

Hans C. Nieder

MS-DOS VERSION 3

**Eine leicht verständliche Einführung
in 3.0 und 3.1 mit Beispielen und Übungen,
alphabetischer Befehlsübersicht und
Hinweisen zur Konfiguration**

MS-DOS Version 3

Hans C. Nieder

MS-DOS Version 3

Eine leichtverständliche Einführung
in 3.0 und 3.1
mit Beispielen und Übungen,
alphabetischer Befehlsübersicht und
Hinweisen zur Konfiguration

Markt & Technik Verlag

CIP-Kurztitelaufnahme der Deutschen Bibliothek

Nieder, Hans C.:

MS-DOS Version 3 : e. leicht verständl. Einf. in 3.0 u. 3.1
mit Beispielen u. Übungen, alph. Befehlsübersicht u. Hinweisen zur Konfiguration / Hans C. Nieder. –
Haar bei München : Markt-und-Technik-Verlag, 1986.
ISBN 3-89090-201-4

Die Informationen im vorliegenden Buch werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht.

Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt.

Bei der Zusammenstellung von Texten und Abbildungen wurde mit größter Sorgfalt vorgegangen.

Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Verlag, Herausgeber und Autoren können
für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine
Haftung übernehmen.

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

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien.

Die gewerbliche Nutzung der in diesem Buch gezeigten Modelle und Arbeiten ist nicht zulässig.

MS-DOS® ist ein Warenzeichen der Microsoft Corp., USA

15 14 13 12 11 10 9 8 7 6
89 88 87

ISBN 3-89090-201-4

© 1986 by Markt & Technik, 8013 Haar bei München

Alle Rechte vorbehalten

Einbandgestaltung: Grafikdesign Heinz Rauner

Druck: Jantsch, Günzburg

Printed in Germany

Inhaltsverzeichnis

	Vorwort	9
1	Was ist MS-DOS?	11
2	Neues in der MS-DOS-Version 3.1 (Vergleich zur Version 2)	13
2.1	Neue Befehle	14
2.2	Neue MS-DOS-Konfigurations-Befehle (CONFIG.SYS)	14
2.3	Geänderte Befehle	15
2.4	Weitere Änderungen	16
3	Die Kompatibilität von MS-DOS	17
4	Die Tastatur	19
4.1	Allgemeines	19
4.2	Die Belegung der Funktionstasten	21
4.3	Beispiele für die Anwendung der Funktionstasten	23
5	Eine Übersicht der MS-DOS-Befehle	25
5.1	Allgemeines	25
5.2	Interne Befehle	27
5.3	Externe Befehle	29
6	Allgemeine Einführung in das Arbeiten mit MS-DOS 3.1	31
6.1	Der neue Computer kommt	31
6.2	Start mit MS-DOS	33
6.3	Kopien von Original Disketten	35
6.4	Die wichtigsten Befehle im Überblick	40
6.5	Drucken von Texten und vom Bildschirm	53
6.6	Erzeugen von Sicherheitskopien (von Diskette und Platte)	54

7	Diskette und Winchester	57
7.1	Aktuelles Laufwerk	57
7.2	Einteilung von Diskette und Festplatte	58
7.3	Disketten	59
7.4	Festplatten	62
8	Das MS-DOS-Inhaltsverzeichnis	63
8.1	Befehle zur Bearbeitung von Directories	66
8.2	Anlegen eines Sub-Directory (MKDIR oder MD)	69
8.3	Anzeigen und Ändern des aktuellen Directory	70
8.4	Löschen eines Sub-Directory (RMDIR oder RD)	71
8.5	Anzeigen des Inhaltsverzeichnisses eines Directories (DIR)	72
8.6	Löschen von Dateien in einem Directory (DEL oder ERASE)	74
8.7	Kopieren von Dateien zwischen Directories (COPY)	75
8.8	Suchen von Dateien in den Directories (SEARCH)	76
8.9	Zusammenfassendes Beispiel	77
9	Datenumleitung und Filter	79
9.1	Datenumleitung	80
9.2	Befehlsverkettung und Filter	81
10	Gerätenamen	85
11	Die MS-DOS-Befehle mit Übungen	87
	APPLY	89
	ASGNPART	91
	ASSIGN	93
	ATTRIB	95
	BACKUP	97
	BOOTF	101
	BREAK	102
	CHDIR	103
	CHKDSK	105
	CHMOD	108
	CIPHER	110
	CLS	112
	COMMAND	113
	COMP	115
	CONFIGUR	117
	COPY	119
	CTTY	122
	DATE	123
	DEL und ERASE	125

DIR	127
DISKCOMP	131
DISKCOPY	133
ECHO	135
EDLIN	137
EXE2BIN	146
EXIT	148
FC	149
FIND	154
FOR	157
FORMAT	158
GOTO	161
GRAFTABL	163
GRAPHICS	164
IF	165
JOIN	167
KEYBxxxx	169
LABEL	171
MKDIR	173
MODE	175
MORE	179
NODEBUG	180
PART und FDISK	181
PATH	182
PAUSE	183
PREP	184
PRINT	185
PROMPT	188
PSC	190
RDCPM	191
RECOVER	193
REM	195
RENAME	196
RESTORE	198
RMDIR	201
SEARCH	202
SELECT	204
SET	206
SHARE	208
SHIFT	209
SHIP und SHIPDISK	211
SORT	212
SUBST	214

	SYS	216
	TIME	217
	TREE	218
	TYPE	220
	VER	222
	VERIFY	223
	VOL	224
12	BATCH-Befehle	225
13	Die Konfiguration von MS-DOS	233
13.1	Die CONFIG.SYS-Befehle	234
	Anhang	
I	MS-NET (Netzwerk)	241
II	Begriffserklärungen	242
III	Umrechnungstabellen	246
IV	Zeichendarstellungen/Code-Tabelle	248
	Übersicht weiter Markt&Technik-Bücher	252

Vorwort

Durch das Erscheinen des IBM PC (Ende 1980 in den USA) gab es den großen Durchbruch auf dem Microcomputermarkt (Microcomputer gibt es seit ca. Mitte der 70er Jahre - Verbreitung fanden diese ungefähr ab Ende der 70er Jahre). In den folgenden Jahren brachten viele andere Hersteller kompatible Computer auf den Markt. Es entstand eine ganze Industrie für die zusätzliche Hard- und Software. Daraus entstand ein Computerstandard, für den heute das wohl größte Angebot an Zusatzhardware und Software zur Verfügung steht.

MS-DOS (PC-DOS) ist heute das Standard-Betriebssystem für 16-Bit-Personal-Computer. MS-DOS ist nicht nur einfach an verschiedene Hardware anpaßbar, sondern auch sehr leicht zu erlernen und zu bedienen.

Das vorliegende Buch ist aus der Durchführung von Seminaren zum Betriebssystem MS-DOS entstanden und entsprechend aufgebaut. Sie finden in den ersten Kapiteln (1 bis 9) eine Einführung mit den wichtigsten Grundlagen zur Bedienung von MS-DOS. Die Kapitel 10 bis 12 enthalten alle Befehle, alphabetisch geordnet, mit einer genauen Beschreibung. In den einzelnen Kapiteln werden viele Beispiele dargestellt, welche Sie an Ihrem Computer durcharbeiten sollten. Dadurch wird der Lerneffekt gesteigert. Für den Fortgeschrittenen bietet das Buch ab Kapitel 10 ein hilfreiches Nachschlagewerk (mit MS-DOS-Konfiguration im Kapitel 13).

Im Anhang werden zu den wichtigsten Begriffen Erläuterungen gegeben. Diese wurden bewußt an das Buchende gesetzt, damit Sie später nicht laufend darüber hinwegblättern müssen.

Ferner finden Sie im Anhang eine Darstellung zu MS-NET (PC-NET) Netzwerk.

Ziel des Buches ist es, Sie zu befähigen, Ihre Arbeiten mit MS-DOS 3.1 am Computer erledigen zu können. Das Buch soll Ihnen ein ständiger Begleiter sein.

Ich bedanke mich bei allen, die mich so freundlich bei der Beschaffung von Unterlagen oder in anderer Weise unterstützt haben.

Durch unbürokratisches Überlassen von Unterlagen verschiedener Hersteller war es möglich, dieses Buch so schnell zu erstellen, damit es kurzfristig nach Erscheinen der MS-DOS-Version 3.1 gedruckt werden konnte.

Mörfelden, im Januar 1986

Hans C. Nieder

1 Was ist MS-DOS?

Eigentlich müßte es heißen: Was ist ein Betriebssystem? MS-DOS ist eine Ansammlung von Programmen, die zum Betrieb eines Computers notwendig ist.

Ein Computer (Hardware) ist allein für sich genommen ein Gerät, welches Sie zwar einschalten, jedoch nicht betreiben können. Erst durch das Betriebssystem "erwacht die Anlage zum Leben". Ein Betriebssystem ist nichts anderes als eine Sammlung von Programmen, welche den Computer und seine Peripherie (Drucker, Plotter, Diskettenlaufwerke, Platte, Bildschirm, usw.) verwaltet. Ferner unterstützt es die Ausführung von Anwenderprogrammen (Finanzbuchhaltung, Textsystem usw.).

Ein Anwenderprogramm wird speziell für die Lösung einer Aufgabe erstellt, unabhängig vom Computersystem. Die Verbindung zur Hardware wird durch das Betriebssystem hergestellt. Zusätzlich können Sie mit MS-DOS viele Aufgaben, wie das Kopieren von Daten und Programmen, durchführen.

Wenn Sie Ihren Computer einschalten, wird das MS-DOS in den Speicher (RAM) geladen. Dieser Vorgang erfolgt automatisch (System-Bootup). Das Programm, welches den Bootup durchführt, ist im Computer enthalten und wird durch das Einschalten aktiviert. Anschließend erscheint am Bildschirm die Befehlseingabe-Bereitschaftsmeldung (der Prompt).

Die Aufgaben von MS-DOS lassen sich in vier Bereiche aufteilen:

1. Verwaltung und Transport von Daten
2. Unterstützung und Verwaltung der/des Benutzer(s)
3. Zuteilung und Verwaltung der Betriebsmittel (Drucker, usw.)
4. Steuerung von Prozessen

MS-DOS ist die Abkürzung von **Microsoft Disk Operating System**. Dieses Betriebssystem konnte sich in kurzer Zeit durchsetzen, vor allem durch die Entscheidung von IBM, ihre PCs damit auszustatten (IBM-Bezeichnung PC-DOS).

MS-DOS kann heute als Quasi-16-Bit-Standard-Betriebssystem bezeichnet werden (bei 8-Bit-Computern CP/M). Das große Angebot an Software ist inzwischen schon allein ein Grund zum Einsatz von MS-DOS.

Welchem Betriebssystem die Zukunft auf den neuen 32-Bit-Computern (z.B. IBM PC/AT, Zenith Z-240) gehört, ist zur Zeit fraglich. Wird das "Rennen" MS-DOS oder XENIX gewinnen? Sicherlich wird entscheidend sein, ob für XENIX künftig mehr Anwendungsprogramme verfügbar sind. Der wesentliche Vorteil von XENIX ist die Mehrplatzfähigkeit. Aber auch MS-DOS entwickelt sich weiter, wird jedoch nie mehrplatzfähig sein (keine Terminals als weitere kostengünstige Arbeitsplätze anschließbar).

