

CMD

HD Series™

ANLEITUNG



Copyright

**Copyright © 1992
von Creative Micro Designs, Inc.**

1. Ausgabe, Erster Druck, Frühjahr 1993

Die Anleitung, Hardware und sämtliche Programme sind urheberrechtlich geschützt. Kein Teil dieser Anleitung darf in irgendeiner Form, ohne schriftliche Erlaubnis von Creative Micro Designs, Inc. gedruckt, vervielfältigt oder kopiert werden.

Das HD DOS ist ebenfalls urheberrechtlich geschützt und darf auch nicht teilweise ohne schriftliche Erlaubnis von Creative Micro Designs, Inc. kopiert werden.

JiffyDOS, RAMLink, und HD Series sind Schutzmarken der Creative Micro Designs, Inc. Commodore 64, 64C, SX64, C-128, C-128D, 1541, 1541-C, 1541-II, 1571 und 1581 sind Schutzmarken der Commodore Electronics Limited. Amiga ist eine Schutzmarke der Commodore Amiga. GEOS, GEOS deskTop, GEORAM und Berkeley Softworks sind Schutzmarken der Berkeley Softworks. CP/M ist eine registrierte Schutzmarke der Digital Research Corp. Apple und Macintosh sind registrierte Schutzmarken der Apple Computer, Inc. IBM ist eine registrierte Schutzmarke der International Business Machines. MS-DOS und Microsoft sind registrierte Schutzmarken der Microsoft Corporation. Lt. Kernal ist eine registrierte Schutzmarke der Xetec, Inc. und Fiscal Information, Inc.

Für Verbesserungsvorschläge und Hinweise auf Fehler sind Verlag und Herausgeber dankbar!

Kapitel 5: Der Gebrauch von GEOS.....	15
Die Einleitung.....	15
Was Sie benötigen.....	15
Installation von GEOS HD	15
Einstellung der HD unter GEOS.....	16
GEOS mit der Festplatte.....	16
Der Gebrauch von QuickMove.....	17
GEOS von der HD booten.....	18
GEOS-Ratschläge.....	18
Kapitel 6: CP/M mit der HD.....	19
Der Gebrauch der HD mit CP/M	19
Die Installation von CP/M.....	19
Partitionswechsel unter CP/M	20
Kapitel 7: Kommerzielle Software mit der HD.....	21
Arten von kommerzieller Software.....	21
Software ohne Kopierschutz	21
Kopiergeschützte Software	22
Kapitel 8: Befehlshinweis	23
Befehlssyntax	23
Befehlsschleife-Elemente.....	23
Befehlsschleife-Beispiel	23
Befehlsschleife-Pfade	24
Unterverzeichnispfade mit JiffyDOS-Befehlen	25
Befehle von BASIC Sendén.....	25
Der Befehlskanal	25
Diskettenfehler-Lesen.....	26
Partitionsnummern in Dateinamen.....	27
Partitionsnummern in Diskettenbefehlen.....	27
Partitionsbefehle	27
Partitionen erstellen.....	27
1581-Unterpitionen erstellen	27
Das Löschen von Partitionen.....	28
Das Wechseln von Partitionen	28
Das Formatieren von Partitionen.....	29
Das Initialisieren der Partitionen.....	30
Das Validieren der Partitionen.....	30
Das Partitions-Verzeichnis	31
Das Ändern von Partitionsnamen.....	31
Das Umbenennen von Verzeichnis-Headern.....	31
Das Beschaffen von Partitionsinformation	32
Automatisches Booten.....	32
Unterverzeichnisbefehle	33
Das Erstellen von Native-Modus-Unterverzeichnissen.....	33
Sich zwischen Native-Modus-Unterverzeichnissen bewegen	33
Das Löschen von Native-Modus-Unterverzeichnissen.....	34
Verzeichnis-Überblick	34
Muster vergleichen	35
Verzeichnisauflisten mit Zeit und Datum stempeln	35
Dateibefehle	36
Das Laden von Dateien.....	37
Das Speichern von Dateien.....	37
Das Verifizieren (Prüfen) von Dateien.....	38
Das Umbenennen von Dateien und Unterverzeichnissen	39
Das Löschen von Dateien (Scratchen).....	39
Das Kopieren von Dateien.....	40

Inhaltsverzeichnis

Kapitel 1: Allgemeine Informationen	1
Einleitung.....	1
Sonderzüge der HD	1
Allgemeines über Festplatten.....	1
Zubehör.....	1
JiffyDOS.....	2
RAMLink.....	2
Ersatz-Netzgeräte.....	2
Kabel.....	2
Bestellungen und Auskunft.....	2
Technische Hilfe	2
Anerkennungen.....	2
Kapitel 2: Inbetriebnahme.....	3
Der Anschluß	3
Verbindungen	3
Das Einschalten der Festplatte.....	3
Einschalt Diagnosen	3
Ausschalten der HD	4
Low-Level Formatieren	4
Upgrades und HD DOS Installation.....	4
Partitionen Erstellen/Konfiguration.....	4
Konfigurations-Parameter.....	5
Partitionen.....	5
Der Anschluß Zusätzlicher Festplatten.....	5
Sonstiges.....	5
Die Echtzeituhr.....	5
Das Einstellen der Uhr.....	5
Eigenschaften der Laufenden Festplatte	6
Kapitel 3: Sonderfunktionen.....	7
Vorgegebene Geräteadresse (Default Device Number).....	7
Vorgegebene Partition (Default Partition).....	7
Das Tauschen der Geräteadressen.....	7
Merken Sie sich bitte beim Geräteadressetauschen folgendes:	7
Schreibschutz	7
Zeit und Datum Einstellen	7
Frontseite-Partitionswahl (FPPS).....	8
Das FPPS-Modus.....	8
Das Wählen einer Partition im FPPS-Modus.....	8
Konfigurationsmodus.....	10
Installationsmodus.....	10
Die Auxiliarybuchse	10
Die Parallelbuchse.....	10
Kapitel 4: Partitionen und Unterverzeichnisse	11
Partitionen.....	11
Native-Modus-Partitionen	11
Emulations-Modus-Partitionen.....	11
Die Anwendung von Partitionen.....	12
Aktuelle Partition.....	13
Native-Modus-Unterverzeichnisse	13

Das Sperren und Entsperren von Dateien	41
Relativdateibefehle	42
Das Öffnen und Erstellen einer Relativdatei	42
Auf einem bestimmten Datensatz positionieren	43
Besondere HD-Befehle	43
Software SWAP- (Tausch-)Befehle	43
Software-Schreibschutz-Befehle	44
Echtzeituhr-Befehle	45
Das Senden von SCSI-Befehlen	47
Direktzugriffs-Befehle	47
Der Direktzugriffskanal	47
Das Lesen und Schreiben von Daten mit Direktzugriff	48
Blockbefehle	49
Speicherbefehle	51
Das Lesen von HD-Speicherbereichen	51
Das Schreiben zu HD-Speicherbereichen	51
Speicherausführung (Memory Execute)	52
User-Befehle	52
U0 Utility-Befehle	52
Blöcke lesen mit U1	53
Blöcke schreiben mit U2	53
User-Sprung-Befehle	54
Burst-Befehle	55
1541/1541 HD-Native-Modus-Burst-Befehle	55
1581/1581 CP/M Burstbefehle	60
Spezielle Lader	65
Job-Queue (Warteschlange) Anweisungen	65
Job-Queue Befehl-Codes	65
Job-Queue-Stellen	66
Anhang A: Utilities (Dienstprogramme)	67
Information über die Utility-Disketten	67
HD UTILITIES	67
GEOS UTILITIES	67
CP/M UTILITIES	67
Programmbeschreibung	67
HD-TOOLS(.64/.128)	68
FCOPY	69
MCOPY	70
SET HD CLOCK (HD-Uhr einstellen)	70
BOOTQ	70
XFER QDATA	70
AUTO-BOOT 128	70
PARK HD (.64/.128)	70
DISK CRACKER HD	71
PARTITION INFO	71
FIX BLOCKS	71
REWRITE DOS(.64/.128) (DOS neu installieren)	71
CREATE SYS(.64/.128)	71
LLFORMAT(.64/.128)	72
ADD DRIVE(.64/128) (Neue SCSI-Geräte anschließen)	72
1541SUB und 1581SUB	72
DRIVE INFO	72
Anhang B: Fehlercodes	73

Anhang C: Partitions- und Dateiformate	76
Gewöhnliche Formate in allen Partitionstypen	76
1541 und 1571 Emulations-Modus-Partitionen	77
1581 Emulations-Modus-Partitionen	79
Native-Modus-Partitionen	80
Datei-Formate	84
Anhang D: HD-Speichertabelle	86
Anhang E: SCSI-Buchse	88
Anhang F: Seriell- und Zusatzbuchsen	89
Anhang G: Parallelbuchse	90
Anhang H: Netzstecker	91
Anhang I: Technische Daten	92
Anhang J: Befehl-Zusammenfassung	93
Begrenzte Garantie	97
Problembericht	99

Kapitel 1

Allgemeine Informationen

Einleitung

Die CMD Festplatte wurde so entworfen, damit sie dem Commodore 64/128 als das, am leichtesten verwendbarste Festplattensystem zur Verfügung steht. Kompatibilität mit der breitmöglichsten Palette von Software wurde eingebaut. Viele Sonderfunktionen und die Fähigkeit, die Speichergröße des Systems zu erweitern sind auch in der Festplatte vorhanden. Zusätzlich darf die HD (Hard Drive, z.D. Festplatte) durch die SCSI-Buchse mit anderen Arten von Computern verbunden werden. Dies erlaubt das "Zusammenleben" von Informationen verschiedenster Computer auf der HD.

Sonderzüge der HD

Neben den gewöhnlichen Festplatten-Eigenschaften, besitzt Die HD-Serie eine Reihe von Sonderzügen. Hier eine kurze Liste dieser Züge:

- Kompatibilität mit Commodore DOS Befehlen, GEOS und CP/M
- Leichter Anschluß über den Serial-Port
- Kompatibilität mit allen Serial-Betriebsarten und Burst-Befehlen
- Echtzeituhr, die alle Dateien mit Zeit und Datum verseht
- Schreibschutzschalter an der Gerätefrontseite
- Tausch-Funktion erlaubt das Ändern der Geräteadresse auf 8 bzw. 9
- Bis zu 254 separaten Partitionen (Speicherbereichen)
- 1541, 1571, 1581 Emulations-Modi
- Native-Modus-Partitionsgrößen bis 16 MB mit echten Unterverzeichnissen
- Frontseite-Partitionswahl ermöglicht manuellen Zugriff zu allen Partitionen.
- SCSI Interfacebuchse ermöglicht die Verbindung mit anderen SCSI-Geräten.
- Die HD darf mit anderen Arten von Computern (Amiga, IBM, Macintosh) betrieben werden.
- Partitionen für andere Arten von Computern dürfen auf der Festplatte "zusammenleben."

Allgemeines über Festplatten

Festplatten, auch "Winchesters" oder Fixierte-Disketten genannt, werden in der Regel als Speicher für Dateien und Programme verwendet. Es gibt zwei Hauptunterschiede zwischen Festplatten und Laufwerken: Festplatten bieten mehr Speicherplatz und die Platte kann nicht vom Laufwerk entfernt werden. Festplatten brauchen auch eine besondere Steuerelektronik, um mit den meisten Systemen zusammenarbeiten zu können. Dies ist besonders der Fall mit dem Commodore-System, wo ein eigebautes DOS in der Festplatte notwendig ist.

Es gibt auch Unterschiede zwischen den verschiedenen Arten von Festplatten. Die wohl bekanntesten Arten sind die RLL und MFM Festplatten. Diese Abkürzungen beschreiben eigentlich die Methoden, die die Festplatten anwenden, um Informationen auf die Platte zu speichern. Diese HDs sind die gewöhnlichen IBM kompatiblen Festplatten, die eine separate Steuerung des DOSes anwenden, um Dateien auf die Platte schreiben zu können. Weil diese HDs diese Steuerung brauchen, jedoch nur den Laufwerkmechanismus und eine kleine Zahl Elektronik besitzen, bezeichnet man sie als "dumme" Platten. Die SCSI-HD von CMD dagegen, nennt man eine "kluge" Festplatte, weil sie nicht nur den Laufwerkmechanismus und die Elektronik besitzt, sondern auch einen On-Board-Prozessor. Dieser Prozessor kontrolliert alle bestimmten Aufgaben der Festplatte. Der große Vorteil der CMD Festplatte liegt darin, daß man sie, wegen des SCSI-Anschlusses und der eingebauten Steuerung, anderen Computertypen anpassen kann.

Zubehör

Eine Vielzahl an Zubehör kann man von CMD oder deren Vertreibern beziehen, die die Fähigkeiten der HD erweitern bzw. die Arbeit damit erleichtern. Hier eine Liste des Zubehörs mit kurzer Beschreibung ihrer Vorteile:

JiffyDOS

JiffyDOS ist ein Floppyspeeder, der Zugriffe auf Laufwerke und zwischen HD und Laufwerken beschleunigt. Es besitzt auch ein DOS-Wedge, um das Senden von Befehlen an die HD zu erleichtern.

RAMLink

RAMLink bietet dem Anwender die Möglichkeit an, die HD mit dem Rechner parallel anzuschließen. Dadurch werden Zugriffszeiten wesentlich schneller. RAMLink besitzt ein sehr schnelles RAMDOS, das die Commodore RAM-Erweiterung und GEORAM mit kommerzieller Software kompatibel macht. Zusätzliches RAM von 1 bis 16MB kann in RAMLink eingesteckt werden. RAMLink hat ein eigenes Netzteil, welches das Erhalten von Dateien nach Ausschalten des Rechners sichert.

Ersatz-Netzgeräte

Ersatz-Netzgeräte für die Festplatte kann man von CMD oder deren Vertreter beziehen. Ein Ersatz-Netzteil kann man zum Beispiel benutzen, wenn man die Festplatte an zwei verschiedenen Orten (daheim und im Büro) einsetzen will. Diese Netzgeräte ersetzen auch die Netzteile der Commodore 1541 II und 1581 Laufwerke.

Kabel

Eine Vielzahl von Kabeln bietet CMD für den Gebrauch mit der Festplatte und anderen Peripheriegeräten an. Wir haben Ersatzseriellkabel, verlängerte Seriellkabel, RAMLink-Parallelkabel und 25 polige Steckerkabel für zusätzliche Laufwerke bzw. Festplatten.

Bestellungen und Auskunft

Falls Sie eine Bestellung aufgeben möchten, oder Auskunft über CMDs Produkte brauchen, können Sie sich an CMDs Vertreter wenden, oder CMD direkt anrufen bzw. faxen. Sie können CMD von Montag bis Freitag von 16 bis 23 Uhr (MEZ) unter der Telefonnummer 001413-5250023 oder der FAXnummer 001413-5250147 erreichen. Falls Sie Probleme mit einer Bestellungen haben, wenden Sie sich bitte an den Vertreter, von dem Sie bestellt haben.

Technische Hilfe

Falls Sie technische Hilfe brauchen, setzen Sie sich mit uns in Verbindung.

Notieren Sie bitte die Laufnummer, Modellnummer und sonstige wichtige Informationen ins Problem-Bericht-Log im Anhang dieser Anleitung. Wenn Sie ein Modem haben, sind wir 24 Stunden durchgehend bei 300/1200/2400 BAUD erreichbar (Parameter: (8-N-1 oder 8-None-1)). Die neuesten Updates können Sie auch downloaden. Die Telefonnummer für dieses Service ist 001 413 525 0148. Eine Antwort bekommen Sie innerhalb 48 Stunden. Teilen Sie uns auch bitte mit, wie Sie Ihre Festplatte erfolgreich eingesetzt haben!

Anerkennungen

Voller Stolz bringt CMD ihre Festplatten auf den Markt. Viele Mitarbeiter, deren Namen wir bekannt geben möchten, haben lange daran gearbeitet. Mark Fellows, Vater von JiffyDOS, schrieb das DOS und übersah die gesamte Entwicklung. Charlie Andrews entwickelte die Hardware; Doug Cotton die Utilities und das Handbuch; Charles A. Christianson Jr. überzeugte uns von dem Projekt, und Charles R. Christianson Sr. hat alles organisiert. Zusätzlich bedanken wir uns bei den folgenden Leuten außerhalb der Firma, ohne deren Hilfe die Verwirklichung dieser Arbeit sicherlich länger gedauert hätte: Paul Bosacki für die GEOS-Patches, Miklos Garamszeghy für seine Einsichten und Dienstprogramme für CP/M, Brian Dougherty und Matt Loveless von Berkeley Softworks für ihre GEOS-Hilfen und Richard Gaudet für seine Übersetzungsarbeit.

Kapitel 2

Inbetriebnahme

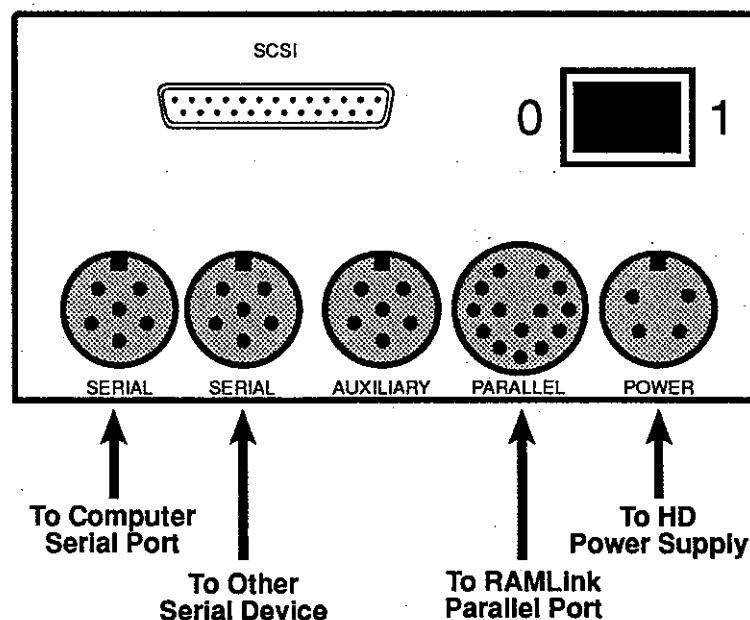
Der Anschluß

VORSICHT: Bevor Sie die Festplatte zum ersten Mal in Betrieb nehmen, sollten Sie diese Anleitung durchlesen! Es wird vielleicht nicht unbedingt notwendig sein, die ganze Anleitung durchzulesen, aber Sie sollten die Teile, die für Sie wichtig sind, gründlich lesen!

Entfernen Sie die Festplatte von der Verpackung und stellen Sie sie so neben Ihren Computer, daß der Luftzufuhr ungehindert ist. Packen Sie auch das Netzteil aus, und stellen Sie auch dieses dorthin, wo es einen guten Luftzufuhr hat.

Verbindungen

Schließen Sie die HD an den Seriellport des Computers mit dem mitgelieferten Seriellkabel an. Beide seriellen Anschlüsse der HD können dafür benützt werden. Falls der Seriellport des Rechners belegt ist, können Sie den freien Anschluß eines angeschlossenen Gerätes benutzen.



Als nächstes, schließen Sie das Netzteil an die HD an. Der runde DIN-Stecker wird in die POWER-Buchse gesteckt.

Bevor Sie das Netzteil an den Strom anschließen, vergewissern Sie sich, daß sowohl das Netzteil, als auch die HD auf "Off" (AUS) stehen. Schließen Sie jetzt das Netzteil an den Strom an. Wir empfehlen den Gebrauch eines Gerätes, das vor schwankenden Netzüberspannungen und Störspannungen schützt.

Das Einschalten der Festplatte

Die HD ist jetzt für den Gebrauch mit Ihrem Computer bereit. Schalten Sie zuerst das Netzteil ein. Das Netzteil können Sie immer eingeschaltet lassen. Dies schont den Akku, der die Echtzeituhr mit Strom versorgt. Schalten Sie jetzt die HD ein. (Manchmal spielt die Reihenfolge beim Einschalten der Geräte eine Rolle. Wenn dies bei Ihnen der Fall sein sollte, folgen Sie immer dieser Reihenfolge.)

Einschaltdiagnosen

Beim Einschalten macht die HD einen Selbsttest, um eventuelle Fehler aufzudecken. Auf der Frontseite leuchten die kleinen Anzeigerlämpchen kurz auf. Falls ein Fehler entdeckt wird, bleibt ein Lämpchen leuchtend und das ERROR-

Lämpchen blinkt. Im Falle eines Fehlers im SCSI-Anschluß blinkt das ERROR-Lämpchen allein. Nur das obere ACTIVITY-Lämpchen wird beim Diagnosentest gebraucht, obwohl das untere ab und zu mitleuchtet (zeigt SCSI-Bewegungen an).

Folgende Liste zeigt welche Funktion der HD getestet wird, wenn ein Anzeigerlämpchen aufleuchtet:

ACTIVITY	-	Serielle Anschlüsse (I/O)
ERROR	-	SCSI-Anschlüsse (I/O)
GEOS	-	Pararellsteuerung (I/O)
SWAP 8	-	RAM (Nullseite)
SWAP 9	-	Boot ROM
WRITE PROTECT	-	RAM (ganze 64K)

Ausschalten der HD

Wenn Sie die Festplatte ausschalten, betätigen Sie einfach den Ein/Aus-Schalter auf ihrer Rückseite. Der Laufmechanismus in der HD ist "auto-parking". Das heißt, die Lese/Schreibköpfe werden automatisch sichergestellt. Mann kann auch die Köpfe der HD und anderer angeschlossenen HDs manuell parken, in dem man das PARK HD-Programm auf der HD-UTILITIES-Diskette gebraucht.

Low-Level Formatieren

Low-Level Formatieren wurde von CMD schon durchgeführt, und darf daher nur dann wiedergemacht werden, wenn Probleme mit der HD auftauchen. Wenn Sie zusätzliche Festplatten an Ihr System anschließen, sollten Sie ein Low-Level-Formatieren durchführen. Die Programme LLFORMAT. 64 und LLFORMAT. 128 sind für diesen Zweck zu verwenden. Die Programmanleitung dazu ist im Anhand A zu lesen.

VORSICHT: LLFORMAT zerstört alle Dateien auf der Festplatte. Verwenden Sie dieses Dienstprogramm nur wenn es notwendig ist!

Upgrades und HD DOS Installation

HD DOS ist vom Haus aus auf der Festplatte installiert. Es sollte keinen Grund dafür geben das DOS neu installieren zu müssen. Falls aber etwas doch im DOS fehlerhaft ist oder Sie von CMD ein Upgrade bekommen, können Sie mit dem Programm REWRITE DOS.64 bzw. 128 das DOS neu installieren. Dieses Programm schreibt nur ein neues DOS und schadet keinen Dateien oder Partitionen auf der HD. Die Anleitung dafür finden Sie im Anhang A.

Im extrem äußersten Fall kann es notwendig sein, das komplette DOS neu schreiben zu müssen. Dies muß immer getan werden, wenn die Festplatte neu formatiert wurde, muß aber nicht gemacht werden, wenn zusätzliche HDs oder SCSI-Geräte ans System angeschlossen werden. Das Programm CREATE SYS.64 bzw. CREATE SYS.128 wird dafür benützt. Die Anleitung dafür finden Sie im Anhang A.

VORSICHT: CREATE SYS zerstört alle Dateien auf der Festplatte. Bitte nur verwenden, wenn absolut notwendig!

Partitionen Erstellen/Konfiguration

Die HD ist mit einigen Partitionen schon versehen. Sie können sie sofort einsetzen, wenn Sie möchten. Aber damit die Festplatte Ihren Wünschen entspricht, wird es wahrscheinlich notwendig sein, einige neue Partitionen zu erstellen, bzw. alte Partitionen zu löschen. Vielleicht möchten Sie auch einige Eigenschaften der HD ändern. Dies können Sie mit dem Programm HD-TOOLS machen. Bitte achten Sie auf die Anleitung dieses Programms im Anhang A.

Konfigurations-Parameter

Folgende Parameter sind von CMD voreingestellt, können aber vom Anwender modifiziert werden:

DEFAULT DEVICE NUMBER (Vorgegebene Geräteadresse)	Die Geräteadresse der HD beim Einschalten. Die Voreinstellung ist 12.
DEFAULT PARTITION (Vorgegebene Partition)	Die gegenwärtige Partition beim Einschalten. Die Voreinstellung ist Partition 1.

Partitionen

Die HD darf in Speicherbereichen - Partitionen genannt - eingeteilt werden. Dies ähnelt dem Besitz mehrerer Laufwerke in einem einzigen Laufwerk. Hier sind einige Richtlinien, die man sich beim Arbeiten mit den Partitionen merken sollte:

- A. Es muß mindestens eine Partition auf der HD vorhanden sein.
- B. Sie können bis zu 254 Partitionen erstellen.
- C. Partitionsgröße hängt von dem Partitionstyp ab. Emulations-Modus-Partitionen umfassen die gleiche Zahl von Blöcken, wie die Laufwerke, die sie emulieren. Native-Modus-Partitionen können von 256 bis 65,280 Blöcken in Werten von 256 Blöcken umfassen.

Mindestens eine Partition wurde von CMD auf der Festplatte erstellt. Falls sie eine größere Festplatte (ab 40MB) bestellt haben, haben wir von jeder Partitionsart so viele wie möglich auf der Festplatte erstellt.

Der Anschluß Zusätzlicher Festplatten

Zusätzliche SCSI Geräte können über den SCSI-Anschluß an der Rückseite der HD angeschlossen werden. Diese Buchse ist eine DB-25S Verbindung, die die gleiche Polkonfiguration wie bei Apple Macintosh Computern besitzt. Diese Zusatzplatten kann man gewöhnlich über Versand oder Computer-Shops beziehen. Bis zu 6 Festplatten können angeschlossen werden. Vergessen Sie nicht, das Programm ADD DRIVE hier zu benutzen. Sonst werden diese Festplatten von der HD nicht anerkannt. Man kann auch separate SCSI-Steuerungen kaufen, die den Anschluß mehrerer Festplatten ermöglichen.

Die SCSI-Wirtssteuerung in der HD hat die SCSI-Geräteadresse 7. Diese Nummer kann man, wenn notwendig, auch später ändern. Die Festplatte selber hat die SCSI- Geräteadresse 0. Diese Adresse läßt sich nur dadurch ändern, indem man die HD öffnet und einen Jumper ändert. Die Garantie wird dadurch ungültig gemacht. Am besten kaufen Sie sich nur SCSI-Geräte, deren Jumpernummern sich ändern lassen. Die HDs totale Systemkapazität darf bis zu 4 Gigabytes betragen. Merken Sie sich bitte auch, daß Zusatzgeräte ihre eigenen Netzteile haben müssen.

Sonstiges

Die Echtzeituhr

Die CMD HD hat eine eingebaute Echtzeituhr, die das Prägen der Zeit und des Datums auf Dateien ermöglicht. Ein Akku versorgt die Uhr mit Strom und hat eine Lebensdauer von ca. 10 Jahren. Dieser Akku kann nur von einem Elektrotechniker ausgetauscht werden.

Das Einstellen der Uhr

Die Uhrzeit wurde von CMD eingestellt, dürfte aber Ihrer Zeitzone nicht entsprechen. Das Dienstprogramm, SET HD CLOCK, wird gebraucht, um die Uhrzeit richtig einzustellen. Dieses Programm ist ein Basic-Programm und darf als Beispiel verwendet werden, wenn Sie selber Programme schreiben und darin die Uhrzeit lesen und erstellen wollen.

Eigenschaften der Laufenden Festplatte

Während die HD läuft, werden Sie einiges merken, was Sie vielleicht beunruhigt. Folgende Auftretungen sind aber normal:

- | | |
|----------------------|---|
| MOTORGERÄUSCH | Im Gegensatz zu Laufwerken, dreht sich der HD-Motor ständig. Dies verursacht ein Schwirrengeräusch, das in manchen HDs lauter ist als in anderen. |
| HAUSHALTEN | Einige der Festplattenmechanismen führen regelmäßige "Haushaltsroutine" durch. Dabei rattert der HD-Kopf einmal alle 5 bis 10 Minuten. |
| MOTORWÄRME | Dadurch, daß der Motor sich immer dreht und in der Nähe der oberen Gehäuserseite montiert ist, wird die Oberseite der HD beim Dauerbetrieb warm, ja sogar heiß. Wenn die Festplatte einen guten Luftzufuhr hat, ist diese Wärme nicht schädlich. (Lassen Sie mindestens 3cm an allen Seiten der HD dafür frei!) |

Kapitel 3

Sonderfunktionen

Vorgegebene Geräteadresse (Default Device Number)

Die Geräteadresse Ihrer Festplatte wurde von CMD mit 12 voreingestellt. Diese Nummer ist nicht durch Hardware gesteuert, sondern liegt in einer Tabelle in der Systempartition der HD. Mit dem Programm HD-TOOLS können Sie jederzeit die Geräteadresse von 8 bis 29 ändern. Wenn aus irgendeinem Grund das Betriebssystem nicht richtig ladet, kann es vorkommen, daß die Festplatte in ihrer Bootkonfiguration zurückfällt und sich die Geräteadresse 30 gibt.

Vorgegebene Partition (Default Partition)

Die Default Partition ist die Partition, in der die HD sich beim Reset bzw. Einschalten befindet. Die vorgegebene Partition ist die Nummer 1. Diese Nummer können Sie auch jederzeit mit dem Programm HD-TOOLS ändern. Die geänderte Partition muß natürlich auf der Festplatte sein.

Das Tauschen der Geräteadressen

Eine der stärksten und einmaligen Funktionen der HD ist ihre Fähigkeit die Geräteadresse mit anderen seriell-angeschlossenen Laufwerken zu tauschen. Diese Funktion wird mit den SWAP 8 und SWAP 9-Schaltern an der vorderen Seite der HD durchgeführt. Ein leichtes Drücken auf den SWAP 8 oder SWAP 9-Schalter genügt, um die Festplattegeräteeadresse auf 8 bzw. 9 zu ändern. Das Laufwerk übernimmt dann die Geräteadresse der HD. Der rote Anzeiger über dem SWAP-Schalter leuchtet auf, wenn die Nummern getauscht wurden. Wenn Sie den SWAP-Schalter erneut drücken, werden die Nummern ihren alten Geräten wieder zugeteilt.

Gewöhnlich wird diese Funktion benötigt, wenn man Software hat, die nur mit den Geräteadresse 8 bzw. 9 laufen. Man kann die Programme auf die Festplatte speichern, die Geräteadresse ändern, und das Programm von der HD laden. Wenn das Programm kopiergeschützt ist, kann man Files, die nachgeladen werden können, auf die HD speichern, das Programm vom Laufwerk 8 laden, den SWAP 8-Schalter drücken und anschließend weiter mit der Festplatte arbeiten.

Merken Sie sich bitte beim Geräteadressetauschen folgendes:

1. Beim Einschalten oder Durchführen eines Resets eines C-64ers (auch wenn man vom 128er auf 64er Modus beim C-128er umschaltet) muß der Seriellbus, bevor man die Tauschfunktion benützt, ausgeschaltet werden. Dies geschieht durch irgendeinen Diskettenzugriff oder mit dem Befehl: POKE 56576,199.
2. Wenn GEOS aktiv ist, können die SWAP-Funktionen nicht eingesetzt werden. Der Grund dafür ist, daß GEOS immer einen aktiven Quittungsbetrieb mit dem Seriell-Bus-Laufwerk durchführt. Wenn Sie unbedingt eine SWAP-Funktion durchführen wollen, müssen Sie dies vor dem Booten tun.
3. Drücken Sie NIEMALS die SWAP-Schalter während eines Laufwerkzugriffs. Sie können wertvolle Dateien dadurch löschen!

Schreibschutz

Sie können die Festplatte mit einem Schreibschutz versehen, indem Sie die WRITE PROTECT-Taste an der Frontseite betätigen. Das rote Anzeigerlämpchen leuchtet dann auf. Durch erneutes Tastedrücken wird diese Funktion deaktiviert. Diese Funktion ist wertvoll, wenn man Software austesten oder eine andere Diskette in einem anderen Laufwerk formatieren und absolut sicher vorgehen will.

Zeit und Datum Einstellen

Im Schaltkreis der HD wurde eine akkugetriebene Echtzeituhr eingebaut. Diese Uhr versieht alle Dateien im Dateiverzeichnis mit der echten Zeit und dem echten Datum. Die Festplatte besitzt eine Menge neuer Befehle, die mit dieser Zeit- und Datumeinstellung zusammenhängen. Diese Befehle sind im Befehlshinweisteil dieser Anleitung.

Frontseite-Partitionswahl (FPPS)

Ab und zu kann es notwendig sein eine neue Partition zu wählen, welches das laufende Programm aber nicht zuläßt. Diese Situation kommt meistens bei Programmen vor, die keine oder nur gewisse Floppybefehle zulassen. Viele Multi-Diskettenprogramme verlangen einen Diskettentausch und überprüfen dann, ob dieser Tausch durchgeführt wurde. Man kann diese Art von Problemen lösen, in dem man mit dem FPPS-Modus die Partition wechselt.

Der FPPS-Modus

FPPS (z.D. Frontseite-Partitionswahl) benötigt die Betätigung von 3 Frontseitenschaltern der HD: SWAP 8, SWAP 9 und WRITE PROTECT. Hier eine kurze Beschreibung deren Funktionen.

- | | |
|---------------|---|
| WRITE PROTECT | Dieser Schalter hat 2 Funktionen im FPPS Modus. Durch das Drücken dieseses Schalters wird der Modus ein- bzw. ausgeschaltet. Er wird auch benötigt, um die Einerstelle der Partitionsnummer einzustellen. |
| SWAP 8 | Mit diesem Schalter stellt man die Zehnerstelle der Partitionsnummer ein. |
| SWAP 9 | Mit diesem Schalter stellt man die Hunderterstelle - zu gleich die höchste Partitionsnummer - ein. |

Merken Sie sich beim Umschalten, daß nur ein leichtes und schnelles Zudrücken der Schalter erforderlich ist. Die Anzeigerlämpchen für GEOS, SWAP 8, SWAP 9 und WRITE PROTECT zeigen die gewählte Partition an.

Das Wählen einer Partition im FPPS-Modus

Folgendes Beispiel zeigt Ihnen wie Sie den FPPS-Modus benützen. Versuchen Sie dieses Beispiel selber und dann den Modus mit anderen Partitionen.

1. Schalten Sie den FPPS-Modus durch Drücken und kurzes Niederhalten (2 Sekunden) des WRITE PROTECT-Schalters ein. Das grüne, obere Lämpchen leuchtet auf. Lassen Sie den Schalter jetzt los. Das Lämpchen blinkt. Sie sind jetzt im FPPS-Modus.
2. Drücken Sie den WRITE PROTECT-Schalter wieder und lassen Sie ihn los. Die Anzeigerlämpchen für GEOS, SWAP 8, SWAP 9 und WRITE PROTECT sollen alle erlöschen.
3. Drücken Sie erneut den WRITE PROTECT-Schalter und lassen Sie ihn schnell wieder los. Das WRITE PROTECT-Anzeigerlämpchen leuchtet auf. Dies bedeutet, daß die Einerstelle der dreizahligen Partitionsnummer auf 1 eingestellt ist. Das Zahlschema liest man von der FPPS Anzeigertabelle ab.
4. Drücken Sie den WRITE PROTECT-Schalter jetzt einige Male. Jedes Mal vergleichen Sie den Anzeigerstatus mit der Tabelle. Kehren Sie zum Wert 1 zurück (WRITE PROTECT Lämpchen leuchtet allein auf.)

Weil Sie bis zu 254 Partitionen erstellen können, sind 3 Zahlen notwendig, um eine Partitionsnummer zu bestimmen. Bis jetzt haben Sie Partition 001 eingestellt. Wenn Sie den FPPS- Modus jetzt verlassen, stellt die HD Partitionsnummer 1 ein.

