7A 00	-- --	-- --	7A 00	-- --	7A 00
21	05	4	05 34	05 24	05 34	05 24	7B 00	-- --	-- --	7B 00	-- --	7B 00
22	06	5	06 35	06 25	06 35	06 25	7C 00	-- --	-- --	7C 00	-- --	7C 00
23	07	6	07 36	07 26	07 36	07 26	7D 00	07 1E	07 00	7D 00	-- --	7D 00
24	08	7	08 37	08 2F	08 37	08 2F	7E 00	-- --	-- --	08 7B	-- --	08 7B
25	09	8	09 38	09 28	09 38	09 28	7F 00	-- --	-- --	09 5B	-- --	09 5B
26	0A	9	0A 39	0A 29	0A 39	0A 29	80 00	-- --	-- --	0A 5D	-- --	0A 5D
27	0B	0	0B 30	0B 3D	0B 30	0B 3D	81 00	-- --	-- --	0B 7D	-- --	0B 7D
28	0C	-	0C E1	0C 3F	0C E1	0C 3F	82 00	-- --	-- --	0C 5C	-- --	0C 5C
29	0D		0D 27	0D 60	0D 27	0D 60	83 00	-- --	-- --	83 00	-- --	83 00
30	29	^	29 5E	29 F8	29 5E	29 F8	29 00	-- --	-- --	29 00	-- --	29 00
31	0E	{	0E 08	0E 08	0E 08	0E 08	0E 08	0E 7F	0E 00	0E 00	-- --	0E 00
52	0F	<-->	0F 09	0F 00	0F 09	0F 00	A5 00	-- --	-- --	0F 00	-- --	0F 00
40	10	Q	10 71	10 51	10 51	10 71	10 00	10 11	10 11	10 40	-- --	10 40
41	11	W	11 77	11 57	11 57	11 77	11 00	11 17	11 17	11 00	-- --	11 00
42	12	E	12 65	12 45	12 45	12 65	12 00	12 05	12 05	12 00	-- --	12 00
43	13	R	13 72	13 52	13 52	13 72	13 00	13 12	13 12	13 00	-- --	13 00
44	14	T	14 74	14 54	14 54	14 74	14 00	14 14	14 14	14 00	-- --	14 00
45	15	Z	15 7A	15 5A	15 5A	15 7A	2C 00	15 1A	15 1A	2C 00	-- --	2C 00
46	16	U	16 75	16 55	16 55	16 75	16 00	16 15	16 15	16 00	-- --	16 00
47	17	I	17 6F	17 4F	17 4F	17 6F	17 00	17 09	17 09	17 00	-- --	17 00
48	18	O	18 6F	18 4F	18 4F	18 6F	18 00	18 0F	18 0F	18 00	-- --	18 00
49	19	P	19 70	19 50	19 50	19 70	19 00	19 10	19 10	19 00	-- --	19 00
50	1A	J	1A 81	1A 9A	1A 9A	1A 81	1A 00	1A 1B	1A 1B	1A 00	-- --	1A 00
51	1B	+ *	1B 2B	1B 2A	1B 2B	1B 2A	1B 00	1B 1D	1B 1D	1B 7E	-- --	1B 7E
73	1C	enter	1C 0D	1C 0D	1C 0D	1C 0D	1C 00	1C 0A	1C 0A	1C 00	-- --	1C 00
39	1D	CTRL	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --
61	1E	A	1E 61	1E 41	1E 41	1E 61	1E 00	1E 01	1E 01	1E 00	-- --	1E 00
62	1F	S	1F 73	1F 53	1F 53	1F 73	1F 00	1F 13	1F 13	1F 00	-- --	1F 00
63	20	D	20 64	20 44	20 44	20 64	20 00	20 04	20 04	20 00	-- --	20 00
64	21	F	21 66	21 46	21 46	21 66	21 00	21 06	21 06	21 00	-- --	21 00
65	22	G	22 67	22 47	22 47	22 67	22 00	22 07	22 07	22 00	-- --	22 00
66	23	H	23 68	23 48	23 48	23 68	23 00	23 08	23 08	23 00	-- --	23 00
67	24	J	24 6A	24 4A	24 4A	24 6A	24 00	24 0A	24 0A	24 00	-- --	24 00
68	25	K	25 6B	25 4B	25 4B	25 6B	25 00	25 0B	25 0B	25 00	-- --	25 00
69	26	L	26 6C	26 4C	26 4C	26 6C	26 00	26 0C	26 0C	26 00	-- --	26 00
70	27	\	27 94	27 99	27 99	27 94	27 00	-- --	-- --	27 00	-- --	27 00
71	28	[28 84	28 8E	28 8E	28 84	28 00	-- --	-- --	28 00	-- --	28 00
72	28	#	28 23	28 27	28 23	28 27	28 00	28 1C	28 1C	28 00	-- --	28 00
77	2A	sh. left	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --
78	56	< >	56 3C	56 3E	56 3C	56 3E	-- --	-- --	-- --	56 7C	-- --	56 7C
79	2C	Y	2C 79	2C 59	2C 59	2C 79	15 00	2C 19	2C 19	15 00	-- --	15 00