MS-DOS besteht aus vier Teilen (und diversen Dienstprogrammen):

MS-DOS-Teil	Kurzbeschreibung
-------------	------------------

BOOT-LOADER	wird direkt nach dem Start des Computers als erstes Programm ausgeführt. Es startet das MS-DOS, indem es IO.SYS in den Speicher des Computers lädt, und das System bootet.
--------------------	--

IO.SYS	enthält alle systemspezifischen Programme zur Abwicklung der Kommunikation mit den angeschlossenen Geräteeinheiten (Drucker usw.).
---------------	--

MSDOS.SYS	enthält den organisatorischen Teil des Betriebssystems zur Verwaltung der Dateien (Daten und Programme).
------------------	--

COMMAND.COM	Interpretiert alle eingegebenen Befehle und führt sie aus, wobei geprüft wird, ob es sich um einen zulässigen Befehl handelt. Ferner enthält es eigene (interne) Befehle.
--------------------	---

2 Neues in der MS-DOS Version 3.1 (im Vergleich zur Version 2)

Wenn Sie bereits mit den Versionen 1.xx und 2.xx von MS-DOS gearbeitet haben, werden Sie sehr schnell feststellen, daß sich im Grunde nicht viel geändert hat. Die wesentlichen Neuerungen sind vor allem:

- Unterstützung der nationalen Zeichensätze mit richtiger Datums- und Uhrzeitdarstellung sowie Dezimalkomma statt -punkt
- Betriebssystem für Netzwerke (Zusammenschluß von mehreren PCs)
- Es können auch Disketten-Laufwerke mit 1,2 MByte Speicherkapazität verwendet werden.

Allein diese Erweiterungen machen es für Sie sinnvoll, die alte Version 2.xx gegen die neue auszutauschen. Bezüglich der Kompatibilität wenden Sie Ihre Aufmerksamkeit dem nächsten Kapitel zu. Sie sollten aber berücksichtigen, daß MS-DOS 3.1 ca. 12 kByte mehr Speicher (RAM) benötigt.

MS-DOS 3.0 oder 3.1?

Wenn Sie planen, ein Netzwerk einzusetzen, sollten Sie ausschließlich mit der Version 3.1 arbeiten. Die Version 3.0 enthält zwar bereits den Support (Unterstützung) für MS-NET, jedoch erst in der Version 3.1 sind alle notwendigen Betriebssystem-Voraussetzungen vorhanden. Hervorzuheben ist vor allem, daß die neue Version auf allen PC-Modellen lauffähig ist (nicht nur auf den sogenannten ATs). Im Rahmen dieses Buches kann aber nicht auf alle Neuerungen eingegangen werden (z.B. neue Funktionsaufrufe - diese werden nur von Programmierern benötigt).

2.1 Neue Befehle

Befehl	ab Version	Kurzbeschreibung
ATTRIB	3.0	Setzen eines Read-Only-Attributs ("Nur lesen") - Software-Schreibschutz
GRAFTABL	3.0	unterstützt erweiterten Zeichensatz im Grafik-Mode (z.B. deutsche Umlaute)
JOIN	3.1	ersetzt eine Laufwerksbezeichnung durch ein Sub-Directory
LABEL	3.0	Disketten-/Plattenname ändern
SELECT	3.0	Nationale Tastaturbelegung und Datums-/Uhrzeit-Darstellung wählen und damit eine neue Diskette erstellen (Original i.d.R: MS-DOS-System-Diskette)
SHARE	3.0	Installieren des File-Sharing-Support (Netzwerk)
SUBST	3.1	ersetzt ein Sub-Directory durch eine Laufwerksbezeichnung

2.2 Neue MS-DOS-Konfigurations-Befehle (CONFIG.SYS)

Beachten Sie zu diesen Befehlen das Kapitel 13 mit näheren Erläuterungen.

Befehl	ab Version	Kurzbeschreibung
COUNTRY	3.0	Setzen von landesabhängigen Informationen (Datums-/Uhrzeit-Darstellung, Dezimalpunkt)
FCBS	3.0	definiert die Anzahl der maximal gleichzeitig geöffneten FCBs
LASTDRIVE	3.0	Maximale Anzahl der logischen (nicht physikalischen) Laufwerke

2.3 Geänderte Befehle

Befehl	Kurzbeschreibung
BACKUP	Sichern von Dateien (z.B. von Platte auf Diskette) Bei einigen Herstellern neue Parameter. Unterstützt wird jetzt auch das Sichern zwischen verschiedenen Medien (Platte auf Diskette, Platte auf Platte, usw.). Datumsdarstellung jetzt auch deutsch (Tag-Monat-Jahr) möglich.
DATE	Datum anzeigen/eingeben Es ist jetzt landesabhängig möglich, das Datum in der Form Monat-Tag-Jahr oder Tag-Monat-Jahr oder Jahr-Monat-Tag einzugeben.
DISKCOMP	Vergleichen von 2 Disketten Es werden auch Laufwerke mit 1,2 MByte unterstützt.
DISKCOPY	Kopieren einer Diskette auf eine zweite Es werden auch Laufwerke mit 1,2 MByte unterstützt.
FORMAT	Formatieren einer Diskette oder Festplatte Es werden auch Laufwerke mit 1,2 MByte unterstützt.
GRAPHICS	Kopieren von einer Grafik-Bildschirmdarstellung auf einen Drucker Es werden jetzt mehrere Drucker unterstützt.
LIB	
LINK	Diese beiden Programme wurden erweitert (nur für Programmierer).
PRINT	Drucken (läuft gleichzeitig zu anderer Arbeit ab) Neue Parameter für den Druck.
RESTORE	Zurückkopieren von mit BACKUP gesicherten Dateien Bei einigen Herstellern neue Parameter. Unterstützt wird jetzt auch das Rückkopieren zwischen verschiedenen Medien (Platte auf Diskette, Platte auf Platte, usw.). Datumsdarstellung jetzt auch deutsch (Tag-Monat-Jahr) möglich.

2.4 Weitere Änderungen

Bei den externen Befehlen können Sie jetzt vor den Befehl den Namen des Directory (Verzeichnis) angeben, in welchem sich dieser befindet (ab Version 3.0).

Bisher war es notwendig, ein Programm, welches in einem Sub-Directory benötigt wurde, in dieses zu kopieren, oder aber mit PATH (siehe Kapitel 13) eine entsprechende Zuweisung durchzuführen.

Das Befehlsformat bis Version 2.xx war:

Laufwerksangabe:Befehlsname Parameter

Das Befehlsformat ab Version 3.xx ist:

Laufwerksangabe:Verzeichnisname\Befehlsname Parameter

Der Befehl ASSIGN wurde bei Zenith angeglichen, so daß dieser jetzt mit denen anderer Hersteller identisch ist (wurde in der Version 2 MAP genannt).

Für Programmierer ist noch zu erwähnen, daß sich einige Funktionsaufrufe geändert haben (38, 3D usw.) und einige neu hinzugekommen sind (59, 5A, usw.). Ferner gibt es neue bzw. geänderte Interrupts (z.B. 24, 2F).

3 Kompatibilität von PCs und MS-DOS

Grundsätzlich muß bei der Kompatibilität zwischen Hardware und Software unterschieden werden. Viele Systeme sind zwar software-, jedoch nicht hardwarekompatibel.

Wer ist schon wirklich kompatibel? Nur sehr wenige Computer. Wenn man von Kompatibilität spricht, so versteht man IBM-kompatibel. Wenn Ihnen oder Ihrem Lieferanten nicht bekannt ist, ob ein Programm oder Hardware-Teil auf Ihrem Computer-System lauffähig ist, sollten Sie dies vor dem Kauf testen.

Kleine Unterschiede in der Bedienung gibt es immer. Viele "Kompatible" sind in der Bedienung oder Bildschirmdarstellung besser.

Die Kompatibilität hat den Vorteil des Standards. Es gibt eine riesige Palette von Programmen, Peripherie und Hardware-Zusätze, welche Sie verwenden können. Dies bedeutet auch, daß die Hersteller wegen der größeren Verkaufszahlen günstiger kalkulieren können.

Fraglich ist jedoch, ob Ihnen als Anwender die Lösung einer Aufgabe oder die Kompatibilität wichtiger ist. Dies sollte bei aller Euphorie bedacht werden.

Kompatibilität zwischen MS-DOS 2 und 3

Programme, welche Sie unter älteren MS-DOS-Version eingesetzt haben, können auch mit 3.1 benutzt werden.

Beachten Sie jedoch, daß MS-DOS 3.1 mehr Speicher als die Versionen 1 und 2 benötigt. War der Speicher Ihres Computers bisher gerade noch ausreichend, so benötigen Sie jetzt eine Erweiterung.

Wenn der Speicher des Computers nicht ausreicht, um ein Programm zu benutzen, so erhalten Sie von MS-DOS eine Fehlermeldung. Wenn Ihre Software bestimmte MS-DOS-Befehle benutzt, so kann es bei geänderten Befehlen zu Schwierigkeiten kommen. Dieser Fall tritt sehr selten auf.

Vor der Umstellung auf MS-DOS Version 3.1 sollten Sie sich mit Ihrem Software-Lieferanten in Verbindung zu setzen, damit Probleme vermieden werden.

Vermeiden Sie vor allem, Teile von MS-DOS 2 unter der Version 3.1 zu benutzen. Es gilt der allgemeine Grundsatz, daß verschiedene Betriebssystem-Versionen nie miteinander "gemischt" werden dürfen, auch wenn dies zufällig einmal funktioniert!

4 Die Tastatur Ihres PCs

4.1 Allgemeines

Die Tastatur Ihres Computers umfaßt mehrere Gruppen:

- Kompletter Buchstabensatz entsprechend einer Schreibmaschine (mit der Betätigung der Taste **Caps Lock** können Sie Ihre Tastatur auf Großschreibung umschalten - dies gilt nur für Buchstaben).
- Die Funktionstasten finden Sie unten als Bild dargestellt (diese können bei den einzelnen Computern teilweise etwas anders plaziert sein). Die Bedeutung der Tasten ist auf den nächsten Seiten näher beschrieben.
- Auf der rechten Seite Ihrer Tastatur finden Sie ein Zifferntastenfeld. Um mit diesem arbeiten zu können, muß vorher die Taste **Num Lck** betätigt werden. Damit wird jedoch gleichzeitig die Möglichkeit der Nutzung als Funktionstasten abgeschaltet. Betätigen Sie **Num Lck** ein weiteres Mal, so ist der ursprüngliche Zustand wiederhergestellt.

Mit Hilfe der Funktionstasten können Sie die Befehlszeilen leicht ändern. Die zuletzt eingegebene Zeile kann aus dem Tastatureingabespeicher (RAM) komplett oder in Teilen zurück in die Befehlszeile (Bildschirmanzeige) kopiert werden. Der Tastatureingabespeicher ist am Bildschirm nicht sichtbar, dort befindet sich immer der zuletzt eingegebene MS-DOS-Befehl.

Eine Befehlszeile können Sie einfach

- durch Tastendruck wieder eingeben,
- korrigieren,
- in geänderter Form eingeben.