5. Drücken Sie schnell den SWAP 8 Schalter und lassen Sie ihn wieder los. Das WRITE PROTECT-Lämpchen ist erloschen. Sobald Sie einen Schalter zum ersten Mal drücken und loslassen (oder nachdem Sie einen von den anderen Schaltern drücken) zeigt der Modus den augenblicklichen Status der Zahl, die von diesem Schalter gesteuert wird. Weil alle Zahlen durch Einschalten des FPPS-Modus' gelöscht werden, wird durch das erstmalige Drücken und Loslassen jedes Schalters, die Zahl 0 eingestellt. Gehen Sie durch alle Nummern mit dem SWAP 9-Schalter bis die Zahl 0 an der Zehnerstelle wieder erreicht wird (alle Lämpchen erlöschen).
6. Wiederholen Sie Punkt 5, dieses Mal mit dem SWAP 8-Schalter. Diese Zahl läßt sich nur bis 2 einstellen. Der Grund dafür ist, daß Partitionsnummern über 254 nicht erlaubt sind. Wenn diese Zahl auf 2 eingestellt ist, wird sowohl die Zehnerstelle begrenzt (auf 5) als auch die Einerstelle (auf 4), wenn die Zehnerstelle auf 5 eingestellt ist. Spielen Sie ein wenig mit diesen Kombinationen, stellen Sie aber zum Schluß die Partitionsnummer 1 wieder ein.
7. Verlassen Sie den FPPS-Modus mit einem Drücken und Halten des WRITE PROTECT-Schalters, bis das obere Activity-Lämpchen erlischt. Lassen Sie den WRITE PROTECT-Schalter los. Sie sollten sich in Partition 1 befinden.

FPPS ANZEIGERTABELLE

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	0
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	1
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	2
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	3
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	4
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	5
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	6
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	7
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	8
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
GEOS	SWAP 8	SWAP 9	WRITE PROTECT	RESET	=	9

ANZEIGERSTAU: EIN ☒ AUS ☐

Konfigurationsmodus

Wann immer Sie die Voreinstellungen Ihres Systems ändern, Partitionen erstellen oder löschen, oder eine neue DOS-Version auf die Festplatte speichern wollen, müssen Sie die HD in den Konfigurationsmodus versetzen. Drücken und **halten** Sie den WRITE PROTECT-Schalter. Drücken Sie den RESET-Schalter und lassen Sie ihn los. Die Anzeigerlampchen blinken kurz auf. Lassen Sie dann den WRITE PROTECT-Schalter los. Die HD befindet sich jetzt im Konfigurationsmodus. Jetzt können Sie die notwendige Software laden und laufen lassen. Nachdem Sie die Änderungen gemacht haben, kehren Sie zum normalen Modus zurück, indem Sie den RESET-Schalter drücken.

Installationsmodus

Falls die Systemparameter fehlerhaft werden und das Booten des Systems nicht möglich ist, muß es wahrscheinlich notwendig sein, das komplette System neu installieren zu müssen. Versuchen Sie zuerst das DOS neu auf die HD zu schreiben. Wenn das nicht hilft, müssen Sie die HD in den Installationsmodus versetzen und das Programm CREATE SYSTEM verwenden. Dieses Programm befindet sich auch auf der HD UTILITIES Diskette.

VORSICHT: Tun Sie dies nur als allerletzte Hilfe! CREATE SYSTEM zerstört alle Dateien auf der Festplatte!

Schalten Sie den Computer und die HD ein. Drücken und halten Sie die SWAP 8 und SWAP 9 Tasten. Drücken Sie jetzt auch die RESET-Taste. Das ACTIVITY-Lämpchen leuchten auf.

Lassen Sie beide SWAP-Tasten wieder los. Die HD ist jetzt im Installationsmodus. Laden Sie jetzt das Program CREATE SYSTEM von Ihrem Laufwerk. Nach einigen Minuten wird eine neue Systempartition und ein neues DOS auf die HD geschrieben. Nach Beenden des Programms erscheint die READY- Meldung. Drücken Sie jetzt die RESET-Taste.

Bevor Sie mit dem System beginnen können, müssen Sie erneut Partitionen erstellen. Verwenden Sie wieder das HD-TOOLS Programm. Die Geräteadresse und Partitionsnummer werden Sie wahrscheinlich auch ändern wollen.

Falls das Problem immer noch nicht behoben wird, setzen Sie sich mit CMD in Verbindung.

Die Auxiliarybuchse

Diese Buchse wird zur Zeit nicht verwendet. Zukünftig können Drucker daran angeschlossen werden. Ein neues DOS ordnet der HD eine Art Puffer zu. Dateien werden in der HD gespeichert und anschließend zum Drucker geleitet.

Die Parallelbuchse

Die Funktion dieser Buchse ist die Schnellübertragung von Dateien zwischen der HD und CMD's RAMLink.

Kapitel 4

Partitionen und Unterverzeichnisse

Partitionen

Der gesamte Speicherbereich der CMD-Festplatte läßt sich in kleineren Bereichen - Partitionen genannt - einteilen. Dadurch werden eigentlich mehrere Laufwerke auf einem einzigen Laufwerk erstellt. Insgesamt können bis zu 254 Partitionen auf der HD gleichzeitig existieren. Wie vorher erwähnt wurde, ist die HD vom Haus aus mit einigen Partitionen bestückt. Partitionen werden erstellt bzw. gelöscht mit dem Dienstprogramm HD-TOOLS. Jede Partition wird als eine bestimmte Art definiert. Vier verschiedene Arten stehen dem HD Anwender zur Verfügung:

- Native-Modus-Art
- Emulation-Modus-Art
- Print-Buffer (Druckerpuffer)-Art
- Foreign-Modus-Art

Native-Modus-Partitionen

Native-Modus-Partitionen ermöglichen den Gebrauch der vielen zusätzlichen Funktionen des CMD HD-DOS, wobei die volle Kompatibilität mit den Standard-Commodore-DOS-Befehlen erhalten bleibt. Mit dieser Art von Partition haben Sie einen Speicherbereich von bis zu 16 Megabytes. Diese hohe Kapazität steht den Commodore DOS-Befehlen auf der HD zu Verfügung. Weil Native-Modus-Partitionsgröße variabel ist (Die kleinste Partition kann 256 Blöcke haben.), hängt die Größe einer Partition allein von Ihnen ab. Nur dieser Modus unterstützt Unterverzeichnisse und dynamische Zuteilung des Verzeichnissesplatzes. Sie können daher Ihre Dateien leicht organisieren und neue Dateien hinzufügen, bis die Partition voll ist.

Native-Modus-Partitionen besitzen 256 Sektoren pro Spur und können von 1 bis 255 Spuren haben. Weil die Größe der Partition variabel ist, müssen alle Vorsatzblock-, BAM-, und Verzeichnisinformationen auf die erste Spur gespeichert werden. Unterverzeichnisse besitzen eine DIR-Dateienart und haben einen Dateienartwert von 6 erhalten.

Emulations-Modus-Partitionen

Bei Emulation-Modus-Partitionen bleibt die Kompatibilität mit Softwareprogrammen erhalten. Die Spuren und Sektoren, sowohl BAM als auch Verzeichnis, werden einem bestimmten Commodore-Laufwerk identisch. Eine Emulation-Modus-Partition hat daher die gleiche Speicherkapazität wie das Laufwerk, das sie emulieren sollte. Es stehen Ihnen vier Emulation-Modus-Partitionsarten zur Verfügung:

- 1541 Emulation-Modus-Art
- 1571 Emulation-Modus-Art
- 1581 Emulation-Modus-Art
- 1581 CP/M Emulation-Modus-Art

Wie die Namen andeuten, emulieren diese Partitionen die gewöhnlichen Commodore-Laufwerke 1541, 1571, und 1581. Diese Laufwerke werden fast haargenau nachgeahmt. Eine vollkommene Nachahmung hätte die Herstellung und den Verkauf der HD unrentabel gemacht. Es folgt eine Beschreibung der verschiedenen Emulation-Modi.

1541 Emulations-Partitions

Hier sind Verzeichnis und BAM an der gleichen Stelle wie beim Commodores 1541-Laufwerk. Alle Bytes innerhalb dieser Blöcke sind ihren Gegenüberbytes im 1541 gleich. Diese Partionart benützt 684 Blöcke Diskettenplatz (342 SCSI Blöcke), wovon 664 frei sind.

Einige Unterschiede gibt es doch zwischen dem 1541-Laufwerk und dem 1541- Emulationslaufwerk. Erstens ist es in diesem Emulationsmodus völlig möglich die schnelle serielle Datenübertragung zu benützen. Dies wird automatisch erledigt, wenn schnelle serielle Geräte verwendet werden. Zusätzlich erlaubt der 1541-Emulationsmodus den Gebrauch von 1571-Burstbefehle. Diese 2 Unterschiede machen aus der 1541-Partition eine einseitige 1571-Diskette.

Die Weise, in der REL-Dateien behandelt werden, kann ab und zu auch anders sein. In einem gewöhnlichen 1541-Laufwerk dürfen nur 6 relative Dateisektoren gebraucht werden. So sieht es auch im 1541-Emulationslaufwerk aus. Man kann aber, der "Supersektorsektor" des REL Dateiformats des 1581-Laufwerks schreiben und von ihr lesen. Es bedarf aber einer Sonderprogrammierung, um solche Datei in 1541-Partitionen zu erstellen. Wenn sie aber erstellt ist, funktioniert sie ohne Probleme.

Noch einen Unterschied gibt es. Alle Dateien werden mit Datum und Zeit versehen. Dies geschieht in der gleichen Weise, wie bei GEOS. Alle Dateien, die auf der HD erstellt werden, zeigen die korrekte Zeit und das korrekte Datum auf, wenn sie mit den Verzeichnisbefehlen aufgerufen werden. Zeit und Datum werden auch geändert, wenn die Datei kopiert, neu benannt oder geändert wird.

1571 Emulations-Partitionen

1571 Emulations-Partitionen sind, mit einigen Ausnahmen, den 1541-Partitionen identisch. Erstens haben 1571-Partitionen zweimal soviel Speicherkapazität wie 1541-Partitionen. Diese Partitionen gebrauchen 1366 Blöcke Diskettenplatz (683 SCSI Blöcke) wovon 1328 für Daten frei sind.

Einige extra Bytes fürs BAM im Headerblock sind auf Spur 53 vorhanden. Diese sind genau wie bei einem normalen 1571-Laufwerk zugewiesen. 1571-Partitionen entsprechen auch dem beidseitigen 1571-Laufwerk. Wenn Sie aus irgend einem Grund ein einseitiges 1571- Emulations-Laufwerk benötigen, benützen Sie ein 1541-Emulations-Laufwerk.

Wie bei 1541-Emulations-Partitions, ist es auch hier möglich die "Supersektorsektor" der REL- Dateien zu lesen und darauf schreiben. Diese Methode erlaubt die REL-Dateien, die innerhalb der 1571-Emulations-Partition erstellt wurden, über die maximale Größe von 726 Blöcken hinaus zu wachsen.

Beim CP/M Modus mit 1571-Partitionen wird nur das Commodore CP/M unterstützt. Programme, die versuchen, den Diskettensteuerchip des normalen 1571-Laufwerk neu zu programmieren, werden nicht laufen, da dieser Chip in der HD nicht vorhanden ist.

1581- und 1581-CP/M-Emulationspartitionen

1581-Emulationspartitionen bieten eine exzellente Emulation der 1581-Floppy. Alle Header-, BAM-, und Verzeichnisinformationen sind dem 1581-Laufwerk identisch. Schnelle serielle und die meisten Burstbefehle sind völlig kompatibel. Der Hauptunterschied zwischen den beiden liegt darin, daß der Spur-Cache-Puffer des 1581-Laufwerks in der HD nicht vorhanden ist. Dieser Puffer beschleunigt das 1581-Laufwerk. Die HD hat diese Beschleunigung nicht nötig! Alle anderen DOS-Befehle werden durchgeführt. Um die volle Kompatibilität mit dem 1581 beizubehalten, startet der DOS-Anfangsbefehl eine Änderung im vorhandenen 1581- Partitionstatus. Diese Änderung führt dazu, daß weitere Diskettenzugriffe im Hauptverzeichnis der 1581-Partition stattfinden.

Wenn Sie eine 1581-Partition als CP/M-Bootpartition benützen wollen, müssen Sie den 1581- CP/M Emulationsmodus verwenden. Weitere Informationen über die Anwendung der HD mit CP/M finden Sie im Kapitel 6.

Druckerpuffer-Partitionen

Druckerpuffer-Partitionen sind zur Zeit mit der HD nicht brauchbar. Diese Funktion wird in zukünftigen DOS-Versionen vorhanden sein.

Foreign-Modus-Partitionen (Fremdmodus-Partitionen)

Foreign-Modus-Partitionen werden benützt um einen Bereich der HD für den Gebrauch mit Dateien von anderen Computerarten zu definieren. Das Dienstprogramm PARTITION INFO zeigt Ihnen genau wo die Partitionen auf der HD zu finden sind. Mit dieser Information können Sie die möglichen Zugriffsbereiche der HD auf anderen Computerarten definieren. Um das zu verwirklichen, müssen Sie Partitionsoftware, die für jene Computerart bestimmt ist, benützen.

Die Anwendung von Partitionen

CMD's Festplatte kann in kleineren Speicherbereichen, Partitionen genannt, eingeteilt werden. Bis zu 254 Partitionen können auf der HD erstellt werden. Partitionen können mit dem Dienstprogramm HD-TOOLS erstellt bzw. gelöscht werden.

Aktuelle Partition

Partition 0 (Null) hat eine Sonderbedeutung unter HD-DOS. Diese Partition 0 zeigt die aktuelle Partition an. Dies ermöglicht Kompatibilität mit Software, die ein "0:" innerhalb der Dateinamen oder Diskettenbefehle aussenden. Vor dem Versuch mit kommerziellen Software ist es ratsam, die HD in die von Ihnen erwünschte Partition zu versetzen. Dadurch findet der Dateizugriff nur innerhalb dieser Partition statt.

Native-Modus-Unterverzeichnisse

Die folgende Information dient als Einführung in die Native-Modus-Unterverzeichnisse und wie diese auf der HD gespeichert sind. Die dazugehörenden Befehle finden Sie im Befehl-Hinweis-Kapitel dieser Anleitung.

Native-Modus-Unterverzeichnisse sind den Unterverzeichnissen MS-DOS Computerarten in der Struktur ähnlich. Wenn Sie ein Unterverzeichnis kreieren, wird ein DIR Dateityp erstellt und dem Verzeichnis hinzugefügt. Unterverzeichnisnamen können, genau wie normale Dateinamen, 16 Zeichen lang sein. Diese Datei ist anfänglich 2 Blöcke groß und enthält einen Verzeichnis-Headerblock und ersten Verzeichnisblock. Beide Blöcke stehen nebeneinander in der gleichen Spur. Wenn dies nicht der Fall ist, wird kein Verzeichnis erstellt.

Dem Unterverzeichnis steht der gleiche Speicherplatz wie beim Mutterverzeichnis zur Verfügung. Alle Blöcke innerhalb einer Native-Modus-Partition werden von allen Verzeichnissen aller Partitionen geteilt. Dies bedeutet, daß wenn 62000 Blöcke in einer Partition frei sind, diese Anzahl freier Blöcke angezeigt werden, egal in welchem Verzeichnis Sie sich befinden. Wenn eine Datei mit 37 Blöcken in irgendein Verzeichnis gespeichert wird, zeigen alle Verzeichnisse innerhalb jener Partition 37 weniger Blöcke an.

Unterverzeichnisse können im Hauptverzeichnis oder innerhalb eines anderen Unterverzeichnisses (Verschachtelung genannt) erstellt werden. Der Menge der Verzeichnisse innerhalb einer Partition sind praktisch keine Grenzen gesetzt. Die einzige Begrenzung ist die Zahl der nebenstehenden Blöcke innerhalb der Partition. Praktische Begrenzungen gibt es natürlich, wenn man durch einen einzigen Befehl in verschiedene Unterverzeichnisse zugreifen will. Der Eingabepuffer der HD ist nur 254 Zeichen lang. Unterverzeichnisse, die zu tief verschachtelt sind, können beim Dateizugriff mehrere Befehle benötigen.

Kapitel 5

Der Gebrauch von GEOS

Die Einleitung

Dieses Kapitel führt Sie schrittweise in die Einstellung der Festplatte für den Gebrauch unter GEOS 2.0, GEOS 128 und das GEORAM-GEOS ein.

In diesem Kapitel lernen Sie, wie Sie die Konfigurationsdatei auf Ihrer Bootdiskette mit der auf der GEOS UTILITIES-Diskette ersetzen, die HDTIME-Datei auf Ihre Bootdiskette kopieren und QUICKMOVE verwenden

Dieses Kapitel bietet auch Hilfe an, wie Sie die Festplatte effektiv unter GEOS benützen können. Auch mögliche Fehler und eine Liste von Ratschlägen erscheinen am Ende des Kapitels.

Was Sie benötigen

Software:

- GEOS 2.0 (GEOS 128 2.0/ GEORAM GEOS)
- Die GEOS UTILITIES- Diskette von CMD

Hardware:

- Commodore 64 oder 128
- 1541 oder 1571 Floppy
- CMD HD
- Maus oder Joystick

Empfohlene Hardware

- Eine RAM-Erweiterung (z.B. RAMLink, RAMDrive, GEORAM, Commodore 17xx REU)
- Zusätzliche Laufwerke (z.B. CMD FD-2000, FD-400, CBM 1571, CBM 1581)
- Drucker

Installation von GEOS HD

Die Installation von GEOS HD ist sehr einfach. Trotzdem bitten wir Sie, diese Schritte aufmerksam durchzulesen.

1. Trennen Sie die HD von Ihrem Computer, schalten Sie ihn ein und booten Sie GEOS.
2. Entfernen Sie Ihre Bootdiskette vom Laufwerk A und legen Sie die GEOS UTILITIES- Diskette ein. Öffnen Sie Laufwerk A.
3. Von der Menüleiste wählen Sie durch Klicken "aufräumen" unter "Diskette." Jetzt wird die Diskette validiert. Falls Sie eine Errormeldung bekommen, machen Sie nicht weiter. Setzen Sie sich mit CMD oder ihrem Vertreiber in Verbindung. Sie erhalten eine neue Diskette.
4. Legen Sie Ihre Bootdiskette erneut in Laufwerk A ein und öffnen Sie sie. Sie müssen die Konfigurationsdatei von der Bootdiskette entfernen. Machen Sie aber zuerst davon eine Kopie!
5. Klicken Sie die Konfigurationsdatei einmal an, kurz warten, und dann wieder anklicken. Bewegen Sie das "Geister-Piktogramm" der Konfigurationsdatei an den Rand und legen Sie es ab.
6. Klicken Sie das Piktogramm wieder an, kurz warten, und wieder anklicken. Ziehen Sie das "Geister-Piktogramm" auf den Papierkorb und klicken Sie wieder. Die Datei ist jetzt gelöscht.
7. Entfernen Sie die Bootdiskette von Laufwerk A und legen Sie die GEOS UTILITIES-Diskette ein. Öffnen Sie sie. Sie sehen jetzt vier CONFIGURE-Dateien. Die Datei, die Sie brauchen entnehmen Sie aus zwei Faktoren:
 - A. Benützen Sie GEOS 128 oder GEOS 64.
 - B. Besitzen Sie eine RAM-Erweiterung, wenn ja, ist sie eine Commodore-Erweiterung oder ein GEORAM?

Wenn Sie GEOS 64 benutzen und **besitzen keine Erweiterung**, wählen Sie **CONFIGURE**. Wenn Sie eine **Commodore-Erweiterung** benutzen, wählen Sie auch **CONFIGURE**. Wenn Sie ein **GEORAM** benutzen, wählen Sie **CONFIGURE_r**.

Wenn Sie GEOS 128 benutzen und **keine Erweiterung** haben, wählen Sie **128 CONFIGURE**. Wenn Sie doch eine **Commodore-Erweiterung** benutzen, wählen Sie auch **128 CONFIGURE**. Wenn Sie ein **GEORAM** benutzen, wählen Sie **128 CONFIGURE_r**.

8. Nachdem sie die richtige CONFIGURE-Datei ausgesucht haben, klicken Sie sie einmal an, kurz warten, und wieder anklicken. Ziehen Sie das "Geister-Piktogramm" auf den Rand und klicken Sie wieder.
9. Entfernen Sie die GEOS UTILITIES-Diskette und legen Sie Ihre Bootdiskette wieder ein. Öffnen Sie die Diskette.
10. Klicken Sie die CONFIGURE-Datei am Rande an und ziehen Sie sie auf das Arbeitsblatt des Desktops. Klicken Sie jetzt wieder. Folgen Sie die Aufforderungen auf dem Bildschirm. Sie werden einige Male die Diskette wechseln müssen.
11. Wenn der Vorgang zu Ende ist, erscheint die neue CONFIGURE-Datei auf dem Desktop. **WICHTIG:** Falls Sie eine CONFIGURE_r-Datei übernommen haben, müssen Sie vor dem Booten, diese Datei auf KONFIGURIEREN bzw. 128 KONFIGURIEREN umbenennen!!

Damit Ihre HD unter GEOS richtig funktioniert, müssen Sie auch die Datei HDTIME auf Ihre Bootdiskette kopieren. HDTIME ist eine autoexec Datei. Wenn GEOS bootet, sorgt diese Datei dafür, daß die Festplatte sich in einer 1581-Partition befindet. Zusätzlich versieht sie die GEOS-Uhr mit der Zeit und dem Datum der HD-Echtzeituhr.

Schalten Sie jetzt Ihren Rechner aus und schließen Sie die HD an Ihren Computer wieder an.

Einstellung der HD unter GEOS

Nachdem Sie die neuen Dateien (CONFIGURE und HDTIME) auf Ihre Bootdiskette kopiert haben, können Sie jetzt die HD für den Gebrauch mit GEOS einstellen. Bevor Sie erneut GEOS booten, merken Sie sich folgende Beschränkungen:

- Zur Zeit werden nur 1581 Emulationspartitions unterstützt.
- CONFIGURES 1581 DIR-Schattenwahl ist nicht mehr vom 1581-Treiber unterstützt.
- GEOS erkennt nur die Geräteadressen 8, 9 und 10 an. Wenn Ihre HD eine andere Geräteadresse hat, kann GEOS sie nicht finden. Sie müssen entweder die SWAP 9 Taste betätigen oder die HD auf die 8, 9 oder 10 mit HD-TOOLS ändern.
- GEOS erwartet, daß Geräteadressen aufeinanderfolgend sind. In anderen Worten, bei einem Zweilaufwerkssystem ohne RAM-Erweiterung muß die HD-Geräteadresse 9 sein.
- Man kann der HD die Geräteadresse 8 nicht geben.

Schalten Sie die Festplatte ein und booten Sie GEOS. Höchstwahrscheinlich wird das Laufwerkspiktogramm für die HD nicht erscheinen.

1. Öffnen Sie die CONFIGURE-Datei mit einem Doppelklicken.
2. Wählen Sie als Laufwerk B ein 1581-Laufwerk. Das Activity-Lämpchen auf der HD leuchtet kurz auf.
3. Speichern Sie diese Konfiguration durch SAVE CONFIGURATION unter dem "File"-Menü.
4. Ihre Festplatte ist jetzt installiert und bereit für den Gebrauch mit GEOS. Wählen Sie jetzt "quit" unter dem "File"-Menü, und entdecken Sie die neuen Möglichkeiten, die Ihre HD anzubieten hat.

GEOS mit der Festplatte

Die Festplatte unter GEOS benimmt sich wie ein 1581-Laufwerk mit einer Ausnahme: Geschwindigkeit. Die CMD HD ist das schnellste Laufwerk, das es in Verbindung mit GEOS je gab. Applikationen und Datendateien laufen mit beeindruckender Geschwindigkeit. Eine RAM- Erweiterung (oder RAMLink) und CMDs HD beseitigen die letzten Zweifel über die Rentabilität der Anschaffung des GEOS-Systems. Niemand wird dann der Meinung sein, daß GEOS zu langsam ist!

Mit QuickMove bietet die Festplatte Zugriff zu allen erstellten 1581-HD-Partitionen. Einzelne Partitionen können einer einzigen Applikation gewidmet werden. Andere können Grafiken oder Zeichensätze enthalten. Mit QuickMove kann man mühelos zwischen Partitionen einzelne oder mehrere Dateien kopieren. Aus diesem Grund soll es eine Kopie von QuickMove und Desktop auf jeder 1581-Partition geben. Wenn Sie keine RAM-Erweiterung benutzen, kopieren Sie das CONFIGURE-Datei auch auf jede 1581-Partition.

Um das Kopieren der Dateien zu erleichtern, kopieren Sie alle drei auf eine Partition. Öffnen Sie das Datei 'QuickMove' und benutzen Sie das untenbeschriebene 'Multi-File' Verfahren um die Dateien auf die anderen Partitionen zu kopieren.

Ein versteckter Vorteil für die Besitzer eines normalen 1581-Laufwerks in Verbindung mit der Festplatte ist auch vorhanden. Besondere serielle Routine beschleunigen die Byteübertragungsgeschwindigkeit des 1581-Laufwerks auf 75% des originalen Berkeley-Treibers.

Der Gebrauch von QuickMove

QuickMove ist der eigentliche Schlüssel der CMD HD/GEOS-Verbindung. Leichter Zugriff zu allen 1581-Partitionen und Multi-File-Kopieren bis zu 9 Dateien werden durch QuickMove erledigt. Durch Doppelklicken wird QuickMove geöffnet. QuickMove holt dann Partitionsinformationen von der Festplatte. Auf Systemen mit einer RAM-Erweiterung werden diese Informationen auf einen unbenutzten Teil in die Erweiterung gespeichert.

Nachdem der QuickMove-Bildschirm sich darstellt, ist der vorhandene Partitionsname oben sichtbar. Unten sehen Sie eine Grafik der CMD Festplatte. Ein kurzes Menü oben links bietet zwei Möglichkeiten an: **geos** und **options**.

Das Options-Menü

In diesem Menü finden Sie 3 Wahlmöglichkeiten: **open partition**, **show directory**, und **quit** (z.D. Partition öffnen, Verzeichnis zeigen, und Beenden).

Open Partition (C= P)

Diese Option zeigt eine Dialogbox in der Mitte des Bildschirms. Die ersten 7 Partitionen werden dargestellt. Ein Pfeil zeigt die erste Partition an. Um eine andere Partition auszuwählen, fahren Sie mit der CRSR DWN Taste hinunter oder klicken Sie einfach die gewünschte Partition an. Drücken Sie jetzt RETURN oder klicken Sie OKAY an. Die CURRENT PARTITION- Box zeigt die gewählte Partition an. Sie können auch die ersten 9 1581-Partitionen durch Drücken der entsprechenden Zahlentasten auswählen.

Durch CANCEL oder C=Q brechen Sie jeden Vorgang ab. Sie kehren dann zum ersten QuickMove-Fenster zurück.

Wenn Sie eine neue Partition öffnen, befinden Sie sich in der "Show Directory"-Option.

Show Directory (C=D)

Diese Option ruft eine Dialogbox hervor, die die ersten 7 Dateien des Verzeichnisses zeigt. Unten stehen 6 Piktogramme, die leichte Bewegung innerhalb des Verzeichnisses erlauben. Von links nach rechts bedeuten die Piktogramme:

- bis zum Ende des Verzeichnisses bewegen
- bis zum Anfang des Verzeichnisses bewegen
- eine Datei hinauf scrollen
- eine Datei hinunter scrollen
- eine ganze Seite (7 Dateien) nach unten bewegen
- eine ganze Seite nach oben bewegen

Ein Pfeil zeigt die gewählte Datei an. Den Pfeil bewegt man gleich wie oben.

Wenn Dateien im Verzeichnis vorhanden sind, stehen rechts von der Dialogbox 2 Optionen: **COPY** und **COPY TO**. Um eine Datei kopieren zu können, klicken Sie sie an oder drücken Sie C= C. Oberhalb COPY lesen Sie jetzt **1 file(s) selected** (1 Datei(en) gewählt). Bis zu 9 Dateien können so gewählt werden. Zur Zeit kann man eine falsch gewählte Datei nur durch CANCEL löschen.

Um die Zielpartition zu wählen klicken Sie **COPY TO** oder drücken Sie C= K. Die Open Partition-Dialogbox erscheint. Suchen Sie sich die Zielpartition aus und klicken Sie OKAY an. Während des Kopierens wird der Name der gerade kopierten Datei oben sichtbar sein.

Wenn die Zielpartition keine GEOS-Partition ist, werden Sie gefragt, ob diese ins GEOS-Format konvertiert werden sollte. Egal ob Sie JA oder NEIN wählen, werden die Dateien auf die Zielpartition kopiert.

Falls die Datei auf der Zielpartition vorhanden ist, werden Sie gefragt ob die Datei überschrieben werden sollte. Falls Sie NEIN antworten und andere Dateien zu kopieren haben, wird diese Datei einfach übersprungen. Wenn das Kopieren fertig ist, befinden Sie sich in der Zielpartition.

GEOS von der HD booten

Wenn Sie ein Programm besitzen, das eine 1581-Bootdiskette erstellen kann, können Sie GEOS von der Festplatte booten.

Es gibt zwei Möglichkeiten, eine GEOS Boot-Partition zu erstellen. Ihre GEOS-Original-Systemdiskette müssen Sie aber zuerst mit den HD-Dateien, wie vorher beschrieben, versehen. Nachdem Ihre Bootdiskette mit der HD richtig funktioniert, können Sie Ihr Programm benutzen, um eine 1581-Boot-Partition zu erstellen. Uns ist es mit MAVERICK V3 und V4 gelungen, eine 1581-Boot-Partition zu erstellen.

Folgendes Verfahren erlaubt das Booten von GEOS auf der Festplatte:

1. Schalten Sie Ihr System ein.
2. Schalten Sie Ihre HD durch den Change Partition Befehl (siehe Befehl-Hinweis- Kapitel) in die 1581-Partition um, die Sie für die Boot-Partition bestimmt haben.
3. Betätigen Sie die SWAP 9-Taste, damit die HD auf Geräteadresse 9 eingestellt wird.
4. Booten Sie Ihr Kopier-Programm und befolgen Sie seine Anweisungen.
5. Die Festplatte ist das 1581-Ziellaufwerk.
6. Nach Beenden des Kopiervorganges, booten Sie GEOS von Ihrem normalen Laufwerk und kopieren Sie die HD-Dateien auf die neu erstellte GEOS-Boot-Partition der HD (CONFIGURE, HDTIME, und QUICKMOVE).

Wenn Sie das nächste Mal GEOS booten, drücken Sie die SWAP 8 Taste an der HD, schalten Sie auf die Boot-Partition um und laden Sie GEOS.

Wenn es Ihnen nach diesem Vorgang nicht gelingt, eine Boot-Partition direkt auf der HD zu erstellen, müssen Sie zuerst auf einem normalen 1581-Laufwerk eine Boot-Diskette erstellen und dann den Inhalt der Boot-Diskette mit dem Dienstprogramm MCOPY auf die Boot-Partition der Festplatte kopieren. Die originale Systemdiskette, die der Festplatte angepaßt ist, müssen Sie unbedingt verwenden. Befolgen Sie dann den obigen Anweisungen unter Punkt 6.

GEOS Ratschläge

- Kopieren Sie ruhig Ihre GEOS-Applikationen auf die HD. Sie ist 100% kompatibel mit allen GEOS-Applikationen.
- Teilen Sie uns bitte Ihre Vorschläge oder Probleme mit.
- Niemals die RESET-Taste innerhalb GEOS drücken! Ihr System wird garantiert abstürzen!
- Gleichfalls die SWAP-Tasten nie innerhalb GEOS drücken.
- Falls Sie unter GEOS Probleme mit der HD haben, rufen Sie nicht Markt & Technik oder Berkeley Softworks an. Rufen Sie entweder uns oder einen von unseren Vertreibern an!

Kapitel 6

CP/M mit der HD

Der Gebrauch der HD mit CP/M

Die CMD HD unterstützt derzeit CP/M Plus V3.0. Diese Version ist am 28. Mai 1987 herausgekommen. Frühere Versionen - obwohl sie nicht getestet wurden - sollten mit 1541- bzw. 1571- Partitionen arbeiten. Mit CP/M kann man auf der Festplatte arbeiten, wenn man folgende Partitionsformate benutzt:

1541	-	C128 einseitig
1571	-	C128 beidseitig
1581 CP/M	-	Commodore 1581 (beidseitig)

Die Installation von CP/M

Wie oben gelistet, ist der Gebrauch der HD mit CP/M auf Commodore-Disk-Emulation begrenzt. Die Menge des Speicherplatzes, die CP/M zur Verfügung steht, wird durch das Dienstprogramm SPORT.COM erweitert. Dadurch kann man sich zwischen verschiedenen Partitionen bewegen. Sie können so viele 1581-Partitionen erstellen, wie Sie auf der HD Platz haben.

Folgende Anweisungen zeigen, wie Sie eine CP/M-Boot-Partition auf der HD erstellen können. Sie dürfen alle oben gelisteten Partitionen dafür benutzen. Wir empfehlen aber eine 1581- CP/M Partition.

1. Geben Sie der Festplatte die Geräteadresse 9, 10 oder 11. Benutzen Sie dafür entweder das 'HD-TOOLS'-Programm oder die SWAP-9 Taste.
2. Bevor Sie CP/M PLUS V3.0 booten, stellen Sie die Partition ein, die Sie für CP/M bestimmt haben.
3. Booten Sie vom Laufwerk 8 Ihre CP/M PLUS V3.0 Systemdiskette. (Geben Sie einfach BOOT ein und drücken Sie RETURN.
4. Auf A> geben Sie den FORMAT-Befehl ein und drücken Sie RETURN.
5. Nachdem die FORMAT.COM-Datei geladen ist, erscheint am Bildschirm ein Menü. Aus diesem Menü wählen Sie "Select Different Drive" und drücken Sie RETURN.
6. Jetzt müssen Sie sich einen Laufwerksdesignator für die HD auswählen. Laufwerk A ist für Geräteadresse 8, B ist für Geräteadresse 9, C ist für Geräteadresse 10, und Laufwerk D ist für Geräteadresse 11. Wählen Sie den passenden Designator und drücken Sie RETURN.
7. CP/M teilt Ihnen jetzt mit, welcher Laufwerkstyp selektiert wurde, und präsentiert Ihnen einige Formattyp-Möglichkeiten. Wählen Sie den Formattyp gemäß der obigen Liste. Wenn Sie eine 1581-CP/M-Partition wählen, müssen Sie das ganze Format benutzen. Drücken Sie zum Schluß RETURN.
8. CP/M verlangt nach der nicht formatierten Diskette. Diese Anweisung ignorieren Sie einfach und drücken Sie die SCHIFT-Taste, halten Sie sie und drücken Sie die Zahl taste 4. CP/M formatiert jetzt die Partition. Falls Sie eine 1541-Partition formatiert haben, kann es vorkommen, das CP/M eine Datei offen läßt. In diesem Fall bleibt das obere ACTIVITY-Lämpchen leuchtend.

Falls Sie eine Kopie der CP/M Systemdatei auf die HD kopieren möchten, machen Sie bei Punkt 9 weiter. Andernfalls können Sie diese Partition sofort benutzen. Kopieren Sie das Programm SPORT.COM auf diese Partition, wenn Sie vorhaben, mehrere Partitionen unter CP/M zu gebrauchen. Anweisungen dafür werden später angeführt.

9. Kopieren Sie die CP/M Systemdatei von Ihrer Bootdiskette auf die HD-Partition mit dem Programm PIP.COM. Folgendes Beispiel zeigt, wie dies gemacht wird. Die Festplatte in diesem Beispiel hat die Geräteadresse 9, also Laufwerk B. Ändern Sie B auf C, wenn Ihre HD die Nummer 10 hat; D, wenn sie die Nummer 11 hat.

```
A>PIP B:=A:CPM+.SYS
```

10. Kopieren Sie die CP/M Befehlsdatei CCP.COM auf die Festplatte. Mit der Festplatte auf Geräteadresse 9 eingestellt, sieht der Befehl so aus:

```
A>PIP B:=A:CCP.COM
```

Sie können auch alle COM-Befehle auf die HD kopieren.

```
A<PIP B:=A:*.COM
```

Jetzt können Sie CP/M von der HD booten. Sie müssen nur noch die SWAP 8-Taste betätigen.

Partitionswechsel unter CP/M

Mit dem Programm SPORT.COM können Sie einen Partitionswechsel auf der HD durchführen. Diese Datei befindet sich auf der mitgelieferten CP/M UTILITIES-Diskette. Kopieren Sie diese Datei auf alle CP/M-Partitionen. Gebrauchen Sie dafür das Programm 'PIP'.