Die Umschalttaste NUM LOCK hat keinen Einfluss auf die obigen Tasten.

*** Anlage 5: Tastatur-Übersicht ***

IT-Nr	SCAN	Gravur deutsch	shift				CAPS LOCK				ALT		CTRL		ALT +
			AH AL		AH AL		AH AL		AH AL		AH AL	AH AL	AH AL	CTRL	
			AH	AL	AH	AL	AH	AL	AH	AL	AH	AL	AH	AL	AH
80	2D	X	2D 78	2D 58	2D 58	2D 78	2D 00	2D 18	2D 00						
81	2E	C	2E 63	2E 43	2E 43	2E 63	2E 00	2E 03	2E 00						
82	2F	V	2F 76	2F 56	2F 56	2F 76	2F 00	2F 16	2F 00						
83	30	B	30 62	30 42	30 42	30 62	30 00	30 02	30 00						
84	31	N	31 6E	31 4E	31 4E	31 6E	31 00	31 0E	31 00						
85	32	M	32 6D	32 4D	32 4D	32 6D	32 00	32 0D	32 00						
86	33	, ;	33 2C	33 3B	33 2C	33 3B	33 00	-- --	33 00						
87	34	. :	34 2E	34 3A	34 2E	34 3A	34 00	-- --	34 00						
88	35	-	35 2D	35 5F	35 2D	35 5F	35 00	35 1F	35 00						
89	36	sh. right	-- --	-- --	-- --	-- --	-- --	-- --	-- --						
95	38	ALT	-- --	-- --	-- --	-- --	-- --	-- --	-- --						
96	39	Space	39 20	39 20	39 20	39 20	39 20	39 20	39 00						
60	3A	CAPS LOCK	-- --	-- --	-- --	-- --	-- --	-- --	-- --						
97	E0 38	ALT 1	-- --	-- --	-- --	-- --	-- --	-- --	-- --						

Die Umschalttaste NUM LOCK hat keinen Einfluss auf die obigen Tasten.

IT-Nr	SCAN	Gravur deutsch	shift		ALT		CTRL	
			AH	AL	AH	AL	AH	AL
			AH	AL	AH	AL	AH	AL
2	3B	F1	3B 00	54 00	68 00	5E 00		
3	3C	F2	3C 00	55 00	69 00	5F 00		
4	3D	F3	3D 00	56 00	6A 00	60 00		
5	3E	F4	3E 00	57 00	6B 00	61 00		
6	3F	F5	3F 00	58 00	6C 00	62 00		
7	40	F6	40 00	59 00	6D 00	63 00		
8	41	F7	41 00	5A 00	6E 00	64 00		
9	42	F8	42 00	5B 00	6F 00	65 00		
10	43	F9	43 00	5C 00	70 00	66 00		
11	44	F10	44 00	5D 00	71 00	67 00		
12	57	F11	85 00	87 00	8B 00	89 00		
13	58	F12	86 00	88 00	8C 00	8A 00		

* Die Umschalttasten CAPS LOCK und NUM LOCK haben keinen Einfluss auf die obigen Tasten.