Bild 4.1 Die PC-Tastatur

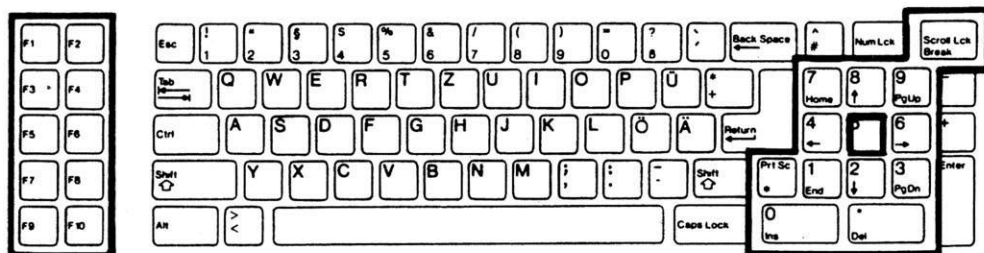


Bild 4.2 Funktionstasten

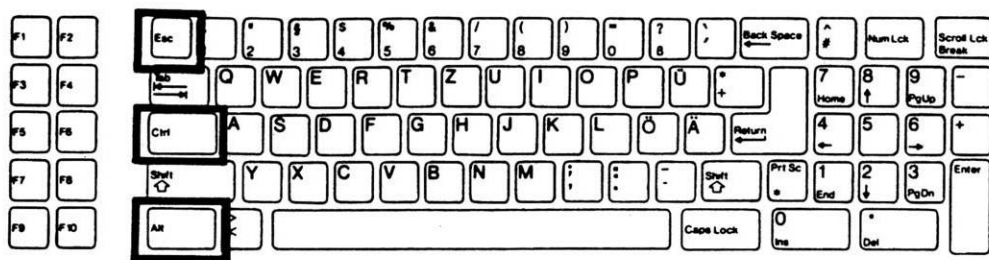


Bild 4.3 Besondere Funktionstasten

4.2 Die Belegung der Funktionstasten (Übersicht)

Die angegebenen Funktionen gelten nur für MS-DOS! Bei der Benutzung einer Anwendungssoftware haben die Tasten meist eine andere Bedeutung.

Taste	Funktion	Beschreibung
F1	kopiert ein Zeichen	kopiert ein Zeichen aus dem Eingabespeicher in die neue Zeile
F2x	kopiert bis Zeichen (x)	kopiert alle Zeichen, bis zum eingegebenen Zeichen (x) vom Eingabespeicher (RAM) in die neue Zeile am Bildschirm
F3	kopiert alle Zeichen	kopiert alle Zeichen des Eingabespeichers in die neue Zeile am Bildschirm
F4x	überspringt Zeichen bis x	überspringt alle Zeichen des Eingabespeichers, bis (x) erkannt wird
F5	neuer Eingabespeicher	schreibt die neue Zeile in den Eingabespeicher
F6	beendet die Eingabe	erzeugt ein Ctrl Z , welches das Zeichen für ein Dateiende darstellt
F7	Nullzeichen	gibt ein Ctrl @ aus, welches den Wert 00 (HEX) darstellt
Alt-Break	Eingabe löschen	löscht den Tastaturspeicher (der eingegebene Befehl wird abgebrochen)
Backspace	Eingabe löschen	löscht ein Zeichen aus der Eingabezeile am Bildschirm und im Eingabespeicher

Ins	einfügen	schaltet die Eingabe auf Einfügemodus um
Del	löschen	löscht ein folgendes Zeichen im Eingabespeicher, jedoch nicht in der Eingabezeile bzw. am Bildschirm
Esc	Eingabe löschen	löscht die gesamte Eingabezeile, nicht jedoch den Eingabespeicher
Alt-Esc	Key-Klick abschalten	schaltet das Klicken der Tastatur ein und aus (nur ZENITH)
Ctrl-Alt-Del	Systemstart	den Computer neu starten (booten, also das MS-DOS komplett neu laden)
Ctrl-Alt-Ins	Monitoraufruf	kehrt zum Monitorprogramm zurück
Ctrl-Alt-Return	Monitoraufruf	unterbricht das laufende Programm, kehrt zum Monitorprogramm zurück und zeigt die aktuellen Registerinhalte an (nur ZENITH)
Ctrl-Break	Befehlsabbruch	unterbricht die Ausführung eines MS-DOS-Befehls (siehe Befehl BREAK)
Ctrl-C	Befehlsabbruch	unterbricht die Ausführung eines MS-DOS-Befehls (siehe Befehl BREAK)
Ctrl-H	Zeichen löschen	löscht das zuletzt eingegebene Zeichen am Bildschirm (wie Taste Back Space)
Ctrl-I	Tabulator	Cursor am Bildschirm zur nächsten Tabulatorposition bewegen (wie Taste Tab)
Ctrl-Return	Zeilenende	fügt ein Zeilenende bzw. Zeilenvorschub ein
Ctrl-J	Zeilenende	fügt ein Zeilenende bzw. Zeilenvorschub ein
Ctrl-NumLck	Bildschirmstopp	unterbricht die Bildschirmanzeige, bis eine beliebige andere Taste betätigt wird
Ctrl-S	Bildschirmstopp	unterbricht die Bildschirmanzeige, bis eine beliebige andere Taste betätigt wird
Ctrl-P	Druckausgabe	Bildschirm- und -ausgaben gleichzeitig drucken (erneutes Betätigen schaltet die Funktion ab)
Ctrl-PrtSc	Druckausgabe	Bildschirm- und -ausgaben gleichzeitig drucken (erneutes Betätigen schaltet die Funktion ab)
Ctrl-Z	beendet die Eingabe	erzeugt ein Ctrl Z (1Ah), welches das Dateiende markiert (wie Taste F6)
Shift-PrtSc	Bildschirm drucken	erstellt eine Kopie der Bildschirmanzeige am Drucker (Hard-Copy-Funktion)

4.3 Beispiele für die Anwendung der Funktionstasten

Sie wollen sich aus dem Inhaltsverzeichnis einer Diskette die Daten der Datei HANDBUCH.TXT anzeigen lassen. Sie tippen ein:

`dir handbuch.txt`

Return-Taste

Der Befehl wird ausgeführt (Ihre Eingabe steht für Sie unsichtbar im Tastatureingabespeicher). Mit nur zwei Tasten ist es jetzt möglich, diesen Befehl zu wiederholen. Betätigen Sie die Tasten:

F3

Return-Taste

Sie wollen Informationen über sämtliche Dateien, die Sie unter dem Namen HANDBUCH gespeichert haben. Der Namenszusatz unterscheidet die einzelnen Dateien. In der Eingabezeile muß dafür nur `.txt` gegen den Stern (*) getauscht werden. Geben Sie ein:

F2-Taste und `t`

Es werden sämtliche Zeichen, bis zum `t` ausschließlich, in die Eingabezeile (am Bildschirm sichtbar) kopiert. Es erscheint:

`dir handbuch.`

Geben Sie dann den Stern (*) ein, und betätigen Sie die Taste **Return**.

Da es sich bei der Datei HANDBUCH.TXT um einen Text handelt, können Sie den Inhalt am Bildschirm anzeigen lassen. Verändern Sie die letzte Eingabe, um folgenden Befehl zu erhalten:

`type handbuch.txt`

Sie überschreiben die ersten drei Zeichen der alten Zeile (`dir`), indem Sie die Buchstaben `typ` tippen. Sie betätigen die Taste **Ins**, fügen ein `e` hinzu, betätigen die Tasten **F2 *** und schreiben als letztes `txt`. Nach Betätigung von **Return** wird der Inhalt auf dem Bildschirm angezeigt.

Ihre Eingaben zum letzten Beispiel noch einmal in der Übersicht:

Tasten	Funktion	Ergebnis am Bildschirm
t	schreibt ein Zeichen	t
y	schreibt ein Zeichen	ty
p	schreibt ein Zeichen	typ
Ins	schaltet um auf Einfügen	
e	schreibt ein Zeichen	type
F2*	kopiert Zeichen bis zum*	type handbuch.
t	schreibt ein Zeichen	type handbuch.t
x	schreibt ein Zeichen	type handbuch.tx
t	schreibt ein Zeichen	type handbuch.txt
Return	gibt die Zeile an das MS-DOS-Programm COMMAND.COM zum Ausführen	Inhalt wird angezeigt

Bei Verwendung einer deutschen Tastatur benötigen Sie bei der Arbeit mit MS-DOS Zeichen, welche Sie nur durch Betätigung mehrerer Tasten gleichzeitig erzeugen können:

Tasten-Kombination	Zeichen	Bemerkung
Ctrl + Alt + ü	[	Eckige Klammer auf
Ctrl + Alt + +	]	Eckige Klammer zu
Ctrl + Alt + <	\	Back-Slash
Ctrl + Alt + 2	@	Klammeraffe
Ctrl + Alt + F1		Umschalten auf ASCII Tastatur
>		
Ctrl + Alt + F2		Umschalten auf deutsche Tastatur

Auf dem Bildschirm wird das |-Zeichen als unterbrochener Strich | dargestellt.

5 Übersicht der MS-DOS-Befehle

5.1 Allgemeines

Befehle sind Anweisungen an das Betriebssystem, bestimmte Aufgaben durchzuführen (Kopieren, Drucken usw.). Sie werden von Ihnen an der Tastatur eingegeben und gleichzeitig am Bildschirm angezeigt. Sie können diese auch in einer speziellen Datei speichern und zu einem späteren Zeitpunkt mit einem einzigen Befehl abrufen (siehe Kapitel 12).

Es werden zwei Arten von Befehlen unterschieden:

Interne Befehle. Diese sind jederzeit verfügbar. Sie sind in der Programm-Datei COMMAND.COM enthalten, die beim Start des Systems in den Speicher geladen wird.

Externe Befehle. Dies sind kleine Programme (sogenannte Dienstprogramme), welche Sie auf der Diskette oder Festplatte unter einem eigenen Namen finden. Wenn Sie einen externen Befehl eingeben, wird dieser zur Ausführung in den Speicher des Computers geladen.

Ausführung der Befehle

Bei Eingabe eines Befehls wird dieser von COMMAND.COM (wird beim Start in den Speicher geladen) gestartet. Die höchste Priorität besitzen die internen Befehle. COM- und EXE-Dateien verfügen über zweite und BAT-Dateien dritte Priorität.

Beispiel:

COPY ist ein interner Befehl. Dazu ist auf der Diskette die Datei COPY.COM vorhanden. Der interne Befehl COPY, nicht COPY.COM wird ausgeführt, wenn Sie den Befehl eingeben.

Beachten Sie also immer folgende Suchreihenfolge bei der Ausführung:

1. Interne Befehle (Bestandteil von COMMAND.COM)
2. COM-Dateien (externe Befehle mit Namenszusatz COM)
3. EXE-Dateien (externe Befehle mit Namenszusatz EXE)
4. BAT-Dateien (externe Befehle mit Namenszusatz BAT)

Achten Sie darauf, daß ein Befehlsname immer nur in einer Form vorliegt, damit die Ausführung des richtigen Befehls gewährleistet ist.

Wenn ein Befehl vom MS-DOS nicht gefunden wird, erhalten Sie die Fehlermeldung:

Bad command or filename

COMMAND.COM

Dieses Programm wird beim Systemstart in den Speicher (RAM - Random Access Memory) geladen. COMMAND.COM besteht aus 3 Teilen:

1. Initialisierungsteil (nur beim Start des Computers notwendig)
2. Beständiger Teil (Programmloader, Fehlerbearbeitung usw.)
3. Überschreibbarer Teil (Interne Befehle, Befehlsinterpreter usw.)

Der erste Teil wird nicht mehr benötigt, wenn er seine Funktion erfüllt hat.

Der beständige Teil bleibt erhalten, bis Sie den Computer neu starten. Der dritte Teil wird erst überschrieben, wenn der freie Speicher schon belegt ist und zur Ausführung eines Programms mehr Speicher benötigt wird.

In den folgenden Übersichten wird nicht angegeben, auf welchen Systemen der verschiedenen Hersteller die einzelnen Befehle lauffähig sind. Diese Angaben finden Sie bei der Befehlsdarstellung im Kapitel 11.

5.2 Interne Befehle

Interne Befehle sind Teil des System-Programms COMMAND.COM. Sie können diese Befehle jederzeit eingeben, wenn Sie sich in keinem Programm befinden, sondern auf Betriebssystemebene (es gibt zwischenzeitlich Programmiersprachen, wie z.B. GW-BASIC, welche einen Aufruf aus einem Programm ermöglichen). Diese wird durch das Befehlsanforderungszeichen, den Prompt (**Laufwerk-Buchstabe und >**) angezeigt.