SPORT.COM erlaubt das Senden von Commodore-DOS Befehlen an alle seriellen Geräte. 'SPORT' wird als Befehl benutzt, mit bis zu 3 Befehl-Parametern. Diese Parameter zeigen Geräteadresse, Sekundäradresse und Befehl. Wenn der Befehl ohne Parameter ausgeführt wird, erscheint ein Hilfsfenster. Hier die Syntax:

```
SPORT [.COM] [/Ddv] [/Ssa] [/C'command']
```

wo: dv	= die Geräteadresse der Zieldiskette ist
sa	= die Sekundäradresse des Befehlkanals ist
command	= die Befehlschleife sind.

'SPORT' funktioniert in der gleichen Weise, wie die meisten DOS-Wedges. Hier ist ein Beispiel, wie Sie die Partionen auf der HD (als Geräteadresse 9 eingestellt) tauschen können:

```
SPORT /D9 /SF /C'CP5'
```

Durch diesen Befehl wird die Partitionsnummer 5 aktiv. In einer neuen Partition sollten Sie die CONTROL-C-Tastenkombination betätigen, um die Partition anzumelden. Wenn Sie sich, zum Beispiel, im Laufwerk A befinden, geben Sie "B:" ein und drücken Sie RETURN. Jetzt drücken Sie zugleich die CONTROL und die C-Tasten. Die Partition ist jetzt angemeldet und funktioniert richtig.

Kapitel 7

Kommerzielle Software mit der HD

Arten von kommerzieller Software

Es gibt viele verschiedene Arten von kommerziellen Programmen, die wir einfach in zwei Hauptgruppen einteilen werden: geschützte und ungeschützte. Die meisten kommerziellen Programme - mit der Ausnahme von Spielen - sind mit der HD kompatibel. Mit Hilfe dieses Kapitels können Sie diese Programme auf die Festplatte speichern und sie verwenden.

Software ohne Kopierschutz

Diese Software läßt sich ohne weiteres auf die HD installieren. Die einzige Frage betrifft die Partitionart, die Sie verwenden sollen. Am besten experimentieren Sie, beginnend mit einer Native-Modus-Partition. Erstellen Sie zuerst für die Dateien ein Native-Modus-Unterverzeichnis. Gebrauchen Sie dafür den MAKE DIRECTORY-Befehl. Kopieren Sie dann mit FCOPY alle Dateien von der Programm-Diskette ins Unterverzeichnis. Testen Sie anschließend das Programm, indem Sie die Festplatte auf Geräteadresse 8 umschalten (SWAP 8), und die richtige Partitionnummer und das richtige Unterverzeichnis einstellen. Falls die Software sich vom Unterverzeichnis nicht laden läßt, versuchen Sie das Laden vom Hauptverzeichnis der Native-Modus-Partition.

Manche Software funktioniert nur mit einem bestimmten Laufwerk. In diesem Fall müssen sie eine Emulations-Modus-Partition verwenden. Versuchen Sie dies zuerst mit einer 1581-Partition, dann mit einer 1571, und zum Schluß mit einer 1541. Vor dem Testen müssen Sie den CHANGE PARTITION-Befehl eingeben um die neue Partitionnummer zu ändern.

Falls die Software immer noch nicht funktioniert, kann das Problem bei "versteckten" Daten liegen. In diesem Fall kopieren Sie mit MCOPY die ganze Diskette auf eine HD-Partition. In seltenen Fällen hilft auch dieser Vorgang nicht. Dabei handelt es sich um ein Dienst-Programm, das spezifische Laufwerksaufgaben durchführt.

Mehr-Disketten-Software

Manche Software werden auf mehreren Disketten geliefert. Kopieren Sie mit FCOPY alle Dateien auf eine einzige Partition. Seien Sie vorsichtig mit Dateinamen, die gleich sind. Sie dürfen nicht identisch sein! Diese Methode funktioniert, wenn alle Dateien standard sind und das richtige Laufwerk (die richtige Emulationspartition) angesprochen wird.

Wenn manche Programminformationen - ohne Namen - direkt auf einer Diskette gespeichert sind, oder der Diskettenname vom Programm überprüft wird, müssen separate Partitionen für jede Diskette benützt werden. Diese Partitionen müssen die gleichen Namen wie die originalen Disketten haben und Sie müssen mit MCOPY die ganze Diskette auf die HD-Partitionen kopieren.

Manchmal hilft eine Kombination von Methoden. Ein Textverarbeitungsprogramm, zum Beispiel, bestehend aus 4 Disketten (3 Standard-Datei-Disketten und 1 Sonder-Datei als Wörterbuch) könnten Sie nach dieser Weise auf die HD bringen: Die 3 Standard-Disketten speichern Sie in eine Native-Modus-Partition und die vierte in eine Emulations-Partition. Die ersten 3 Disketten kopieren Sie mit FCOPY und die vierte mit MCOPY.

Wenn ein Programm in verschiedene Partitionen gespeichert werden muß, benützen Sie den FPPS (Front-Seite-Partition-Wahl) an der HD und wechseln Sie die Partition, wenn das Programm danach fragt. Andere Programme lassen Laufwerksbefehle zu. Hier können Sie den "CP"-Befehl (Partition ändern) dafür einsetzen.

Weitere Lösungen

Manche Software erwarten, daß das Verzeichnis einen bestimmten Platz auf der Diskette einnimmt. Normalerweise genügt hier der Gebrauch von einer Emulationspartition. In einem Native-Modus-Unterverzeichnis kann man das Verzeichnis eines 1541- bzw. 1581-Laufwerks simulieren. Diese Unterverzeichnisse kann man mit den Dienstprogrammen 1541SUB und 1581SUB erstellen. Diese Programme erweitern die Menge des brauchbaren Diskettenplatzes mit Programmen wie Superbase. Einige dieser Programme untersuchen die Diskettenspuren oder Sektoren und stellen dabei die Art des Laufwerkes fest. Die Entscheidung, welches Unterverzeichnis man emulieren soll, kann daher ein wenig schwierig sein. So lange Sie eine Native-Modus-Partition mit 40 oder mehr Spuren

benützen, werden 1581-Emulations- Unterverzeichnisse immer funktionieren. Andere Programme werden die vorhandene Partition als eine 1541 annehmen, egal wie viele Spuren die Partition besitzt.

In der Software eingebaute Floppybeschleuniger (z.B. V-MAX!) können auch problematisch sein. Wenn Sie diese Software-Funktion nicht ausschalten können, können Sie diese Software auf der HD nicht anwenden.

Kopiergeschützte Software

Kopiergeschützte Software können Sie normalerweise nicht direkt auf die HD abspeichern. Leicht geschützte Programme können vielleicht mit MCOPY auf die Festplatte kopiert werden. Diese sind meistens Programme, die auf der 1581 laufen.

Wenn ein Programm sich von der HD nicht laden läßt, können Sie vielleicht die Daten- dateien mit FCOPY auf die Festplatte kopieren und das Hauptprogramm dann von dem normalen Laufwerk laden. Nachdem das Programm geladen ist, drücken Sie SWAP 8. Diskettenzugriff findet jetzt auf der HD statt.

Bei manchen Programmen findet kein Diskettenzugriff nach dem Laden statt. Hier kann man kaum die Festplatte anwenden. Sie müssen den Schutz zuerst mit Hilfe eines entsprechenden Programmes entfernen. So kann man auch bootfähige Kopien von allen Software machen.

Kapitel 8

Befehlshinweis

Befehlssyntax

Dieses Kapitel dokumentiert viele der HD-Befehle. Die Syntax dieser Befehle sind in einem Standardformat angegeben. Die notwendigen und wahlfreien Parameter sind daher leicht erkennbar. Beispiele werden überall angeführt. Wenn Sie Probleme haben, lesen Sie erneut folgende Informationen.

Befehlsschleife-Elemente

Die Befehlsschleife enthält verschiedene Elemente. Bei Befehlen, die direkt aus BASIC kommen, ist der erste Teil der Schleife meistens der Befehl selber. Bei Befehlen, die über den Laufwerksbefehlskanal kommen, ist der Befehl selber am Anfang der Schleife zu finden. Die Elemente der Befehlsschleife in dieser Anleitung sind folgende:

Literals sind Zeichen, die wie gezeigt, genau eingegeben werden müssen. Sie haben einen Text.

User-ergänzt-Werte sind Werte, die der User eingeben muß. Diese Werte und Typen hängen mit einem bestimmten Befehl zusammen. Diese Werte erscheinen in *Schrägschrift*.

Freie Parameter und **Optionen** sind Werte oder Literals, die nicht unbedingt benötigt werden, aber vom User erwünscht sind. Wenn diese weggelassen werden, gibt es vorgegebene Parameter, die automatisch eingestellt werden. Freie Parameter erscheinen in eckigen Klammern [].

Wahl-Parameter erlauben oder verlangen die Wahl von mehr als einem Parameter innerhalb der Befehlsschleife. Wenn diese Parameter vorhanden sind, erscheinen sie in geschweiften Klammern { } und bei mehreren Parametern durch einen vertikalen Strich (|) getrennt. Nur ein Wahl-Parameter darf in einer Befehlsschleife gebraucht werden.

Ab und zu sehen Sie eine Syntax, in der ein (odere mehrere) Element(e) von drei Punkten (...) gefolgt wird (werden). Diese bedeuten, daß der zuletzt gezeigte Parameter wiederholt werden darf. Weitere Einzelheiten lesen Sie im folgenden Text.

Befehlsschleife-Beispiel

Folgendes Beispiel illustriert die Befehlssyntax, die in dieser Anleitung gebraucht wird:

```
HEADER"partitionsname"[ ,Id ][ ,Dn ][ {ON| , }Udv]
```

In dieser Befehlsschleife ist HEADER der Literalbefehl. Die Beistriche zusammen mit I, D, ON und U sind freie Literals innerhalb eckiger Klammern. Die Parameter 'Partitionsname', id, n, und dv sind user-ergänzte Variablen. Diese Variablen werden in einer Tabelle nach der Befehlssyntax beschrieben. Die Tabelle für die obige Befehlsschleife würde so aussehen:

wo:	partitionsname	=	der von Ihnen ausgewählte Partitionsheader ist
	id	=	ein zweistelliges Zeichen für den Partitionsheader ist
	n	=	die Partitionsnummer ist, welche Sie formatieren wollen (0 oder 1)
	dv	=	die Geräteadresse der Festplatte ist

Weil diese Parameter alle in Klammern sind, sind sie Optionsparameter. Wenn Sie sich für einen oder mehrere Parameter entscheiden, merken Sie sich, daß sie von einem Literal begleitet sind. Das Literal muß in der gezeigten Reihenfolge zusammen mit der user-ergänzten Variable stehen. Merken Sie sich auch, daß 'ON' und der Beistrich innerhalb geschweiften Klammer und durch einen vertikalen Strich getrennt sind. Dies bedeutet, daß wenn Sie die Geräteadresse zufügen wollen, muß vorher entweder das Literal 'ON' oder ein Beistrich stehen. Hier ist der vollständige Befehl:

```
HEADER"PARTITION 1",IP1,D0,U12
```

Weil die Partition-(oder Laufwerk) Null(0:) in BASIC 7.0 sowieso angenommen ist, kann man diesen Befehl, wie folgendes kürzen:

```
HEADER"PARTITION 1",IP1,U12
```

Befehlsschleifen-Pfade

In dieser Anleitung sehen Sie Befehle, die in der Befehlssyntax als [path] (z.D. Pfad) vorkommen. Pfade gebraucht man nur, wenn auf Native-Modus-Partitionen zugegriffen werden. Sie bestimmen in welchem Unterverzeichnis der Zugriff stattfinden soll. Hier ein Beispiel:

```
VERIFY"[[n][pfad]:]dateiname",dv[,sa]
```

Die meisten gewöhnlichen HD-Befehle erlauben den Gebrauch von einem Unterverzeichnis-Pfad innerhalb der Befehlsschleife, vorausgesetzt, daß sich die Zieldatei innerhalb einer Native-Modus-Partition befindet. Dieser Pfad folgt unmittelbar der Partitionsnummer innerhalb der Befehlsschleife und wird in dieser Anleitung innerhalb der Befehlssyntax als [Pfad] angeführt. Wenn mehrere Unterverzeichnisse sich innerhalb einer Befehlsschleife befinden, wird jedes Unterverzeichnis durch einen Schrägstrich / eingeklammert, und dem letzten Schrägstrich folgt ein Doppelpunkt.

Wenn Sie, zum Beispiel, eine Datei mit dem Namen COPY in einem Unterverzeichnis namens UTILITIES in Partition 1 hätten, könnten Sie diese Datei mit folgendem Befehl laden:

```
LOAD"1/UTILITIES/:COPY",12
```

Der Teil dieses Befehls, der den Pfad enthält, ist: /UTILITIES/.

Wenn Sie verschachtelte Unterverzeichnisse haben, und die Datei COPY sich in einem Unterverzeichnis namens COPIERS befindet, die sich auch in einem Unterverzeichnis mit dem Namen UTILITIES in Partitionsnummer 1 befindet, können Sie diese Datei so laden:

```
LOAD"1/UTILITIES/COPIERS/:COPY",12
```

Der Teil dieses Befehls, der den Pfad enthält, ist: /UTILITIES/COPIERS/.

Wenn die Festplatte sich schon in Partitionsnummer 1 befindet, und das Hauptverzeichnis das aktuelle ist, können Sie die Partitionsnummer in der Befehlsschleife weglassen:

```
LOAD"/UTILITIES/COPIERS/:COPY",12
```

Wenn die vorhandene Partition Partitionsnummer 1 ist, und das vorhandene Verzeichnis UTILITIES ist, schreiben Sie einfach:

```
LOAD"/COPIERS/:COPY",12
```

Wenn Ihre vorhandene Partition die Nummer 2 ist, aber das vorhandene Verzeichnis in Partition 1 UTILITIES ist, schreiben Sie einfach:

```
LOAD"1/COPIERS/:COPY",12
```

Wenn die vorhandene Partition, Partition 2 ist, aber das vorhandene Verzeichnis in Partition 1 COPIERS ist, schreiben Sie:

```
LOAD"1:COPY",12
```

Es gibt eine weitere Syntax, die den Beginn ihres Pfades immer im Hauptverzeichnis erlaubt. Dies hilft vor allem, wenn Sie sich innerhalb eines anderen Unterverzeichnisses der gleichen Partition befinden. Wenn das vorhandene Verzeichnis in Partition 1, zum Beispiel, GAMES heißt und Sie das oben angeführte Programm COPY laden möchten, können Sie ihren Pfad mit zwei Schrägstrichen beginnen:

```
LOAD"//UTILITIES/COPIERS/:COPY",12
```

Wenn zwei Schrägstriche am Anfang eines Unterverzeichnissespfades stehen, fängt der Pfad beim Hauptverzeichnis an. Wenn Sie sich in einer anderen Partition befinden, oder nicht sicher sind, welches Verzeichnis in der Partition vorhanden ist, das sie verwenden möchten, ist es empfehlenswert zwei Schrägstriche zu verwenden. Wenn Sie sich in einer anderen Partition befinden, vergessen Sie nicht die neue Partitionsnummer zu benutzen.

```
LOAD"1//UTILITIES/COPIERS/:",12
```

Unterverzeichnispfade mit JiffyDOS-Befehlen

Die oben gezeigten Beispiele sind die normalen Commodore-DOS Ladebefehle. Falls Ihr Computer mit JiffyDOS ausgestattet ist, können Sie die JiffyDOS Ladebefehle verwenden. So würden die oben angeführten Befehle mit JiffyDOS aussehen:

```
/"1/UTILITIES/:COPY",12
/"1/UTILITIES/COPIERS/:COPY",12
/"/UTILITIES/COPIERS/:COPY",12
/"/COPIERS/:COPY",12
///UTILITIES/COPIERS/:COPY
/1//UTILITIES/COPIERS/:COPY
```

Sie dürfen auch die Anführungszeichen, den Beistrich und die Geräteadresse unter JiffyDOS auslassen, indem Sie die CONTROL-D Tasten drücken. Dadurch wird das vorgegebene Gerät geändert. Die letzten zwei Beispiele zeigen diese Option. Das vorletzte Beispiel zeigt auch die einzige Situation, in der drei Schrägstriche hintereinander stehen können. Der erste Schrägstrich ist der JiffyDOS-BASIC-Ladebefehl. Ihre JiffyDOS Anleitung gibt Ihnen darüber mehr Informationen.

Befehle von BASIC Senden

Die meisten an die HD gesandten Befehle sind BASIC-Befehle. Wir empfehlen BASIC 2.0 oder DOS Befehlskanal-Befehle wenn Sie mit einem C-128er arbeiten, weil BASIC 7.0 den Gebrauch von Partitionsnummern in seiner Befehlssyntax begrenzt. Der Geräteadresseparameter in BASIC 7.0 kann nur durch eine Null oder eine Eins dargestellt werden. Die HD gebraucht diese Nummern als ein Hinweis dafür, welche Partition für einen bestimmten Befehl gebraucht wird. Weil die HD bis zu 254 Partitionen haben kann, kann es vorkommen, daß die BASIC 7.0-Befehle die gewünschte Partition nicht ansprechen können. BASIC 7.0-Befehle lassen auch keine Unterverzeichnispfade zu. Wenn Sie Unterverzeichnispfade benutzen wollen, verwenden Sie die BASIC 2.0- oder DOS Kanal-Befehle.

Der Befehlskanal

Viele der Befehle in diesem Kapitel benötigen das Senden des Befehls über den Befehlskanal des Laufwerks. Um einen Kanal zum Laufwerk zu öffnen, müssen Sie folgenden BASIC-Befehl eingeben:

```
OPENlf,dv,sa
```

wo: lf = die logische Dateinummer ist
 dv = die Geräteadresse ist und
 sa = die Sekundäradresse ist

Die logische Dateinummer kann irgendeine Zahl zwischen 1 und 127 sein. Andere Zahlen sind auch "legal" (128-255), verursachen aber unerwünschte Nebenwirkungen.

Die Geräteadresse ist die gleiche Nummer, die der HD zugeteilt wurde. Diese ist von der Hersteller als 12 zugeteilt worden, kann aber mit HD-TOOLS oder durch die SWAP-Tasten geändert werden.

Die Sekundäradresse ist die Kanalnummer. Adressen von 0 bis 14 werden benutzt, um Dateien zu öffnen. Adresse 15 teilt dem Laufwerk mit, daß Daten, die über diesen Kanal gesendet werden, als Befehle verstanden werden. Kanal 15 ist daher der Befehlskanal. Dieses Beispiel zeigt, wie der Befehlskanal geöffnet wird:

```
OPEN15,12,15
```

Der Befehlskanal darf auch innerhalb eines Programms oder im Direktmodus geöffnet werden. Wann immer Sie einen Befehl eingeben, der sofort ausgeführt werden soll, ohne vorher eine Zeilennummer einzugeben, ist dieser Befehl in BASIC-Direkt-Modus einzugeben. Hier ein Beispiel:

```
OPEN15,12,15:PRINT#15,"I":CLOSE15
```

Diese Art von Befehlen kann man auch ohne PRINT# eingeben.

```
OPEN15,12,15,"I":CLOSE15
```

Andere Befehle verlangen, daß andere Parameter als Zeichenschleifen mit CHR\$(-Funktionen ans Laufwerk gesandt werden. Diese Befehlsarten können mit dem PRINT#-Befehl gesandt werden. Hier ein Beispiel:

CMD Hard Drive Anleitung

```
OPEN15,12,15:PRINT#15,"U0>";CHR$(10):CLOSE15
```

Der Strichpunkt zwischen dem Befehlsteil in Anführungszeichen und dem Zeichenschleifecode ist wahlfrei. Bei einigen Befehlen müssen numerische Werte oder wirkliche Zahlen als Befehlsparameter benutzt werden. Dies ist bei Direktzugriffsbefehlen meistens der Fall. Hier ein Beispiel im U1 Befehl:

```
OPEN15,12,15:PRINT#15,"U1";2;0;1;34:CLOSE15
```

Die gezeigten Zahlen hinter den Strichpunkten können auch als Variablen eingegeben werden:

```
OPEN15,12,15:PRINT#15,"U1";C;D;T;S:CLOSE15
```

Zahlen können auch innerhalb des Schleifeteils verwendet werden, wenn jede Zahl von der Schleife und von den anderen Zahlen durch einen Leerraum getrennt wird.

```
OPEN15,12,15:PRINT#15,"U1 2 0 1 34":CLOSE15
```

In diesem Fall steht das End-Anführungszeichen hinter der letzten Zahl oder dem letzten Parameter. Sie dürfen wahlweise einen Doppelpunkt am Ende des Befehls stellen, egal wie der Befehl gebraucht wird.

```
OPEN15,12,15:PRINT#15,"U1:";2;0;1;34:CLOSE15
```

```
OPEN15,12,15:PRINT#15,"U1: 2 0 1 34":CLOSE15
```

Diskettenfehler-Lesen

Diskettenfehler werden durch das Lesen des Befehlskanals entdeckt. Die GET#- oder INPUT#- Befehle werden dafür eingesetzt. INPUT# ist die schnellere Methode. GET# wird meistens für das Holen von Nicht-Fehler-Informationen benutzt. Beide Befehle brauchen den BASIC-INPUT-Puffer und können daher im Direkt-Modus nicht gebraucht werden.

Programme suchen normalerweise nach Fehlern sofort nach einem Diskettenzugriffsversuch oder Diskettenbefehl. Hier haben Sie ein kurzes Programm, welches zeigt, wie der Fehlerkanal gelesen wird:

```
10 OPEN15,12,15:INPUT#15,E,E$,T,S:CLOSE15
20 PRINTE,E$,T,S
```

Wie Sie sehen, werden 4 Parameter über den Befehlskanal nach einer Fehlersuche zurückgesandt. Diese sind, der Reihenfolge nach: Fehlernummer, Fehlermeldung, die Spur und der Sektor, wo der Fehler passierte. Viele Fehler finden in keiner Spur oder keinem Sektor statt. In diesem Fall stehen in den Spur- und Sektorvariablen Nullen.

Manchmal ist es sinnvoller Fehlerdaten oder andere Informationen vom Befehlskanal bytewise zu holen. In diesem Fall können Sie so ein Programm einsetzen:

```
10 OPEN15,12,15
20 GET#15,E$:PRINTE$;:IFST<>64THEN20
30 CLOSE15
```

Wenn Sie JiffyDOS dafür anwenden, können Sie den Fehlerkanal ohne Programm lesen. Drücken Sie einfach die @ Taste und dann RETURN.

Im vorherigen Beispiel haben wir die Statusvariable benutzt, um festzustellen wann das Ende der Datei erreicht wird. Die BASIC-Statusvariable ST ist bei serielltem Gerätezugriff sehr nützlich. Deshalb zeigen wir hier die Aufstellung der einzelnen Bitwerte. Die Beschreibungen treffen dort zu, wo das spezifizierte Bit eingestellt ist, oder eins gleich ist.

BIT	STATUSBESCHREIBUNG	STATUSWERT
7	Gerät nicht angeschlossen/Kassette zu Ende	128
6	Dateiende (EOI)	64
5	Kassette-Kontrollsumme-Fehler	32
4	Verifizierungsfehler/Kassetten-Lesefehler	16
3	Daten-Block zu lang (Kassette)	8
2	Daten-Block zu kurz (Kassette)	4
1	Hörerzeit abgelaufen	2
0	Sprecherzeit abgelaufen	1

Durch die BASIC 7.0-Variablen DS und DSS bietet der C-128er eine andere Methode an, Diskettenfehler zu finden. Die Variable DS zieht die Fehlernummer, DSS die Fehlermeldungsschleife an. Diese Variablen kann man mit einem PRINT-Befehl anschauen.

```
PRINTDS,DSS
```

Es ist nicht notwendig beide Variablen gleichzeitig einzugeben, aber die Fehlermeldungsschleife hilft zu verstehen, warum der Fehler passiert ist. Der Fehler kann dann im Anhang B nachgeschlagen werden.

Partitionsnummern in Dateinamen

Innerhalb eines Dateinamens kann eine Partitionsnummer anstelle der Laufwerksnummer spezifiziert werden. Die Laufwerksnummer ist die Nummer, die vor dem Doppelpunkt (:) im Commodore-DOS-Dateinamen steht. Diese Nummer wird oft weggelassen, weil sie nur dann gebraucht wird, wo ein zweites Laufwerk angeschlossen ist. Wo sie weggelassen wird, wird Laufwerk 0 angenommen. Diese Regel trifft auch bei der Festplatte zu, wobei anstatt 2 Laufwerke ein einziges Laufwerk mit bis zu 254 Partitionen vorhanden ist. Wann immer Sie mit der aktuellen Partition einen Befehl oder Arbeitsgang durchführen, können Sie 0 verwenden oder die Partitionsnummer ganz weglassen. Wenn Ihre vorhandene Partition anders ist als die Partition, in der Sie den Arbeitsgang durchführen wollen, müssen Sie entweder zuerst in diese Partition wechseln oder die Partitionsnummer innerhalb des Dateinamens spezifizieren.

```
LOAD "1:MCOPY",12
OPEN2,12,2,"3:TESTFILE,S,W"
```

Ladebefehle können mit JiffyDOS auch abgekürzt werden:

```
/1:MCOPY
```

Partitionsnummern in Diskettenbefehlen

Partitionsnummern können Laufwerksnummern in Dateinamen ersetzen. Partitionsnummern können auch in dieser Weise eingesetzt werden, indem man Diskettenbefehle eingibt. Eigentlich kann jeder Befehl, der das Einschließen einer Laufwerksnummer erlaubt (Direkt-Zugriffsbefehle ausgenommen), mit einer Partitionsnummer ersetzt werden. Sie dürfen beim Formatieren, Kopieren, Umbenennen, Löschen, Validieren und Initialisieren Partitionsnummern benutzen. Die Kompatibilität der HD mit Commodores DOS liegt größtenteils darin.

Bei Direkt-Zugriffsbefehlen ist die aktuelle Partition der Direkt-Zugriffsdatei, wenn die Datei geöffnet ist, angesprochen. Dies hat den Gebrauch des CHANGE-PARTITION-Befehls vor dem Öffnen einer Direkt-Zugriffsdatei, um die verlangte Partition zu selektieren, zu Folge. Wenn die Datei offen ist, werden alle Befehle, die zur Direkt-Zugriffsdatei gesandt werden, der anfänglichen Partition zugewiesen, auch wenn die vorhandene Partition gewechselt wird.

Partitionsbefehle

Viele der HD-Befehle sind mit Partitionen verknüpft, und werden benutzt, um Partitionen zu formatieren, initialisieren und wechseln.

Partitionen erstellen

Mit dem Programm HD-TOOLS werden Partitionen auf der Festplatte erstellt. Eine genaue Beschreibung dieses Programms finden Sie im Anhang A.

1581-Unterpitionen erstellen

Innerhalb 1581-Emulations-Modus-Partitionen können Sie Partitionen in der 1581 Art erstellen. Weil die HD schon in Partitionen geteilt wird, und weil diese 1581-Unterverzeichnisse in 1581-Emulations-Modus-Partitionen verschachtelt sind, nennen wir sie Unterpitionen.

Gewisse Einschränkungen gelten dem Erstellen von Unterpitionen, die dem Speichern von Dateien dienen sollen. Wegen der Art, in der reale Spuren in Commodores 1581-DOS behandelt werden, und weil Unterpitionen Header- und Verzeichnisblöcke haben müssen, ist die kleinste Größe einer Unterpition 120 Blöcke. Der Startsektor muß 0

CMD Hard Drive Anleitung

sein und der Endsektor muß ein Vielfaches von 40 sein. Unterpartitionen können nicht auf Spur 40 anfangen, enden oder darauf sein. Hier ist die Syntax, die gebraucht wird, um 1581- Unterpartitionen erstellen zu können:

```
OPENlf,dv,15:PRINT#lf,"/[n]:partitionname,"CHR$(st)CHR$(ss)CHR$(sl)CHR$(sh)
",C":CLOSE15
```

wo:	lf	=	der logische Dateiname für den Befehlskanal ist
	dv	=	die vorhandene Geräteadresse der HD ist
	n	=	die Ziel-1581Emulations-Modus-Partition ist
	partitionname	=	der Name der 1581-Unterpartition ist, die erstellt wird
	st	=	die Startspur der Unterpartition ist
	ss	=	der Startsektor der Unterpartition ist
	sl	=	das niedrigste Byte der Unterpartitionsgröße in Sektoren ist
	sh	=	das höchste Byte der Unterpartitionsgröße in Sektoren ist

Beispiel:

```
OPEN15,12,15:PRINT#15,"/4:SUB1,"CHR$(1)CHR$(0)CHR$(160)CHR$(0)","C":CLOSE15
```

Für diesen Befehl gibt es keinen gleichwertigen JiffyDOS-Befehl.

VERMERK: Neue 1581-Unterpartitionen müssen vor dem Gebrauch formatiert werden. Sehen Sie unter "Formatieren der 1581-Unterpartitionen" nach.

Das Löschen von Partitionen

Partitionen werden mit Hilfe des Programms HD-TOOLS gelöscht. Das Programm ist im Anhang A beschrieben.

Das Löschen von 1581 Unterpartitionen

1581-Unterpartitionen werden ganz anders als normale HD-Partitionen behandelt. Mit DOS oder BASIC 7.0 SCRATCH-Befehlen kann man eine Unterpartition löschen. Das Löschen von 1581-Partitionen wird genau wie beim Löschen von Dateien gemacht. Alle Dateien in der Unterpartition gehen dabei verloren. Um dafür die richtige Befehlsyntax zu gebrauchen, schauen Sie unter "Dateien löschen" nach. Anstelle vom Dateinamen müssen Sie den Unterpartitionsnamen einsetzen.

Das Wechseln von Partitionen

Sie können mit dem CHANGE-PARTITION-Befehl (CP) die Partitionen wechseln. Hier die Syntax:

```
OPENlf,dv,15:PRINT#15lf,"CPn":CLOSElf
```

wo:	lf	=	die logische Dateinummer für den Befehlskanal ist
	dv	=	die Geräteadresse der Festplatte ist
	n	=	die gewählte Partitionsnummer (1-254) ist

Beispiel:

```
OPEN15,12,15:PRINT#15,"CP4":CLOSE15
```

JiffyDOS Beispiel:

```
@CP11
```

Der Befehl - 'C' <SHIFT> 'P' - ist eine Variation des Partitionswechsel-Befehls, die von BASIC- Programmen leichter verwendbar ist. Wenn man gleichzeitig SHIFT und 'P' drückt, ergibt sich das Symbol '□'. Mit diesem Befehl kann man mit einer Zeichenschleife auf die Partition hinweisen. Die Syntax lautet:

```
OPENlf,dv,15:PRINT#lf,"C□"+CHR$(n):CLOSElf
```

wo:	lf	=	die logische Dateinummer für den Befehlskanal ist
	dv	=	die Geräteadresse der HD ist
	n	=	die gewählte Partitionsnummer (1-254)

Beispiel:

```
OPEN15,12,15:PRINT#15,"C□"CHR$(11):CLOSE15
```

Sich zwischen 1581-Unterpitionen bewegen

Weil die 1581-Emulations-Modus-Partitionen auf der HD 1581-Unterpitionen unterstützen, dürfen Sie die normalen DOS-Befehle benutzen, um sich von einer Unterpition zur anderen zu bewegen. Die Syntax dafür ist:

```
OPENlf,dv,15:PRINT#lf,"[n]:[partitionsname]":CLOSElf
```

wo: lf = der logische Dateiname für den Befehlskanal ist
 dv = die Geräteadresse der HD ist
 n = die Partitionsnummer der 1581-Emulationspartition ist
 partitionsname = der Partitionsname der 1581 Unterpition ist

Beispiel:

```
OPEN15,12,15:PRINT#15,"/4:SUB1":CLOSE15
```

JiffyDOS-Beispiel:

```
@/4:SUB1
```

Wenn die 1581 Emulations-Modus-Partition die vorhandene Partition ist, kann die Partitionsnummer "n" weggelassen werden. Die Unterpitionen können ineinander verschachtelt werden, aber um eine Partition zu erreichen, die zwei Stufen unter der vorhandenen Partition verschachtelt ist, muß man den Befehl zweimal eingeben. Um zum Hauptverzeichnis wieder zu gelangen, gibt man erneut den Befehl ein. Durch einen INITIALIZE-Befehl an die 1581 Emulations-Modus-Partition gelangt man auch ins Hauptverzeichnis.

Merken Sie sich bitte, daß das Verlassen einer 1581 Emulation-Modus-Partition und das spätere Zurückkehren in die gleiche Partition mit dem 'CP'-Befehl in jene Unterpition zurückführt, in der Sie sich beim Verlassen der Partition befunden haben.

Das Formatieren von Partitionen

Der normale Commodore-DOS NEW-Befehl (nicht mit BASIC NEW zu verwechseln) darf von BASIC 2.0 oder 7.0 benutzt werden, um Partitionen auf der HD zu formatieren. Der BASIC 7.0 HEADER-Befehl wird zwar von der HD akzeptiert, ist aber nur für die vorhandene Partition (0) oder Partition 1 gültig.

Die DOS NEW-Befehle können benutzt werden, um alle Dateien von einer Partition zu löschen. Partitionen werden automatisch beim Erstellen formatiert. Wenn Sie eine Partition neu formatieren möchten, um, sagen wir, den Headernamen oder die Diskette-ID zu ändern, benutzen Sie den DOS NEW-Befehl. Sie können entweder die lange oder kurze Version dafür verwenden. Die Syntax dafür ist:

```
OPENlf,dv,15:PRINT#lf,"N[n]:partitionsname[,id]":CLOSElf
```

wo: lf = die logische Dateinummer für den Befehlskanal ist
 dv = die vorhandene Geräteadresse der HD ist
 n = die gewählte Partition, die Sie formatieren wollen (0-254) ist
 Partitionsname = der Name für den Partitionsheader ist
 id = eine 2 stellige ID für den Partitionsheader ist

Folgende Syntax betrifft dem BASIC 7.0 HEADER-Befehl:

```
HEADER"partitionsname"[,Id][,Dn][{ON|,}Udv]
```

wo: partitionsname = der gewählte Partitionsheader-Name ist
 id = eine 2 stellige ID für den Partitionheader ist
 n = die Partitionsnummer, die Sie formatieren wollen, ist (0 oder 1)
 dv = die Geräteadresse der HD ist

Beispiele:

```
OPEN15,12,15:PRINT#15,"N3:PARTITION 3,P3":CLOSE15
HEADER"PARTITION 1",IP1,D1 ON U12
```

JiffyDOS-Beispiele mit dem DOS NEW-Befehl

```
@N3:PARTITION 3,P3
@"N3:PARTITION 3,P3",12
```

ACHTUNG! Die DOS NEW- und BASIC 7.0 HEADER-Befehle werden nicht innerhalb eines Native-Modus-Unterverzeichnisses akzeptiert.

Das Formatieren 1581-Unterpitionen

Die DOS NEW-Befehle werden auch benützt, um die 1581-Unterpitionen zu formatieren. Diese Unterpitionen müssen vor ihren Gebrauch formatiert werden. Ehe eine Unterpition formatiert wird, muß sichergestellt werden, daß diese Unterpition tatsächlich die Aktuelle ist, die in der "Mutter-Partition" erstellt wurde. Benützen Sie dafür den Befehl, der unter "Sich zwischen 1581 Unterpitionen bewegen" beschrieben ist. Um vor dem Formatieren eines falschen Bereiches der HD zu bewahren, soll vorher die richtige 1581-Emulation- Modus-Partition als vorhandene Partition eingestellt werden.