*** Anlage 5: Tastatur-Übersicht ***

T-Nr	SCAN	Gravur deutsch	NUM LOCK				ALT
			shift		shift		
			AH AL	AH AL	AH AL	AH AL	
35	45	NUM LOCK	-- --	-- --	-- --	-- --	-- --
36	E0 35	↑	E0 2F	E0 2F	E0 2F	E0 2F	A4 00
37	37	*	37 2A	37 2A	37 2A	37 2A	37 00
56	47	7	47 00	47 37	47 37	47 00	-- --
57	48	8	48 00	48 38	48 38	48 00	-- --
58	49	9	49 00	49 39	49 39	49 00	-- --
38	4A	-	4A 2D	4A 2D	4A 2D	4A 2D	4A 00
74	4B	4	4B 00	4B 34	4B 34	4B 00	-- --
75	4C	5	4C 00	4C 35	4C 35	4C 00	-- --
76	4D	6	4D 00	4D 36	4D 36	4D 00	-- --
59	4E	+	4E 2B	4E 2B	4E 2B	4E 2B	4E 00
91	4F	1	4F 00	4F 31	4F 31	4F 00	-- --
92	50	2	50 00	50 32	50 32	50 00	-- --
93	51	3	51 00	51 33	51 33	51 00	-- --
101	52	0	52 00	52 30	52 30	52 00	-- --
102	1)	00	2)	2)	2)	2)	2)
103	53	,	53 00	53 2C	53 2C	53 00	-- --
94	E0 1C	ENTER	E0 0D	E0 0D	E0 0D	E0 0D	A6 00

Bei Umschalttaste CAPS LOCK hat keinen Einfluss auf die obigen Tasten.

T-Nr	SCAN	Gravur deutsch	ALT				CTRL
			shift		shift		
			AH AL	AH AL	AH AL	AH AL	
14	E0 37	PRT SC	-- --	-- --	-- --	72 00	
15	46	SROLL	-- --	-- --	-- --	-- --	
16	3)	PAUSE	-- --	-- --	-- --	00 00	
32	E0 52	INS	52 E0	52 E0	A2 00	92 E0	
53	E0 53	DEL	53 E0	53 E0	A3 00	93 E0	
33	E0 47	HOME	47 E0	47 E0	97 00	77 E0	
54	E0 4F	END	4F E0	4F E0	9F 00	7F E0	
34	E0 49	PG UP	49 E0	49 E0	99 00	84 E0	
55	E0 51	PG DN	51 E0	51 E0	A1 00	76 E0	
98	E0 4B	<--	4B E0	4B E0	9B 00	73 E0	
100	E0 4D	-->	4D E0	4D E0	9D 00	74 E0	
90	E0 48	↑	48 E0	48 E0	98 00	8D E0	
99	E0 50	↓	50 E0	50 E0	A0 00	91 E0	
1	54	SYS	-- --	-- --	-- --	-- --	

Die Umschalttasten CAPS LOCK bzw. NUM LOCK haben keinen Einfluss auf die obigen Tasten.

PAUSE: 1) E1 52 E1 52 Doppel-Null (00): 3) E1 1D 45 E1 9D C5
2) 52 30 52 30 E0 46 E0 C6 mit SROLL LOCK

-- = Taste liefert keinen Kode

T-Nr = Tasten-Nummer entsprechend Anlage 6.