Befehle	Funktion
BREAK	aktiviert/deaktiviert Programmunterbrechung
CHDIR oder CD	Directory wechseln
CLS	Bildschirm löschen
COPY	Dateien kopieren
CTTY	Peripheriegerät ändern, von welchem Befehle eingegeben werden
DATE	Datum anzeigen/eingeben
DIR	Dateien anzeigen
ECHO	unterdrückt die Anzeige von Befehlszeilen in einer Batch-Datei; zeigt einen Kommentar an
ERASE oder DEL	Datei(en) löschen
EXIT	Verlassen Betriebssystem und Rückkehr in ein Anwenderprogramm
FOR	mehrmaliges Wiederholen eines MS-DOS-Befehls in einer Batch-Datei
GOTO	unbedingter Sprungbefehl in einer Batch-Datei

IF	macht die Befehlsausführung von der Erfüllung einer Bedingung abhängig (in Batch-Datei)
MKDIR oder MD PATH	Sub-Directory anlegen benennt Wege, die zu einem Verzeichnis führen, in dem ein externer Befehl gesucht werden soll
PAUSE	unterbricht die Bearbeitung einer Batch-Datei und gibt einen Text auf dem Bildschirm aus (Fortsetzung mit Betätigung einer Taste)
PROMPT	ändert Befehlsanforderungszeichen (Prompt)
REM	Kommentare in einer Batch-Datei
RENAME oder REN	Datei umbenennen
RMDIR oder RD	Sub-Directory löschen
SET	setzt den Wert einer Zeichenkette gleich dem Wert einer anderen Zeichenkette
SHIFT	erhöht die Zahl der Variablen in einer Batch-Datei auf mehr als 10
TIME	Uhrzeit anzeigen/ändern
TYPE	Datei-Inhalt anzeigen
VER	zeigt die Versionsnummer von IO.SYS und MS-DOS an
VERIFY	überprüft kopierte Dateien (Schalter)
VOL	zeigt den Namen einer Diskette oder Festplatte an (Label-Name)
d:	Arbeitslaufwerk umschalten

5.3 Externe Befehle

Externe Befehle sind Dienst-Programme, die vor jedem Start von der Diskette/Festplatte in den Speicher (RAM) geladen werden (führt COMMAND.COM automatisch durch). Diese überschreibt ein nachfolgend aufgerufenes Programm (Befehl) wieder.

Befehle	Funktion
APPLY	führt einen Befehl mehrere Male aus
ASGNPART	logische Laufwerksbezeichnung einer Festplatte-Partition zuweisen
ASSIGN	leitet Laufwerksanfragen um
ATTRIB	Datei auf "Nur lesen" setzen
BACKUP	sichert Dateien von Diskette oder Festplatte
BOOTF	Ausführung von Programmen ohne Festplatte
CHKDSK	überprüft die Platte und/oder eine Datei auf Fehler
CHMOD	Anzeigen und Ändern von Dateiattributen
CIPHER	ver- und entschlüsselt Dateien
COMMAND	führt interne Befehle wie externe Befehle aus
COMP	vergleicht den Inhalt zweier Dateien auf Gleichheit
CONFIGUR	modifiziert MS-DOS für Peripheriegeräte
DEBUG	für Programmtest und -modifikation
DETECT	untersucht Platte auf fehlerhafte Sektoren
DISKCOMP	vergleicht zwei Disketten
DISKCOPY	kopiert Disketten komplett
EDLIN	einfaches Textprogramm
EXE2BIN	formt Dateien von .EXE auf .COM um
FC	zeigt Unterschiede der Dateien
FDISK	verwaltet die Bereiche einer Festplatte (IBM)
FIND	sucht einen Text
FORMAT	formatiert Diskette oder Festplatte
GRAFTABL	ASCII-Tabelle für Farbgrafik-Mode laden
GRAPHICS	Grafik- und Sonderzeichen auf angeschlossenen Drucker ausgeben
JOIN	temporäres Laufwerk durch Sub-Directory ersetzen
KEYxxxx	Zeichensätze für verschiedene Länder laden
LABEL	Zuweisen/Anzeigen von Namen für Disks

LIB	zur Verwaltung von MS-DOS-Programm-Bibliotheken
LINK	Programm zum Binden von Programm-Modulen
MODE	modifiziert MS-DOS für Peripheriegeräte
MORE	zeigt die Bildschirmausgabe seitenweise an
NODEBUG	"ausschalten" des ROM-Debuggers
PART	teilt Festplatte in verschiedene Partitionen auf, und legt Boot-Partition fest.
PREP	Initialisieren einer Festplatte
PRINT	Ausgabe von Texten auf den Drucker
PSCdrucker	Grafik- und Sonderzeichen auf angeschlossenem Drucker ausgeben (Druckerauswahl)
RDCPM	kopiert CP/M-Dateien
RECOVER	erneuert fehlerhafte Dateien
RESTORE	schreibt gesicherte Dateien zurück
SEARCH	sucht Dateien auf der gesamten Platte
SELECT	Tastatur-Layout und Datums-Format installieren
SHARE	File-Sharing-Support laden (für MS-NET)
SHIP oder SHIPDISK	Festplatte sichern
SORT	sortiert, alphabetisch oder numerisch
SUBST	temporär Sub-Directory durch Laufwerk ersetzen
SYS	kopiert Systemdateien IO.SYS und MSDOS.SYS
TREE	zeigt das strukturierte Verzeichnis einer Platte

6 Allgemeine Einführung in das Arbeiten mit MS-DOS 3

Dieses Kapitel ist für den Anfänger bestimmt, der noch keinerlei Kenntnisse im Umgang mit MS-DOS hat. Grundlage für den Inhalt ist die Erfahrung, daß ca. 80 % der Anwender nur 20 % der Befehle des MS-DOS für die tägliche Arbeit benötigen.

Sie werden sehr schnell feststellen, daß man kein erfahrener Programmierer oder "EDV-Mann" sein muß, um mit MS-DOS umzugehen. Ich habe versucht, das Vorliegende mit möglichst einfachen und verständlichen Worten zu beschreiben. Sollten Sie Vorschläge und Anregungen für dieses Kapitel bzw. das Buch haben, so würde ich mich über Ihre Zuschrift freuen.

6.1 Der neue Computer kommt

Sie erhalten zusammen mit Ihrem Computer einen Bildschirm, Drucker und die Handbücher mit MS-DOS. Vergessen Sie bitte nicht, das sogenannte Enduser-Agreement (Lizenzvertrag) ausgefüllt an den Computer-Hersteller oder Händler zu senden. Dies ist sehr wichtig, da Sie als registrierter Anwender von MS-DOS immer benachrichtigt werden, wenn eine neue und korrigierte Version erscheint (Sie sollten nur die neueste Version benutzen, damit Sie bei Programm-Neuanschaffungen keine Probleme mit der Software bekommen). Diesen Vertrag müssen Sie allerdings auch aus rechtlichen Gründen unterschrieben absenden, da erst damit für Sie das Recht auf Nutzung des MS-DOS eintritt.

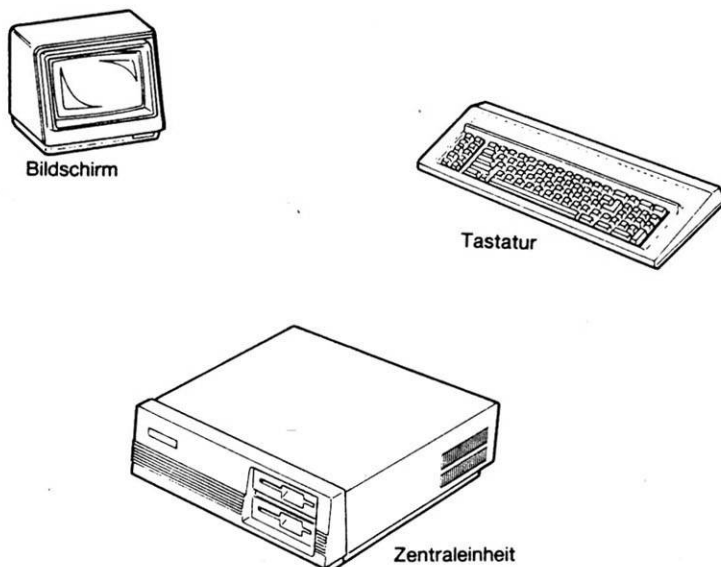


Bild 6.1 Die Hardware

MS-DOS verbindet die einzelnen Teile Ihres Personal Computers (PC) logisch zu einem einheitlichen System. Dies ist erforderlich, damit Sie Anwendungsprogramme auch benutzen können. MS-DOS stellt die Verbindung zwischen Programm und Hardware dar.

Sie haben entweder ein Disketten- oder Festplatten-System erworben. Eine Festplatte ist im Grunde genommen nichts anderes als eine "fest eingebaute Diskette mit hoher Speicherkapazität". Vergleichbar sind diese Speicher mit einem Aktenschrank. Wenn Sie bisher Unterlagen, Statistiken, Briefe, usw. abgelegt haben, so werden diese künftig auf der Festplatte oder Diskette gespeichert (der Verwalter der Daten ist das MS-DOS bzw. Ihr Anwendungsprogramm). Dabei wird in der EDV eine Kartei (z.B. Kunden, Lieferanten) als Datei bezeichnet. Beachten Sie hierzu auch das Kapitel 7 "Diskette und Festplatte".

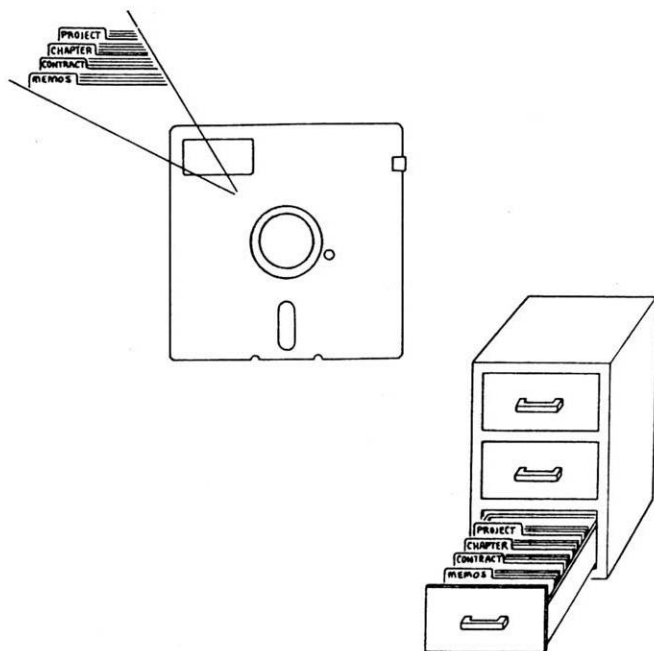


Bild 6.2 Die Diskette als Aktenschrank

6.2 Start mit MS-DOS

Schalten Sie bitte Ihren Computer ein. Als erstes kopieren Sie die MS-DOS-Diskette(n) - verwenden Sie niemals Original-Disketten.

Führen Sie Ihre MS-DOS-Diskette (wenn Sie über mehrere verfügen, die erste) in das erste Laufwerk ein, und schließen Sie dieses. Die meisten Computer starten dann automatisch das MS-DOS (wenn nicht, beachten Sie bitte Ihr Handbuch zum PC). Bei Festplatten-Systemen wird von der Diskette gestartet, wenn eine solche eingelegt ist. Bei Zenith-Computern kann auch wahlweise automatisch oder manuell von Diskette oder Festplatte gestartet werden.