Das Initialisieren der Partitionen

Der DOS 'INITIALIZE'-Befehl wird oft benützt, wenn eine neue Diskette ins Laufwerk eingelegt wird. Dieser Befehl sorgt dafür, daß der Laufwerkpuffer mit einer Kopie des Disketten-BAMs versehen wird. Diese Funktion wird auf der HD automatisch durchgeführt, aber der Befehl wurde, um die Kompatibilität zu gewährleisten, implementiert. Das Initialisieren einer 1581-Emulations-Modus-Partition führt sie ins Hauptverzeichnis zurück. Die Syntax dafür ist:

```
OPENlf, dv, 15:PRINT#lf, "I[n] [:]":CLOSElf
```

wo: lf = die logische Dateinummer für den Befehlskanal ist
 dv = die vorhandene Geräteadresse der HD ist
 n = die Partition ist, die man initialisieren will

Beispiel:

```
OPEN15, 12, 15:PRINT#15, "I3:":CLOSE15
```

JiffyDos-Beispiel:

```
@I3:
```

Das Validieren der Partitionen

Die DOS VALIDATE und BASIC 7.0 COLLECT-Befehle überprüfen alle Dateien in einer Partition, um den Speicherplatz zu bestätigen, falsch zugewiesene Blöcke zu befreien, und nicht geschlossene Dateien zu löschen. Benützen Sie **nicht** diese Befehle mit Partitionen, die Blöcke enthalten, die mit dem BLOCK-ALLOCATE-Befehl zugeordnet wurden. Sonst können Informationen verloren gehen. Der VALIDATE-Befehl muß mit BASIC 7.0 benützt werden, falls Sie eine Partition - außer der Vorhandene oder Partition 1 - validieren wollen. Die Syntax dafür heißt:

```
OPENlf, dv, 15:PRINT#lf, "V[n] [:]":CLOSElf
```

wo: lf = die logische Dateinummer für den Befehlskanal ist
 dv = die Geräteadresse der HD ist
 n = die Partition ist, die validiert werden soll (0-254)

Die Syntax für den BASIC 7.0 COLLECT-Befehl heißt:

```
COLLECT[, Dn] [{ON|, }Udv]
```

wo: n = die Partition ist, die validiert werden soll (0 oder 1)
 dv = die Geräteadresse der HD ist

Beispiele:

```
OPEN15, 12, 15:PRINT#15, "V2:":CLOSE15  
COLLECT, D0, U12
```

Das JiffyDOS-Beispiel des DOS VALIDATE-Befehls lautet:

```
@V2
```

Das Partitions-Verzeichnis

Eine Vielzahl von HD-Partitionen erfordert einen Überblick des Partitions-Verzeichnisses. Das Partitions-Verzeichnis kann man in jeder Partition betrachten und es informiert über die Nummer, den Namen, und die Art jeder HD-Partition. Dieser Befehl beinhaltet auch Optionen, die Ihnen erlauben, spezifische Partitionen aufzulisten. Die Syntax dafür heißt:

$$\text{LOAD} \$=P[:*][=tp], dv$$

wo:	tp	= die Partitionart ist	N	= Native
			4	= 1541
			7	= 1571
			8	= 1581
			C	= 1581 CP/M
	dv	= die vorhandene HD-Geräteadresse ist		

Beispiele:

```
LOAD"$=P",12
LOAD"$=P:*=8",12
```

JiffyDOS-Beispiele:

@\$=P
@"\$=P:*=N", 12

Das Ändern von Partitionsnamen

Wenn Sie eine Partition mit dem DOS NEW-Befehl neu formatieren, können Sie auch ihren Namen ändern. Dafür gibt es den RENAME-PARTITION-Befehl. Dieser Befehl ist dem DOS-RENAME-Befehl ähnlich. Die Syntax lautet so:

```
OPENlf,dv,15:PRINT#lf,"R-P:neuer name=alter name":CLOSElf
```

wo: lf = der logische Dateiname ist
dv = die vorhandene Geräteadresse der HD ist
neuer name = der gewünschte neue Partitionsname ist
alter name = der alter Partitionsname ist

Beispiel:

```
OPEN15,12,15,"R-P:WORK=NATIVE 1":CLOSE15
```

JiffyDOS-Beispiel:

@R-P:WORK=NATIVE 1

Das Umbenennen von Verzeichnis-Headers

Nachdem eine Vielzahl von Dateien in einer bestimmten Partition, oder eines bestimmten Unterverzeichnisses vorhanden sind, kann es vorkommen, daß der Verzeichnis-Header umbenannt werden sollte. Zwar könnte man dafür den DOS NEW-Befehl benutzen, würden aber dabei alle Dateien innerhalb der Partition verlorengehen. Der HD-Befehl - RENAME HEADER - erlaubt das Umbenennen eines Headers ohne Dateiverlust. Die Syntax lautet wie folgt:

```
OPENlf,dv,15:PRINT#lf,"R-H[n][pfad]:neuer name":CLOSElf
```

wo: lf = die logische Dateinummer ist
dv = die vorhandene HD-Geräteadresse ist
n = die Partition ist, wo der Header umbenannt werden soll
pfad = der Unterverzeichnispfad ist
neuer Name = der neue Headername ist

Beispiele:

```
OPEN15,12,15,"R-H:WORK":CLOSE15
OPEN15,12,15,"R-H1//ASSEM/:BUDDY64":CLOSE15
```

JiffyDOS-Beispiele:

@R-H:WORK
@ "R-H//ASSEM/:BUDDY64",12

Das Beschaffen von Partitionsinformation

Der DOS Get-Partition-Info-Befehl wurde geschaffen, um Informationen über die Partitionen zu beschaffen. Dieser Befehl ist dem Programmierer, dessen Software auf verschiedene Arten von Partitionen auch verschieden reagiert, besonders wertvoll. Die Partitionsnummer der verlangten Information darf als Variable eingesetzt werden und als Zeichenschleife in den Befehl eingegeben werden. Die Syntax für den 'G-P'-Befehl ist wie folgt:

```
OPENlf,dv,15:PRINT#lf,"G-P"[+CHR$(n)]:CLOSElf
```

wo: lf = die logische Dateinummer für den Befehlskanal ist
 dv = die HD-Geräteadresse ist
 n = die Partitionsnummer ist (0-255)

Wenn der 'G-P'-Befehl ohne Zeichenschleife oder einen Wert von 255 eingegeben wird, gilt die erhaltene Information der vorhandenen Partition. Der Wert "Null" (0) verlangt Systempartitionsinformation. Dreißig Bytes (0-29) von Informationen, die der verlangten Partition gelten, und ein CHR\$(13), werden über den Laufwerksfehlerkanal angezeigt. Anschließend folgt ist eine Liste dieser Informationen:

Byte 0	- Partitionsart	0 = nicht erstellt
		1 = Native-Modus
		2 = 1541 Emulations-Modus
		3 = 1571 Emulations-Modus
		4 = 1581 Emulations-Modus
		5 = 1581 CP/M Emulations-Modus
		6 = Druckerpuffer
		7 = Foreign-Modus
		255 = System
Byte 1	- (reserviert)	
Byte 2	- Partitionsnummer	
Byte 3-18	- Partitionsname, wie im Partitionsverzeichnis angezeigt	
Byte 19	- Beginnende Systemadresse der Partition (höchstes Byte)	
Byte 20	- Beginnende Systemadresse der Partition (mittleres Byte)	
Byte 21	- Beginnende Systemadresse der Partition (niedrigstes Byte)	
Byte 22-36	- (reserviert)	
Byte 27	- Partitionsgröße (höchstes Byte)	
Byte 28	- Partitionsgröße (mittleres Byte)	
Byte 29	- Partitionsgröße (niedrigstes Byte)	
Byte 30	- CHR\$(13)	

Merken Sie sich: Die Werte in Bytes 19-21 und 27-29 sind in 512 Byte-Blöcken dargestellt. Auch können nicht definierte Bytes später für besondere Zwecke benutzt werden. Um Probleme mit dem Lesen der Informationen in Partition 13 zu vermeiden, soll der 'G-P'-Befehl immer mit einem folgenden Wagenrücklauf (CHR\$(13)) eingegeben werden. Der BASIC PRINT#-Befehl erledigt dies automatisch, so lange er nicht einem Strichpunkt (;) folgt.

Automatisches Booten

Man kann im Zusammenhang mit einem C-128er oder 128D im 128er Modus ein Programm von der HD automatisch booten. Die HD muß entweder als Geräteadresse 8 konfiguriert werden oder der BASIC 7.0 BOOT-Befehl eingegeben werden. Die vorhandene Partition muß auch einen gültigen Bootsektor haben. Dieser Bootsektor ist auf Spur 1, Sektor 0 in allen Partitionen. Ein interessanter Vorteil bei Native-Modus-Partitionen ist die Tatsache, daß der Bootsektor immer zugeteilt wird. Dadurch besteht niemals die Gefahr einer Überschreibung des Bootsektors und der Bootsektor kann durch den DOS VALIDATE-Befehl nicht befreit werden. Die folgende Syntax betrifft den BOOT-Befehl:

```
OPEN [ [Dn] {ON|,} [Udv] ]
```

wo: n = die Partition ist, wo sich die Datei befindet (0 oder 1)
 dv = die HD-Geräteadresse ist

Die Partitionsnummer dieses Befehls darf nur eine Null oder eine Eins sein. Die Null zeigt die vorhandene Partition, während die Eins Partition 1 zeigt. Dieser Befehl akzeptiert keine anderen Partitionsnummern, da eine Beschränkung in den BASIC 7.0-Befehlssyntax-analyseroutine vorhanden ist. Die Struktur des Bootsektors ist die gleiche wie bei den normalen Commodore-Laufwerken.

Beispiel:

```
BOOT U12
```

Unterverzeichnisbefehle

Drei neue DOS-Befehle wurden hinzugefügt, um das Erstellen und Löschen von Unterverzeichnissen zu ermöglichen. Mit ihnen läßt sich auch das vorhandene Verzeichnis ändern. Beide Befehle benutzen eine ähnliche Syntax, während die Syntax für das Löschen begrenzt wurde, um Probleme zu vermeiden.

Das Erstellen von Native-Modus-Unterverzeichnissen

Der MAKE DIRECTORY-Befehl erlaubt das Erstellen von Native-Modus-Unterverzeichnissen. Die normale Pfadsyntax wird benutzt. Mit einem Pfad können Sie ein Unterverzeichnis in irgendeiner Native-Modus-Partition erstellen, egal in welcher Partition oder in welchem Verzeichnis Sie sich befinden. Hier die Syntax dafür:

```
OPENlf,dv,15:PRINT#lf,"MD[n][pfad]:name":CLOSElf
```

wo: lf = die logische Dateinummer für den Befehlskanal ist
 dv = die vorhandene HD-Geräteadresse ist
 n = die Native-Modus-Partition ist, wo das Unterverzeichnis erstellt werden soll
 pfad = der Pfad zum Unterverzeichnis ist, in der das neue Unterverzeichnis erstellt werden soll
 name = der Name des neuen Unterverzeichnisses ist

Diese Syntax mag ein wenig verwirrend erscheinen, daher erklären wir kurz, wie sie funktionieren soll:

1. Der Name des Unterverzeichnisses muß von dem Rest des Befehls durch einen Doppelpunkt (:) getrennt werden.
2. Wenn Sie ein Unterverzeichnis innerhalb eines anderen Unterverzeichnis erstellen, müssen Sie dieses Unterverzeichnis, falls es nicht das Aktuelle ist, im Pfad angeben.
3. Wenn Unterverzeichnisse im Befehlspfad angegeben werden, muß jeder Unterverzeichnisname zwischen Schrägstrich (/) und Zeichen vorhanden sein. (Nur ein Schrägstrich wird zwischen Unterverzeichnisnamen gebraucht.)
4. Pfade fangen normalerweise beim vorhandenen Verzeichnis an. Wenn der Pfad beim Hauptverzeichnis beginnen soll, muß er mit zwei Schrägstrichen anfangen.
5. Wenn das Unterverzeichnis in einer Partition, die nicht die Vorhandene ist, erstellt werden soll, muß die Partitionsnummer am Anfang des Pfades stehen (vor den Schrägstrichen).

Folgende Beispiele dienen zur Aufklärung:

```
OPEN15,12,15:PRINT#15,"MD:TEMP":CLOSE15
OPEN15,12,15:PRINT#15,"MD1:TEMP":CLOSE15
OPEN15,12,15:PRINT#15,"MD1//:TEMP":CLOSE15
OPEN15,12,15:PRINT#15,"MD1//TEMP/:TEMP2":CLOSE15
OPEN15,12,15:PRINT#15,"MD/TEMP/:TEMP2":CLOSE15
```

JiffyDOS-Beispiele:

```
@MD:TEMP
@MD1:TEMP
@MD1//:TEMP
@"MD1//TEMP/:TEMP2"
@"MD/TEMP/:TEMP2",12
```

Sich zwischen Native-Modus-Unterverzeichnissen bewegen

Der CHANGE-DIRECTORY-Befehl erlaubt die Bewegung zwischen Native-Modus-Unterverzeichnissen. Dieser Befehl verwendet die gleiche Syntax wie der MAKE-DIRECTORY-Befehl. Der Gebrauch von einem Pfad ermöglicht die Bewegung in irgendeinem Unterverzeichnis in der vorhandenen Native-Modus-Partition. Dieser Befehl läßt auch das Ändern des vorhandenen Verzeichnisses auf jede andere Partition zu. Um sich ins vorhandene Verzeichnis einer anderen Partition zu bewegen, müssen Sie einen CHANGE-PARTITION-Befehl eingeben. Die CHANGE-DIRECTORY-Syntax sieht so aus:

```
OPENlf,dv,15:PRINT#lf,"CD[n]{[←]||[[pfad][:]untername]}":CLOSElf
```

wo: lf = die logische Dateinummer für den Befehlskanal ist
 dv = die vorhandene HD-Geräteadresse ist
 n = die Native-Modus-Partition ist, wo das Unterverzeichnis, das Sie als vorhandenes Verzeichnis machen möchten, vorhanden ist
 pfad = der Unterverzeichnispfad ist, der zum...
 untername = Namen des Unterverzeichnisses führt

Der Rückpfeil dient dazu, die HD um ein Verzeichnis zurückzubewegen. Der Rückpfeil kann nicht mit sonstigen Unterverzeichnispfadinformationen kombiniert werden. Sehen Sie die unterstehenden Beispiele.

Sie müssen nicht unbedingt den Doppelpunkt (:) vor den Unterverzeichnisnamen einsetzen, wenn ein Schrägstrich vor dem Unterverzeichnisnamen steht.

Hier sind einige Beispiele des CHANGE-DIRECTORY-Befehls:

```
OPEN15,12,15:PRINT#15,"CD:TEMP":CLOSE15
OPEN15,12,15:PRINT#15,"CD1//:TEMP":CLOSE15
OPEN15,12,15:PRINT#15,"CD1//TEMP/:TEMP2":CLOSE15
OPEN15,12,15:PRINT#15,"CD1←":CLOSE15
```

JiffyDOS-Beispiele:

```
@ "CD:TEMP",12
@CD1//TEMP
@CD//TEMP/TEMP2
@CD←
```

Das Löschen von Native-Modus-Unterverzeichnissen

Der REMOVE DIRECTORY-Befehl erlaubt das Löschen von Native-Modus-Unterverzeichnissen. Diesen Befehl können Sie nicht verwenden, um Pfade zu benützen, die ein Unterverzeichnis, das als Mutterverzeichnis dient, löschen sollen. Sie können mit diesem Befehl auch kein Unterverzeichnis, die Dateien beinhaltet, löschen. Diese Dateien müssen zuerst mit dem DOS-SCRATCH oder einem BASIC 7.0-Befehl gelöscht werden. Folgende Syntax bezieht sich auf den 'Remove Directory'-Befehl:

```
OPENlf,dv,15:PRINT#lf,"RD[n]:Untername":CLOSElf
```

wo: lf = die logische Dateinummer für den Befehlskanal ist
 dv = die vorhandene HD-Geräteadresse ist
 n = die Partition ist, wo das zu löschende Unterverzeichnis existiert
 Untername = der Name des zu löschenden Unterverzeichnisses ist

Beispiele:

```
OPEN15,12,15:PRINT#15,"RD3:TEMP":CLOSE15
OPEN15,12,15:PRINT#15,"RD:TEMP2":CLOSE15
```

JiffyDOS-Beispiele:

```
@ "RD3:TEMP",12
@RD:TEMP2
```

Verzeichnis-Überblick

Einen Verzeichnis-Überblick hat man mit dem BASIC 2.0-Befehl:

```
LOAD "$",12
```

Dieser Befehl ladet das vorhandene Verzeichnis von der HD, die als Geräteadresse 12 eingestellt wurde. Geben Sie LIST ein, um den Verzeichnis-Überblick auf dem Bildschirm zu listen. Sie können auch eine Partitionsnummer eingeben, um ein bestimmtes Verzeichnis zu laden, wie in diesem Beispiel:

```
LOAD "$2",12
```


Muster vergleichen

Selektive Verzeichnisse können auch nach gewöhnlicher Art geladen werden. Man braucht nur einen Doppelpunkt am Ende der Partitionsnummer oder des Pfades zu stellen und einen Dateinamen oder Mustergleichheitszeichen verwenden, um festzustellen, welche Dateien in eine Liste eingerechnet werden. Das (=) Zeichen und ein Dateienartbezeichner dürfen auch nach der Dateienart eingegeben werden. Die Dateienartzeichen sind: **P** für Programm (PRG), **S** für sequentiell (SEQ), **U** für User (USR), **R** für relative (REL) und **B** für Unterverzeichnis- Branch (DIR).

Beispiele:

```
LOAD "$2:S*=P",12
```

Nach diesem Beispiel wird ein Verzeichnis alle PRG-Dateien, die mit S aus Partition 2 anfangen, geladen. Sie können auch den Stern(*) am Anfang eines Dateinamens wie folgt benutzen:

```
LOAD "$1/UTILS/*E=P",12
```

Nach diesem Beispiel wird ein Verzeichnis von allen PRG-Dateien, die mit E enden geladen. Das Fragezeichen (?) darf anstelle eines unbekannten Zeichens in einem Dateinamen gebraucht werden. Möglich ist auch der Gebrauch eines Sterns mitten in einem Muster:

```
LOAD "$1/UTILS/:R*E=P",12
```

Dieses Muster gleicht Dateinamen wie RIDE und RUE. Nur ein Stern darf im Muster verwendet werden. Sie können auch das Fragezeichen als Ersatzzeichen an irgendeine Stelle in der Dateiname benutzen.

```
LOAD "$2:B?RE=P",12
```

Nach diesem Beispiel ladet ein Verzeichnis alle PRG-Dateien, die vier Zeichen lang sind, mit **B** anfangen und mit **RE** enden. Mehrere Fragezeichen dürfen in einem Muster eingesetzt werden. Man kann auch Fragezeichen mit Sternern in einem Muster zusammenstellen.

Verzeichnisauflisten mit Zeit und Datum stempeln

Weil die CMD-HD mit einer Echtzeituhr ausgestattet ist, können Sie problemlos die Zeit und das Datum der erstellten Dateien ins Verzeichnis eintragen. Dies wird bei allen Dateiarten unabhängig von Partitionsart erledigt. Die HD verwendet dafür die gleiche Methode wie GEOS. Um dem User das Anschauen der fixierten Zeit und des fixierten Datums zu erleichtern, wurden neue Optionen für das Laden der Verzeichnisse hinzugefügt. Diese Optionen erlauben nicht nur das Anschauen der geprägten Daten, sondern auch das Selektieren von Dateien, die innerhalb eines bestimmten Zeitrahmens erstellt wurden. Die Syntax dafür sieht so aus:

```
LOAD "$=T[n] [pfad] [:muster[={tp|option}[,option...]]],dv
```

wo:	n	= die Partitionsnummer des Verzeichnisses ist, der geladen werden soll
	pfad	= das gewünschte Unterverzeichnis ist
	muster	= der Name der zu gleichender Datei oder des Musters ist
	tp	= das erste Zeichen der Dateiart (P, S, R,U, oder B) ist
	option	= eine der unten aufgelisteten Optionen ist
	dv	= die vorhandene HD-Geräteadresse ist

Optionen:

L	= das lange Format
N	= Zeit und Datum in der Auflistung nicht einschließen
>stamp	= größer oder gleich stamp
<stamp	= weniger oder gleich stamp
(stampformat = MM/DD/YY HH:MM xM)	

Obwohl die Syntax dieses Befehls kompliziert aussieht, ist sie in Wirklichkeit einfach.

Die Partitionsnummer (n) kann, wenn erwünscht, angegeben werden. Fehlt dieser Parameter, trifft der Befehl die vorhandene Partition.

Der Dateiname ist der Name einer Datei oder einer gewöhnlichen mustergleichenden Schleife. Man kann also einen Stern (*) benützen, um eine Vielzahl von Zeichen vergleichen zu können, und ein Fragezeichen, um einzelne Zeichen zu vergleichen. Sehen Sie die Beispiele unter 'Muster zusammenpassen' noch einmal an.

Die Dateiart (*tp*) ist eine Option, die aber als erste Option nach dem Dateinamenmuster stehen muß, wenn erwünscht wird. Dies kann entweder ein P (PRG), S (SEQ), R (REL), oder ein U (USR) sein. Wenn Sie alle Dateiarten ansehen möchten, lassen Sie diese Option aus.

Die Optionen erlauben das Vergleichen spezifischer Zeiten und Daten (>*stamp*, <*stamp*), und das lange Format (L). Sie können auch nach Verzeichniseingaben innerhalb eines bestimmten Zeit- oder Datumsrahmens verlangen, aber auch daß diese Informationen nicht in einer Auflistung erscheinen (N). Alle diese Optionen müssen durch Beistriche getrennt werden.

Das lange Zeitformat gibt das volle Datum und die Zeit:

```
112 "TESTFILE"          PRG    02/02/90    04.44 PM
```

Das kurze Zeitformat gibt Datum und Zeit wie folgt:

```
112 "TESTFILE"          P 02/02 04.44 P
```

Wenn die NICHT-LISTEN-Option eingegeben wird, werden die Verzeichnisangaben, egal ob das lange Format angegeben wird oder nicht, normal geladen. Die NICHT-LISTEN-Option existiert, um den Gebrauch der Zeit und des Datums fürs Mustergleichen bei Programmen, die die extra Zeit- und Datumzeichen nicht akzeptieren, zu erlauben.

Die <*stamp* Option listet alle Dateien, die eine erstellte Zeit und ein erstelltes Datum haben, die weniger oder gleich der Zeit und dem Datum, die im *stamp* angegeben werden, sind. Die >*stamp* Option listet alle Dateien, die eine erstellte Zeit und ein erstelltes Datum haben, die größer oder gleich der Zeit und dem Datum, die im *stamp* angegeben werden, sind.

Werden beide *stamp* Optionen im gleichen Befehl benützt, so sind in der folgenden Liste die Dateien, welche zwischen die zwei Rahmen fallen, vorhanden.

Das *stamp* Format muß, wie gezeigt, genau eingegeben werden. Monat, Tag und Jahr müssen mit zwei Zeichen, getrennt mit einem Schrägstrich, eingegeben werden. Stunde und Minute müssen mit zwei Zeichen im 12 Stundenformat, getrennt mit einem Doppelpunkt oder Punkt, eingegeben werden. Der letzte Parameter muß AM oder PM sein. Datum und Zeit müssen mit einem einzigen Leerraum, so wie Zeit und AM/PM-Parameter, getrennt werden. Hier sind einige Beispiele:

```
LOAD"$=T",12
LOAD"$=T2",12
LOAD"$=T2:*=P",12
LOAD"$=T2:*=P,L",12
LOAD"$=T2:*=P,L,>12/21/89 04:15 PM",12
LOAD"$=T:*=L,<12/21/89 04:15 PM",12
LOAD"$=T4:*=S,N,>12/01/89 12:01 AM,<12/31/89 12:00 PM",12
```

JiffyDOS-Beispiele:

```
@"$=T",12
@"$=T2"
@$=T2:*=P
@$=T4:*=S,N,>12/01/89 12:01 AM,<12/31/89 12:00 PM
```

Dateibefehle

Dateibefehle sind die wohl am meisten gebrauchten Befehle. Sie beinhalten das Laden und Speichern, Verifizieren, Umbenennen, Löschen, Kopieren und Sperren von Dateien, und werden in einer ähnlichen Art, wie bei Commodore-Laufwerken, eingegeben. Obwohl die BASIC 2.0-Versionen dieser Befehle in BASIC 7.0 unterstützt werden, besitzt BASIC 7.0 andere Befehle, die die gleichen Funktionen durchführen. Die BASIC 2.0 Versionen dieser Befehle sind vielseitiger, wenn man mit Partitionen arbeitet.

Das Laden von Dateien

Programme werden mit Hilfe folgender Syntax in BASIC 2.0 und 7.0 geladen:

```
LOAD "[n] [pfad] : dateiname", dv[, sa]
```

wo: n = ein legaler Partitionsname von 1 bis 254 ist
 pfad = der Unterverzeichnispfad zur Datei ist
 dateiname = ein legale Dateiname bis 16 Zeichen ist
 dv = die vorhandene HD-Geräteadresse ist
 sa = die (falls notwendig) Sekundäradresse ist

Um ein Maschinensprachprogramm laden zu können, muß die Sekundäradresse 1, getrennt von der Geräteadresse durch einen Beistrich, am Ende des Befehls hinzugefügt werden.

Beispiel:

```
LOAD "2:BASEBALL",12
```

JiffyDOS-Beispiele:

```
/"2:BASEBALL",12  
%3/TERMS/:TERMBOOT
```

Es ist empfehlenswert, die BASIC 2.0-Syntax zu benutzen, wenn Sie Partitionen bestimmen. BASIC 7.0 erlaubt den Zugriff nur zur vorhandenen Partition(0) oder zur Partition 1 und die Unterverzeichnispfade werden nicht berücksichtigt.

Der BASIC 7.0 BLOAD-Befehl kann benutzt werden, um Maschinensprach- oder Datendateien in den Speicher zu laden. Der DLOAD-Befehl wird hauptsächlich angewendet, um BASIC-Programme zu laden. Die Syntax für diese Befehle lautet so:

```
BLOAD "dateiname" [, Dn] [{ON|, Udv] [Bb] [, Pa]
```

```
DLOAD "dateiname" [, Dn] [{ON|, } Udv]
```

wo: dateiname = der Name der Maschinensprachprogramm ist, der geladen werden soll
 n = die Partitionsnummer ist, wo sich die Datei befindet
 dv = die vorhandene HD-Geräteadresse ist
 b = die Speicherbank ist, zu der die Datei geladen werden soll
 a = die Startadresse der Datei ist

Beispiele:

```
BLOAD "SPRITE", D0, U9, B0, P3584  
DLOAD "TEST", D0 ON U9
```

Das Speichern von Dateien

Folgende Syntax kann benutzt werden, um Programme in BASIC 2.0 bzw. 7.0 zu speichern:

```
SAVE "[[@] [n] [pfad] : dateiname", dv
```

wo: n = irgendeine legale Partitionsnummer von 1-254 ist
 pfad = der Unterverzeichnispfad ist, wo die Datei gespeichert wird
 dateiname = irgendein legaler Dateiname von bis zu 16 Zeichen ist
 dv = die vorhandene HD-Geräteadresse ist

Das gezeigte "@" Symbol in der Befehlssyntax wird benutzt, um zeigen zu können, daß eine Datei durch eine neue gleichnamige Datei ersetzt wird. Diese SAVE WITH REPLACE-Option muß von einer Partitionsnummer und einem Doppelpunkt (:) gefolgt werden. Um ein Maschinensprachprogramm von einem C-64 (oder 128 im 64er-Modus) zu speichern, müssen Sie einen Maschinensprachmonitor gebrauchen, bzw. einige BASIC-Hinweisadressen ändern.

Beispiele:

```
SAVE "2:BASEBALL",12  
SAVE "/TERMS/:TERMBOOT",12
```

JiffyDOS-Beispiele:

```
←"2:BASEBALL",12  
←/TERMS/:TERMBOOT
```

Sie dürfen die BASIC 7.0-, BSAVE- und DSAVE-Befehle benutzen, wenn Sie vorhaben, mit der vorhandenen Partition (0) bzw. im Verzeichnis, oder im vorhandenen Verzeichnis von Partition 1, zu arbeiten. BSAVE gebraucht man für nicht-BASIC-Programmdateien, während DSAVE mit BASIC-Programmen verwendet wird.

```
BSAVE"[@]dateiname"[,Dn][{ON|,}Udv][,Bb],Pa TO Pe
```

```
DSAVE"[@]dateiname"[,Dn][{ON|,}Udv]
```

wo: dateiname = der Name, der zu ladender Datei ist
 n = die Partitionsnummer ist, wo die Datei gespeichert werden soll
 dv = die vorhandene HD-Geräteadresse ist
 b = die Speicherbank ist, wo die Datei gespeichert werden soll
 a = die Anfangsadresse der zu speichernden Datei ist
 e = die Endadresse der zu speichernden Datei ist

Das gezeigte "@" Symbol in der Befehlssyntax wird benutzt, um zeigen zu können, daß eine Datei durch eine neue gleichnamige Datei ersetzt wird. Dieser Vorgang nennt man 'SAVE WITH REPLACE'.

Beispiele:

```
BSAVE"SPRITE",D0,U9,B0,P3584
DSAVE"TEST",D1, ON U9
DSAVE"TEST2"
```

Das Verifizieren (Prüfen) von Dateien

BASIC 2.0- und 7.0-Befehle enthalten Befehle, die Ihnen das Prüfen gespeicherter Programme erlauben. Diese Befehle vergleichen das gespeicherte Programm mit dem Speicherinhalt. Weil die kleinste Speicheränderung zu einer Prüf-Fehlfunktion führen kann, empfehlen wir ein Prüfen der Dateien sofort nach dem Speichern der Dateien. Beide BASIC- Versionen unterstützen das Spezifizieren einer Partition innerhalb des Dateinamenteils dieses Befehls.

```
VERIFY"[n][pfad]:]dateiname",dv[,sa]
```

wo: n = die Partition ist, wo die Datei zu finden ist
 pfad = der Unterverzeichnispfad ist, der zu der Datei führt, die geprüft werden soll
 dateiname = der Name der zu prüfenden Dateien ist
 dv = die vorhandene HD-Geräteadresse ist
 sa = die Sekundäradresse 1 einer zu prüfenden Nicht-BASIC-Datei ist

Beispiele:

```
VERIFY"NEWSTATS",12
VERIFY"1/UTILS/TERMS/:XLATOR",12
```

JiffyDOS-Beispiele:

```
' "NEWSTATS",12
```

In BASIC 7.0 wird der normale VERIFY-Befehl akzeptiert. Sie können aber den DVERIFY- Befehl benutzen. Dieser Befehl ist nur für den Gebrauch mit Partition 0, Partition 1, und das vorhandene Unterverzeichnis bestimmt.

```
DVERIFY"dateiname"[,Dn] [{ON|,}Udv]
```

wo: dateiname = der Dateiname ist
 n = die Partition ist, wo die Datei sich befindet (0 oder 1)
 dv = die HD-Geräteadresse ist

Beispiel:

```
DVERIFY"NEWSTATS",D1,U12
```

Das Umbenennen von Dateien und Unterverzeichnissen

Mit den DOS RENAME oder BASIC 7.0 RENAME-Befehlen können Sie Dateien und Unterverzeichnisse umbenennen. Der BASIC 7.0-Befehl unterstützt nur das vorhandene Verzeichnis in der vorhandenen Partition 0 oder 1. Bei beiden Versionen müssen die Partitionen für die zwei Dateinamen die gleichen sein, oder die gleiche Partition bestimmen müssen. Die Syntax für den DOS RENAME-Befehl lautet:

```
OPENlf,dv,15:PRINT#lf,"R[n][pfad]:neuer name=[[n][pfad]:]dateiname:CLOSElf
```

wo: lf = die logische Dateinummer des Befehlskanal ist
 dv = die vorhandene HD-Geräteadresse ist
 n = die Partition ist, wo die zu umbenennender Datei sich befindet
 pfad = der Unterverzeichnispfad zur gesuchten Datei ist
 neuer name = der neue Name der Datei ist
 dateiname = der Name der zu umbenennenden Datei ist

Beispiele:

```
OPEN15,12,15:PRINT#15,"R1:BOOT1=BOOT":CLOSE15
OPEN15,12,15,"R1/UTILS/:NEWT=1/UTILS/:WW":CLOSE15
```

JiffyDOS-Beispiel:

```
@ "R1:BOOT1=BOOT",12
```

Die BASIC 7.0 RENAME-Befehlsyntax ist:

```
RENAME[Dn]"dateiname"TO[Dn]"neue datei"[,Udv]
```

wo: n = die Partition ist, wo sich der Dateiname befindet
 dateiname = der Name der zu umbenennender Datei ist
 neue datei = der neue Dateiname ist
 dv = die vorhandene HD-Geräteadresse ist

Das Löschen von Dateien (Scratchen)

Die normalen DOS und BASIC 7.0 SCRATCH-Befehle können benutzt werden, um von einer Partition Dateien zu löschen. Der BASIC 7.0-SCRATCH-Befehl ist nur mit Partitionsnummern 0 (vorhandener) und 1 effektiv. Auch der normale Commodore DOS-SCRATCH-Befehl kann anstelle von den BASIC 7.0- und 2.0-Versionen gebraucht werden. Folgende Syntax deckt die DOS-Version dieses Befehls.

```
OPENlf,dv,15:PRINT#lf,"S[n][pfad]:dateiname[, [n][pfad]:]dateiname...":CLOSElf
```

wo: lf = die logische Dateinummer des Befehlskanals ist
 dv = die vorhandene HD-Geräteadresse ist
 n = die Partition(s) ist (sind), wo sich die Datei(en) befindet(en)
 pfad = der(die) Unterverzeichnispfad(e) ist (sind), der (die) zu den Dateien führt(en)
 dateiname = der Name der Datei(en) ist (sind)

Man kann mit diesem Befehl mehrere Dateien löschen, da er bis zu fünf separaten Dateinamenparametern akzeptiert. Verschiedene Partitionen können mit den separaten Dateinamen bestimmt werden. Die Dateinamenparameter dürfen auch Stellvertretersymbole, die das Löschen von mehreren Dateien innerhalb einer einzelnen Partition erlauben, anwenden.

Beispiele:

```
OPEN15,12,15:PRINT#15,"S1:JUNK,3:C?*.BAS":CLOSE15
OPEN15,12,15,"S1/UTILS/:CO*":CLOSE15
```

JiffyDOS-Beispiele:

```
@ "S1:JUNK,3:C?*.BAS",12
@S1/UTILS/:CO*
```

Die BASIC 7.0 SCRATCH-Befehlsyntax ist:

```
SCRATCH"dateiname" [,Dn] [{ON|,}Udv]
```

wo: dateiname = der Name der zu löschender Datei ist
 n = die Partition ist, wo sich die zu löschender Datei befindet
 dv = die HD-Geräteadresse ist

Mehrere Dateien können auch mit diesem Befehl durch Mustergleichen gelöscht werden, obwohl das Bestimmen mehrerer Dateinamen im Gegensatz zur DOS-Version nicht zugelassen ist. Vergessen Sie nicht, daß dieser Befehl nur für den Gebrauch mit dem vorhandenen Verzeichnis in Partition 0 oder 1 funktioniert.

Das Kopieren von Dateien

Mit den mitgelieferten Programmen und DOS-Befehlen können Sie zwischen Partitionen Dateien kopieren. Ein weiteres Programm erlaubt das Kopieren einer ganzen Diskette zu einer ähnlichen HD-Partition und umgekehrt. Weitere Informationen darüber finden Sie im Anhang A.

Das Kopieren zwischen Laufwerken

Mit einem gewöhnlichen Kopierprogramm können Sie Dateien zwischen HD und Laufwerken kopieren. Nur auswählbare (generisch) Kopierprogramme, die nicht versuchen, die Laufwerksart durch Überprüfen der ROM-Belegung zu finden, können mit der HD benützt werden.

Das Programm FCOPY assistiert beim Kopieren. FCOPY läuft mit allen Dateiartern und Laufwerksarten, inklusiv RAM-Erweiterungen, die unter RAMDOS laufen.

Wenn Sie JiffyDOS im Rechner eingebaut haben, können Sie den eingebauten JiffyDOS- Dateikopierer dafür anwenden.

Das Kopieren und Kombinieren von Dateien zwischen Partitionen

Mit dem gewöhnlichen Commodore DOS COPY-Befehl kann man Dateien von einer Partition zur anderen kopieren. Dieser Befehl erlaubt das Plazieren einer Partitionsnummer vor jeden, im Befehl bestimmten, Dateinamen. Die Syntax lautet wie folgt:

```
OPENlf,dv,15:PRINT#lf,"C[n][pfad]:neue datei=[n][pfad]:]dateiname[, [n:]  
dateiname...]" :CLOSElf
```

wo: lf = die logische Dateinummer für den Befehlskanal ist
 dv = die vorhandene HD-Geräteadresse ist
 n = die Partition ist, in der sich die Datei befindet oder die Datei erhalten soll
 pfad = der (die) Unterverzeichnispfad(e) ist (sind), wo die Datei(en) sich befindet
 neue datei = der Name der neuen Datei ist
 dateiname = der Name der zu kopierenden Datei(en) ist(sind)

Bis zu fünf Dateien dürfen mit diesem Befehl in eine einzige Datei kombiniert werden, obwohl es nur bei Textdateien sinnvoll ist. Wenn Sie diesen Befehl zum Kopieren einer einzigen Datei auf eine andere Partition benützen, dürfen Sie für beide Dateien den gleichen Namen verwenden.