*** Anlage 6: Kode - Tabelle ***

Anlage 6: Kode - Tabelle

A 6.1: Kode - Tabelle 650

HEX	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0		▶	0	0	P	"	p	Q	Q	Q	Q	Q	Q	Q	Q	Q
1	◻	▶	!	1	A	Q	a	q	ü	ä	í	◻	◻	◻	◻	◻
2	◻	▶	"	2	B	R	b	r	é	ó	◻	◻	◻	◻	◻	◻
3	◻	▶	#	3	C	S	c	s	á	ó	◻	◻	◻	◻	◻	◻
4	◻	▶	\$	4	D	T	t	ä	ü	◻	◻	◻	◻	◻	◻	◻
5	◻	▶	%	5	E	U	u	á	ó	◻	◻	◻	◻	◻	◻	◻
6	◻	▶	&	6	F	V	v	á	ó	◻	◻	◻	◻	◻	◻	◻
7	◻	▶	'	7	G	W	w	á	ó	◻	◻	◻	◻	◻	◻	◻
8	◻	▶	(8	H	X	x	é	ó	◻	◻	◻	◻	◻	◻	◻
9	◻	▶)	9	I	Y	y	é	ó	◻	◻	◻	◻	◻	◻	◻
a	◻	▶	*	:	J	Z	j	é	ó	◻	◻	◻	◻	◻	◻	◻
b	◻	▶	+	:	K	L	k	í	◻	◻	◻	◻	◻	◻	◻	◻
c	◻	▶	,	<	L	/	/	/	/	/	/	/	/	/	/	/
d	◻	▶	-	=	M	J	m	y	í	ó	í	é	=	í	ó	2
e	◻	▶	.	>	N	^	n	ä	x	◻	◻	◻	◻	◻	◻	◻
f	◻	▶	/	?	0	_	o	o	ä	f	◻	◻	◻	◻	◻	◻

A 6.2: Kode - Tabelle 437

IRRX	0	1	2	3	4	5	6	7	8	9	n	l	c	d	e	f
0	▶	◀	0	P	'	p	Q	é	á	▨	▩	▪	▫	▬	▭	▮
1	◀	▶	1	A	Q	a	q	ü	ä	▨	▩	▪	▫	▬	▭	▮
2	◀	▶	2	B	R	b	r	é	á	▨	▩	▪	▫	▬	▭	▮
3	◀	▶	3	C	S	c	s	á	ó	▨	▩	▪	▫	▬	▭	▮
4	◀	▶	4	D	T	d	t	ü	ö	▨	▩	▪	▫	▬	▭	▮
5	◀	▶	5	E	U	e	u	á	ó	▨	▩	▪	▫	▬	▭	▮
6	◀	▶	6	F	V	f	v	á	ó	▨	▩	▪	▫	▬	▭	▮
7	◀	▶	7	G	W	g	w	á	ó	▨	▩	▪	▫	▬	▭	▮
8	◀	▶	8	H	X	h	x	á	ó	▨	▩	▪	▫	▬	▭	▮
9	◀	▶	9	I	Y	i	y	á	ó	▨	▩	▪	▫	▬	▭	▮
0	◀	▶	:	J	Z	j	z	á	ó	▨	▩	▪	▫	▬	▭	▮
1	◀	▶	:	K	C	k	c	á	ó	▨	▩	▪	▫	▬	▭	▮
2	◀	▶	:	L	V	l	v	á	ó	▨	▩	▪	▫	▬	▭	▮
3	◀	▶	:	M	N	m	n	á	ó	▨	▩	▪	▫	▬	▭	▮
4	◀	▶	:	O	W	o	w	á	ó	▨	▩	▪	▫	▬	▭	▮
5	◀	▶	:	P	R	p	r	á	ó	▨	▩	▪	▫	▬	▭	▮
6	◀	▶	:	Q	A	q	a	á	ó	▨	▩	▪	▫	▬	▭	▮

*** Anlage 7: Umrechnungs-Tabellè ***

Anlage 7: Umrechnungstabelle (dezimal <-> hexadezimal)

1. Byte		2. Byte		3. Byte	
6. Tetrade	5. Tetrade	4. Tetrade	3. Tetrade	2. Tetrade	1. Tetrade
hex = dez	hex = dez	hex = dez	hex = dez	hex=dez	hex=dez
0	0	0	0	0	0
1	1.048.576	1	65.536	1	4.096
2	2.097.152	2	131.072	2	8.192
3	3.145.728	3	196.608	3	12.288
4	4.194.304	4	262.144	4	16.384
5	5.242.880	5	327.680	5	20.480
6	6.291.456	6	393.216	6	24.576
7	7.340.032	7	458.752	7	28.672
8	8.388.608	8	524.288	8	32.768
9	9.437.184	9	589.824	9	36.864
A	10.485.760	A	655.360	A	40.960
B	11.534.336	B	720.896	B	45.056
C	12.582.912	C	786.432	C	49.152
D	13.631.488	D	851.968	D	53.248
E	14.680.064	E	917.504	E	57.344
F	15.728.640	F	983.040	F	61.440

III-12-12 Kv 825/89

NOTIZEN

NOTIZEN

NOTIZEN

2