Am Bildschirm erhalten Sie etwa folgende Anzeige:

```
Current date is Thu 8-22-1985
Enter new date (mm-dd-yy):
```

Geben Sie an dieser Stelle das Tagesdatum ein (bitte beachten Sie hierzu die Befehle **DATE** und **SELECT** in Kapitel 11 und **COUNTRY** im Kapitel 13). Wenn Sie ohne weitere Eingabe nur die Taste Return drücken, so bleibt das angezeigte Datum unverändert. Das Datum in der Anzeige (es kann bei Ihnen ein anderes sein) ist lediglich ein Beispiel (bei dem angegebenen handelt es sich um den 22. August 1985).

Als nächstes erscheint am Bildschirm:

```
Current time is 7:27:40.15
Enter new time:
```

Geben Sie an dieser Stelle die Uhrzeit ein. Wenn Sie ohne weitere Eingabe nur die Taste Return drücken, so bleibt die angezeigte Uhrzeit unverändert. Beachten Sie hierzu auch den Befehl **TIME** im Kapitel 11. Die Uhrzeit in der Anzeige kann bei Ihnen anders lauten (hier: 7 Uhr 27 Minuten 40 Sekunden 15 Hundertstel).

Wenn Sie sich bei dem Datum oder der Uhrzeit vertippen (vorausgesetzt, Sie haben noch nicht Return gedrückt), so können Sie mit der Taste Back Space wieder rückwärts löschen und die Eingabe korrekt vornehmen.

Zum Schluß erscheint am Bildschirm:

```
MS-DOS Version 3.10
Copyright(C) 1981,82,83,84,85 Microsoft Corp.
```

A>

bei IBM:

```
The IBM Personal Computer DOS
Version 3.10 (C)Copyright International Business Machines Corp 1981, 1985
              (C)Copyright Microsoft Corp 1981, 1985
```

A>

Die Angabe A> ist der MS-DOS-Prompt, die System-Bereitschaftsmeldung. Damit erwartet das Betriebssystem von Ihnen Befehle.

Wenn Sie Datum und/oder Uhrzeit nicht richtig eingegeben haben, so korrigieren Sie dies durch Eingabe der Befehle **DATE** und/oder **TIME**.

6.3 Kopien von Original-Disketten

Wie bereits erwähnt, sollten Sie niemals mit Original-Programm-Disketten arbeiten. Für die Erstellung von Kopien ist eine unterschiedliche Handhabung bei Disketten- und Festplatten-Systemen erforderlich.

6.3.1 Kopieren auf Disketten-Systemen

Legen Sie Ihre MS-DOS-Diskette in Laufwerk A ein. Zum Kopieren benötigen Sie den Befehl DISKCOPY (siehe auch Kapitel 11):

```
A>diskcopy a: b:
```

Nach dem Befehl und **a:** betätigen Sie die Leertaste (Leerschritt) und nach **b:** die Return-Taste. Damit wird der Befehl von der MS-DOS-Diskette gelesen und in den Speicher (RAM) geladen, um diesen dort abzuarbeiten. Sie werden vom Programm aufgefordert, das Original in Laufwerk A: und eine neue Diskette in Laufwerk B: einzulegen und Return zu drücken. Damit wird der Kopiervorgang gestartet. Die neue Diskette wird formatiert und anschließend eine 1:1-Kopie vom Original erstellt.

Wenn der Kopiervorgang beendet ist, werden Sie noch gefragt, ob Sie weitere Disketten kopieren wollen. Drücken Sie bitte **y** (YES = JA) und Return, wenn Sie dies wünschen (dazu eine weitere Original-Diskette in Laufwerk A: und eine neue in B: einlegen), im anderen Falle **n** und Return.

Bitte verwenden Sie nur Kopien von MS-DOS und anderen Programmen für die Arbeit am Computer. Empfehlenswert ist es, noch eine zweite Kopie zu erstellen. Dies ist erforderlich, da Disketten nicht "ewig" halten, sondern einer Abnutzung unterliegen, und im Falle von Zeitmangel oder im Interesse eines reibungslosen Ablaufs dann sofort weitergearbeitet werden kann. Sie können dann später problemlos weitere Kopien vom Original erstellen.

6.3.2 Kopieren auf Festplatten-Systemen bzw. Computern mit einem Laufwerk

Da Ihnen in der Regel bei einem Festplatten -System nur ein Disketten-Laufwerk zur Verfügung steht, kann der Kopiervorgang nicht von Laufwerk zu Laufwerk erfolgen. Die meisten Hersteller bieten mit DISKCOPY ein Kopierprogramm an, welches in der Lage ist, eine Diskette mit nur einem Laufwerk zu kopieren (siehe 6.3.2.1). Sollte dies nicht möglich sein, so muß die Festplatte eingerichtet und das MS-DOS auf die Platte kopiert werden (6.3.2.2 und 6.3.2.3).

6.3.2.1 Kopieren von Disketten mit nur einem Laufwerk

Legen Sie Ihre MS-DOS-Diskette in Laufwerk A ein. Zum Kopieren benötigen Sie den Befehl DISKCOPY (siehe auch Kapitel 11):

```
A>diskcopy  
A>diskcopy a: b: (bei Zenith)
```

Nach dem Befehl und **a:** die Leertaste (Leerschritt) und nach **b:** Return-Taste betätigen (im ersten Fall nur Return nach dem Befehl). Damit wird der Befehl von der MS-DOS-Diskette gelesen und in den Speicher (RAM) geladen, um diesen dort abzuarbeiten.

Sie werden vom Programm jeweils aufgefordert, das Original (Source) oder die neue Diskette (target oder destination) in das Laufwerk einzulegen und eine beliebige Taste zu betätigen. Als erstes wird die neue Diskette formatiert. Anschließend folgt der Kopiervorgang. Sie werden wechselweise aufgefordert, die Disketten einzulegen (es wird immer ein Teil von der Original-Diskette in den Speicher gelesen und nach dem Diskettenwechsel auf die neue geschrieben).

Wenn der Kopiervorgang beendet ist, werden Sie noch befragt, ob weitere Disketten kopiert werden sollen. Drücken Sie bitte **y** und Return, wenn Sie weitere kopieren möchten, im anderen Falle **n** und Return.

Bitte verwenden Sie nur Kopien von MS-DOS und anderen Programmen für die Arbeit am Computer. Empfehlenswert ist es, noch eine zweite Kopie zu erstellen. Dies ist erforderlich, da Disketten nicht "ewig" halten, sondern einer Abnutzung unterliegen, und im Falle von Zeitmangel oder im Interesse eines reibungslosen Ablauf dann sofort weitergearbeitet werden kann. Sie können dann später problemlos weitere Kopien vom Original erstellen.

6.3.2.2 Kopieren von MS-DOS auf eine neue Festplatte (IBM, usw.)

Bevor Sie die Platte verwenden können, muß diese eingerichtet werden. Beachten Sie auch das Kapitel 7 "Diskette und Festplatte".

Legen Sie die MS-DOS-Diskette ein, und rufen Sie das Programm **FDISK** auf (Eingabe **fdisk** und Return). Anschließend erhalten Sie eine sogenannte Programmauswahl am Bildschirm. Wählen Sie die 1. Position, indem Sie 1 und Return eingeben. Sie werden anschließend befragt, ob die gesamte Platte für MS-DOS verwendet werden soll. Geben Sie **y** und Return ein, um die gesamte Platte für MS-DOS einzurichten. Nachdem die MS-DOS-Diskette bereits eingelegt ist, können Sie bei der folgenden Meldung ebenfalls Return betätigen.

Der Computer startet erneut das MS-DOS von der Diskette. Tippen Sie den Befehl **FDISK** ein. Wählen Sie jetzt die Position 2 aus. Ihnen werden am Bildschirm die einzelnen Bereiche der Festplatte angezeigt. Geben Sie die Ziffer 1 ein, um den MS-DOS-Bereich aktiv zu setzen. Damit ist es möglich, beim Einschalten des Computers MS-DOS automatisch von der Festplatte zu laden, ohne dabei eine MS-DOS-Diskette einzulegen.

Betätigen Sie die Taste **Esc** zweimal, um das Programm **FDISK** zu verlassen. Jetzt muß die Festplatte noch formatiert werden, damit Sie diese benutzen können. Geben Sie dazu ein:

format c:/s/v

Return-Taste

Bitte beachten Sie, daß bei der Einrichtung der Festplatte mit **FDISK** und **FORMAT** alle eventuell vorher vorhandenen Dateien gelöscht werden. Nach dem Start des Programms **FORMAT** erhalten Sie eine entsprechende Warnung.

Wenn Sie das Programm abbrechen wollen, betätigen Sie **n** und Return. Für die Ausführung die Tasten **y** und Return drücken. Der Formatierungsvorgang kann je nach Größe der Festplatte mehrere Minuten dauern. Anschließend können Sie noch einen Namen mit maximal 11 Stellen zuweisen, oder Sie betätigen nur die Taste Return. Am Bildschirm erscheint nach dem Formatieren und der Übertragung des MS-DOS:

xxxxxxx bytes total disk space
xxxxx bytes used by system
xxxxxxx bytes available on disk

Totalkapazität
durch MS-DOS belegter Platz
freie Kapazität

Nach Beendigung des Programms wird der Prompt **A>** angezeigt.

Beim Formatieren wurde das MS-DOS-System durch die Angabe von **/S** (**IO.SYS**, **MSDOS.SYS** und **COMMAND.COM** bzw. bei IBM: **IBMBIO.COM**, **IBMDOS.COM** und **COMMAND.COM**) übertragen. Die Dienstprogramme können mit folgendem Befehl auf die Festplatte kopiert werden:

copy a:*. * c:/v

Return-Taste

Wenn wieder **A>** erscheint, haben Sie eine komplette Kopie von MS-DOS auf der Festplatte, und der Vorgang ist abgeschlossen.

Um die Richtigkeit der Kopie zu überprüfen, können Sie wie folgt vorgehen. Damit Ihr Computer von der Festplatte starten kann, entnehmen Sie bitte die Diskette (im anderen Falle versucht das System MS-DOS von der Diskette zu starten). Betätigen Sie **Ctrl/Alt/Del** gleichzeitig, um das System neu zu starten.

Geben Sie Datum und Uhrzeit ein, und tippen Sie den Befehl für die Anzeige des **Directorys** (Inhaltsverzeichnis der Festplatte) ein:

dir

Return-Taste

Überprüfen Sie anhand der Anzeige, ob alle Dateien vorhanden sind (vergleichen Sie diese mit den Befehlen in Ihrem Handbuch).

Beachten Sie bitte hierzu auch die Befehle **FDISK**, **FORMAT** und **COPY** im Kapitel 11.

6.3.2.3 Kopieren von MS-DOS auf eine neue Festplatte (für Zenith-Computer)

Bevor Sie die Festplatte verwenden können, muß diese eingerichtet werden. Beachten Sie auch das Kapitel 7 "Diskette und Festplatte".

Legen Sie nach dem Starten Ihres Computers (mit der MS-DOS-Diskette Nr. 1) die Diskette Nr. 2 ein. Rufen Sie das Programm **PART** auf (Eingabe part und Return). Sie erhalten am Bildschirm Hinweise zum Programm - drücken Sie die Tasten y und anschließend F1.

Wenn ein Bereich mit der gesamten Kapazität eingerichtet werden soll, so richten Sie nur mit **F1** das Boot-Laufwerk (Partition/Bereich 1) ein. Mit **F2** können Sie die Festplatte für MS-DOS einteilen (maximal 4 Bereiche mit Größen nach Ihrer Wahl).

Das Boot-Laufwerk wird mit F1 festgelegt. Geben Sie den Bereich an (normalerweise 1), von welchem beim Start das MS-DOS geladen werden kann (ohne dabei eine MS-DOS-Diskette verwenden zu müssen). Beenden Sie mit der Taste E und speichern Sie durch Betätigung der Taste M. Anschließend starten Sie das System erneut von der Diskette Nr. 1.