Beispiele:

```
OPEN15,12,15:PRINT#15,"C1:FCOPY=3:FCOPY":CLOSE15  
OPEN15,12,15,"C:FULLSTATS=STAT1,3:STAT3":CLOSE15  
OPEN15,12,15,"C2:MCOPY=1/COPIERS/:MCOPY":CLOSE15
```

JiffyDOS-Beispiele:

```
@ "C1:FCOPY=3:FCOPY",12  
@ "C2:MCOPY=1/COPIERS/:MCOPY"  
@ C:FULLSTATS=STATS1,3:STATS3
```

Mit dem gewöhnlichen BASIC 7.0-COPY-Befehl können Sie auch Dateien von einer Partition auf eine andere kopieren. Der Befehl ist aber auf Dateien, die im vorhandenen Verzeichnis der vorhandenen Partition (0) bzw. Partition 1 sind, begrenzt. Benützen Sie den DOS COPY- Befehl, wenn Sie Dateien auf andere Partitionen kopieren möchten. Die Syntax für diesen Befehl lautet:

```
COPY[Dn,]"dateiname"TO[Dn]"neue datei"[,Udv]
```

wo: n = die Quell- bzw. Zielpartition ist
 dateiname = der (die) Name(en) der zu kopierenden Datei(en) ist(sind)
 neue datei = der Name der neu erstellten Datei ist
 dv = die vorhandene HD-Geräteadresse ist

Wenn Sie diesen Befehl benützen, um eine Datei auf eine andere Partition zu kopieren, dürfen beide Dateien den gleichen Namen tragen. Wenn die Datei aber auf die gleiche Partition kopiert wird, muß sie einen neuen Namen tragen.

Mit dem BASIC 7.0-CONCAT-Befehl dürfen zwei Dateien zu einer Datei kombiniert werden. Dies ist nur bei Textdateien sinnvoll. Auch hier ist der Befehl auf dem vorhandenen Verzeichnis der vorhandenen Partition (0) bzw. Partition 1 begrenzt. Die Syntax lautet:

```
CONCAT[Dn,]"dateiname"TO[Dn]"neue datei"[,Udv]
```

wo: n = die Partition ist, wo sich die Datei befindet
 dateiname = der Name der Datei ist, die hinzugefügt wird
 neue datei = der neue Dateiname ist
 dv = die vorhandene HD-Geräteadresse ist

Die Datei, die vorher Neue Datei hieß, wird durch die neu erstellte Datei ersetzt.

```
CONCAT "NEWNUMBERS" TO "ALLNUMBERS"
```

Das Sperren und Entsperren von Dateien

Um vor unbeabsichtigtem Löschen zu schützen, können Dateien, die sich auf der HD befinden, gesperrt werden. Bevor Sie eine gesperrte Datei löschen können, muß sie zuerst entsperrt werden. Gesperrte Dateien merkt man in einer Auflistung durch ein "<" Zeichen, das direkt nach dem Dateinamen steht. Hier ein Beispiel.

```
33 "JIFFYMON" PRG<
```

Man kann auch Native-Modus-Unterverzeichnisse und 1581 Unterpertitionen sperren. Wenn ein Unterverzeichnis gesperrt ist, kann es nur mit dem REMOVE DIRECTORY-Befehl gelöscht werden, wenn es zuerst entsperrt wurde.

Der Sperrbefehl ist eine Kippfunktion. Wenn er bei einer gesperrten Datei angewendet wird, wird die Datei entsperrt. Bei einer entsperrten Datei wird sie gesperrt. Die Syntax dieses Befehls sieht so aus:

```
OPENlf,dv,15:PRINT#lf,"L[n][pfad]:name":CLOSElf
```

wo: lf = die logische Dateinummer ist
 dv = die vorhandene HD-Geräteadresse ist
 n = die Partitionsnummer ist, in der die Datei sich befindet
 pfad = der Native-Modus-Unterverzeichnispfad ist
 name = der Datei- bzw. Unterverzeichnisname ist, den Sie sperren bzw. entsperren

Hier einige Beispiele:

```
OPEN15,12,15:PRINT#15,"L:TEST":CLOSE15
OPEN15,12,15:PRINT#15,"L1//:TEST":CLOSE15
OPEN15,12,15:PRINT#15,"L/UTILS/:TEST":CLOSE15
```

JiffyDOS-Beispiele:

```
@L:TEST
@"L1//:TEST"
@"L/UTILS/:TEST",12
```

VERMERK: JiffyDOS enthält seine eigene Version des LOCK-Befehls (Sperrbefehls). Diesen Befehl können Sie natürlich auch verwenden. Weitere Information entnehmen Sie aus dem JiffyDOS-Handbuch.

Relativdateibefehle

Relativdateien sind Dateien, die eine Indextabelle enthalten. Dadurch wird bei der Datensatzsuche der Zugriff schneller gemacht. Datensätze werden in einem Seitensektor der Relativdatei untergebracht. Zwei Arten von Seitensektoren existieren auf der Festplatte: gewöhnliche Seitensektoren (wie bei den 1541 und 1571-Emulations-Modus-Partitionen) und Superseitensektoren (wie bei den 1581 Emulations-Modus und Native-Modus-Partitionen).

Relativdateien können bis zu 720 Blöcken groß sein, wenn gewöhnliche Seitensektoren verwendet werden, über 60.000 Blöcke groß sein, wenn Superseitensektoren verwendet werden. Bis zu 65.535 Datensätze von 2 bis 254 Zeichen lang sind erlaubt. Alle Datensätze in der gleichen Datei haben die gleiche Länge, obwohl es nicht notwendig ist, alle Zeichen in jedem Datensatz zu benutzen.

Der Hauptvorteil der Relativdatei ist die Geschwindigkeit. Wenn Sie sich den Datensatz der gesuchten Information merken, können Sie sofort die Datei finden. Sonst müssen Sie am Anfang der Datei anfangen, die Information zu suchen. Einen separaten Index hält man meistens in einer anderen Dateiart, wo die Datensatzinformation aufbewahrt wird.

Das Öffnen und Erstellen einer Relativdatei

Die gleiche Syntax kann benutzt werden, um eine Relativdatei zu erstellen oder eine bereits vorhandene Relativdatei zu öffnen. Dafür verwenden Sie den BASIC 2.0- oder BASIC 7.0 DOPEN-Befehl. Wenn Sie eine neue Relativdatei erstellen wollen, schreiben Sie mit dem 'PRINT#'-Befehl einen neuen Datensatz zu der Datei, nachdem sie geöffnet wurde. Wenn Sie mit der Relativdatei fertig sind, schließen Sie sie mit dem CLOSE- oder DCLOSE-Befehl. Hier ist die BASIC 2.0-Syntax fürs Öffnen oder Erstellen einer neuen Relativdatei:

```
OPENlf,dv,sa,"[n][pfad]:dateiname[",L"+CHR$(rl)]
```

wo: lf = die logische Dateinummer ist
 dv = die vorhandene HD-Geräteadresse ist
 sa = die Sekundäradresse (2-14) ist
 n = die Partition ist, in der die Datei existiert oder erstellt wird
 pfad = der Native-Modus-Unterverzeichnis ist, der zur Datei führt
 dateiname = der Name der Relativdatei ist
 rl = die Datensatzlänge ist (nur notwendig, wenn eine neue Relativdatei erstellt wird)

Beispiele:

```
OPEN2,12,2,"1/DATA/:CUSTOMERS,L"+CHR$(127)
OPEN2,12,2,"ADDRESS"
```

Diese Syntax dürfen Sie auch in BASIC 7.0 benutzen, aber für diese neue BASIC-Version wurde eine alternative Syntax geschaffen. Wie bei allen BASIC 7.0-Befehlen, ist auch dieser auf Partition 0 oder 1 begrenzt. Hier ist die alternative Syntax:

```
DOPEN#lf,"dateiname"[,Lrl][,Dn][,Udv]
```

wo: lf = die logische Dateinummer ist
 dateiname = der Relativdateiname ist
 rl = die Datensatzlänge ist (nur notwendig, wenn eine neue Relativdatei erstellt wird)
 n = die Partition ist, wo die Datei zu finden ist oder wo sie erstellt wird
 dv = die vorhandene HD-Geräteadresse ist

Beispiele:

```
DOPEN#2,"CUSTOMERS",L127,D1,U12
DOPEN#2,"ADDRESS"
```


Auf einem Bestimmten Datensatz Positionieren

Wenn Sie an einen bestimmten Datensatz schreiben, bzw. einen bestimmten Datensatz lesen wollen, müssen Sie entweder den DOS POSITION- oder den BASIC 7.0 RECORD-Befehl anwenden. Die Relativdatei muß vorher geöffnet werden, um diese Befehle zu benutzen, und im Falle des DOS POSITION-Befehls, muß der Laufwerkbefehlskanal geöffnet werden. Mit diesen Befehlen können Sie auch eine Vielzahl von leeren Datensätzen erstellen, wenn Sie eine Relativdatei erstellen. Dadurch wird das Datenschreiben erheblich schneller. Hier die Syntax für den DOS POSITION-Befehl:

```
PRINT#lf, "P"+CHR$(ch)+CHR$(lr)+CHR$(lh) [+CHR$(of)]
```

wo: lf = die logische Dateinummer des Befehlskanals ist
 ch = die Sekundäradresse ist, die beim Öffnen der Relativdatei plus ein Wert von 96 gebraucht wird
 lr = das niedrigste Byte der Datensatznummer ist, zu der Sie zugreifen möchten
 lh = das höchste Byte der Datensatznummer ist, zu der Sie zugreifen möchten
 of = die Bytenummer des Datensatzes ist, wo Sie anfangen lesen bzw. schreiben wollen

Beispiele:

```
PRINT#15, "P"+CHR$(98)+CHR$(30)+CHR$(0)+CHR$(10)
PRINT#15, "P"+CHR$(98)+CHR$(30)+CHR$(0)
```

Hier die Syntax für den BASIC 7.0 RECORD-Befehl:

```
RECORD#lf, rn[, of]
```

wo: lf = die logische Dateinummer für die Relativdatei ist
 rn = die Datensatznummer ist, die Sie erstellen oder zu der Sie zugreifen wollen
 of = die Bytenummer im Datensatz ist, wo Sie anfangen lesen bzw. schreiben wollen

Beispiele:

```
RECORD#2, 30, 10
RECORD#2, 30
```

Wenn Sie zuvor eine Relativdatei erstellen, benutzen Sie den PRINT#-Befehl, um ein CHR\$(255) in den letzten Datensatz hineinzuschreiben. Dieses Zeichen hat eine Sonderbedeutung in Relativdateien. Es wird als das erste Zeichen in leeren Datensätzen verwendet. Sie können später die Größe einer Relativdatei ändern, indem Sie einfach dem Datensatz eine höhere Nummer geben.

Wann immer ein neuer Datensatz erstellt wird, meldet sich das Laufwerk mit einer Fehlermeldung (Fehlermeldung 50 'RECORD NOT PRESENT'). Diese Fehlermeldung ist zu ignorieren.

VERMERK: Obwohl es nicht notwendig ist, die RECORD oder POSITIONS-Befehle zweimal zu senden, sollten Sie es trotzdem tun, um Probleme mit anderen Laufwerken zu vermeiden. Geben sie daher den Befehl vor und nach dem Schreiben eines Datensatzes.

Besondere HD-Befehle

Mit den folgenden HD-DOS-Befehlen können Sie die Sonderfunktionen der HD voll ausnützen.

Software SWAP- (Tausch-)Befehle

Die HD-SWAP 8- und SWAP 9-Funktionen können auch innerhalb Softwareprogrammen benutzt werden. Sie können auch einen aktiven SWAP-Zustand mit einem SWAP TO DE- FAULT-Befehl aufheben. Diese Befehle lassen das Tauschen der HD-Geräteadresse bei Lade- bzw. Bootprogrammen zu. Ein Beispiel eines Ladeprogramms finden Sie auf der HD-Utility- Diskette (BOOTQ).

Wann immer einer der SWAP-Befehle innerhalb eines Programms benutzt wird, ist es sehr wichtig nach dem Befehl und vor dem Schließen eine Verzögerungsschleife einzubauen. Diese Befehle richten der HD aus, Befehle an andere seriell-angeschlossenen Laufwerke zu senden, führt Jeder Versuch des Computers, den Seriellbus während dieser Aktivität zu benutzen, führt zum Absturz. Wir empfehlen daher, Werte um die 500 in der FOR/NEXT-Schleife anzuwenden. Dadurch haben diese Befehle genügend Zeit, ihre Aufgaben zu erfüllen.

Die SWAP-Funktionen funktionieren bei geöffneten Dateien - mit der Ausnahme des Befehls-kanals - nicht.

Ein SWAP TO DEFAULT-Befehl ist nicht notwendig, um von einem SWAP 8 zu einem SWAP 9-Zustand, (oder umgekehrt) - zu wechseln. SWAP TO DEFAULT ist nur dann notwendig, wenn die HD zu ihrer vorgegebenen Geräteadresse zurückkehren sollte. Sie können auch die SWAP-Befehle vom Direkt-Modus aussenden. Mit JiffyDOS können sie auch den Befehl abkürzen. Die folgende Syntax ist für die SWAP-Befehle.

```
OPENlf, dv, 15:PRINT#lf, "S-x":FOR t=1 TO 500:NEXT:CLOSElf
```

wo:	lf	= die logische Dateinummer ist
	dv	= die vorhandene HD-Geräteadresse ist
	x	= 8 mit Geräteadresse 8 zu tauschen ist
		9 mit Geräteadresse 9 zu tauschen ist
		D, um die HD auf die vorgegebene Geräteadresse zurückzubringen
	t	= eine Variable für die Zeitschleife ist

Hier folgt ein Musterprogramm mit allen SWAP-Befehlen. Dieses Programm nimmt an, daß die

HD-Geräteadresse 12 ist.

```
100 OPEN15, 12, 15
110 PRINT#15, "S-8":FORI=1TO500:NEXT:CLOSE15
120 OPEN15, 8, 15
130 PRINT#15, "S-9":FORI=1TO500:NEXT:CLOSE15
140 OPEN15, 9, 15
150 PRINT#15, "S-D":FORI=1TO500:NEXT:CLOSE15
```

Direkt-Modus-Beispiel:

```
OPEN15, 12, 15, "S-8"
CLOSE15
```

VERMERK: Im Direkt-Modus brauchen Sie keine Zeitschleife, solange Sie den SWAP- Befehl zuerst senden und in einer zweiten Zeile den Kanal schließen.

JiffyDOS-Beispiele:

```
@S-8
@"S-9", 8
```

Software-Schreibschutz-Befehle

Die HD läßt das Setzen und Löschen eines Schreibschutzes softwaremäßig zu. Wichtig ist es sich zu merken, daß aus Datenschutzgründen ein Schreibschutz-Zustand softwaremäßig nicht umfahren werden kann, wenn er hardwaremäßig (durch Drücken des WRITE PROTECT- Schalters) aktiviert wurde. Die Syntax fürs softwaremäßige Setzen und Löschen der WRITE PROTECT- Funktion sieht so aus:

```
OPENlf, dv, 15:PRINT#lf, "W-x":CLOSElf
```

wo:	lf	= die logische Dateinummer ist
	dv	= die vorhandene HD-Geräteadresse ist
	x	= 1 setzt WRITE PROTECT, 0 löscht WRITE PROTECT

Beispiele:

```
OPEN15, 12, 15, "W-1":CLOSE15
OPEN15, 12, 15, "W-0":CLOSE15
```

JiffyDOS-Beispiele:

```
@W-1
@"W-1", 12
```

Echtzeituhr-Befehle

Es gibt 3 Befehlsarten, womit die Echtzeituhr der HD gelesen und eingestellt werden kann. Jede Art benötigt ein anderes Format für das Senden und Empfangen von Uhrdaten. Die Datenarten sind ASCII, BCD (Binar-Coded-Dezimal) und dezimal.

Zeit und Datum im ASCII-Format lesen

Der 'T-RA'-Befehl erlaubt das Lesen der HD-Uhr und Zurücksenden des Datums und der Zeit als eine ASCII-Schleife über den Fehlerkanal. Die Syntax dafür ist:

```
OPEN!f, dv, 15:PRINT#1f, "T-RA":CLOSE!f
```

wo: lf = die logische Dateinummer ist
 dv = die HD-Geräteadresse ist

Nachdem der 'T-RA'-Befehl gesandt wurde, antwortet der HD-Fehlerkanal mit dem Datum und der Zeit im folgenden Format:

```
"dow. mo/da/yr hr:mi:se xM"+CHR$(13)
```

wo: dow. = der Wochentag ist (4 Zeichen gefolgt von einem Leerraum)
 (SUN./MON./TUES/WED./THUR/FRI./SAT.)
 mo = der Monat (01-12) ist
 da = der Tag ist
 yr = das Jahr ist
 hr = die Stunde (01-12) ist
 mi = die Minute (00-59) ist
 se = die Sekunde (00-59) ist
 x = A oder P (Vormittag/Nachmittag) ist

Um den Fehlerkanal vom BASIC aus zu lesen, benützen Sie die folgende GET-Schleife:

```
10 GET#1f, A$:T$=T$+A$:IF ST<>64 THEN 10
```

Das Einstellen der Zeit und des Datums im ASCII-Format

Der 'T-WA'-Befehl erlaubt das Einstellen der HD-Echtzeituhr über den Befehlskanal durch das Senden einer ASCII-Schleife, die die aktuelle Zeit darstellt. So sieht die Syntax aus:

```
OPEN!f, dv, 15:PRINT#1f, "T-WAdow. mo/da/yr hr:mi:se xM":CLOSE!f
```

wo: lf = die logische Dateinummer ist
 dv = die HD-Geräteadresse ist

Die übrigen Parameter (dow., mo, da, usw.) folgen dem gleichen Format wie beim 'T-RA'-Befehl. Achten Sie genau auf die Leerräume und Abgrenzungen in der Syntax. Der Wochentag muß 4 Zeichen lang sein und ein Leerraum muß angehängt werden.

Das Lesen von Zeit und Datum im Dezimal-Format

Der T-RD-Befehl erlaubt das Lesen der HD-Uhr und gibt dann über den Fehlerkanal das Datum und die Zeit in einer Reihe von Dezimalbytewerten weiter. BASIC- und MS-Programmierer können dann innerhalb eines Programms die Zeit und das Datum im numerischen Format lesen. So sieht der 'T-RD'-Befehl dafür aus:

```
OPEN!f, dv, 15:PRINT#1f, "T-RD":CLOSE!f
```

wo: lf = die logische Dateinummer ist
 dv = die HD-Geräteadresse ist

Nachdem der 'T-RD'-Befehl gesandt wurde, antwortet der HD-Fehlerkanal mit Datum und Zeit im folgenden Format:

Byte0	=	der Wochentag (00=SUN.,01=MON., usw.)
Byte1	=	das Jahr
Byte2	=	der Monat (01-12)
Byte3	=	der Tag (01-xx)
Byte4	=	die Stunde (01-12)
Byte5	=	die Minute (00-59)
Byte6	=	die Sekunde (00-59)
Byte7	=	AM/PM Flags (00=Vormittag, keine 0=Nachmittag)
Byte8	=	CHR\$(13)

Das Einstellen der Zeit und des Datums im Dezimal-Format

Der 'R-WD'-Befehl erlaubt das Einstellen der HD-Uhr durch das Senden einer Reihe von Dezimal-Byte-Werten, die die Zeit darstellen, über den Befehlskanal. Hier die Syntax:

```
OPEN!f, dv, 15
```

```
PRINT#1f, "T-WD"+CHR$(byte0)+CHR$(byte1)+CHR$(byte2)+CHR$(byte3)+CHR$(byte4)+  
CHR$(byte5)+CHR$(byte6)+CHR$(byte7):CLOSE!f
```

wo:	!f	=	die logische Dateinummer ist
	dv	=	die vorhandene HD-Geräteadresse ist
	bytes 0-7	=	die vorhandene Zeit und das vorhandene Datum, die durch 8 Dezimalbytes dargestellt werden, ist

Das Lesen von Zeit und Datum im BCD-Format

Der 'T-RB'-Befehl erlaubt das Lesen und Wiedergeben der HD-Uhr und des Datums über den Fehlerkanal in einer Reihe von BCD-Bytes. Die Syntax lautet:

```
OPEN!f, dv, 15:PRINT#1f, "T-RB":CLOSE!f
```

wo:	!f	=	die logische Dateinummer ist
	dv	=	die HD-Geräteadresse ist

Der Fehlerkanal gibt das Datum und die Zeit in BCD-Bytes im folgenden Format weiter:

Byte0	-	der Wochentag (00=SUN.,01=MON., usw.)
Byte1	-	das Jahr (z.B. 1990=\$90)
Byte2	-	der Monat (\$01=\$12)
Byte3	-	der Tag (\$01-xx)
Byte4	-	die Stunde (\$00-\$12)
Byte5	-	die Minute (\$00-\$59)
Byte6	-	die Sekunde (\$00-\$59)
Byte7	-	AM/PM-Flagge (00=Vormittag, keine 0=Nachmittag)
Byte8	-	\$0D

Das Schreiben von Zeit und Datum im BCD-Format

Der 'T-WB'-Befehl erlaubt das Einstellen der HD-Uhr durch das Senden einer Reihe von BCD-Bytes, die die Zeit darstellen, über den Befehlskanal. Dieser Befehl wird normalerweise innerhalb eines Maschinenspracheprogramms gesandt. Die Syntax für diesen Befehl ist:

```
"T-WB"+BCD time
```

wo:	BCDtime	=	die vorhandene Zeit/ das vorhandene Datum, die durch 8 BCD-Bytes dargestellt werden, ist (Sehen Sie das 'T-RB'-Format)
-----	---------	---	--

Das Senden von SCSI-Befehlen

Die HD besitzt einen eingebauten Befehl, der denen, die mit dem SCSI-Allgemeinkommando- satz vertraut sind, erlaubt, diese Kommandos direkt an die HD zu schicken. Wir können nicht alle SCSI-Kommandos hier beschreiben. Daher beschreiben wir nur den für das Senden der SCSI-Befehle an die Wirtsteuerung der HD nötige DOS-Befehl. Wir nennen ihn den "SCSI-Send"- Befehl, und hier ist seine Syntax:

```
OPENlf, dv, 15:PRINT#lf, "S-C"CHR$(de) CHR$(bl) CHR$(bh) CHR$(cb) ...
```

wo:	lf	= der logische Dateiname des Befehlskanals ist
	dv	= die HD-Geräteadresse ist
	de	= die SCSI-Geräteadresse ist
	bl	= das niedrigste Byte des SCSI-Datenpuffer im HDRAM ist
	bh	= das höchste Byte des SCSI-Datenpuffer im HDRAM ist
	cb	= die SCSI-Befehlsbytes sind

Beim Lesen des Fehlerkanals gleich nach dem Senden eines SCSI-Befehls wird ein einziges Byte, welches den SCSI-Befehlungsstatus darstellt, weitergegeben. Statusbytewerte und Beschreibungen sind unten aufgelistet.

00	= OK
02	= Zustand prüfen
08	= Ziel beschäftigt
16	= sofortiger Status IK
24	= Buchungskonflikt
48	= DOS Syntaxfehler

Direktzugriffs-Befehle

Bei den meisten Direktzugriffs-Befehlen müssen Dateien sowohl zu dem Befehlskanal als auch zu einer Direktzugriffs-Datei geöffnet werden. Bevor Sie diese Befehle anwenden, sollten Sie sich mit dem Zugreifen auf und Öffnen von Befehls- und Dateidatenkanale(n) vertraut machen.

Der Direktzugriffskanal

Um einen Direktzugriffskanal zu öffnen, müssen Sie folgenden BASIC-Befehl eingeben:

```
OPENlf, dv, sa, "[bu]"
```

wo:	lf	= die logische Dateinummer ist
	dv	= die HD-Geräteadresse ist
	sa	= die Sekundäradresse ist
	bu	= der Laufwerkpuffer ist

Die logische Dateinummer kann jede Zahl von 1 bis 127 sein, aber nicht jene, die zur Zeit benützt wird.

Die Geräteadresse ist die vorhandene HD-Geräteadresse (8-29).

Die Sekundäradresse (Kanalnummer) darf jede Zahl zwischen 2 und 14 sein, sollte aber anders, als jede Zahl der aktiven HD-Kanalnummer, sein. Wir empfehlen die gleiche Nummer für die logische Dateinummer und die Kanalnummer zu verwenden. Dadurch werden die Dateien für den Programmierer leichter zu finden sein.

Die Laufwerkspuffernummer darf zur Zeit jede Zahl zwischen 0 und 29 haben. Dies ist ein optionaler Parameter. Wenn er ausgelassen wird, wird der nächste freier Puffer angesprochen. Am Besten läßt man die Puffernummer aus, es sei denn Sie brauchen eine bestimmte Nummer. Falls Sie eine Nummer, die Sie schon verwendet haben, benützen, werden Sie einen Fehler verursachen. Hier ein Beispiel, wie Sie eine Direktzugriffsdatei öffnen:

```
OPEN2, 12, 2, "[#]"
```

VERMERK: Direktzugriffsdateien sind immer bei der vorhandenen Partitionsnummer geöffnet. Wenn Sie in eine andere Partition hineingreifen wollen, müssen Sie die Partition wechseln (CP). Weiterer Zugriff in diese Datei findet in dieser Partitionsnummer statt, auch wenn Sie nach dem Öffnen der Datei die Partition wechseln.

Das Lesen und Schreiben von Daten mit Direktzugriff

Wenn BLOCK-READ oder BLOCK-WRITE-Befehle benutzt werden, werden Informationen mit BASIC-Befehlen GET#, INPUT#, und PRINT# gelesen und geschrieben. Die Daten, worauf zugegriffen werden, werden nicht direkt auf die oder von der Diskette geschrieben oder gelesen, sondern vorübergehend in einen Puffer gespeichert. Dies mag ein wenig verwirrend klingen, deshalb haben wir hier eine einfachere Beschreibung angegeben.

Das Lesen der Daten von der HD

Mit den BLOCK-READ oder U1-Befehlen kann man Daten direkt von der HD ablesen. Ein bestimmter Block wird abgelesen. Dieser Block wird in einen 256-Byte RAM-Puffer der HD gesteckt. Um diese Daten vom Laufwerk zum Computer zu transferieren braucht man die BASIC-Befehle GET# oder INPUT#. Wenn Sie nur einen Teil der Blockdaten brauchen, können Sie den BUFFER-POINTER-Befehl benutzen, um dem Laufwerk mitzuteilen, welches Byte als erstes an den Computer mit GET# oder INPUT# gesandt werden soll.

Hier ist ein kurzes Beispiel, das zeigt, wie das 7. Byte der 1. Spur, Sektor 0 der Geräteadresse 12 gelesen wird. Der Puffer zeigt auf Byte 6, weil die Bytenummern bei 0 anfangen.

```
10 OPEN15,12,15:REM OPEN COMMAND CHANNEL
20 OPEN2,12,2,"#":REM DIRECT ACCESS CHANNEL
30 PRINT#15,"U1";2;0;1;0: REM TRACK 1 SEKTOR 0
40 PRINT#15,"B-P";2;6: REM SEVENTH BYTE (6)
50 GET#2,A$:REM READ THE BYTE INTO A$
60 CLOSE2: REM CLOSE DIRECT ACCESS CHANNEL
70 CLOSE15:REM CLOSE COMMAND CHANNEL
```

Wenn Sie wissen, welcher Puffer benutzt wird, können Sie auch einen MEMORY-READ- Befehl anwenden. Hier ein Beispiel dieser Methode:

```
10 OPEN15,12,15:REM OPEN COMMAND CHANNEL
20 OPEN2,12,2,"#2": REM USE BUFFER #2
30 PRINT#15,"U1";2;0;1;0:REM TRACK 1 SEKTOR 0
40 PRINT#15,"M-R"CHR$(6)CHR$(5):REM LOC $0506
50 GET#15,A$:REM READ THE BYTE INTO A$
60 CLOSE2:REM CLOSE DIRECT ACCESS CHANNEL
70 CLOSE15: REM CLOSE COMMAND CHANNEL
```

Das Schreiben von Daten auf die HD

Mit dem PRINT#-Befehl werden Daten zuerst auf einen 256-Byte-Puffer der HD geschrieben. Der Puffer selber wird dann auf die gewünschte Spur und den Sektor der Diskette mit dem BLOCK-WRITE oder U2-Befehl transferiert.

Hier das gleiche Beispiel wie vorher:

```
10 OPEN15,12,15:REM OPEN COMMAND CHANNEL
20 OPEN2,12,2,"#":REM DIRECT ACCESS CHANNEL
30 PRINT#15,"B-P";2;6: REM SEVENTH BYTE (6)
40 PRINT#2,CHR$(65):REM WRITE BYTE TO BUFFER
50 PRINT#15,"U2";2;0;1;0:REM TRACK 1 SEKTOR 0
60 CLOSE2: REM CLOSE DIRECT ACCESS CHANNEL
70 CLOSE15:REM CLOSE COMMAND CHANNEL
```

Das obige Beispiel schreibt einen Sektor auf die Diskette, ganz unabhängig von dem, was auf diesem Sektor vorher war. Wenn Sie ein einziges Byte des Sektors ändern wollen, ohne den Rest des Sektors zu ändern, lesen Sie es zuerst in den Puffer ein, ändern Sie das Byte auf einen neuen Wert, und schreiben Sie den Sektor zur Diskette zurück.

Wenn Sie wissen, welcher Puffer benutzt wird, um die Daten zu schreiben, dürfen Sie einen MEMORY-WRITE-Befehl für diesen Zweck auch benutzen. Hier ein Beispiel dieser Methode:

```
10 OPEN15,12,15:REM OPEN COMMAND CHANNEL
20 OPEN2,12,2,"#2":REM USE BUFFER #2
30 PRINT#15,"M-W"CHR$(6)CHR$(5)CHR$(1)CHR$(65):REM LOC $0506, ONE BYTE,
VALUE 65
40 PRINT#15,"U2";2;0;1;0:REM TRACK 1 SEKTOR 0
50 CLOSE2:REM CLOSE DIRECT ACCESS CHANNEL
60 CLOSE15:REM CLOSE COMMAND CHANNEL
```

Das obige Beispiel schreibt einen Sektor auf die Diskette egal was auf diesem Sektor vorher war. Wenn Sie ein einziges Byte des Sektors ändern wollen, ohne den Rest des Sektors zu ändern, lesen Sie es zuerst in den Puffer ein, ändern Sie das Byte auf einen neuen Wert, und schreiben Sie den Sektor zur Diskette zurück.

Blockbefehle

Die BLOCK-Befehle erlauben das Lesen, Schreiben, Zuordnen und Freigeben der Diskettensektoren. Wenn Sie diese Befehle benutzen, sollten Sie die Partitionsart, an der Sie arbeiten, im Gedächtnis halten. Jede Partition hat eine andere legale Spur und Sektorwerte. Verzeichnis, BAM, und Header-Information sind in anderen Bereichen gespeichert. Viele der BLOCK-Befehlsfunktionen dürfen auch durch die USER-Befehle ersetzt werden. Die USER-Befehle sind auch bei einigen bevorzugt, weil die BLOCK-Befehle unerwünschte "Nebenwirkungen" mit sich bringen.

Das Zuordnen von Blöcken

Der BLOCK-ALLOCATE-Befehl wird benutzt um einen Block direkt zuzuordnen. Beim Speichern und Löschen von Dateien, wird diese Funktion automatisch durchs DOS erledigt. BLOCK-ALLOCATE wird normalerweise gebraucht, nachdem Information durch einen BLOCK-WRITE-Befehl auf eine Diskette geschrieben wurde. Er kann zusätzlich benutzt werden, um den nächsthöheren, nicht zugeordneten, freien Block festzustellen, nachdem Sie versucht haben, einen Block zuzuordnen, der schon zugeordnet wurde. Wenn der Fehlerkanal gelesen wird, entdecken Sie, daß die Spur- und Sektorvariablen der Fehlermeldung, die Spur und der Sektor des nächsten freien Blocks beinhalten. Die Syntax des BLOCK-ALLOCATE-Befehls lautet:

```
PRINT#lf, "B-A: "; n; t; s
```

wo:	lf	= die logische Dateinummer des Befehlskanals ist
	n	= die Partition (0) ist, in der der zuzuordnende Block ist
	t	= die Spur ist, auf der der Block existiert
	s	= der Blocksektor ist

WICHTIGER VERMERK: Die Block-Allocate-Fehler, die in einigen Commodore-Laufwerken existieren, existieren nicht in der CMD-HD-Version dieses Befehls.

Das Aufheben der Blockzuordnung

Der BLOCK-FREE-Befehl erlaubt das Aufheben eines zugeordneten Blocks. Wenn Dateien gelöscht werden, findet diese Funktion durchs DOS automatisch statt. BLOCK-FREE wird normalerweise benutzt, um einen Block, in dem Information durch einen BLOCK-WRITE-Befehl vorher angegeben wurde, aber nicht mehr gebraucht wird, frei zu machen. Die Syntax für den BLOCK-FREE-Befehl lautet:

```
PRINT#lf, "B-F: "; n; t; s
```

wo:	lf	= die logische Dateinummer des Befehlskanals ist
	n	= die Partition (0) ist, in der der Block existiert
	t	= die Spur ist, worauf der Block existiert
	s	= der Sektor des Blocks ist

Die Puffer-Hinweisadresse

Der BUFFER-POINTER-Befehl weist auf ein bestimmtes Byte innerhalb eines Sektors, von, bzw. zu dem, gelesen bzw. geschrieben wird. Diese Aktion findet im Diskettenpuffer, der die Daten dafür hält, statt. Hier die Syntax:

```
PRINT#lf, "B-P"; ch; pt
```

wo:	lf	= die logische Dateinummer des Befehlskanals ist
	ch	= die Kanalnummer der Direktzugriffsdatei ist
	pt	= die Bytenummer des gewünschten Blocks ist, der als erstes Byte des nächsten GET#-, INPUT#- oder PRINT#-Befehls benutzt

Diese Syntax nimmt an, daß der Befehl- und Direktzugriffskanal schon offen sind. Die Kanalnummer ist die gleiche wie die Sekundäradresse, die beim Öffnen des Direktzugriffskanals verwendet wurde. Wenn Sie diesen Befehl benutzen, vergessen Sie nicht, daß Bytes von 0 bis 255 nummeriert sind. Wenn Sie aufs erste Byte zugreifen wollen, trägt es die nummer 0.

Das Lesen von Blöcken

Wegen eines Fehlers in seiner Durchführung wird der BLOCK-READ-Befehl selten benutzt. Die meisten Programme verwenden deshalb den 'U1'-Befehl, der dem BLOCK-READ-Befehl sehr ähnlich ist. Die Syntax dafür ist:

```
PRINT#lf, "B-R"; ch; n; t; s
```

wo:	lf	= die logische Dateinummer des Befehlskanals ist
	ch	= die Kanalnummer der Direktzugriffsdatei ist
	n	= die Partitionsnummer ist (immer 0)
	t	= die Spur ist, von der Sie einen Block lesen wollen
	s	= der Sektor ist, von dem sie einen Block lesen wollen

Die Kanalnummer ist die gleiche Nummer, wie die Sekundäradresse, die benutzt wird, um die Direktzugriffsdatei zu öffnen.

Die Partitionsnummer soll immer 0 (Null) sein. Beim Öffnen des Kanals, reifen die Direktzugriffsbefehle immer in die aktuelle Partition.

Das Schreiben von Blöcken

Wegen eines Fehlers in seiner Durchführung wird der BLOCK-READ-Befehl selten benutzt. Die meisten Programme verwenden deshalb den 'U2'-Befehl, der dem BLOCK-READ-Befehl sehr ähnlich ist. Die Syntax dafür ist:

```
PRINT#lf, "B-W"; ch; n; t; s
```

wo:	lf	= die logische Dateinummer des Befehlskanals ist
	ch	= die Kanalnummer der Direktzugriffsdatei ist
	n	= die Partitionsnummer ist (immer 0)
	t	= die Spur ist, zu der Sie den Block schreiben wollen
	s	= der Sektor ist, zu dem Sie schreiben wollen

Die Kanalnummer ist die gleiche Nummer, wie die Sekundäradresse, die benutzt wird, um die Direktzugriffsdatei zu öffnen.

Die Partitionsnummer soll immer 0 (Null) sein. Direktzugriffsbefehle greifen immer in die beim Öffnen des Kanals vorhandene Partition, hinein.

Blockausführung (BLOCK-EXECUTE)

Noch ein selten gebrauchter Befehl, BLOCK-EXECUTE, ladet einen bestimmten Block in den Laufwerkspeicher, wo er, wie ein Maschinensprachprogramm abläuft. Das DOS führt ein JSR am Anfang des Puffers, der vom Direktzugriffskanal benutzt wird, durch. Hier sollten Sie eine Puffernummer beim Öffnen des Direktzugriffskanal angeben, es sei denn die Maschinensprache wurde geschrieben, um verschoben zu werden. Die Maschinensprachroutine sollte auch mit einem RTS enden, wenn Sie zur DOS-Steuerung zurückkehren wollen. Hier die Syntax:

```
PRINT#lf, "B-E"; ch; n; t; s
```

wo:	lf	= die logische Dateinummer des Befehlskanals ist
	ch	= die Kanalnummer der Direktzugriffsdatei ist
	n	= die Partitionsnummer ist (immer 0)
	t	= die Spur ist, wo der Block gespeichert ist
	s	= der Sektorbereich des Blocks ist

Folgendes Beispiel nimmt an, daß ein Block auf Spur 2, Sektor 0 im Puffer#2 (\$0500) angesprochen werden soll:

```
10 OPEN15, 8, 15
20 OPEN2, 8, 2, "#2"
30 PRINT#15, "B-E"; 2; 0; 2; 0
40 CLOSE2:CLOSE15
```


Speicherbefehle

Die Speicherbefehle erlauben das Lesen oder Schreiben von bzw. zu Laufwerkspeichern und führen Maschinensprachprogramme im Laufwerksspeicher aus. Die Speicherbefehle sind für viele Anwendungsprogramme brauchbar, sollten aber vorsichtig benützt werden, um die vom DOS gebrauchten wichtigen Bereiche oder andere Daten nicht zu verfälschen.

Das Lesen von HD-Speicherbereichen

Der MEMORY-READ-Befehl läßt das Lesen von HD-Speicherbereichen zu. Dieser Befehl ist dem BASIC PEEK-Befehl ähnlich, mit dem einen Unterschied, daß MEMORY-READ den Laufwerksspeicher statt Computerspeicher weitergibt. Das Lesen von dem Laufwerksspeicher ist in vielen Anwendungsprogrammen nützlich, bei der die Art des Laufwerks für eine bestimmte Geräteadresse festzustellen ist. Die Syntax für MEMORY-READ ist:

```
PRINT#lf, "M-R"CHR$(ml)CHR$(mh)CHR$(nb)
```

wo: lf = die logische Dateinummer des Befehlskanals ist
 ml = das niedrigste Byte der Anfangsspeicheradresse ist
 mh = das höchste Byte der Anfangsspeicheradresse ist
 nb = die Zahl der zu lesenden Bytes ist

Der Wert der Bytezahl kann zwischen 0 und 255 sein. Es ist eher unwahrscheinlich, daß Sie von einer Diskette 0 Bytes lesen wollen. Auf der anderen Seiten kann es oft vorkommen, daß Sie von der Diskette 256 Bytes lesen möchten. Deshalb wurde der Wert 0 (Null) auf 256 geändert.

Nachdem ein MEMORY-READ-Befehl gesandt wurde, werden die bestimmten Bytes über den Fehlerkanal zurückgesandt. Mit einem GET#-Befehl, können Sie diese Bytes einzeln lesen.

Um das niedrigste bzw. höchste Byte einer Dezimaladresse festzustellen, benützen Sie folgende Formel:

```
HB=INT(AD/256):LB=AD AND 255
```

wo: AD = die Dezimaladresse ist, wo Sie lesen anfangen wollen
 HB = das höchste Byte der Anfangsspeicheradresse ist
 LB = das niedrigste Byte der Anfangsspeicheradresse ist

Hier ist ein Beispielprogramm, das feststellt, ob das Laufwerk, vom dem Sie lesen wollen, eine Festplatte ist:

```
10 OPEN15,12,15
20 PRINT#15,"M-R"CHR$(160)CHR$(254)CHR$(6)
30 FORI=1TO6:GET#15,B$:A$=A$+B$:NEXT
70 CLOSE15
80 IFA$="CMD HD"THENPRINT"HD PRESENT":END
90 PRINT"NOT A CMD HD":END
```

VERMERK: Der MEMORY-READ-Befehl wird nach einer Seitengrenze nicht weitergelesen.

Das Schreiben zu HD-Speicherbereichen

Der MEMORY-WRITE-Befehl ermöglicht das Schreiben zu HD-Speicherbereichen. Dieser Befehl ist dem BASIC-POKE-Befehl ähnlich, mit dem einen Unterschied, daß MEMORY-WRITE in den Laufwerksspeicher statt in den Computerspeicher schreibt. MEMORY-WRITE ist dort nützlich, wo man Maschinensprachroutine ins Laufwerk stecken, oder Daten direkt zu einem Puffer schreiben will. Die Syntax für MEMORY-WRITE ist folgende:

```
PRINT#lf, "M-W"CHR$(ml)CHR$(mh)CHR$(nb)CHR$(d) ...
```

wo: lf = die logische Dateinummer des Befehlskanals ist
 ml = das niedrigste Byte der Anfangsspeicheradresse ist
 mh = das höchste Byte der Anfangsspeicheradresse ist
 nb = die Bytezahl ist, die geschrieben werden soll
 d = der Wert des Datenbytes ist, das geschrieben werden soll (Wenn mehr als ein Wert, benützen Sie mehrere CHR\$(Befehle) oder Schließenvariablen.)

Der Wert der Bytezahl kann zwischen 1 und 248 sein. Im MEMORY-WRITE-Befehl werden die Bytes am Ende des Befehls gesetzt.

Um die niedrigsten bzw. höchsten Bytes ausfindig zu machen, benützen Sie die Formel im obigen MEMORY-WRITE-Befehl.

Hier haben Sie ein Beispielprogramm, das den Gebrauch des MEMORY-WRITE-Befehls illustriert:

```
10 FORI=1TO248
20 A$=A$+CHR$(I)
30 NEXT
40 OPEN15,12,15
50 OPEN#15,"M-W"CHR$(0)chr$(8)CHR$(248)A$
60 CLOSE15
70 END
```

Speicherausführung (Memory Execute)

Mit dem MEMORY-EXECUTE-Befehl wird dem Laufwerk (der HD) diktiert, die Ausführung eines Maschinensprachprogramms zu beginnen. Wenn dieser Befehl gegeben wird, wird ein JSR von der Laufwerksteuerungsschleife durchgeführt. Um die Steuerung des Laufwerks an das DOS zurück zu geben, soll das durchgeführte Maschinensprachprogramm mit einem RTS- Befehl enden. Die Syntax für diesen Befehl ist:

```
PRINT#lf,"M-E"CHR$(ml)CHR$(mh)
```

wo: lf = die logische Dateinummer des Befehlskanals ist
ml = das niedrigste Byte der Speicheradresse ist, wo die Ausführung beginnen soll
mh = das höchste Byte der Speicheradresse ist, wo die Ausführung beginnen soll

Dieses Beispiel führt ein Maschinensprachprogramm bei \$0800 im Laufwerkspeicher aus:

```
PRINT#15,"M-E"CHR$(0)CHR$(8)
```

User-Befehle

Mit den User-Befehlen können Sie auf Speicherbereiche innerhalb der HD springen, BURST- UTILITY-Befehle ausführen, und nützliche Ersätze für die BLOCK-READ- bzw. BLOCK-WRITE-Befehle beschaffen. Der folgende Text bietet einen Überblick über die Burst-Utility-Befehle. Zusätzliche Informationen bietet später der Burst-Befehl-Abschnitt. Alle USER-Befehle werden über den DOS-Befehlskanal gesandt (Sekundäradresse 15).

U0 Utility-Befehle

Der U0-Befehl hat viele Anwendungsmöglichkeiten. Ohne Parameter, führt er ein User-Vektor-Reset durch. Wenn er mit bestimmten Parametern kombiniert wird, wird ein Weg angeboten, um Burst und Utility(Dienst)-Befehle zu senden. Hier die Syntax:

```
PRINT#lf,"U0[+|-|>[bu]]"[+CHR$(d1)[+CHR$(d2)]]
```

wo: lf = die logische Dateinummer des Befehlskanals ist
bu = das (die) Dienstbefehlszeichen ist (wenn notwendig)
d1 = das Burst-Dienstdatenbyte ist (wenn notwendig)
d2 = das zweite Burst-Dienstdatenbyte ist (wenn notwendig)

Weil viele dieser Befehle im Absatz mit dem Titel "CHGUTL-Utility" in Detail behandelt werden, geben wir hier die allgemeine Syntax jedes Befehls weiter. Nehmen Sie an, daß jeder Befehl mit einem "PRINT#lf," beginnt, und lf mit der logischen Dateinummer für den Befehlskanal ersetzt werden soll.

"U0"	Reset Uservektore auf Default
"U0+"	*Serielle-Geschwindigkeit für den VIC-20 einstellen
"U0-"	*Serielle-Geschwindigkeit für den C64/128 einstellen
"U0>"+CHR\$(d)	Gerätadresse ändern (d=Gerätadresse)
"U0>B0"	Schnell-Seriell ausschalten
"U0>B1"	Schnell-Seriell einschalten
"U0>R"+CHR\$(r)	DOS-Neuversuche einstellen (r= Zahl der Neuversuche)

"U0>S"+CHR\$(i)	Sektorüberlappen für die 1571 Burstbefehle einstellen (i=überlappen)
"U0>T"	*Test ROM-Kontrollsumme
"U0>V0"	*Verifizieren ein
"U0>V1"	*Verifizieren aus
"U0>M0"	*1571 Einseit-Modus einstellen
"U0>M1"	*1571 Doppelseit-Modus einstellen
"U0>MR"+CHR\$(sp)+CHR\$(np)	Burstspeicher lesen (sp = Anfangsseite, np=Seitenzahl)
"U0>MW"+CHR\$(sp)+CHR\$(np)	Burstspeicher schreiben (sp=Anfangsseite, np=Seitenzahl)

* Befehle, die zwar akzeptiert werden, aber keine Funktion haben.

Blöcke lesen mit U1

Der U1-Befehl wird benutzt, um bestimmte Blöcke innerhalb einer Partition zu lesen. Der U1-Befehl wird anstelle von BLOCK-READ benutzt und ist daher ein Direktzugriffsbefehl. Der U1-Befehl liest einen ganzen Block in den Puffer, der vom Direktzugriffskanal gebraucht wird. Die Syntax für den U1-Befehl ist:

```
PRINT#lf, "U1"; ch; n; t; s
```

wo:	lf	= die logische Dateinummer des Befehlskanals ist
	ch	= die Kanalnummer der Direktzugriffsdatei ist
	n	= die Partitionsnummer ist (immer 0)
	t	= die Spur ist, von der Sie einen Block lesen wollen
	s	= der Sektor ist, von dem Sie einen Block lesen wollen

Die Kanalnummer ist die gleiche wie die Sekundäradresse, die beim Öffnen der Direktzugriffsdatei benutzt wurde.

Die Partitionsnummer muß immer 0 (Null) sein, weil die Direktzugriffsbefehle immer auf die vorhandene Partition zugreifen.

Nachdem der Block mit Hilfe des U1-Befehls gelesen wurde, können Sie die Daten aus dem Puffer mit dem GET#-Befehl herausholen. Davor können Sie den BUFFER-POINTER-Befehl einsetzen, um bestimmte Bytes herauszuholen. Die Zeichen 'UA' können mit 'U1' ersetzt werden. Ein Beispiel dafür finden Sie unter dem Abschnitt: "Das Lesen und Schreiben von Daten mit Direktzugriff".

Blöckeschreiben mit U2

Mit dem U2-Befehl kann man Blöcke zu einer Partition dazuschreiben. Der U2-Befehl wird anstelle von BLOCK-WRITE benutzt, und ist daher, ein Direktzugriffsbefehl. Der U2-Befehl schreibt einen ganzen Block von dem Puffer, der vom Direktzugriffskanal gebraucht wird. Die Syntax für den U2-Befehl ist:

```
PRINT#lf, "U2"; ch; n; t; s
```

wo:	lf	= die logische Dateinummer des Befehlskanals ist
	ch	= die Kanalnummer der Direktzugriffsdatei ist
	n	= die Partitionsnummer ist (immer 0)
	t	= die Spur ist, zu der Sie einen Block schreiben wollen
	s	= der Sektor ist, zu dem Sie schreiben wollen

Die Kanalnummer ist identisch mit der Nummer der Sekundäradresse, die beim Öffnen der Direktzugriffsdatei benutzt wurde.

Die Partitionsnummer muß immer 0 (Null) sein, weil die Direktzugriffsbefehle immer auf die vorhandene Partition zugreifen.

Bevor Sie den U2-Befehl benutzen, sollten Sie die Daten mit dem PRINT#-Befehl in den Direktzugriffskanalphuffer hineinschreiben. Daten können vorher in bestimmte Bytes mit dem BUFFER- POINTER-Befehl eingegeben werden. Sie können auch die Zeichen UB mit U2 ersetzen. Sehen Sie das Beispiel unter dem Abschnitt: "Das Lesen und Schreiben von Daten mit Direktzugriff".

User-Sprung-Befehle

Die restlichen USER-Befehle sind besonders für Maschinensprachprogrammierer nützlich. Mit diesen Befehlen springt man zu bestimmten Speicherstellen im Laufwerkspeicher. Sechs dieser Sprünge finden in der Speicherstelle des dritten Puffers statt. Sie werden bei jeder dritten Stelle durchgeführt. Jeder Befehl hat eine Parallelbezeichnung(Alias). Die folgende Tabelle zeigt die User-Sprung-Befehle, ihre Alias und die Speicherstellen, zu denen sie springen:

BEFEHL	ALIAS	ZIELADRESSE
U3	UC	\$0500
U4	UD	\$0503
U5	UE	\$0506
U6	UF	\$0509
U7	UG	\$050C
U8	UH	\$050F

Die restlichen 2 USER-Befehle werden benutzt, um das Laufwerk zu reseten. Hier sind ihre Definitionen:

BEFEHL	ALIAS	DEFINITION
U9	UI	Warm-Reset(Laufwerksvariablen wenig be-einflusst)
U9	UJ	Kalt-Reset (ändert Partitionsnummer nicht)

Die Syntax für die USER-Befehle ist:

```
PRINT#lf, "Ux"
```

wo: lf = die logische Dateinummer des Befehlskanals ist
 x = das Zeichen des verlangten USER-Befehls

Burst-Befehle

Die HD gebraucht 2 verschiedene Burst-Befehlssätze, die von der Partitionsart abhängig sind. Mit dem 'HD-Burst-Command-Instruction-Set' (HDBCIS) kann der User eigene Quittungsbetriebs- Routine (Handshaking genannt) durchführen. Diese Befehle sind wegen der Kompatibilität, und nicht wegen der Geschwindigkeit, vorhanden.

1541/1541 HD-Native-Modus-Burst-Befehle

Diese Befehle emulieren so nahe wie möglich den 1571-BCIS. Befehle oder Befehlsschalter, die für MFM-Formatieren bestimmt sind, werden von der HD nicht emuliert.

Lesen

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	T	E	B	X	0	0	0	N
03	Zielspur							
04	Zielsektor							
05	Zahl der Sektoren							
06	Nächste Spur (optional)							

BEREICH: Sehen Sie ANHANG C (C3,C4,C14)

FLAGS: T - Daten transferieren (1=kein transferieren)
 E - Fehler ignorieren (1=ignorieren)
 B - Puffer-Transfer alleine (1=Puffer-Transfer alleine)
 X - nicht belegt
 N - Partitionsnummer (nur 0)

PROTOKOLL: Burst-Quittungsbetrieb

HINWEIS: Vor dem Lesen oder Schreiben einer Partition, muß die Partition zuerst entweder mit dem INQUIRE DISK- oder QUERY DISK FORMAT-Befehl protokolliert werden. Dies muß auch dann getan werden, wenn Sie Partitionen wechseln.

AUSGABE: Ein Burst-Status-Byte, gefolgt von Burstdaten wird für jeden Sektor, der transferiert wird, gesandt. Ein Fehler verhindert das Senden der Daten, es sei denn, das IGNORE ERROR-Bit im Byte 02 hat den Wert von 1 bekommen.

Schreiben

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	T	E	B	X	0	0	1	N
03	ZIELSPUR							
04	ZIELSEKTOR							
05	ZAHL DER SEKTOREN							
06	NÄCHSTE SPUR							

BEREICH: Sehen Sie Anhang C (C3,C4,C14)

FLAGS:

- T - Daten transferieren (1=kein transferieren)
- E - Fehler ignorieren (1=ignorieren)
- B - Puffer-Transfer alleine (1=Puffer-Transfer alleine)
- X - nicht belegt
- N - Partitionsnummer (nur 0)

PROTOKOLL: Schnell-Seriell-Ausgabe, Schnell-Seriell-Eingabe setzen, Uhr hinunterziehen, warten auf Status, Uhr loslassen. Vom Anfang für mehrere Sektorausgabe wiederholen.

HINWEIS: Vor dem Lesen oder Schreiben einer Partition, muß die Partition zuerst entweder mit dem INQUIRE DISK- oder QUERY DISK FORMAT-Befehl protokolliert werden. Dies muß dann getan werden, wenn Sie Partitionen wechseln

EINGABE: Wirt muß Burstdaten transferieren.

AUSGABE: Ein Burst-Status-Byte folgt jeder WRITE-Durchführung

Diskettenabfrage

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	X	X	X	X	0	1	0	N

FLAGS:

- X - nicht belegt
- N - Partitionsnummer (nur 1 unterstützt)

PROTOKOLL: Burst-Quittungsbetrieb

AUSGABE: Ein Burst-Status-Byte folgt jeder INQUIRE DISK-Durchführung

MF-Format

Der FORMAT MF-Format-Befehl wird von der HD nicht unterstützt.

GCR-Format (ohne Verzeichnis)

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	X	X	X	X	0	1	1	N
03	M	X	X	X	X	X	X	X
04	ID (LOW BYTE)							
05	ID (HIGH BYTE)							

FLAGS: X - nicht belegt
 N - Partitionsnummer (nur 0)
 M - Modus (nur 0)

PROTOKOLL: Allgemein

HINWEIS: Dieser Befehl muß von einem INQUIRE DISK- oder QUERY DISK FORMAT-Befehl gefolgt werden, um protokolliert zu werden.

AUSGABE: keine Ausgabe

VERMERK: Die ID-Information wird in Wirklichkeit nicht auf die Platte geschrieben, da die HD diese Information nicht auf jede Spur, wie bei Floppy- Disketten, speichert.

Sektor Interleave (]berlappen)

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	W	X	X	0	1	0	0	N
03	INTERLEAVE							

FLAGS: W - Lesen/Schreiben (0=Schreiben,1=Lesen vorhanden)
 X - nicht belegt
 N - Partitionsnummer (nur 0)

PROTOKOLL: Burst-Quittungsbetrieb (W=1)

HINWEIS: Dies ist ein Software-Überlappen (interleave), das für Multi-Sektorburst READ und WRITE-Befehle gebraucht wird.

AUSGABE: kein Output (W=0) Vorhandenes Interleave-Burstbyte (W=1)

Query(Abfrage) Disketten-Format

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	F	X	X	X	1	0	1	N
03	OFFSET (optional-F=1)							

FLAGS: F - Forcierte Flags (F=1 Der Kopf wird mit dem Offset im Byte 03 gezwungen.)
 X - nicht belegt
 N - Partitionsnummer (nur 0)

PROTOKOLL: Burst-Quittungsbetrieb

HINWEIS: Partitionen werden protokolliert.

AUSGABE: Burst-Status-Byte

Abfrage (Inquire)-Status

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	W	C	X	0	1	0	1	N
03	NEU STATUS (optional-W=0)							

FLAGS: W - Schreib Flags (W=0-Schreiben/W=1-zum vorhandenen Status zurückkehren)
 C - Ändern (C=1 und W=0 - Partition protokollieren/C=1 und W=1 - Zurückkehren)
 X - nicht belegt
 N - Partitionsnummer (nur 0)

PROTOKOLL: Burst-Quiterungsbetrieb (W=1)

HINWEIS: Diese Methode liest den vorhandenen Status.

AUSGABE: keine Ausgabe (W=0) Burst-Status-Byte (W=1)

CHGUTL Utility

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	X	X	X	1	1	1	1	0
03-??	UTILITY-BEFEHL UND PARAMETER							

FLAGS: X - nicht belegt

BEFEHLE: CHR\$(n) Gerätadresse ändern
 n=neue Gerätadresse

Bn Schnell-Seriell-Modus
 n=0 - Schnell-Seriell aus
 n=1 - Schnell-Seriell ein

Hn Headauswahl (Fehler wird angezeigt)
 n=0 - Seite 0
 n=1 - Seite 1

Mn Modusauswahl (ignoriert)
 n=0 - einseitig
 n=1 - doppelseitig

MR+CHR\$(a)+CHR\$(p) Speicher lesen*
 a=Anfangsadresse des höchsten Bytes
 p=Seitenzahl zum Lesen

MW-CHR\$(a)+CHR\$(p) Speicher schreiben*
 a=Anfangsadresse des höchsten Bytes
 p=Seitenzahl zum Lesen

R+CHR\$(n) DOS-Neuersuche
 n=Zahl der Neuversuche bis Fehler

S+CHR\$(n) DOS-Sektor-Interleave
 n=Sektor-Interleave-Offset

T Test-ROM-Kontrollsumme (ignoriert)

Vn Verifizieren (ignoriert)
 n=0 - Verifizieren ein
 n=1 - Verifizieren aus

*VERMERK: Burst MR und MW benutzen gewöhnliche Burst-Protokolle (ohne Satusbyte).

Schnell-Laden Utility

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	P	X	X	1	1	1	1	1
03-??	DATEINAME							

FLAGS: P - Nicht-Programm-Dateiflags (P=0 - Programmdatei/ P=1 - Nicht-Programm-Datei)
 X - nicht belegt

PROTOKOLL: Burst-Quittungsbetrieb

AUSGABE: Burst-Status-Byte vor jedem Sektor, der transferiert wird. In einer Programm-Datei soll die Ladeadresse normal behandelt werden.

STATUS: 0000000X = OK
 00000010 = Datei nicht gefunden
 00011111 = EOI Eingabeende (letzter Sektor)

Jeder andere Statuswert ist ein Datei-Lese-Fehler. Das Byte nach dem EOI zeigt die übrigen Bytezahl in der Datei.

1581/1581 CP/M Burstbefehle

Diese Befehle sollen den 1581 BCIS so nahe wie möglich emulieren. Befehle oder Befehlsflags, die für das MFM-Formatieren bestimmt sind, sind von der HD nicht emuliert.

Lesen

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	L	E	X	S	0	0	0	N
03	ZIELSPUR							
04	ZIELSEKTOR							
05	ZAHL DER SEKTOREN							
06	NÄCHSTE SPUR (optional)							

BEREICH:	Logisches Format:	Spuren:	01-80 (einseitig)
		Sektoren:	00-39 (je 256 Bytes)
	Physikalisches Format:	Spuren:	00-79 (beidseitig)
		Sektoren:	01-10 (je 512 Bytes)

FLAGS: L - Daten transferieren (1=kein transferieren)
 E - Fehler ignorieren (1=ignorieren)
 X - nicht belegt
 S - Seite (0 oder 1)
 N - Partitionsnummer (nur 0)

PROTOKOLL: Burst-Quittungsbetrieb

HINWEIS: Vor dem Lesen oder Schreiben einer Partition, muß die Partition zuerst entweder mit dem INQUIRE DISK- oder QUERY DISK FORMAT-Befehl protokolliert werden. Dies muß dann getan werden wenn Sie Partitionen wechseln.

AUSGABE: Ein Burst-Status-Byte, gefolgt von Burstdaten wird für jeden Sektor, der transferiert wird, gesandt. Ein Fehler verhindert das Senden der Daten, es sei denn, das IGNORE ERROR-Bit im Byte 02 hat den Wert von 1 bekommen.

Schreiben

[illegible]

BEREICH:

Logisches Format:	Spuren: 01-80 (einseitig)
	Sektoren: 00-39 (je 256 Bytes)
Physikalisches Format:	Spuren: 00-79 (beidseitig)
	Sektoren: 01-10 (je 512 Bytes)

FLAGS:

L	-	Logische Flags (0=physikal/1=logisch)
E	-	Fehler ignorieren (1=ignorieren)
X	-	nicht berücksichtigt
S	-	Seite (0 oder 1)
N	-	Partitionsnummer (nur 1)

PROTOKOLL: Ausgabedaten, Schnell-Seriell einsetzen, Uhr hinunterziehen, warten auf Statusbyte, Uhr loslassen.
Für mehrere Sektoren wiederholen.

HINWEIS: Vor dem Lesen oder Schreiben einer Partition, muß die Partition zuerst entweder mit dem INQUIRE DISK- oder QUERY DISK FORMAT-Befehl protokolliert werden. Dies muß dann getan werden, wenn Sie Partitionen wechseln.

EINGABE: Wirt muß Burstdaten transferieren.

AUSGABE: Ein Burst-Status-Byte folgt jeder WRITE-Durchführung.

Diskettenabfrage

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	X	X	X	X	0	1	0	N

FLAGS: X - nicht belegt
N - Partitionsnummer (nur 1 unterstützt)

PROTOKOLL: Burst-Quittungsbetrieb

AUSGABE: Ein Burst-Status-Byte folgt jeder INQUIRE DISK-Durchführung

Format

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	M	X	X	X	0	1	1	N
03	SEKTORGRÖSSE (optional)							
04	LETZTE SPURNUMMER (optional)							
05	SEKTORENTZAHL (optional)							
06	ANFANGSSPUR (optional)							
07	FÜLLBYTE (optional)							
08	ANFANGSSEKTOR (optional)							

FLAGS: M - Modus (0=logisches Format mit Verzeichnis und BAM/
1=physikalisches Format mit wahlfreien Bytes 03-08)
X - nicht belegt
N - Partitionsnummer (nur 0)

OPTIONEN: Bytes 03-08 sind optional und nur angenommen, wenn physikalisches Formatieren bestimmt ist (M=1). Wenn Sie für diese Bytes keine bestimmten Werte wählen, werden systemvorgegebene Werte benutzt. Folgende Liste zeigt den legalen Bereich und vorgegebene Werte dieser Bytes.

Byte 03 - muß 2 sein
Byte 04 - 00 bis 79 (79 vorgegeben)
Byte 05 - muß 10 sein
Byte 06 - 00 bis 79 (79 vorgegeben)
Byte 07 - \$00 bis \$FF (\$E5 vorgegeben)
Byte 08 - muß 1 sein

PROTOKOLL: Gewöhnlich

HINWEIS: Dieser Befehl muß von einem INQUIRE DISK- oder QUERY DISK FORMAT-Befehl gefolgt werden, um in die Partition protokolliert zu werden.

AUSGABE: Keine Ausgabe

VERMERK: Der Verzeichnis-Header dieses Befehls beinhaltet die 'CMD HD'-Schleife für den Diskettenamen und '89' für die ID.

Query (Abfrage) Disketten-Format

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	F	X	T	S	1	0	1	N
03	OFFSET (optional-F=1)							

FLAGS: F - Forciert (F=1 Der Kopf wird mit dem Offset im Byte 03 gezwungen.)
X - nicht belegt
T - Sektortabelle (T=1 sendet Sektortabelle)
S - Seite
N - Partitionsnummer (nur 0)

PROTOKOLL: Burst-Quittungsbetrieb

HINWEIS: Partitionsformat bestimmt

AUSGABE: Burst-Status-Byte, Sektorenzahl (10). logische Spurnummer (0 oder Spurnummer, die in 03 bestimmt ist), niedrigste logische Sektornummer (01), höchste logische Sektornummer (10), Interleave-Wert (1), Sektortabelle (wenn T=1)

VERMERKE: Die Sektortabelle ist eine Reihe von 10 Bytes (1,2,3,4,5,6,7,8,9,10). Statusbyte ist vom Spuroffset 0, es sei denn F-Bit gesetzt wird. Dann ist Status von der Offsetspur im Byte 03. Wenn ein Fehler passiert, folgen keine Bytes dem Statusbyte.

Abfrage-(Inquire)status

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	W	C	M	0	1	1	0	N
03	NEUER STATUS (W=0)							
04	NEU ODER MASKE (M=1)							
05	NEU UND MASKE (M=1)							

FLAGS: W - Schreiben (W=0 - schrei neuen Statuswert/W=1 Status wieder-geben)
C - Ändern (C=1 und W=0 - Partitionen protokollieren/C=1 und W=1 - Status wiedergeben)
M - Schreib AND/OR Masken (M=1)
N - Partitionsnummer (nur 0)

PROTOKOLL: Burst-Quittungsbetrieb (W=1), gewöhnlich (W=0)

HINWEIS: Dies ist eine Methode, den vorhandenen Status zu lesen, und die Status-Maskenwerte zu ändern.

AUSGABE: keine Ausgabe (W=0). Burst-Status-Byte (W=1)

CHGUTL Utility

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	X	X	X	1	1	1	1	0
03-??	UTILITY-BEFEHL UND PARAMETER							

FLAGS: X - nicht belegt

BEFEHLE: CHR\$(n) Gerätadresse ändern
 n=neue Gerätadresse
 Bn Schnell-Seriell-Modus
 n=0 - Schnell-Seriell aus
 n=1 - Schnell-Seriell ein
 Hn Headauswahl (Fehler wiedergegeben)
 n=0 - Seite 0
 n=1 - Seite 1
 Mn Modusauswahl (ignoriert)
 n=0 - einseitig
 n=1 - doppelseitig
 MR+CHR\$(a)+CHR\$(p) Speicher lesen*
 a=Anfangsadresse des höchsten Bytes
 p=Seitenzahl zum Lesen
 MW-CHR\$(a)+CHR\$(p) Speicher schreiben*
 a=Anfangsadresse des höchsten Bytes
 p=Seitenzahl zum Lesen
 R+CHR\$(n) DOS-Neuversuche
 n=Zahl der Neuversuche bis Fehler
 S+CHR\$(n) DOS-Sektor-Interleave
 n=Sektor-Interleave-Offset
 T Test-ROM-Kontrollsumme (ignoriert)
 Vn Verifizieren (ignoriert)
 n=0 - Verifizieren ein
 n=1 - Verifizieren aus

*VERMERK: Burst MR und MW benötigen gewöhnliche Burst-Protokolle (ohne Statusbyte).

Dump Schnellspurpuffer

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	F	S	X	1	1	1	0	1
03	PHYSIKALISCHE SPURNUMMER							

FLAGS: F - Immer schreiben (F=1 schreiben auch wenn nicht "schmutzig")
 S - Seite
 X - nicht belegt

VERMERK: Dieser Befehl wird von der HD angenommen, hat aber keine Bedeutung, da die HD keinen Schnellspurpuffer hat.

Schnell-Laden Utility

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	0	0
02	P	X	X	1	1	1	1	1
03-??	DATEINAME							

FLAGS: P - Nicht-Programm-Dateiflags (P=0 - Programmdatei/ P=1 - Nicht-Programm-Datei)
X - nicht belegt

PROTOKOLL: Burst-Quittungsbetrieb

AUSGABE: Burst-Status-Byte vor jedem Sektor, der transferiert wird. In einer Programm-Datei soll die Ladeadresse normal behandelt werden.

STATUS: 0000000X = OK
00000010 = Datei nicht gefunden
00011111 = EOI Eingabeende (letzter Sektor)

Statuswerte von 3 bis 15 sind Datei-Lese-Fehler. Das Byte nach dem EOI zeigt die übrigen Bytezahl in der Datei.

Burst Burst-Status-Byte

BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	M	N	S1	S0	C3	C2	C1	C0

BIT-DEFINITIONEN: M - Modus: 0 normales Format
1 fremdes Diskettenformat (nicht vorgegebenes physikalisches Format oder vorgegebenes physikalisches Format ohne Verzeichnis und BAM)

N - Partitionsnummer (nur 0)

S - Sektorgröße: S1 S0 Definition

0 0 256 Bytes pro Sektor (1541/1571/HD Native)

1 0 512 Bytes pro Sektor (1581)

C - Kontroll(Regler)-Status: C3 C2 C1 C0 Definition

0 0 0 X OK

0 0 1 0 Prüfen-Verlangen (SCSI)

0 0 1 1 Ziel beschäftigt (SCSI)

0 1 0 0 Nicht legaler Block

0 1 0 1 Kein Selektieren (SCSI)

0 1 1 0 Formatierfehler/ Beschäftigung beendet (SCSI)

0 1 1 1 Nicht legale Phase (SCSI)

1 0 0 0 Schreibschutzfehler

1 0 0 1 Prüfen-Verlangen versagt (SCSI)

1 0 1 0 Nicht definiert

1 0 1 1 Zwischenstufestatus OK

1 1 0 0 Reservierungskonflikt (SCSI)

1 1 0 1 Hardwarefehler

1 1 1 0 Syntaxfehler/ illegaler Job

1 1 1 1 Laufwerk nicht vorhanden

Spezielle Lader

Commodores DOS-Utility-Loader und Autoboot-Loader sind ins HD-DOS eingebaut worden. Sie können daher eine Datei in den HD-Speicher laden und sofort laufen lassen. Das 'und' Zeichen (&) vor dem Dateinamen wird über den Befehlskanal gesandt. Wenn ein Programm mit dem Namen "COPYRIGHT CMD 89" auf der vorhandenen Partition darauf ist, wird es nachdem ein Kaltstart oder Reset durchgeführt wird, geladen und sofort gestartet. Die Syntax für diesen Lader ist:

```
OPENIf, dv, 15:PRINT#If, "&[n] [pfad]:dateiname":CLOSEIf
```

wo: If = die logische Dateinummer des Befehlskanals ist
 dv = die HD-Gerätadresse ist
 pfad = der Native-Modus-Partition-Unterverzeichnispfad ist
 dateiname = der Dateiname ist, der im HD-Speicher geladen und gestartet werden soll

VERMERK: Weil das HD-Speicher-Layout anders ist als das von anderen Laufwerken, müssen Programme, die in die Speicher anderer Laufwerke geschrieben wurden, modifiziert werden.

Job-Queue (Warteschlange) Anweisungen

Die Job-Queue-Puffer sind bestimmte Stellen, die benützt werden, um Befehle und Parameter an den Laufwerksregler zu leiten, die dann Diskettenzugriffsfunktionen durchführen. Der Programmierer kann direkt auf die Job-Queue zugreifen. Verschiedene Bereiche der HD sind für die Job-Queues bestimmt. Die HD kann daher, die Job-Queue-Stellen der 1541,1571 und der 1581 emulieren. In allen Fällen werden die Job-Codes und Parameter bei der tatsächlichen Durchführung auf die Native-Modus-Job-Queue gesetzt. Deshalb empfehlen wir den Gebrauch von Native-Modus-Job-Queues, wenn Sie Routine für die HD schreiben.