Jetzt muß die Platte noch formatiert werden, um diese benutzen zu können. Geben Sie dazu ein:

`format c:/s/v`

Return-Taste

Wenn Sie jetzt das Programm noch abbrechen wollen, betätigen Sie nur **Ctrl** und **Break** gleichzeitig.

Den Formatierungsvorgang starten Sie durch Betätigung von Return. Anschließend können Sie der Festplatte noch einen Namen mit maximal 11 Stellen zuweisen, oder Sie betätigen nur die Taste Return. Am Bildschirm erscheint wieder der Prompt A>.

Beim Formatieren wurde bereits das MS-DOS-System (IO.SYS, MSDOS.SYS und COMMAND.COM) übertragen. Um die Dienstprogramme auf die Festplatte zu kopieren, geben Sie bitte folgenden Befehl ein:

`copy a:*. * c:/v`

Return-Taste

Wenn wieder A> erscheint, haben Sie eine komplette Kopie von MS-DOS auf der Festplatte, und der Vorgang ist abgeschlossen.

Die maximal weiteren 3 Bereiche anschließend ebenfalls formatieren. Bevor die weiteren Bereiche mit MS-DOS angesprochen werden können, sind folgende Befehle notwendig, und zwar nach jedem Computer-Start:

`asgnpart 0:2 d:`

`asgnpart 0:3 e:`

`asgnpart 0:4 f:`

Um die Richtigkeit der Kopie zu überprüfen, entnehmen Sie die Diskette aus Laufwerk A und drücken **Ctrl/Alt/Del** gleichzeitig, um das System neu zu starten.

Geben Sie Datum und Uhrzeit, ein und tippen Sie den Befehl für die Anzeige des Directory (Inhaltsverzeichnis der Festplatte) ein:

dir

Return-Taste

Überprüfen Sie anhand der Anzeige, ob alle Dateien vorhanden sind (vergleichen Sie diese mit den Befehlen in Ihrem Handbuch).

Beachten Sie bitte hierzu auch die Befehle PREP, PART, ASGNPART, FORMAT und COPY im Kapitel 11.

6.4 Die wichtigsten Befehle im Überblick

6.4.1 Allgemeines (Aufbau der Dateinamen)

Hier stelle ich Ihnen als erstes den Aufbau der Datei/Programm-Namen vor. Ferner die Befehle DIR, MODE, COPY, RENAME, DEL.

Wenn Sie den Prompt (z.B. A>) am Bildschirm mit dem Cursor daneben sehen, erwartet MS-DOS die Eingabe eines Befehls. Diese können Sie in Klein- oder Großbuchstaben eingeben (Kleinbuchstaben behandelt das MS-DOS nach der Eingabe automatisch wie Großbuchstaben - ohne daß dies angezeigt wird).

Wenn Sie sich ansehen wollen, was auf Ihrer Diskette oder Platte an Programmen vorhanden ist, so geben Sie den Befehl **DIR** und Return ein. Dabei erscheint z.B. am Bildschirm:

Volume in drive B is MS-DOS 3-1

Directory of B:\

KAPITEL 11A	32128	31.07.85	22:39
KAPITEL 0	2432	2.10.85	19:52
KAPITEL 1	3328	2.10.85	19:53
INHALT	1408	2.10.85	19:52
BUCH	1152	2.10.85	19:50
IMPRESSU	640	2.10.85	19:51
KAPITEL 2	5632	2.10.85	21:45
KAPITEL 11B	36480	31.07.85	22:25
KAPITEL 11C	18560	22.07.85	15:40
KAPITEL 3	2304	2.10.85	22:00
KAPITEL 4	7680	2.10.85	22:54
KAPITEL 5	8064	6.10.85	16:58
KAPITEL 7	128	15.09.85	22:09
MSDOS3	36992	9.10.85	20:59
KAPITEL 6B	7936	16.09.85	7:44
KAPITEL 6A	27776	15.09.85	16:24
ELITE	3	18.09.85	20:49
MSDOS3 BAK	36480	7.10.85	22:35
19 File(s)	120832	bytes free	

Sie finden auf der linken Seite der Anzeige eine Reihe von Datei-Namen. Diese bestehen aus 2 Teilen, dem File-Namen (maximal 8 Stellen) und File-Typ (maximal 3 Stellen - dieser muß nicht vorhanden sein). Der Namenszusatz (File-Typ) gibt meist Auskunft über den Inhalt einer Datei. Es haben sich dabei die folgenden wichtigsten Typen-Bezeichnungen ergeben (diese sind nicht alle zwingend, jedoch gebräuchlich):

Typ Verwendung für

COM	Programm
EXE	Programm
BAT	Arbeits-Datei, welche eine Reihe von Programmaufrufen- beinhaltet (siehe Kapitel 12)
ASM	Assembler-Quellen-Programme
COB	Cobol-Quellen-Programme
BAS	Basic-Quellen-Programme
FOR	Fortran-Quellen-Programme
LST	Liste/Protokoll einer Programmübersetzung
OBJ	Objektdatei; erzeugt durch einen Compiler
OVr	Overlay-Datei
TXT	Texte (z.B. Briefe, Berichte) oder
DOC	Texte
BAK	Sicherungsdatei
BIN	Binärdatei
DAT	ASCII-Datei (Daten wie z.B. Adressen, Artikel)
DIF	Datenübergabe-Datei (und Datei des Programmes FC)
SYS	System-Datei
DVD	MS-DOS Geräte-Treiber
TMP	Temporäre Datei des Programmes LINK
\$\$\$	Arbeitsdatei (Temporäre Datei)

Der Datei-Name (8 Stellen maximal für File-Namen und 3 für den Typ), kann aus beliebigen Buchstaben und/oder Ziffern bestehen. Sonderzeichen (Leerzeichen, Punkt, usw.) sollten grundsätzlich nicht verwendet werden.

Erlaubte Zeichen:

A-Z	0-9	\$	&	#	%	'	(	)
-	@	^	{	}	~	`	!	_

Nicht erlaubte Zeichen:

?	.	,	;	:	=	*	/	\
+	"	<	>					

Bei Angabe des Datei-Namens wird der File-Name vom File-Typ durch einen Punkt getrennt:

```
TEXT.TXT
PROGRAMM.COM
PROGRAMM.EXE
BASIC.BAS
10-10-85.TXT
```

Wichtig ist, daß die Datei-Namen in einem Directory nicht gleich sein dürfen, damit man Texte, Programme usw. vom Namen her unterscheiden kann.

Zusätzlich hat jedes Disketten-/Platten-Laufwerk unter MS-DOS ebenfalls einen Namen:

Anzahl der Diskettenlaufwerke	Disketten- Laufwerk-Namen	Festplatte- Laufwerk-Namen	Festplatte- Laufwerk-Namen (nur Zenith)
1 oder 2	A (und B)	C	C, D, E und F
3	A, B und C	D	D, E, F und G
4	A, B, C und D	E	E, F, G und H

Wenn ein Programm oder eine Datei nicht im aktuellen Laufwerk (steht A> am Bildschirm, so ist das aktuelle Laufwerk A) zu finden ist, so muß vor dem Datei-Namen der Laufwerksname mit einem zusätzlichen Doppelpunkt (:) angegeben werden:

```
C:PROGRAMM
B:PROGRAMM
```

Wie an den obigen Beispielen erkennbar, werden Programme immer ohne Angabe des File-Typs zum Starten angegeben.

Wie kann das aktuelle Laufwerk gewechselt werden?

Tippen Sie den Namen des neuen aktuellen Laufwerks (z.B. B:) ein, und betätigen Sie die Taste Return. Anschließend erscheint als Bestätigung ein neuer Prompt (z.B. B:).

Sollte ein von Ihnen angegebener Datei-Name von MS-DOS nicht gefunden werden, so erscheint am Bildschirm die Meldung:

Bad command or file name

Überprüfen Sie in diesem Fall, ob Sie den Namen richtig eingetippt haben und die Datei auf dem aktuellen Laufwerk oder ggfs. einem anderen vorhanden ist.

Beachten Sie auch das Kapitel 8 "Das MS-DOS-Inhaltsverzeichnis".

6.4.2 Zeichen zur Zusammenfassung mehrerer Dateien

Bisher haben Sie sich ausschließlich mit genau spezifizierten Datei-Namen befaßt. Manchmal ist es jedoch erforderlich, Gruppen von Dateien anzusprechen (z.B. wenn alle BASIC-Programme oder Texte kopiert werden). Dazu benötigen Sie sogenannte Wildcard-Zeichen (* und ?).

Auf einer Diskette in Laufwerk A: sind die Dateien

```
TEST1DAT.TXT
TEST2DAT.TXT
TESTADAT.TXT
TESTNEU1.TXT
TESTNEU2.TXT
TESTNEUE.DOC
PROGRAMM.COM
```

gespeichert. Sie kopieren alle 5 TXT-Dateien von Laufwerk A: nach B:

copy a:test1dat.txt b:/v	Return-Taste
copy a:test2dat.txt b:/v	Return-Taste
copy a:testadat.txt b:/v	Return-Taste
copy a:testneu1.txt b:/v	Return-Taste
copy a:testneu2.txt b:/v	Return-Taste

oder

copy a:*.txt b:/v	Return-Taste
-------------------	--------------

oder

copy a:test*.txt b:/v	Return-Taste
-----------------------	--------------

Ein Stern kann für mehrere Zeichen benutzt werden. Er kann für ein oder aber auch für alle Zeichen des File-Namens und -Typs benutzt werden. Ab der Stelle, an welcher er eingegeben wird, ersetzt er alle Zeichen.

Im vorhergehenden zweiten Beispiel (*.txt) steht der * für beliebige 8 Stellen vor dem Punkt.

Werden vor dem Stern Namensteile aufgeführt (text*.txt), so werden diese mit den entsprechenden Stellen der gespeicherten Datei-Namen verglichen und alle weiteren (bis einschl. der 8. Stelle) können beliebig sein. Da der Stern ab der Stelle, an der er eingesetzt wird, alle Zeichen ersetzt, ist eine Eingabe wie *dat.txt nicht erlaubt (MS-DOS akzeptiert die Eingabe, behandelt dies jedoch wie *.txt).

Das gleiche gilt für den File-Typ. Wenn Sie

copy a:test*.* b:/v

Return-Taste

eingeben, so wird jede Datei kopiert, welche mit den Buchstaben test beginnt (also zusätzlich zu obigen Dateien auch noch TESTNEUE.DOC). Der File-Typ kann beliebig sein.

Mit *.* werden alle Dateien angesprochen.

Die Eingabe des Kopierbefehls

copy a:test*.* b:/v

Return-Taste

führt im vorliegenden Beispiel zu der Meldung 0 file(s) copied, da ein File-Name "TEST" nicht existiert. Richtig müsste der Befehl lauten:

copy a:test*.* b:/v

Return-Taste

Das Fragezeichen (?) kann für einen einzelnen Buchstaben im Dateinamen stehen:

copy a:test?dat.txt b:/v

Return-Taste

Es werden alle gespeicherten Dateien kopiert, die bis auf den fünften Buchstaben des Dateinamens mit der Angabe des Namens im Kopier-Befehl übereinstimmen:

TEST1DAT.TXT
TEST2DAT.TXT
TESTADAT.TXT

Das Fragezeichen darf an jeder beliebigen Stelle des File-Namens und File-Typs benutzt werden.

Die beiden Kopierbefehle

copy a:test????.txt b:/v Return-Taste

copy a:test*.txt b:/v Return-Taste

führen somit im vorliegenden Beispiel zum gleichen Ergebnis.