Job-Queue Befehl-Codes

<u>Code</u>	<u>Name</u>	<u>Beschreibung</u>
\$80	READ	Liest logischen Block mit Spur und Sektorparametern
\$82	CTRL_RESET	Reset an Diskettenregler
\$84	MOTOR_ON	nicht implementiert (OK-Status wiedergegeben)
\$86	MOTOR_OFF	nicht implementiert (OK-Status wiedergegeben)
\$88	MOTOR_ON	nicht implementiert (OK-Status wiedergegeben)
\$8A	MOTOR_OFF	nicht implementiert (OK-Status wiedergegeben)
\$8C	SEEK_PHYS_OFF	nicht implementiert (OK-Status wiedergegeben)
\$8E	FRMT_TRK	nicht implementiert (OK-Status wiedergegeben)
\$90	WRITE	Schreibt logischen Block mit Spur und Sektorparametern
\$92	DSK_IN_DRV	nicht implementiert (OK-Status wiedergegeben)
\$94	ACTIV_ON	Schaltet ACTIVITY LED ein
\$96	ACTIV_OFF	Schaltet ACTIVITY LED aus
\$98	ERR_ON	Schaltet ERROR LED ein
\$9A	ERR_OFF	Schaltet ERROR LED aus
\$9C	SID_SEL	nicht implementiert (OK-Status wiedergegeben)
\$9E	CACHE_MOV	nicht implementiert (OK-Status wiedergegeben)
\$A0	VERIFY	nicht implementiert (OK-Status wiedergegeben)
\$A2	WRT_CACHE	nicht implementiert (OK-Status wiedergegeben)
\$A4	PHYS_BUF0	Liest physikalischen Sektor zu \$0300
\$A6	BUF0_PHYS	Schreibt \$0300 zum physikalischen Sektor
\$A8	SEEK_PHYS	nicht implementiert (OK-Status wiedergegeben)
\$AA	READ_LOG_NQ	Liest den logischen Block, ohne daß er in den Job-Queue-Puffer gesetzt wird.
\$AC	WRT_LOG_NQ	Schreibt den logischen Block, ohne daß er vom Job-Queue-Puffer entfernt wird.
\$B0	SEEK	nicht implementiert (OK-Status wiedergegeben)
\$B2	READ_PHYS_NQ	Liest den logischen Block, ohne daß er in den Job-Queue-Puffer gesetzt wird.
\$B4	WRT_LOG_NQ	Schreibt den logischen Block, ohne daß er vom Job-Queue-Puffer entfernt wird.
\$B6	WRT_PROT_ON	Prüft den Schreibschutz-Status.
\$B8	SEEK_LOG(N LOG)	nicht implementiert (OK-Status wiedergegeben)
\$C0	BUMP	Sucht logische Spur 0 in einer Partition.
\$CC	RD_LOG_ADDR	Liest logischen Block zu der Zieladresse.

\$CE	WRT_LOG_ADDR	Schreibt logischen Block von der Zieladresse.
\$D0	JUMP	Führt Code in Puffer durch.
\$E0	EXEC	Führt Code in Puffer durch.
\$EC	READ_MULTI	Liest mehrere Systemblöcke (Blockzahl) zur Zieladresse.
\$EE	WRT_MULTI	Schreibt mehrere Systemblöcke (Blockzahl) von der Zieladresse.
\$F0	FRMT_PART	nicht implementiert (OK-Status wiedergegeben)
\$F2	SCSI_RESET	Am SCSI-Bus wird ein RESET durchgeführt.
\$F4	UNIT_RDY	nicht implementiert (OK-Status wiedergegeben)
\$F6	REZERO_UNIT	Bestimmter SCSI-Gerätkopf und LU auf Spur 0 gesetzt.
\$F8	START_UNIT	Gerätesekopf und LUN entparkt
\$FA	STOP_UNIT	Gerätesekopf/schreibkopf und LUN parken.
\$FC	RD_PHYS_MULTI	Liest mehrere physikalische Blöcke (Blockzahl) zu der Zieladresse.
\$FE	WRT_PHYS_MULTI	Schreibt mehrere physikalische Blöcke (Blockzahl) von der Zieladresse.

Job-Queue-Stellen

Adresse	Reihe	Beschreibung
\$0000	- \$0004	1541/1571-Emulations-Modus-Job-Queue
\$0006	- \$000F	1541/1571-Emulations-Modus-Spur und Sektor
\$0002	- \$000A	1581-Emulations-Modus-Job-Queue
\$000B	- \$001C	1581-Emulations-Modus-Spur und Sektor
\$0020	- \$003F	Native-Modus-Job-Queue
\$2800	- \$283F	Native-Modus-Spur und Sektor
\$2840	- \$285F	'Seite'
\$2860	- \$287F	Zieladresse (hoch)
\$2880	- \$289F	Zieladresse (niedrig)
\$28A0	- \$28BF	Blockzahl
\$28C0	- \$28DF	SCSI-Gerät
\$28E0	- \$28FF	SCSI LU

Anhang A

Utilities (Dienstprogramme)

Information über die Utility-Disketten

Eine Vielzahl an Dienstprogramme wurde von CMD erstellt und mit der HD mitgeliefert. Die meisten Programme sind auf den Disketten, einige aber sind auf Partition 1 der HD. Machen Sie sofort Kopien von diesen Programmen! Der Diskettenaufkleber weist auf den Namen und Inhalt der Diskette. Hier sehen Sie eine Liste der mitgelieferten HD-Dienstprogramme:

HD UTILITIES

HD-TOOLS.64	Partitionshilfe für den C64 bzw. C128 im 64-Modus
HD-TOOLS.128	Partitionshilfe für den C128
FCOPY	Dateikopierprogramm
MCOPY	Ganz-Diskette-/Partitions-Kopierer
1541SUB	Unterverzeichnis-Erstellprogramm
1581SUB	Unterverzeichnis-Erstellprogramm
SET HD CLOCK	Zeit/Datum-Einstellprogramm
AUTO-BOOT.128	Boot-Block-Erstellprogramm für den C128
DISK CRACKER HD	Spur- und Sektoreditor und Laufwerkmonitor
FIX BLOCKS	Bad-Block-Ersatzprogramm
PARK HD.64	Lese-/Schreibkopf-Parkprogramm für den C64
PARK HD.128	Lese-/Schreibkopf-Parkprogramm für den C128
BOOTQ	Nicht für den Gebrauch in Europa geeignet
XFER QDATA	Nicht für den Gebrauch in Europa geeignet
REWRITE DOS.64	Programm, um ein neues DOS auf die HD zu speichern
REWRITE DOS.128	Programm, um ein neues DOS auf die HD zu speichern
CREATE SYS.64	Programm, um das ganze HD-DOS-System zu erstellen
CREATE SYS.128	Programm, um das ganze HD-DOS-System zu erstellen
LLFORMAT.64	HD-Low-level-Formatieren (C64)
LLFORMAT.128	HD-Low-level-Formatieren (128)
ADD DRIVE.64	Programm, um ein extra Laufwerk ans HD-System anzuschließen (C64)
ADD DRIVE.128	Das gleiche, nur beim (C128)
DRIVE INFO	Laufwerkmechanismus-Information weitergeben

GEOS UTILITIES

HTime	Autoexec-Datei, die die GEOS-Uhr einstellt
QuickMove	Partitions-Utility für GEOS 64
QuickMove128	Partitions-Utility für GEOS 128
CONFIGURE	Ersetzt GEOS 64 KONFIGURIERE
CONFIGURE_r	Ersetzt GEOS 64 KONFIGURIERE_r
128CONFIGURE	Ersetzt GEOS 128 KONFIGURIERE
128CONFIGURE_r	Ersetzt GEOS 128 KONFIGURIERE_r

CP/M UTILITIES

SPORT.COM	Seriell-Port-Befehlsprogramm
-----------	------------------------------

Programmbeschreibung

Die verschiedenen Programme auf den mitgelieferten Disketten sind zum Teil schon beschrieben worden. Wo dies nicht der Fall ist, finden Sie in diesem Anhang die Beschreibung. Einige dieser Programme sind PD-Software, daher liegt keine vollständige Beschreibung dabei. Für Fachleute, die sich mit derartigen Dienstprogrammen auskennen, wird es kaum Probleme geben.

HD-TOOLS(.64/.128)

Laden und starten Sie Ihre Version dieses Programms. Befolgen Sie die Bildschirmanweisungen. Vergewissern Sie sich, daß das WRITE PROTECT-Lämpchen erloschen ist. Durch einen beliebigen Tastendruck gelangen Sie ins Hauptmenü.

Default Device Number (Vorgegebene Geräteadresse)

Wählen Sie die CHANGE DEFAULT DEVICE NUMBER-Option im Hauptmenü. Das Programm zeigt den Status aller Seriell-Geräte sowohl die vorhandene HD-Geräteadresse. Jetzt können Sie mit den (+) und (-) Tasten und RETURN die neue Geräteadresse wählen.

Wir empfehlen Geräteadresse 8, wenn die HD Ihr einziges bzw. Primär-Laufwerk ist. Wählen Sie eine von den anderen Nummern (9-29), wenn andere Laufwerke am System angeschlossen sind. Wenn Sie 2 andere Laufwerke benützen, empfehlen wir die Geräteadresse 10. Benützen Sie nie die gleiche Nummer für 2 Geräte!

Default Partition Number (Vorgegebene Partitionsnummer)

Diese Option bestimmt welche Partition die aktive Partition beim Einschalten sein wird.

Wählen Sie die CHANGE DEFAULT PARTITION NUMBER-Option im Hauptmenü. Die vorhandene aktive Partitionsnummer wird angeführt. Mit Hilfe der (+) , (-) und RETURN- Tasten wird die neue Partitionsnummer gewählt.

View Partition Table (Partitionstabelle anschauen)

Diese Option zeigt den vorhandenen Status aller Partitionen. Mit (+) und (-) blättern Sie durch die Partitions durch. RETURN führt Sie ins Hauptmenü zurück.

Create a New Partition (Neue Partition erstellen)

Mit dieser Option können Sie neue Partitionen auf der Tabelle erstellen. Das Programm geht automatisch zur nächsten freien Partition, gibt Ihnen aber auch die Möglichkeit aus der Reihenfolge neue Partitionen zu erstellen. Erneut werden die (+) ,(-) und RETURN-Tasten dafür verwendet.

Jetzt müssen Sie die Partitionsart wählen. Wenn Sie keine Emulations-Modus-Partition gewählt haben, müssen Sie die Partitionsgröße bestimmen. Die Größe wird in 256 Block-Inkrementen bestimmt.

Zum Schluß müssen Sie der Partition einen Namen geben (maximal 16 Zeichen). Später können Sie den Namen mit dem HD-DOS-Rename-Partition-Befehl ändern. Jetzt werden Sie aufgefordert die Partition zu erstellen oder den Vorgang abubrechen.

Delete an Old Partition (Alte Partition löschen)

Nach der Wahl dieser Option wird Ihnen die erste Partition gezeigt. Mit den (+), (-) und RETURN-Tasten wählen Sie jene Partition aus, die Sie löschen möchten. Vor dem Löschen wird Ihnen die Möglichkeit gegeben, den Vorgang abubrechen.

VORSICHT: Wenn das Programm beim Löschen einem schlechten Block begegnet, können Sie alle HD-Daten verlieren! Benützen Sie daher vor dem Löschen das Program FIX BLOCKS und lassen Sie alle Partitionen prüfen.

Das Löschen einer Partition kann von 2 Sekunden bis zu über einer Stunde dauern. Dies ist von der Stelle der Partition auf der HD abhängig. Wenn Sie mehrere Partitionen löschen möchten, fangen Sie mit der zuletzt erstellten Partition an und löschen Sie rückwärts. Falls Sie alle Partitionen löschen möchten, benützen Sie das CREATE SYS-Programm.

WARNUNG: Alle Partitionsdaten werden beim Löschen zerstört. Machen Sie zuerst Kopien der Daten, die Sie behalten wollen.

Quit

Erlaubt das Verlassen des HD-TOOLS. Drücken Sie die RESET-Taste auf der HD.

Ratschläge

- Partitionen vorausplanen! Denken Sie beim Erstellen der Partitionen logisch.
- System-Konfigurieren kann für Ihre Daten gefährlich sein! Immer Kopien machen!
- Niemals die RESET-Taste während eines HD-Zugriffs oder bei geöffneter Datei drücken!!
- Partitionen erstellen und ändern nur wenn notwendig.

FCOPY

Dieses Program wurde speziell von CMD entworfen um Dateien jeder Art zwischen 2 Laufwerken oder 2 Partitionen zu kopieren. FCOPY unterstützt Native-Modus-Unterverzeichnisse, 1581-Unterpitionen und REUs, die unter RAMDOS laufen. Dieses Programm kann auch benützt werden um Dateien zu löschen, Verzeichnisse anzuschauen und Diskettenbefehle zu senden.

Set Source Device (F1) (Quellgerät selektieren)

Mit dieser Option wird das Quellgerät, wovon man kopieren will, selektiert. Falls der Laufwerktyp nicht erkannt wird, werden Fragezeichen an Stelle vom richtigen Typ angeführt. Nicht- Commodore-Laufwerke werden hier auch akzeptiert, wenn ihr DOS dem vom Commodore gleicht.

Set Source Device (F5) (Zielgerät selektieren)

Diese Option selektiert das Zielgerät worauf die Dateien kopiert werden.

Set Source Partition (F3) (Quellpartition selektieren)

Wenn die Quellpartition eine CMD-HD-Partition ist, können Sie mit dieser Option die Partition selektieren, wovon die Dateien kopiert werden. Die Partitionsnummer, der Partitionsname und Typ werden im Quellbereich angeführt.

Set Target Partition(F7) (Zielpartition selektieren)

Wenn die Zielpartition eine CMD-HD-Partition ist, können Sie mit dieser Option die Partition selektieren, worauf die Dateien kopiert werden. Die Partitionsnummer, der Partitionsname und Typ werden im Zielbereich angeführt.

Set Source Path(S) (Quellpfad eingeben)

Wenn das Quellgerät eine CMD-HD ist, und der Partitionstyp eine Native- oder 1581-Emulation ist, oder das Quellgerät ein echtes 1581 ist, können Sie mit dieser Option einen Pfad für Unterverzeichnisse oder Unterpitionen eingeben. Jeder Unterverzeichnis- oder Unter- partitionsname muß mit einem Schragstrich getrennt werden (/).

Set Target Path (T) (Zielpfad eingeben)

Gleiche Funktion wie beim Set Source Path, mit der Ausnahme, daß der Zielpfad eingegeben wird.

Source/Target Directory (A/B) (Quell/Zielverzeichnis)

Mit dieser Option können Sie das Verzeichnis der Quell- bzw. Zieldiskette anschauen. Das Programm verlangt nach einem Suchmuster, damit Sie Dateien anschauen können, die zu einem bestimmten Namen oder Typ passen. Für HD-Verzeichnisse, können Sie auch Dateien nach ihrer Zeit oder ihren Datum suchen.

Select Files (F) (Dateien selektieren)

Hier wird das Quellverzeichnis in den Wahlpuffer, nachdem Sie einen Suchmuster eingegeben haben, eingelesen. Dateiinformation wird in einer dynamischen Methode gespeichert, damit Sie, wenn notwendig, von 700 bis zu weit über 1000 Dateien anschauen können. Falls Ihr Verzeichnis mehr Dateien hat, die sichtbar sind, müssen Sie die Zahl mit einem Muster, der wenige Dateien vergleicht, begrenzen. Wenn Sie sich im Selektiermodus befinden, können Sie mit RETURN die Dateien selektieren. Ein Stern (*) bestimmt die selektierten Dateien. Die 'T'-Taste schaltet die Selektionen aus und ein. Wenn Sie fertig sind, führt der 'Rückpfeil' ins Hauptmenü zurück.

Reselect Files (R) (Dateien neu-selektieren)

Sie können jede Zeit, nachdem Sie Dateien selektiert haben, zum Select-Files-Modus zurückkehren ohne das Verzeichnis lesen zu müssen. Sie können daher vor dem Kopieren Ihre Selektionen ändern.

Send Disk Command (@) (Diskettenbefehl senden)

Mit dieser Option können Sie einen Diskettenbefehl entweder ans Quell- oder Ziellaufwerk senden. Wichtig: Befehle folgen dem Quell- oder Zielpfad nicht, sondern sind an das vorhandene Verzeichnis gesandt. Es ist empfehlenswert in der richtigen Partition, Unterpartition oder im richtigen Unterverzeichnis zu sein, bevor Sie einen Befehl senden. Wenn Sie an die CMD-HD Befehle senden, können Sie einen Partitionsnamen und Pfad inden Befehl eingeben.

Begin Copying (C) (Kopieren starten)

Diese Option startet den Kopierprozeß. Nur Dateien, die vorher selektiert wurden, werden kopiert. Wenn das Kopieren fertig ist, haben Sie die Wahl, die Dateien erneut zu kopieren.

Begin Scratching (#) (Löschen starten)

Diese Option startet das Löschen der selektierten Dateien. Dateien können nur von der Quelldiskette gelöscht werden. Sie werden gefragt "ARE YOU SURE (Y/N)? (Sind sie sicher?) bevor der Prozeß beginnt.

Exit Programm (←) (Programm verlassen)

Mit dieser Option verlassen Sie das Programm.

MCOPY

Dies ist ein 'Ganz-Disketten'-Kopierprogramm. Damit können Sie eine ganze Diskette zwischen zwei Laufwerken der gleichen Art, einem Laufwerk und gleicher HD-Partition, oder zwei gleichen Partitionstypen auf der gleichen oder separaten HD kopieren. Das Programm unterstützt sowohl alle Commodore Laufwerke als auch alle andere Laufwerke, die den Commodore-Lauf- werken völlig gleich sind.

MCOPY ist nützlich wenn Sie Partitionen auf eine Floppydiskette oder andere HD-Partitionen kopieren wollen. Es ist auch möglich Native-Modus-Partitionen auf andere Native-Modus- Partitionen verschiedener Größen zu kopieren.

MCOPY ist selbst-erklärend. Ein Hilfsmenü, das mögliche Optionen anbietet, ist immer am Bildschirm.

SET HD CLOCK (HD-Uhr einstellen)

Mit diesem Programm wird die Echtzeituhr der HD eingestellt. Dieses Programm läuft nur auf der HD. Der Versuch, das Programm von einem Laufwerk zu laden und starten, führt zu einem Fehler.

BOOTQ

Dieses Programm ist nur für US-User geeignet.

XFER QDATA

Dieses Programm ist nur für US-User geeignet.

AUTO-BOOT 128

Mit diesem PD Dienstprogramm können Sie Bootsektoren für den Gebrauch mit dem C128 erstellen. Ein Bootsektor darf in jedem Emulation-Modus oder jeder Native-Modus-Partition erstellt werden. Native-Modus-Partitionen schützen den Bootsektor und halten ihn immer zugeteilt. Wenn Sie das Programm auf einer Native-Modus-Partition anwenden, bekommen Sie die Meldung, daß der Sektor schon besetzt ist. Wenn Sie jetzt fortfahren, wird das Programm den Bootsektor trotzdem erstellt.

Ändern Sie vorher die Parition, damit die Partition worauf sie den Bootsektor erstellen wollen, die aktuelle Partition wird. Wenn das Programm um das Laufwerk fragt, geben Sie 0 ein.

PARK HD (.64/.128)

Dieses Utility ist nur für diejenigen, die zusätzliche HDs an ihre HD anschließen, die die Auto-Park-Funktion nicht besitzen, gedacht. Wenn gestartet, parkt das Programm die Lese-/ Schreibköpfe aller an der CMD-HD angeschlossenen HDs. Mit diesem Programm können Sie auch manuell Ihre HD parken.

DISK CRACKER HD

Dieses PD-Programm ist eine Adaptation eines Programms, das von Mike Henry geschrieben wurde. DISK CRACKER HD ist ein Disketteneditor und -monitor, der nur Anwendung bei denen findet, die sich mit solchen Programmen gut auskennen. Um das Programm in Betrieb zu setzen, müssen Sie die HD-Geräteadresse auf 8 ändern.

PARTITION INFO

Dieses Programm bietet Information über alle Partitionen auf Ihrer HD. PARTITION INFO bietet auch denen Hilfe an, die ihre HD mit anderen Typen von Computern teilen wollen. Das Erstellen von Partitionen von einem anderen Computern erfordert Information über die Stelle, wo die Foreign-Modus-Partitionen zu finden sind.

FIX BLOCKS

Dieses Programm scannt die selektierten Partitionen für schlechte Blöcke. Schlechte Blöcke werden mit guten Blöcken ersetzt. Dabei versucht das Programm die Daten vom schlechten Block in den neuen Block einzuschreiben. Prüfen Sie danach die Blöcke auf ihre Vollständigkeit und kopieren Sie die fehlenden Daten auf die neuen Blöcke.

REWRITE DOS(.64/.128) (DOS neu installieren)

Ihre HD ist vom Haus aus mit dem HD-DOS installiert. Normalerweise werden Sie das HD-DOS nur dann neu installieren müssen, wenn Sie von CMD ein Upgrade bekommen. REWRITE DOS wird Dateien oder Partitionen weder ändern noch zerstören. Es ersetzt einfach die alte DOS-Version mit der neuen. Gehen Sie wie folgt vor, um das DOS neu zu installieren:

1. Legen Sie die HD UTILITIES-Diskette in das Laufwerk 8 ein.
2. Laden und starten Sie das Programm REWRITE DOS (.64 oder .128)

LOAD"REWRITE DOS.64",8	(C64 ohne JiffyDOS)
RUN	
RUN"REWRITE DOS.128"	(C128 ohne JiffyDOS)
↑"REWRITE DOS.64"	(C64 mit JiffyDOS)
↑"REWRITE DOS.128"	(C128 mit JiffyDOS)
3. Schalten Sie die HD in CONFIGURATION MODE um und drücken Sie RETURN. Das neue DOS wird installiert.
4. Drücken Sie die RESET-Taste auf der HD.

VERMERK: REWRITE DOS soll nur von einem Laufwerk geladen und gestartet werden. Andere Programme werden nachgeladen, also die Diskette bitte im Laufwerk lassen. NIEMALS DIE RESET-TASTE WÄHREND NEU-INSTALLIERENS DRÜCKEN!

CREATE SYS(.64/.128)

Es kann vorkommen, daß Sie die HD-DOS-Systemtabellen Ihrer HD neu erstellen müssen. Dies wird nur dann der Fall sein, wenn das System fehlerhaft wird oder der Mechanismus ersetzt werden muß. Benützen Sie aber vorher das Programm LLFORMAT. Sie können mit diesem Programm auch alle Partitionen von der Festplatte schnell löschen.

WARNUNG: CREATE SYS zerstört alle HD-Daten. Nur benützen wenn äußerst notwendig!

Das folgende Programm beschreibt den Gebrauch von CREATE SYS.

1. Legen Sie die HD UTILITIES-Diskette ins Laufwerk 8 ein.
2. Laden und starten Sie das CREATE SYS (.64 oder .128)-Programm.

LOAD"CREATE SYS.64",8	(C64 ohne JiffyDOS)
RUN	
RUN"CREATE SYS.128"	(C128 ohne JiffyDOS)
↑"CREATE SYS.64"	(C64 mit JiffyDOS)
↑"CREATE SYS.128"	(C128 mit JiffyDOS)

3. Schalten Sie die Festplatte in INSTALL MODE um und drücken Sie RETURN.

VERMERK: CREATE SYS soll nur von einem Floppy-Laufwerk geladen und gestartet werden. Andere Programme, gespeichert auf der Diskette, werden nachgeladen. Entfernen Sie die Diskette vom Laufwerk nicht. NIEMALS DIE HD-RESET-TASTE WÄHREND DES PROGRAMMABLAUFS DRÜCKEN!!!

LLFORMAT(.64/.128)

Ihre HD wurde vor der Lieferung schon formatiert und besitzt die neueste Version des HD- DOSes. Sollte es einmal notwendig sein, können Sie die gesamte HD mit LLFORMAT neu formatieren.

Nach dem Neu-Formatieren müssen Sie das HD-DOS neu installieren. Dafür müssen Sie das CREATE SYS-Programm benutzen.

LLFORMAT soll auch dann benutzt werden, wenn Sie eine neue HD an der CMD-HD anschließen.

WARNUNG: Benützen Sie LLFORMAT nur wenn es unbedingt notwendig ist. Alle Daten auf der HD werden zerstört.

ADD DRIVE(.64/128) (Neue SCSI-Geräte anschließen)

Mit diesem Programm werden neue an der CMD-HD angeschlossene SCSI-Geräte mit der CMD-HD lauffähig. Bitte die Geräte einzeln anschließen und bearbeiten. Nachdem das Gerät angeschlossen ist, laden und starten Sie dieses Programm. Ein SCSI-Bus wird für andere Geräte gesucht. Befolgen Sie die Anweisungen, um das Gerät vorzubereiten. Mit dem LLFORMAT formatieren Sie das neue Gerät und mit HD-TOOLS erstellen Sie darauf neue Partitionen.

1541SUB und 1581SUB

Diese Programme sind in Funktion fast identisch und werden gebraucht, um Native-Modus- Unterverzeichnisse, die die Verzeichnisse der 1541 und 1581 emulieren, zu erstellen. Mit diesen Unterverzeichnissen können Sie Programme benutzen, die in einer Native-Modus- Partition nicht laufen.

WARNUNG: Benützen Sie diese Programme nur mit leeren Native-Modus- Partitionen! Die Daten gehen dabei verloren!

1541SUB erstellt ein Unterverzeichnis, das an der gleichen Stelle wie bei einem 1541- Verzeichnis (Spur 18, Sektor 1) beginnt. Ein Header-Block für dieses Verzeichnis wird auch erstellt und dort gesetzt, wo sich der Header-Block auf einem 1541 befindet (Spur 18, Sektor 0).

1581SUB erstellt ein Unterverzeichnis, der an der gleichen Stelle wie bei einem 1581- Verzeichnis (Spur 40, Sektor 3) beginnt. Ein Header-Block für dieses Verzeichnis wird auch erstellt und dort gesetzt, wo sich der Header-Block auf einem 1541 befindet (Spur 40, Sektor 0). 1581SUB muß 2 Unterverzeichnisse erstellen, um Platz für die Headers und das Verzeichnis. Dieses Programm wurde mit Superbase und Superbase 128 getestet.

Laden und Starten Sie dieses Programm und befolgen Sie die Anweisungen. Benützen Sie dieses Programm nicht mit einer Partition, die wertvolle Daten beinhaltet, da diese Partition neu formatiert wird. Vergewissern Sie sich, daß die Partition groß genug ist, um das Unterverzeichnis zu unterstützen (18 Spuren für 1541SUB/40 Spuren für 1581SUB).

DRIVE INFO

Dieses Programm untersucht den SCSI-Mechanismus in der HD und gibt Information über Hersteller, Modell, Speicherkapazität und Seriellnummer des Mechanismus weiter. Nachdem das Programm startet, fragt es nach der HD-Geräteadresse. Dann liest es die Information und gibt sie weiter.

Anhang B

Fehlercodes

Die Fehlercodes sind den Commodore-Fehlercodes kompatibel. Einige Fehler sind nicht vorhanden, weil sie auf der HD nicht auftreten können. Andere neue Codes sind hinzugekommen, die auf der HD einzigartig sind. Wenn immer Fehler auftreten, helfen diese Codes das Problem zu finden.

Fehler werden über den Befehlskanal im folgenden Format weitergegeben:

ec,estring, tv, sv

wo:	ec	= ein zweistelliges Fehlernummer ist
	estring	= eine ASCII-Schleife ist, die den Fehler beschreibt
	tv	= die logische Spur ist, wo der Fehler passierte
	sv	= der logische Sektor ist, wo der Fehler passierte

VERMERK: Einige Fehler werden unten beschrieben.

Befehlkanal Fehlercodes

- 00** **OK** (kein Fehler)
- 01** **FILES SCRATCHED** (Dateien gelöscht/kein Fehler)
Die Zahl der gelöschten Dateien wird in der Spurvariable gezeigt.
- 02** **PARTITION SELECTED** (Partition selektiert/kein Fehler)
Geschieht nach dem Partitionswechsel mit HD-DOS 'CP'-Befehl und nach dem 1581-Unterpertitionswechsel mit dem '/'-Befehl. Nach dem Gebrauch des 'CP'-Befehls beinhaltet die Spurvariable die Partitionsnummer der neu selektierten Partition, während die Sektorvariable eine Null beinhaltet. Nach dem "/"-Befehl beinhaltet die Spurvariable die Zahl der ersten Spur in der 1581-Unterpertition, während die Sektorvariable die Zahl der letzten Spur in der 1581-Unterpertition beinhaltet.
- 26** **WRITE PROTECT ON** (Schreibschutz aktiviert)
Zeigt den Versuch an, auf der HD zu schreiben, während die Schreibschutzfunktion aktiviert wurde.
- 30** **SYNTAX ERROR** (allgemein)
Zeigt an, daß der DOS-Befehl-Interpreter den letzten Befehl nicht identifizieren konnte.
- 31** **SYNTAX ERROR** (Befehl nicht erkennbar)
Zeigt meistens an, daß das erste Zeichen der letzten Befehlsschleife als Teil eines legalen Befehls nicht erkannt wurde.
- 32** **SYNTAX ERROR** (Befehlsschleife zu lang)
Zeigt an, daß die Befehlsschleife länger als 254 Zeichen ist. Dieser Fehler wird selten vorkommen.
- 33** **SYNTAX FEHLER** (nicht legaler Dateiname)
Zeigt meistens an, daß 'Wildcards' oder Mustervergleiche in einem Befehl vorhanden sind, der diese Funktion nicht akzeptiert.
- 34** **SYNTAX ERROR** (Dateiname fehlt)
Zeigt an, daß der Dateiname oder ein Doppelpunkt vor dem Dateiname fehlt.

40 SCSI ERROR

Zeigt an, daß der letzte Diskettenzugriff einen SCSI-Fehler verursachte. Die Meldung wird von extra Parametern in der Fehlermeldungsschleife begleitet. Diese Parameter sind PETASCII-Darstellungen der SCSI-Fehler-Information und werden so angezeigt:

40, "SCSI ERROR d l cl ek ec", tv, sv

wo: d	= die SCSI-Geräteadresse (0-7) ist
l	= die SCSI logische Unitnummer ist (0-7)
cl	= die SCSI Fehlerklasse ist (in hexadezimal)
ck	= der SCSI Fehlerschlüssel ist (in hexadezimal)
ec	= der SCSI Fehlercode ist (in hexadezimal)
tv	= die logische Spur ist, wo der Fehler passierte
sv	= der logische Sektor ist, wo der Fehler passierte

41 STATUS ERROR (Ziel beschäftigt)

Zeigt an, daß das SCSI-Gerät beschäftigt war, als die Funktion durchgeführt wurde.

42 STATUS ERROR (Zwischenstatus OK)

Zeigt an, daß ein Vorgang in einer SCSI-Befehlskette erfolgreich abgeschlossen wurde.

43 STATUS ERROR (Reservierungskonflikt)

Zeigt an, daß ein Versuch gemacht wurde, auf einen Bereich des SCSI-Gerätes zuzugreifen, der für einen anderen Wirtregler reserviert wurde.

44 CONTROLLER ERROR (Reglerfehler) (nicht selektiert)

Zeigt an, daß das SCSI-Gerät nicht selektiert werden darf.

45 CONTROLLER ERROR

Zeigt an, daß das SCSI-Belegt-Signal nicht legal vom Zielgerät fallengelassen wurde.

46 CONTROLLER ERROR (nicht legale Phase)

Zeigt an, daß das SCSI-Zielgerät eine nicht legale Information-Transfer-Phase verlangte. Es ist meistens ein Hardwarefehler.

47 CONTROLLER ERROR

Zeigt an, daß der SCSI-Regler den Sinn des letzten Fehlers nicht verstehen kann. Es ist meistens ein Hardwarefehler im Zielgerät.

48 ILLEGAL JOB

Zeigt an, daß der letzte Job-Code in der Job-Queue nicht legal war.

50 RECORD NOT PRESENT

Zeigt an, daß der letzte Versuch auf einen Relativ-Dateirecord zuzugreifen, eine Zahl angab, die nicht existiert.

51 OVERFLOW IN RECORD

Zeigt an, daß der Relativ-Record zu klein ist für die Datenmenge ist, die geschrieben wird.

52 FILE TOO LARGE

Zeigt an, daß zu wenig Speicherplatz für diese Datei in der Partition vorhanden ist.

60 WRITE FILE OPEN

Zeigt an, daß der Versuch eine Datei zu öffnen nicht gelang, weil die Datei schon offen war.

61 FILE NOT OPEN

Zeigt an, daß der Versuch eine Datei zu öffnen nicht richtig durchgeführt wurde.

62 FILE NOT FOUND

Zeigt an, daß das DOS die gesuchte Datei nicht finden konnte. Das geschieht auch wenn die Dateitypen nicht übereinstimmen.

63 FILE EXISTS

Zeigt an, daß 2 Dateien den gleichen Namen haben.

- 64 FILE TYPE MISMATCH**
Es handelt sich hier, um einen Fehler in der Dateintypübereinstimmung.
- 65 NO BLOCK**
Zeigt an, daß beim Versuch einen Block zuzuteilen, das DOS einen schon zugeteilten Block fand. Der nächste freie Block wird angezeigt. Eine Null zeigt keine freie Blöcke an.
- 66 ILLEGAL BLOCK**
Zeigt an, daß der Block nicht existiert.
- 67 ILLEGAL BLOCK**
Zeigt an, daß der Block fehlerhaft ist.
- 70 NO CHANNEL**
Zeigt an, daß der bestimmte Kanal schon besetzt ist oder alle Puffer besetzt sind. Das kann auch bedeuten, daß zu viele Dateien offen sind.
- 71 DIRECTORY ERROR**
Zeigt an, daß das BAM nicht validiert ist. Die Diskette muß validiert werden.
- 72 PARTITION FULL**
Zeigt an, daß die Zielpartition (oder ihr Verzeichnis) voll sind.
- 74 DRIVE NOT READY**
Zeigt an, daß eine Partition nicht richtig formatiert wurde oder daß die Partition nicht legal ist.
- 75 FORMAT ERROR**
Zeigt an, daß im SCSI-Zielgerät ein Formatierfehler aufgetreten ist.
- 76 CONTROLLER ERROR**
Zeigt an, daß ein nicht zu richtbaren Hardwarefehler im SCSI-Wirtregler aufgetreten ist.
- 77 SELECTED PARTITION ILLEGAL**
Zeigt an, daß die Partition, worauf man zugreifen will, nicht existiert.

Anhang C

Partitions- und Dateiformate

Beim HD-Spur -und Sektorzugriff ist es wichtig, sich zu merken, daß das Spur- und Sektorlayout der Norm der Partitionsart, in der der Zugriff stattfindet, entspricht. Durch diese Methode kann die HD ihre Kompatibilität erreichen. Wenn ein Zugriff in einer Emulations-Modus-Partition geschieht, ist das Spur- und Sektorlayout der Partition mit dem Laufwerk, das sie emuliert, identisch. Die Tabellen in diesem Anhang bestimmen welche Spuren und Sektoren in jedem Partitionstyp vorhanden sind. Weiters erfahren Sie wo sich der Header, das BAM und die Verzeichnisblöcke befinden. Dieser Anhang bietet auch Informationen über BAM-Format und Verzeichniseingaben, sowohl das Format verschiedener Dateitypen an.