6.4.3 Befehle DIR, COPY, RENAME und DEL (ERASE)

Die Angaben im Kapitel 6.4.2 zur Zusammenfassung von Dateien gelten auch für den Befehl **DIR**. Die folgenden beiden Beispiele führen zum gleichen Resultat:

dir a: Return-Taste

dir a:*,* Return-Taste

Als Ergebnis des Befehles **DIR** wird am Bildschirm ein Directory (hier von der Diskette) angezeigt:

```
Volume in drive A is MS-DOS 3.1D
Directory of A:\

IO          SYS      21357  20.05.85  11.35
MSDOS      SYS      27776  19.05.85  14.42
COMMAND   COM       20703   1.01.80   0.04
COMPILER  TXT        2048  26.08.85  15.40
          4 File(s)  288768 bytes free
```

In der ersten Zeile finden Sie den Namen der Diskette/Festplatte, soweit vorhanden (hier: **MS-DOS 3.1D**). Dieser wird beim Formatieren eingegeben oder kann nachträglich mit dem Befehl **LABEL** verändert werden (siehe Kapitel 11, Befehle **FORMAT** und **LABEL**). Sollte kein Name vorhanden sein, so wird der Text "no label" angezeigt.

In der zweiten Zeile finden Sie den Namen des Laufwerks (in unserem Beispiel A:) und des Directory (Näheres siehe Kapitel 8 "Das MS-DOS-Inhaltsverzeichnis"). Die Anzeige \ bedeutet das erste (ROOT) Directory.

Am Ende erhalten Sie noch die Daten, wie viele Dateien angezeigt wurden (nicht unbedingt die Anzahl der im Directory gespeicherten) und wieviel freie Speicherkapazität noch auf der Diskette/Festplatte ist (in unserem Beispiel sind auf der Diskette noch 288768 Zeichen frei für das Speichern von weiteren Daten/Dateien).

In den Zeilen dazwischen finden Sie die Datei-Einträge.

```

COMMAND COM  20703  16.04.85  16:09      (deutsche Form)

COMMAND COM  20703  4-16-85  4:09p      (US-Form)

1234567890123456789012345678901234567890  (Spaltenangaben)
      10      20      30      40
    
```

```

COMMAND      Dateiname
COM          Dateityp
20703        Dateigroesse (in Bytes bzw. Zeichen)
16.04.85     Datum der Erstellung bzw. letzten Aenderung
16.09        Uhrzeit der Erstellung bzw. letzten Aenderung
4:09p        4 Uhr 9 Minuten nach Mittag (a = vor Mittag)
    
```

Die Spalten werden z.B. fuer folgenden Befehl benoetigt:

DIR | SORT /+14

Es wird das Inhaltsverzeichnis sortiert ab der Spalte 14
angezeigt (also nach Dateigroesse aufsteigend anzeigen).

Bild 6.3 Zeilenaufbau im MS-DOS-Inhaltsverzeichnis

Wenn für die Anzeige eines Directory die Zeilen am Bildschirm nicht ausreichend sind (mehr als 25 Zeilen), so können Sie einen der beiden folgenden Befehle ("seitenweise vorwärts blättern") verwenden:

```
dir *.* | more
```

oder

```
dir *.* /p
```

Den Befehl **COPY** haben Sie bereits im Kapitel 6.4.2 kennengelernt. Weitere Informationen finden Sie im Kapitel 6.5 (Kopieren von Texten auf den Drucker) sowie eine detaillierte Beschreibung im Kapitel 11.

Als erstes wird bei COPY "woher" und als zweites "wohin" angegeben:

```
copy          autoexec.bat  exec.bat
```

```
kopiere      woher         wohin
```

Im vorliegenden Beispiel sucht das Programm COPY die Datei AUTOEXEC.BAT und erstellt davon eine Kopie, welche unter dem Namen EXEC.BAT gespeichert wird. Nach Ausführung des Befehls existiert von AUTOEXEC.BAT eine Kopie gleichen Inhalts mit dem Namen EXEC.BAT. Wenn Sie am Ende des Befehls noch /v angeben, so erfolgt nach dem Kopiervorgang eine Überprüfung, ob dies fehlerfrei durchgeführt wurde.

Normalerweise wird von einem Laufwerk auf ein anderes kopiert. Die Laufwerks-Bezeichnung müssen Sie nur angeben, wenn diese vom angezeigten Prompt (z.B. A> für aktuelles Laufwerk A) abweicht. Diese Angabe ist immer vor dem Dateinamen (ohne Zwischenraum) erforderlich.

Erhält "wohin" den gleichen Namen wie "woher", so kann auf eine zweite Dateiangabe verzichtet werden.

Beispiele zum Kopierbefehl:

```
copy c:autoexec.bat a:exec.bat
```

AUTOEXEC.BAT von Laufwerk C nach A kopieren - neuer Name EXEC.BAT

```
copy autoexec.bat a:
```

AUTOEXEC.BAT vom aktuellen nach Laufwerk A mit dem gleichen Namen kopieren

```
copy b:keybgr.com
```

KEYBGR.COM vom Laufwerk B ins aktuelle mit dem gleichen Namen kopieren ("wohin" muß hier nicht angegeben werden).

Oft soll keine Kopie erstellt werden, sondern eine Datei einen anderen Namen erhalten. Hierzu verwenden Sie statt COPY den Befehl **RENAME**. Dieser wird wie COPY angewandt. Die Angabe von /v ist nicht möglich bzw. notwendig, da nur der Dateiname verändert wird:

```
rename autoexec.bat exec.bat
```

Im folgenden noch 4 Beispiele, wie in einer möglichst kurzen Form eine oder mehrere Dateien umbenannt werden können:

```
ren b:ansi.doc *.txt
```

Im Laufwerk B die Datei ANSI.DOC in ANSI.TXT umbenennen. (Wenn der File-Name oder File-Typ gleich bleiben soll, so kann dieser mit * abgekürzt werden).

```
ren b:*.doc *.txt
```

Der File-Typ DOC wird in TXT geändert (bei allen gefundenen Dateien auf der Diskette in Laufwerk B)

```
ren b:ansi bild
```

Die Datei ANSI in BILD umbenennen (Diskette in Laufwerk B)

```
ren programm.* neuprog.*
```

Alle Dateien mit dem File-Namen PROGRAMM in NEUPROG ändern
(der File-Typ bleibt unverändert)

Wie ersichtlich, kann RENAME auch in einer Kurzform REN verwendet werden. Wird nur bei der ersten Dateibezeichnung eine Laufwerksbezeichnung angegeben, so gilt diese auch für die zweite Angabe.

Natürlich benötigen Sie auch einen Befehl zum Löschen von Dateien. Mit **DEL** oder **ERASE** (beide Befehle sind in der Bedienung und Auswirkung gleich), können Dateien gelöscht werden:

```
del b:*,*
```

Alle Dateien im Directory auf der Diskette in Laufwerk B werden gelöscht (wenn diese nicht mit dem Programm ATTRIB geschützt sind). Vor dem Löschen werden Sie noch befragt:

```
Are you sure (Y/N)?
```

(sind Sie sicher?). Die Eingabe Y löscht alles.

```
del b:exec.bat
```

Auf der Diskette in Laufwerk B wird EXEC.BAT gelöscht

```
del *.doc
```

Alle Dateien mit dem File-Typ DOC im aktuellen Laufwerk löschen

```
del neuprog.*
```

Alle Dateien mit dem File-Namen NEUPROG im aktuellen Laufwerk löschen

```
del a:test?dat.txt
```

Alle Dateien wie angegeben werden gelöscht, wobei an der 5. Stelle ein beliebiges Zeichen stehen kann (Laufwerk A).

Wichtige Fehlermeldungen

MS-DOS Fehlermeldung	Beschreibung
Bad command or file name	Befehl nicht vorhanden oder falsch angegebener Dateiname nicht vorhanden oder falsch
Duplicate file name	neuer Dateiname bereits vorhanden (RENAME)
File not found	Dateiname nicht gefunden
File cannot be copied onto itself 0 File(s) copied	beim Kopieren haben Sie 2 gleiche Datei-Namen in einem Befehl angegeben
Insufficient disk space	auf der Diskette/Platte ist kein Platz mehr vorhanden
Invalid device	angegebener Gerätenamen nicht möglich
Invalid drive specification	angegebenes Laufwerk nicht vorhanden

6.4.4 Befehl MODE für die Anpassung des Druckers, usw.

Mit dem Befehl kann folgendes eingestellt werden:

1. Einstellung des "Weges" zum Drucker
2. Einstellung der Video(Bildschirm)-Karte (Grafik, usw.)
3. Einstellung der Schnittstelle für die Übertragung von Daten zu anderen Computern (über Akustikkoppler, Postmodem, usw.)

Wir wollen uns hier nur mit dem Drucker befassen. Weitere Erklärungen zum Befehl MODE finden Sie im Kapitel 11.

Als erstes klären Sie bitte ab, ob Ihr Drucker an der parallelen (Normalfall) oder seriellen Schnittstelle angeschlossen ist. Wenn Ihr Drucker an der seriellen Schnittstelle angeschlossen ist, so können Sie die standardmäßige Druckausgabe von der parallelen dorthin umleiten:

```
mode lpt1=com1
```

Es können bis zu 3 parallele Schnittstellen an Ihrem Computer vorhanden sein, welche MS-DOS benutzt. Diese bezeichnet man als LPT1, LPT2 und LPT3. Mit MODE können diese eingestellt werden:

1. Drucker mit 80 Zeichen pro Zeile
Drucker mit 132 Zeichen pro Zeile
2. Druck mit 6 Zeilen pro Inch (1 Inch = 2,54 cm)
Druck mit 8 Zeilen pro Inch
3. Normalerweise erscheint eine Fehlermeldung, wenn etwas gedruckt werden soll und dies nicht erfolgen kann (z.B. weil der Drucker nicht eingeschaltet ist). Dies kann abgeschaltet werden, indem der Buchstabe P beim Befehl MODE mitangegeben wird (dann wartet der Computer so lange, bis gedruckt werden kann).

Die Standardeinstellung wurde oben entsprechend gekennzeichnet (80 Zeichen, 6 Zeilen).

Bitte beachten Sie, daß verschiedene Programmiersprachen (z. B. GW-BASIC, BASICA) Ihre Einstellungen mit MODE überschreiben (also unwirksam machen). Es muß also mit der Programmiersprache die Einstellung erneut vorgenommen werden.

Beispiele:

```
mode lpt1:132,8,P
```

- 1.Parallel-Schnittstelle
132 Zeichen pro Zeile
8 Zeilen pro Zollwarten bis Drucker "empfangsbereit"

`mode lpt1:80,8`

1. Parallel-Schnittstelle

80 Zeichen pro Zeile

8 Zeilen pro Zoll

Fehlermeldung, wenn Drucker nicht "empfangsbereit"

`mode lpt1:80,6,P`

1. Parallel-Schnittstelle

80 Zeichen pro Zeile

6 Zeilen pro Zoll

warten bis Drucker "empfangsbereit"

`mode lpt2:132,6,P`

2. Parallel-Schnittstelle

132 Zeichen pro Zeile

6 Zeilen pro Zoll

warten bis Drucker "empfangsbereit"

6.5 Drucken von Texten und vom Bildschirm

Sie können alles, was am Bildschirm angezeigt wird, ausdrucken (dies ist auch möglich, wenn Sie gerade ein Anwendungsprogramm, wie z.B. die Finanzbuchhaltung, bearbeiten). Betätigen Sie dazu nur die Tasten Shift und gleichzeitig PrtSc. Damit wird alles 1 zu 1 auf das Papier im Drucker "kopiert". Jedes Zeichen kann jedoch nur dargestellt werden, wenn Sie einen kompatiblen Drucker mit 8-Bit-Zeichensatz verwenden.