Gewöhnliche Formate in allen Partitionstypen

VERZEICHNIS-DATEI-TABELLEN-FORMAT (ALLE PARTITIONSTYPEN)	
1541 & 1571 PARTITIONS-SPUR 18 SEKTOR 1	
1581 PARTITION-SPUR 40 SEKTOR 3	
NATIVE-PARTIONS-SPUR 1 SEKTOR 34	
VERSCHIEDEN BEI UNTERVERZEICHNISSEN	
BYTE	BESCHREIBUNG
0	Spur-Pointer auf den nächsten Verzeichnisblock(0=letzter Block)
1	Sektor-Pointer auf den nächsten Verzeichnisblock(255=letzter Block)
2 - 31	Dateieingabe 1 (siehe Bild C2)
32 - 33	Zwei 0 Bytes (reserviert)
34 - 63	Dateieingabe 2 (siehe Bild C2)
64 - 65	Zwei 0 Bytes (reserviert)
66 - 95	Dateieingabe 3 (siehe Bild C2)
96 - 97	Zwei 0 Bytes (reserviert)
98 - 127	Dateieingabe 4 (siehe Bild C2)
128 - 129	Zwei 0 Bytes (reserviert)
130 - 159	Dateieingabe 5 (siehe Bild C2)
160 - 161	Zwei 0 Bytes (reserviert)
162 - 191	Dateieingabe 6 (siehe Bild C2)
192 - 193	Zwei 0 Bytes (reserviert)
194 - 223	Dateieingabe 7 (siehe Bild C2)
224 - 225	Zwei 0 Bytes (reserviert)
226 - 255	Dateieingabe 8 (siehe Bild C2)

C1

VERZEICHNIS-DATEIEINGABE-FORMAT		
ALLE PARTITIONSTYPEN UND NATIVE-MODUS-UNTERVERZEICHNISSE		
BYTE	INHALT	BESCHREIBUNG
0		Dateityp: 0 DEL (gelöscht) 1 SEQ (sequentiell) 2 PRG (Programm) 3 USR (USER-Datei) 4 REL (relativ) 5 CBM (1581 Unterpartitionsart) 6 DIR (Native-Modus-Unterverzeichnis) VERMERK: Richtig geschlossene Dateien werden mit \$80 verknüpft oder (OR) mit \$C0 wenn sie gesperrt ist.
1		Spur des ersten Datenblockes (Headerblockes, wenn DIR)
2		Sektor des ersten Datenblockes (Headerblockes, wenn DIR)
3 - 18		Dateiname, mit geschifteten Leerzeichen aufgefüllt.
19		Spur des ersten Seite-Sektor-Blocks (Super-Seite-Spur/REL)
20		Sektor des ersten Seite-Sektor-Blocks (Super-Seite-Spur/REL)
21		Satzlänge, wenn Dateityp REL ist
22	0	Reserviert
23		Jahr der erstellten Datei (letzte 2 Zeichen)
24		Monat der erstellten Datei
25		Tag der erstellten Datei
26		Stunde der erstellten Datei
27		Minute der erstellten Datei
28		Zahl, der von Datei genützten Blöcken (low Byte)
29		Zahl, der von Datei genützten Blöcken (high Byte)

C2

1541 und 1571 Emulations-Modus-Partitions

SEKTOREN PRO SPUR (1541 EMULATIONS-MODUS)		
SPURBEREICH	SEKTORBEREICH	GESAMT
1 - 17	0 - 20	21
18 - 24	0 - 18	19
25 - 30	0 - 17	18
31 - 35	0 - 16	17

C3

SEKTOREN PRO SPUR (1571 EMULATIONS-MODUS)		
SPURBEREICH	SEKTORBEREICH	GESAMT
1 - 17	0 - 20	21
18 - 24	0 - 18	19
25 - 30	0 - 17	18
31 - 35	0 - 16	17
36 - 52	0 - 20	21
53 - 59	0 - 18	19
60 - 65	0 - 17	18
66 - 70	0 - 16	17

C4

HEADER & BAM (1541 & 1571 EMULATIONS-MODI) SPUR 18 SEKTOR 0		
BYTE	INHALT	BESCHREIBUNG
0	18	Spur des ersten Verzeichnisblockes
1	1	Sektor des ersten Verzeichnisblockes
2	65	ASCII-Zeichen A für Formattyp
3	0 128	1541 Emulations-Modus 1571 Emulations-Modus
4 - 143		BAM
144 - 161		Diskettenname, aufgefüllt mit geshifteten Leerzeichen
162 - 163		Disketten-ID
164	160	Geshiftete Leerzeichen
165	50	ASCII-Darstellung von 2 für die DOS-Version
166	65	ASCII-Darstellung von A für den Formattyp
167 - 170	160	Geshiftete Leerzeichen für Trennung
171 - 220	0	Nullen - reserviert
221 - 255	0	1541 Emulations-Modus - Nullen- reserviert 1571 Emulations-Modus - Zahl der freien Sektoren für Spuren 36-70 - ein Byte pro Spur

C5

BAM FÜR 1571 (SEITE 2) EMULATIONS-MODUS SPUR 53 SEKTOR 0		
BYTE	INHALT	BESCHREIBUNG
0 - 104		BAM für Spuren 36-70 (3 Bytes pro Spur)
105 - 255	0	Nullen - reserviert

C6

BAM-EINGABE-FORMAT 1541 & 1571 (Seite 1) EMULATIONS-MODI Byteformat 4-143 auf Spur 18, Sektor 0 (Siehe Tabelle C5) 4 Bytes pro Spur: Bytes 4-7 decken Spur 1, Bytes 8-11 decken Spur 2,...	
BYTE	BESCHREIBUNG
0	Zahl der freien Sektoren auf der Spur
1	Zahl der nicht belegten Blöcke für Sektoren 0-7
2	Zahl der nicht belegten Blöcke für Sektoren 8-15
3	Zahl der nicht belegten Blöcke für Sektoren 16-23
VERMERKE:	Das LSB (low Bit) in jedem Byte (1-3) zeigt den Status des niedrigsten Sektors, der von dem Byte gedeckt ist, an. Ein Binarwert von 1 zeigt an, daß der Sektor frei ist, 0 zeigt an, daß er belegt ist.

C7

BAM-EINGABE-FORMAT 1571 (Seite 2) EMULATIONS-MODUS Byteformat 0-104 auf Spur 53, Sektor 0 (Siehe Tabelle C6) 3 Bytes pro Spur: Bytes 0-2 decken Spur 36, Bytes 3-5 decken Spur 37,...	
BYTE	BESCHREIBUNG
0	Zahl der nicht belegten Blöcke für Sektoren 0-7
1	Zahl der nicht belegten Blöcke für Sektoren 8-15
2	Zahl der nicht belegten Blöcke für Sektoren 16-23
VERMERKE:	Das LSB in jedem Byte (0-2) wird benutzt, um den Status des niedrigsten Sektors, der von dem Byte gedeckt ist, anzuzeigen. Ein Binarwert von 1 zeigt an, daß der Sektor frei ist. Ein Binarwert von 0 zeigt an, daß der Sektor belegt ist. In Bytes 221-225 von Spur 18, Sektor 0 ist das zugehörige Byte für die Zahl der freien Sektoren gespeichert. (Siehe Tabelle C5)

C8

1581 Emulations-Modus-Partitions

SEKTOREN PRO SPUR		
BEREICH	SEKTORBEREICH	GESAMTZAHL
1 - 80	0 - 39	40

C9

VERZEICHNISHEADER SPUR 40, SEKTOR 0		
BYTE	INHALT	BESCHREIBUNG
0	40	Spur des ersten Verzeichnisblockes
1	3	Sektor des ersten Verzeichnisblockes
2	68	ASCII-Zeichen von D für Formattyp
3	0	Reserviert
4 - 21		Diskettenname, aufgefüllt mit geshifteten Leerzeichen
22 - 23		Disketten-ID
24	160	Geshiftete Leerzeichen für Trennung
25	51	ASCII-Zeichen von 3 für DOS-Version
26	68	ASCII-Zeichen von D für Formattyp
27 - 28	160	Geshiftete Leerzeichen für Trennungen
29 - 255	0	Nullen- reserviert

C10

BAM BLOCK1 SPUR 40, SEKTOR 1		
BYTE	INHALT	BESCHREIBUNG
0	40	Spur des nächsten BAM-Blockes
1	2	Sektor des nächsten BAM-Blockes
2	68	ASCII-Zeichen von D für DOS-Version
3	187	Komplement der Versionsnummer
4 - 5		Disketten-ID
6	192	Nicht benützt von der HD (1581 Default)
7	0	Flag für Auto-Ladedatei
8 - 15	0	Reserviert
16 - 255		BAM für Spuren 1-40 (6 Bytes pro Spur)

C11

BAM 2 SPUR 40, SEKTOR 2		
BYTE	INHALT	BESCHREIBUNG
0	0	Zeigt den letzten BAM-Sektor an
1	255	Zeigt alle Bytes im Sektor an
2	68	ASCII-Zeichen von D für DOS-Version
3	187	Komplement der Versionsnummer
4 - 5		Disketten-ID (Kopie)
6	192	Nicht von der HD benützt
7	0	Flag für Auto-Laderdatei (Kopie)
8 - 15	0	Reserviert
16 - 255		BAM für Spuren 41-80 (6 Bytes pro Spur)

C12

BAMEINGABEFORMAT Byteformat 16-255 auf Spur 40, Sektoren 1 und 2 (Tabellen C11 u. C12)	
BYTE	BESCHREIBUNG
0	Zahl der freien Sektoren auf der Spur
1	Freie Blöcke für Sektoren 0-7
2	Freie Blöcke für Sektoren 8-15
3	Freie Blöcke für Sektoren 16-23
4	Freie Blöcke für Sektoren 24-31
5	Freie Blöcke für Sektoren 32-39
VERMERKE: Das LSB (low Bit) in jedem Byte (1-5) zeigt den Status des niedrigsten Sektors, der von dem Byte gedeckt ist, an. Ein Binarwert von 1 zeigt an, daß der Sektor frei ist, 0 zeigt an, daß er belegt ist.	

C13

Native-Modus-Partitions

SEKTOREN PRO SPUR		
SPUR BEREICH	FREIE SEKTOREN	GESAMT
1 - 255	0 - 255	256

C14

HAUPTVERZEICHNIS UND UNTERVERZEICHNIS-HEADER SPUR 1, SEKTOR 1 FÜR HAUPTVERZEICHNIS VERSCHIEDEN BEI UNTERVERZEICHNISSEN		
BYTE	INHALT	BESCHREIBUNG
0		Spur des ersten Verzeichnisblockes
1		Sektor des ersten Verzeichnisblockes
2	72	ASCII-Zeichen von H für Formattyp
3	0	Reserviert
4 - 21		Diskettenname, aufgefüllt mit geshifteten Leerzeichen
22 - 23		Disketten-ID
24	160	Geshiftete Leerzeichen für Trennung
25	49	ASCII-Zeichen von 1 für DOS-Version
26	72	ASCII-Zeichen von H für Formattyp
27 - 28	160	Geshiftete Leerzeichen für Trennungen
29 - 31	0	Reserviert
32	1	ROOT-Headerspur
33	1	ROOT-Headersektor
34		Spur zu DIR PARENT-Header
35		Sektor zu DIR PARENT-Header
36		Spur zu DIR-Eingabe in PARENT-Verzeichnis
37		Sektor zu DIR-Eingabe in PARENT-Verzeichnis
38		Index zum Anfangsbyte der DIR-Eingabe im PARENT-Verzeichnis
39 - 255	0	Nullen - reserviert

C15

NATIVE-MODUS-BAM (erster BAM-Block) SPUR 1 SEKTOR 2		
BYTE	INHALT	BESCHREIBUNG
0	0	Reserviert
1	0	Reserviert
2	72	ASCII-Zeichen von H für Formattyp
3	183	Komplement des Formattypes
4 - 5		Disketten-ID
6	192	Von der HD nicht benutzt (1581 Default)
7	0	Flag für die Auto-Ladedatei
8		Spurnummer der letzten freien Spur in Partition
9 - 31	0	Reserviert
32 - 255	0	BAM für Spuren 1 bis 7 (32 Bytes pro Spur)

C16

NATIVE-MODUS-BAM (Blöcke 2-32) SPUR 1 SEKTOR 3-28		
SPUR	SEKTOR	BESCHREIBUNG
1	3	BAM für Spuren 8 -15 (32 Bytes pro Spur)
1	4	BAM für Spuren 16-23 (32 Bytes pro Spur)
1	5	BAM für Spuren 24-31 (32 Bytes pro Spur)
1	6	BAM für Spuren 32-39 (32 Bytes pro Spur)
1	7	BAM für Spuren 40-47 (32 Bytes pro Spur)
1	8	BAM für Spuren 48-55 (32 Bytes pro Spur)
1	9	BAM für Spuren 56-63 (32 Bytes pro Spur)
1	10	BAM für Spuren 64-71 (32 Bytes pro Spur)
1	11	BAM für Spuren 72-79 (32 Bytes pro Spur)
1	12	BAM für Spuren 80-87 (32 Bytes pro Spur)
1	13	BAM für Spuren 88-95 (32 Bytes pro Spur)
1	14	BAM für Spuren 96-103 (32 Bytes pro Spur)
1	15	BAM für Spuren 104-111 (32 Bytes pro Spur)
1	16	BAM für Spuren 112-119 (32 Bytes pro Spur)
1	17	BAM für Spuren 120-127 (32 Bytes pro Spur)
1	18	BAM für Spuren 128-135 (32 Bytes pro Spur)
1	19	BAM für Spuren 136-143 (32 Bytes pro Spur)
1	20	BAM für Spuren 144-151 (32 Bytes pro Spur)
1	21	BAM für Spuren 152-159 (32 Bytes pro Spur)
1	22	BAM für Spuren 160-167 (32 Bytes pro Spur)
1	23	BAM für Spuren 168-175 (32 Bytes pro Spur)
1	24	BAM für Spuren 176-183 (32 Bytes pro Spur)
1	25	BAM für Spuren 184-191 (32 Bytes pro Spur)
1	26	BAM für Spuren 192-199 (32 Bytes pro Spur)
1	27	BAM für Spuren 200-207 (32 Bytes pro Spur)
1	28	BAM für Spuren 208-215 (32 Bytes pro Spur)
1	29	BAM für Spuren 216-223 (32 Bytes pro Spur)
1	30	BAM für Spuren 224-231 (32 Bytes pro Spur)
1	31	BAM für Spuren 232-239 (32 Bytes pro Spur)
1	32	BAM für Spuren 240-247 (32 Bytes pro Spur)
1	33	BAM für Spuren 248-255 (32 Bytes pro Spur)

C17

NATIVE-MODUS-BAM-EINGABE-FORMAT Format der Bytes 32-255 auf Spur 1 Sektor 2 und Bytes 0-255 auf Spur 1 Sektoren 3-33 (Siehe C11 u.C12)	
BYTE	BESCHREIBUNG
0	Block-Verfügbarkeit für Sektoren 0-7
1	Block-Verfügbarkeit für Sektoren 8-15
2	Block-Verfügbarkeit für Sektoren 16-23
3	Block-Verfügbarkeit für Sektoren 24-31
4	Block-Verfügbarkeit für Sektoren 32-39
5	Block-Verfügbarkeit für Sektoren 40-47
6	Block-Verfügbarkeit für Sektoren 48-55
7	Block-Verfügbarkeit für Sektoren 56-63
8	Block-Verfügbarkeit für Sektoren 64-71
9	Block-Verfügbarkeit für Sektoren 72-79
10	Block-Verfügbarkeit für Sektoren 80-87
11	Block-Verfügbarkeit für Sektoren 88-95
12	Block-Verfügbarkeit für Sektoren 96-103
13	Block-Verfügbarkeit für Sektoren 104-111
14	Block-Verfügbarkeit für Sektoren 112-119
15	Block-Verfügbarkeit für Sektoren 120-127
16	Block-Verfügbarkeit für Sektoren 128-135
17	Block-Verfügbarkeit für Sektoren 136-143
18	Block-Verfügbarkeit für Sektoren 144-151
19	Block-Verfügbarkeit für Sektoren 152-159
20	Block-Verfügbarkeit für Sektoren 160-167
21	Block-Verfügbarkeit für Sektoren 168-175
22	Block-Verfügbarkeit für Sektoren 176-183
23	Block-Verfügbarkeit für Sektoren 184-191
24	Block-Verfügbarkeit für Sektoren 192-199
25	Block-Verfügbarkeit für Sektoren 200-207
26	Block-Verfügbarkeit für Sektoren 208-215
27	Block-Verfügbarkeit für Sektoren 216-223
28	Block-Verfügbarkeit für Sektoren 224-231
29	Block-Verfügbarkeit für Sektoren 232-239
30	Block-Verfügbarkeit für Sektoren 240-247
31	Block-Verfügbarkeit für Sektoren 248-255
VERMERKE:	Das LSB (low Bit) in jedem Byte (0-31) zeigt den Status des höchsten Sektors, der von dem Byte gedeckt ist, an. Ein Binarwert von 1 zeigt an, daß der Sektor frei ist, 0 zeigt an, daß er belegt ist.

Datei-Formate

PROGRAM-DATEI-FORMAT	
BYTE	BESCHREIBUNG
0	Spur des nächsten Dateiblockes (0 wenn vorhandener Block der letzte Block ist)
1	Sektor des nächsten Dateiblockes (Hinweis auf das zuletzt benützte Byte, wenn das vorhandene Byte der letzte Dateiblock in der Datei ist)
2 - 255	Programmdaten (Bytes 2 und 3 beinhalten die Ladeadresse des Programms in Low-Byte/High/Byte-Format, wenn der vorhandene Block der erste Block in der Datei ist.)

C19

SEQUENTIELL-DATEI-FORMAT	
BYTE	BESCHREIBUNG
0	Spur des nächsten Dateiblockes (0 wenn der vorhandener Block der letzte Block ist)
1	Sektor des nächsten Dateiblockes (Hinweis auf das zuletzt benützte Byte, wenn der vorhandene Byte der letzte Dateiblock in der Datei ist)
2 - 255	Programmdaten

C20

RELATIV-DATEI-DATENBLOCK-FORMAT	
BYTE	BESCHREIBUNG
0	Spur des nächsten Dateiblockes (0 wenn der vorhandene Block der letzte Block ist)
1	Sektor des nächsten Dateiblockes (Hinweis auf das zuletzt benützte Byte, wenn vorhandener Byte der letzte Dateiblock in der Datei ist)
2 - 255	Programmdaten (Leere Datensätze beginnen mit \$FF im ersten Byte. Die Restlichen beinhalten \$00 Bytes.

C21

RELATIV-DATEI SUPER-SEITEN-SEKTOR-BLOCK-FORMAT	
BYTE	BESCHREIBUNG
0	Spur des nächsten Dateiblockes (0 wenn der vorhandene Block der letzte Block ist)
1	Sektor des nächsten Dateiblockes (Hinweis auf das letzt benützte Byte, wenn das vorhandene Byte der letzte Dateiblock in der Datei ist)
2	Super-Seiten-Sektor-ID-Byte (\$FE)
3 - 254	Spur und Sektor des ersten Seiten-Sektor der 126 Gruppen (Gruppen 0-125, 2 Bytes pro Hinweis). Unbenützte Grupp-Hinweise beinhalten \$00 Bytes.

C22

RELATIV-DATEI-SEITEN-SEKTOR-BLOCK-FORMAT	
BYTE	BESCHREIBUNG
0	Spur des nächsten Seiten-Sektors in dieser Gruppe (0 wenn der vorhandene Block der letzter Seiten-Sektor-Block ist)
1	Sektor des nächsten Seiten-Sektors in dieser Gruppe (Hinweis, wenn der vorhandene Block der letzte Seiten-Sektor-Block ist.)
2	Seiten-Sektornummer (0-5)
3	Record-Länge der zugehörigen Relativdatei
4	Spur der ersten Seiten-Sektor dieser Gruppe (0)
5	Sektor der ersten Seiten-Sektor dieser Gruppe (0)
6 - 7	Spur und Sektor der zweiten Seiten-Sektor dieser Gruppe (1)
8 - 9	Spur und Sektor der dritten Seiten-Sektor dieser Gruppe (2)
10 - 11	Spur und Sektor der vierten Seiten-Sektor dieser Gruppe (3)
12 - 13	Spur und Sektor der fünften Seiten-Sektor dieser Gruppe (4)
14 - 15	Spur und Sektor der sechsten Seiten-Sektor dieser Gruppe (5)
16 - 255	Spur und Sektor zu 120 Datenblöcken (2 Bytes pro Hinweis) Unbenützte Hinweise beinhalten \$00 Bytes.

C23

Anhang D

HD-Speichertabelle

<p>\$FFFF</p> <p>HD DOS</p> <p>Betriebssystem</p> <p>\$9400</p>
<p>\$93FF</p> <p>System I/O, Tabellen und</p> <p>Parameter</p> <p>\$8000</p>
<p>\$7FFF</p> <p>Diskette- und BAM-Puffer</p> <p>\$5000</p>
<p>\$4FFF</p> <p>8K Freies BAM</p> <p>\$3000</p>
<p>\$2FFF</p> <p>System BAM</p> <p>\$2800</p>
<p>\$27FF</p> <p>DOS- und Fehler-Puffer</p> <p>\$0300</p>
<p>\$02FF</p> <p>Befehlspeicher</p> <p>\$0200</p>
<p>\$01FF</p> <p>Stack</p> <p>\$0100</p>
<p>\$00FF</p> <p>Nullseite</p> <p>\$0000</p>

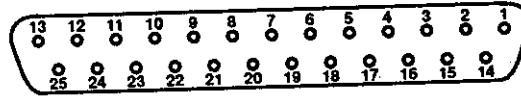
D1

CMD HD Erweiterte Speichertabelle

<u>Adressenbereich</u>	<u>Beschreibung</u>
\$0000 - \$0004	1541/1571 Emulations-Modus-Job-Queue
\$0006 - \$000F	1541/1571 Job-Queue Spur- u. Sektorvariablen
\$0002 - \$000A	1581 Emulations-Modus-Job-Queue
\$000B - \$001C	1581 Job-Queue Spur- u. Sektorvariablen
\$0020 - \$003F	Native-Modus-Job-Queue
\$0040 - \$00FF	Nullseite-Variablen
\$0100 - \$01FF	Prozessor-Stack
\$0200 - \$02FF	Eingabe-Befehlspeicher (254 Bytes)
\$0300 - \$22FF	DOS-Puffer (32 Puffer, je 254 Bytes)
\$2300 - \$27FF	Fehlerpuffer
\$2800 - \$283F	Native-Modus-Job-Queue Spur- u. Sektorvariablen
\$2840 - \$285F	Seitenvariablen
\$2860 - \$287F	Zieladressenvariable (hohe)
\$2880 - \$289F	Zieladressenvariable (niedrige)
\$28A0 - \$28BF	Blockzahlvariable
\$28C0 - \$28DF	SCSI-Gerät für Job
\$28E0 - \$28FF	SCSI-LU für Job
\$2900 - \$291F	SCSI-Blockadressenvariable (hohe, Fehlerklasse)
\$2920 - \$293F	SCSI-Blockadressenvariable (mittel, Fehlerschlüssel)
\$2940 - \$295F	SCSI-Blockadressenvariable (niedrig, Fehlercode)
\$2960 - \$297F	Letzter Job in Job-Queue
\$2980 - \$29FF	Reserviert
\$2A00 - \$2AFF	SCSI-Steuerungsvariablen
\$2B00 - \$2FFF	DOS-Variablen
\$3000 - \$4FFF	Freies BAM
\$5000 - \$5FFF	Reserviert für Diskettenpuffer
\$6000 - \$7FFF	BAM-Puffer
\$8000 - \$81FF	Serielle I/O (6522)
\$8200 - \$82FF	Partitionstyp-Tabelle
\$8300 - \$83FF	Partitionsgröße-(hohe) Tabelle
\$8400 - \$85FF	SCSI I/O (6522)
\$8600 - \$86FF	Partitionsgröße-(niedrige) Tabelle
\$8700 - \$87FF	Partitionsanfangs-Adressentabelle (hohe)
\$8800 - \$89FF	Parallel I/O (8255)
\$8A00 - \$8AFF	Partitionsanfangs-Adressentabelle (mittlere)
\$8B00 - \$8BFF	Partitionsanfangs-Adressentabelle (niedrige)
\$8C00 - \$8DFF	Echtzeituhr I/O
\$8F00 - \$8FFF	LED I/O
\$9000 - \$93FF	DOS-Tabellen
\$9400 - \$FFFF	Betriebssystem

VERMERK: Alle Partitionstabellen enthalten ein Byte pro Partition, beginnend mit Partition 0 (Systempartition) und aufhörend mit Partition 254.

Anhang E SCSI-Buchse



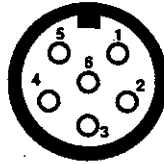
E1

SCSI-BUCHSE PINBELEGUNG			
Pinnummer	Bezeichnung	Pinnummer	Bezeichnung
1	$\overline{\text{REQ}}$	14	GND
2	$\overline{\text{MSG}}$	15	$\overline{\text{C/D}}$
3	$\overline{\text{I/O}}$	16	GND
4	$\overline{\text{RESEST}}$	17	$\overline{\text{ATN}}$
5	$\overline{\text{ACK}}$	18	GND
6	$\overline{\text{BSY}}$	19	$\overline{\text{SEL}}$
7	GND	20	$\overline{\text{DBP}}$
8	$\overline{\text{DBO}}$	21	$\overline{\text{DB1}}$
9	GND	22	$\overline{\text{DB2}}$
10	$\overline{\text{DB3}}$	23	$\overline{\text{DB4}}$
11	$\overline{\text{DB5}}$	24	GND
12	$\overline{\text{DB6}}$	25	N/C
13	$\overline{\text{DB7}}$		

E2

Anhang F

Seriell- und Zusatzbuchsen



F1

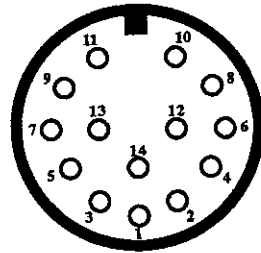
SERIELLENBUCHSE-PINBELEGUNG		
PINNUMMER	PINNAME	BESCHREIBUNG
1	$\overline{\text{SRQ}}$	SERVICE REQUEST
2	GND	MASSE
3	$\overline{\text{ATN}}$	ATTENTION
4	$\overline{\text{CLK}}$	UHR
5	$\overline{\text{DATA}}$	SERIELLDATEN
6	$\overline{\text{RESET}}$	SERIELLBUS-RESET

F2

ZUSATZBUCHSE-PINBELEGUNG		
PINBELEGUNG	PINNAME	BESCHREIBUNG
1		NICHT BELEGT
2	$\overline{\text{GND}}$	MASSE
3	$\overline{\text{AUX ATN}}$	ZUSATZ-ATTENTION
4	$\overline{\text{AUX CLK}}$	ZUSATZ-UHR
5	$\overline{\text{AUX DATA}}$	ZUSATZ-DATEN
6	$\overline{\text{AUX RESET}}$	ZUSATZ-BUS-RESET

F3

Anhang G Parallelbuchse



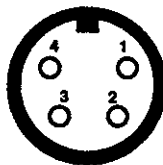
G1

Pinnummer	Pinname	Beschreibung
1	PCLK	Paralleluhr
2	GND	Masse
3	PEXT	Reserviert
4	PD1	Parallel-Datenbit 1
5	PD4	Parallel-Datenbit 4
6	GND	Masse
7	PRDY	Parallel-READY
8	PD0	Parallel-Datenbit 0
9	PD7	Parallel-Datenbit 7
10	PATN	Parallel-Attention
11	PD5	Parallel-Datenbit 5
12	PD3	Parallel-Datenbit 3
13	PD6	Parallel-Datenbit 6
14	PD2	Parallel-Datenbit 2

G2

Anhang H

Netzstecker



H1

PINNUMMER	BESCHREIBUNG
1	+5VDC
2	Masse
3	Masse
4	+12VDC

H2

Anhang I

Technische Daten

Speicher

Systemkapazität (maximum)	4,244,766,720 Bytes
(typisch)	>20,000,000 Bytes (20Mb HD)
	>40,000,000 Bytes (40Mb HD)
	>100,000,000 Bytes (100Mb HD)
	>200,000,000 Bytes (200Mb HD)
Maximale Zahl der Partitions:	254 (Userpartitions)
Partitionsgröße (log. Sektoren):	256 bis 65280 Blöcke (Native)
	684 (1541 - 683 verfügbar)
	1366 (1571)
	3200 (1581)
Maximale Seq.-Dateigröße:	16,564,864 Bytes (Native)
Maximale Rel.-Dateigröße:	≈16,000,000 Bytes (Native)
Datensätze pro Rel.-Datei:	65,535
Dateien pro Partition:	>59,000 (Native)
	144 (1541 u. 1571)
	296 (1581)
Logische Sektorgröße	256 Bytes
Physikalische Sektorgröße	512 Bytes

IC Chips

Mikroprozessor:	6502A (2MHz Uhr)
Eingabe/Ausgabe:	6522A(2), 8255
ROM:	16K(27128, oder 1/2 27256)
RAM:	TMS 4464-12 (2, 64K Gesamtbytes)
Zeituhr:	RTC 72421A
Sonder Chips:	1618PAL(3)
Glue:	16 MOS u. TTL

Abmessungen:

Höhe:	70mm
Breite:	127mm
Tiefe:	254mm
Gewicht:	ca. 3,6Kilo

Betriebsdaten

Spannung	220 Wechselspannung
Frequenz	50-60 Hz
Leistungsaufnahme	30W

Anhang J

Befehl-Zusammenfassung

Partitionsbefehle

BOOT

```
BOOT[[Dn]{ON|,}[Udv]]
```

PARTITION WECHSELN

```
OPENlf,dv,15:PRINT#lf,"CPn":CLOSElf
```

```
OPENlf,dv,15:PRINT#lf,"C□"+CHR$(n):CLOSElf (□ = <SHIFT>+<P>)
```

UNTERPARTITION WECHSELN (1581)

```
OPENlf,dv,15:PRINT#lf,"/[n]:[partitionsname]":CLOSElf
```

UNTERPARTITION ERSTELLEN (1581)

```
OPENlf,dv,15:PRINT#lf,"/[n]:partitionsname,CHR$(st)CHR$(ss)CHR$(sl)CHR$(sh)  
",C":CLOSElf
```

FORMATIEREN

```
OPENlf,dv,15:PRINT#lf,"N[n]:partitionsname[,id]":CLOSElf
```

```
HEADER"partitionsname"[,Iid][,Dn][{ON|,}Udv]
```

GET PARTITIONSINFORMATION

```
OPENlf,dv,15:PRINT#lf,"G-P"[+CHR$(n)]:CLOSElf
```

INITIALISIEREN

```
OPENlf,dv,15:PRINT#lf,"I[n][:]":CLOSElf
```

PARTITIONSVERZEICHNIS

```
LOAD"$=P[:*][=tp]",dv
```

PARTITION UMBENENNEN

```
OPENlf,dv,15:PRINT#lf,"R-P:neuer name=alter name":CLOSElf
```

HEADER UMBENENNEN

```
OPENlf,dv,15:PRINT#lf,"R-H[n][pfad]:neuer name":CLOSElf
```

VALIDIEREN

```
OPENlf,dv,15:PRINT#lf,"V[n][:]":CLOSElf
```

```
COLLECT[,Dn][{ON|,}Udv]
```

Unterverzeichnisbefehle

VERZEICHNIS WECHSELN

```
OPENlf,dv,15:PRINT#lf,"CD[n][←|[[pfad][:]unter. name]]":CLOSElf
```

VERZEICHNIS ERSTELLEN

```
OPENlf,dv,15:PRINT#lf,"MD[n][pfad]:name":CLOSElf
```

VERZEICHNIS LÖSCHEN

```
OPENlf,dv,15:PRINT#lf,"RD[n]:unter. name":CLOSElf
```

Dateibefehle

KOPIEREN

```
OPENlf,dv,15:PRINT#lf,"C[n][pfad]:neue datei=[n][pfad]:dateiname[, [n:]dateiname...]:CLOSElf
```

```
COPY[Dn,]"dateiname"TO[Dn]"neue datei"[,Udv]
```

```
CONCAT[Dn,]"dateiname"TO[Dn]"neue Datei"[,Udv]
```

LADEN

```
LOAD"[n][pfad]:dateiname",dv[,sa]
```

```
BLOAD"dateiname"[,Dn][{ON|,}Udv][,Bb][,Pa]
```

```
DLOAD"dateiname"[,Dn][{ON|,}Udv]
```

LADEN (Verzeichnisdatei)

```
LOAD"$=[T][n][pfad]:muster[={tp|option}[,option]]",dv
```

LOCK / UNLOCK (SPERREN / ENTSPERREN)

```
OPENlf,dv,15:PRINT#lf,"L[n][pfad]:name":CLOSElf
```

UMBENENNEN (Dateien und Unterverzeichnisse)

```
OPENlf,dv,15:PRINT#lf,"R[n][pfad]:neuer name=[n][pfad]:dateiname":CLOSElf
```

```
RENAME[Dn,]"dateiname"TO[Dn]"neue datei"[,Udv]
```

SPEICHERN

```
SAVE"[[@][n][pfad]:dateiname",dv
```

```
BSAVE"[@]dateiname"[,Dn][{ON|,}Udv][,Bb][,Pa TO Pe
```

```
DSAVE"[@]dateiname"[,Dn][{ON|,}Udv]
```

LÖSCHEN

```
OPENlf,dv,15:PRINT#lf,"S[n][pfad]:dateiname[, [n][pfad]:dateiname...]:CLOSElf
```

```
SCRATCH"dateiname"[,Dn][{ON|,}Udv]
```

VERIFIZIEREN

```
VERIFY"[n][pfad]:dateiname",dv[,sa]
```

```
DVERIFY"dateiname"[,Dn][{ON|,}Udv]
```

Relativ-Dateibefehle

OPEN (REL-Datei)

```
OPENlf,dv,sa,"[n][pfad]:dateiname[{",L"+CHR$(rl)}]
```

```
DOPEN#lf,"Dateiname"[,Lrl][,Dn][,Udv]
```

POSITION

```
PRINT#lf, "P"+CHR$(ch)+CHR$(lr)+CHR$(lh) [+CHR$(of)]
RECORD#lf, rn[, of]
```

Sonder-HD-Befehle**S-C (SCSI-Befehl-Senden)**

```
OPENlf, dv, 15:PRINT#lf, "S-C"CHR$(de) CHR$(bl) CHR$(bh) CHR$(cb) ...
```

SWAP (TAUSCHEN)

```
OPENlf, dv, 15:PRINT#lf, "S-x":FORT=1TO500:NEXT:CLOSElf
```

Echtzeituhr-Befehle**T-RA (Zeit lesen in ASCII)**

```
OPENlf, dv, 15:PRINT#lf, "T-RA"
```

T-WA (Zeit schreiben in ASCII)

```
OPENlf, dv, 15:PRINT#lf, "T-WAdw. mo/da/yr hr:mi:se xM":CLOSElf
```

T-RD

```
OPENlf, dv, 15:PRINT#lf, "T-RD"
```

T-WD

```
OPENlf, dv, 15:PRINT#lf, "T-WD"+CHR$(Byte0)+CHR$(Byte1)+CHR$(Byte2)+CHR$(Byte3)
+CHR$(Byte4)+CHR$(Byte5)+CHR$(Byte6)+CHR$(Byte7):CLOSElf
```

T-RB

```
OPENlf, dv, 15:PRINT#lf, "T-RB"
```

T-WB

```
"T-WB"+BCD Zeit
```

SCHREIBSCHUTZ

```
OPENlf, dv, 15:PRINT#lf, "W-x":CLOSElf
```

Direktzugriffsbefehle**OPEN (Direktzugriffskanal)**

```
OPENlf, dv, sa, "[bu]"
```

BLOCKZUTEILUNG

```
PRINT#lf, "B-A: "; n; t; s
```

BLOCKAUSFÜHRUNG

```
PRINT#lf, "B-E "; ch; n; t; s
```

BLOCKFREIGABE

```
PRINT#lf, "B-F: "; n; t; s
```

CMD Hard Drive Anleitung

CMD übernimmt keine Arbeitskosten von unautorisierten Reparaturstellen. Die Kosten der autorisierten Reparaturstellen werden nur innerhalb der Garantiezeit übernommen. CMD übernimmt keine Haftung für Geräte, die von autorisierten Reparaturstellen beschädigt werden. CMD behält das Recht an ihren Geräten Änderungen durchzuführen, ohne daß schon gekaufte Geräte in den Genuß dieser Änderungen kommen. Kunden erhalten gratis ein Upgrade der nächsten Version des Betriebssystems. CMD kann das Porto dem Kunden in Rechnung stellen.

Das Zurücksenden der HD

Die HD kann an CMD innerhalb 30 Tagen zurückgesandt werden. Der Kunde erhält den Kaufpreis - 10% zurück. Frachtspesen und Steuern werden nicht zurückerstattet.

Die Ware muß im Originalkarton und mit kompletten Zubehör zurückgesandt werden. Frachtspesen werden vom Kunden bezahlt. Ein Brief mit Begründung der Rücksendung muß der Ware beigelegt werden. Der Brief soll auf eine 'RAN' (Return-Authorization-Number) hinweisen, die von CMD vorher ausgeteilt wurde. Die RAN muß auf dem Karton sichtbar sein. Diese Nummer bekommt der Kunde nach einem Telefonat mit CMD.

Problembericht

Falls ein Problem mit ihrer HD auftaucht, Sie Auskunft brauchen oder uns Erfahrungen mit Ihrer HD mitteilen wollen, füllen Sie folgenden Bericht aus und senden Sie ihn an Creative Micro Designs. Wir bedanken uns!

NAME: _____

ADRESSE: _____

TEL.NR. _____

HD-TYP: _____ SERIELLNUMMER: _____

Detaillierte Beschreibung des Zustands:

Anweisungen um den Fehlerzustand wieder hervorbringen: (Wenn möglich, senden Sie uns eine Kopie des Softwares.)