Durch Ctrl und gleichzeitig PrtSc (oder Ctrl P) wird der Drucker "zugeschaltet". Alles was eingegeben oder angezeigt wird, erscheint gleichzeitig am Bildschirm und Drucker. Abgeschaltet werden kann diese Funktion durch wiederholte Betätigung der gleichen Tasten.

Diese Funktion ist meist während des Ablaufs von Anwendungsprogrammen gesperrt.

Mit den Befehlen COPY und PRINT können Sie Texte drucken (siehe Kapitel 11). Einfaches Beispiel zum Ausdruck der Datei CONFIG.SYS:

```
copy config.sys prn
```

copy	Kopierbefehl
config.sys	Inhalt dieser Datei soll gedruckt ("ausgegeben") werden
prn	symbolischer Name für den Drucker

Die günstigste Möglichkeit für den Ausdruck von Texten bietet der Befehl PRINT. Wenn mit diesem Befehl dem Computer mitgeteilt wurde, welche Texte gedruckt werden sollen, können gleichzeitig mit dem PC andere Tätigkeiten durchgeführt werden (natürlich ausgenommen Drucken). Dies ist vor allem bei langen Texten hilfreich. Eine komplette Beschreibung zu PRINT finden Sie in Kapitel 11.

```
print autoexec.bat ansi.doc
```

Die beiden Dateien AUTOEXEC.BAT und ANSI.DOC hintereinander drucken

Wenn Sie mit PRINT einen weiteren Text drucken wollen und der Drucker noch beschäftigt ist, so merkt das Programm den neuen Text für den Ausdruck vor.

6.6 Erzeugen von Sicherheitskopien (von Diskette und Platte)

Wenn Sie neue Daten (z.B. Buchungen bei einem Finanzbuchhaltungsprogramm) in den Computer eintippen, sollte täglich vor Arbeitsende eine Kopie der Daten erstellt werden. Dies dient der Sicherheit. Sie benötigen eine Kopie, wenn der Datenbestand auf dem Original verlorengeht, oder nicht mehr bearbeitbar ist. Dies kommt zwar sehr selten vor, aber wenn, dann benötigen Sie eine Kopie, damit nicht alles wieder neu eingetippt werden muß (unter Umständen kann dies Wochen dauern).

Auch wenn es Jahre gut geht - fertigen Sie Sicherheitskopien, die Praxis zeigt, daß dies notwendig ist!

6.6.1 Sicherheitskopien bei Computern mit 2 Laufwerken

Der Vorgang läuft wie in Kapitel 6.3.1 beschrieben ab. Bevor Sie jedoch eine neue Kopie vom Original fertigen, sollten Sie von Ihrer vorherigen Kopie eine Kopie fertigen, so daß Sie immer eine 1. und eine 2. Kopie zusätzlich zum Original haben, falls einmal Ihre aktuelle Kopie defekt sein sollte.

Legen Sie Ihre MS-DOS-Diskette in das erste Laufwerk ein (Laufwerk A), und geben Sie den Befehl

```
diskcopy a: b:
```

ein. Das Programm fordert Sie auf, die Disketten einzulegen. Legen Sie jetzt die zu kopierende Diskette, in Laufwerk A und die Diskette auf welche kopiert werden soll, in Laufwerk B ein und betätigen Sie die Taste Return. Damit wird der Kopiervorgang gestartet.

6.6.2 Sicherheitskopien bei Computern mit einem Disketten-Laufwerk

Legen Sie Ihre MS-DOS-Diskette in das erste Laufwerk ein (Laufwerk A), und geben Sie den Befehl

```
diskcopy
```

```
diskcopy a: b: (bei Zenith)
```

ein. Legen Sie die zu kopierende Diskette in Laufwerk A ein. Sie werden abwechselnd aufgefordert, die Original-Diskette bzw. die Ziel-Diskette einzulegen, bis der Vorgang abgeschlossen ist. Bei einigen Computern erscheint die Meldung in der Form, daß Sie aufgefordert werden, Laufwerk-A- bzw. Laufwerk-B-Diskette einzulegen.

6.6.3 Sicherheitskopien von Daten auf einer Festplatte

Der Sicherungsvorgang erfolgt von einer Winchester auf Disketten (oder wenn vorhanden auf ein Band bzw. Streamer-Tape). Da die zu sichernden Daten meist die Kapazität einer Diskette übersteigen, stehen hierfür zwei spezielle Befehle zur Verfügung:

BACKUP
RESTORE

Sicherungskopie erstellen
Zurückkopieren auf Winchester

Mit BACKUP gesicherte Dateien dürfen auf keinen Fall mit dem COPY-Befehl zurückkopiert werden, sondern nur mit RESTORE.

Hier will ich nur auf den Standard-Fall eingehen. Weitere Möglichkeiten finden Sie im Kapitel 11 unter BACKUP bzw. RESTORE.

Gesichert werden die Daten von der Winchester mit dem Befehl:

```
backup c: a:/s
```

Sichern aller Dateien aus Laufwerk C (aus allen Directories) auf Disketten in Laufwerk A (wenn eine Diskette voll ist, werden Sie automatisch aufgefordert, eine neue einzulegen).

Wenn nur bestimmte Dateien gesichert werden sollen, so muß hinter C: noch die entsprechende Dateiangabe folgen:

```
backup c:*.txt a:
```

Es werden nur die Dateien mit dem File-Typ TXT aus dem aktuellen Inhaltsverzeichnis gesichert.

Die Angabe /S bedeutet, daß alle Directories berücksichtigt werden. Ist diese Angabe nicht vorhanden, so wird nur das aktuelle Directory berücksichtigt (beachten Sie dazu auch das Kapitel 8 "MS-DOS-Inhaltsverzeichnis").

Für das Zurückholen der Dateien von den Disketten benötigen Sie den Befehl RESTORE. Für die beiden obigen Beispiele lauten die Befehle:

```
restore a: c:\*.* /s
```

```
restore a: c:*.txt
```

BACKUP und RESTORE können für folgende Kopierrichtungen verwendet werden:

Platte auf Diskette

Diskette auf Diskette

Diskette auf Platte

Platte auf Platte

7 Diskette und Festplatte

Ein interner Speicher ist in den Computer eingebaut und nennt sich RAM (Random Access Memory). Er wird bei Abschalten des Computers gelöscht. Damit Sie Daten speichern können, ist ein externer Speicher notwendig. Dieser kann eine Diskette und/oder Festplatte sein.

Eine Festplatte besteht, im Gegensatz zur einzelnen Diskette, aus mehreren nicht biegsamen Platten (darum wird diese auch Hard-Disk genannt). Beide Speichermedien arbeiten im Prinzip auf die gleiche Art und Weise.

Damit Sie gezielt auf einen externen Speicher zugreifen können, werden die einzelnen Laufwerke mit Buchstaben bezeichnet. Zusätzlich ist ein Doppelpunkt zur Kennzeichnung notwendig (z.B. B:).

Anzahl der Diskettenlaufwerke	Disketten- Laufwerks-Namen	Festplatte- Laufwerks-Namen
1 oder 2	A (und B)	C
3	A, B und C	D
4	A, B, C und D	E

7.1 Aktuelles Laufwerk

Geben Sie bei einem Befehl keinen Laufwerks-Namen an, so verwendet MS-DOS das aktuelle Laufwerk. Am Bildschirm ist es durch den Prompt erkenntlich. Dieser besteht immer aus dem Namen (z.B. A) und dem Zeichen >.

Sie können das aktuelle Laufwerk durch die Eingabe eines Namens (z.B. B:) jederzeit ändern.

A>b:

Return-Taste

B>

Sie dürfen kein Laufwerk angeben, das nicht vorhanden ist, da Sie sonst folgende Fehlermeldung erhalten:

Invalid drive specification

Bei dem von Ihnen angewählten Laufwerk ist es nicht erforderlich, daß sich darauf MS-DOS befindet, da dieses bereits im internen Speicher ist.

7.2 Einteilung von Disketten und Festplatte

Die Informationen werden in konzentrischen Ringen auf der magnetisierten Oberfläche gespeichert. Diese Ringe nennt man Spuren (Tracks), welche in einzelne Bereiche (Sektoren) unterteilt sind. Die Größe eines Bereichs ist bei den einzelnen Systemen unterschiedlich.

Die Kapazität der einzelnen Bereiche wird in Bytes ausgedrückt. Jedes Byte besteht aus 8 Bit (1 Bit ist die kleinste Informationseinheit in Ihrem Computer). Um einen Buchstaben oder ein anderes Zeichen darstellen zu können, benötigt man 8 Bits (also 1 Byte).

Die Größe eines Speichers wird immer in Kilobytes (kB) oder Megabytes (MB) angegeben. Ein Kilobyte besteht aus 1024 Bytes, ein Megabyte aus 1024 kB (1.048.576 Bytes).

Ein Elektromagnet im Laufwerk liest und schreibt die Informationen. Man nennt ihn Schreib-Lese-Kopf. Die Steuerung erfolgt durch MS-DOS, und er kann an jeder beliebigen Stelle der Diskette/Festplatte positioniert werden. Die genaue Adresse des gewünschten Bereiches erhält das Betriebssystem aus dem Directory (Inhaltsverzeichnis). In diesem Verzeichnis stehen die Dateinamen und die Angaben, wo die dazugehörigen Daten zu finden sind.

MS-DOS speichert die Daten einer Datei nicht in einem Stück zusammenhängend. Um eine möglichst optimale Platzausnutzung zu erreichen, wird die sogenannte dynamische Speicherung angewendet. Es wird der nächste freie Sektor gesucht und dort gespeichert. Dieser Vorgang wird so lange wiederholt, bis alle Daten gespeichert sind. Dabei kann es vorkommen, daß die Daten einer Datei über eine ganze Diskette oder Festplatte verstreut sind.

7.3 Disketten

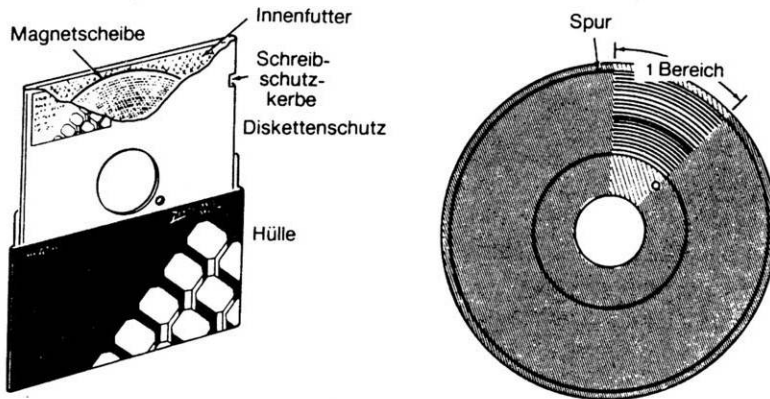


Bild 7.1 Die Diskette

Eine Diskette ist eine dünne, fein magnetisierte Plastikscheibe. Die Scheibe wird in einer Hülle aufbewahrt, die sie fast vollständig bedeckt. Nur ein schmaler Schlitz bleibt für das Lesen und Schreiben offen. Da die Diskette und die darauf gespeicherten Daten sehr leicht zerstört werden können, sollten Sie diese immer sehr sorgfältig behandeln. Bitte beachten Sie vor allem folgendes:

- Berühren Sie niemals die Disketten-Oberfläche (vor allem nicht den Schreib-Lese-Schlitz)
- Wenn Sie die Diskette dem Laufwerk entnehmen, stecken Sie sie sofort wieder in die Verpackung
- Laufwerke erst öffnen, wenn die rote Kontroll-Lampe erloschen ist
- Auf die Diskette dürfen kein Staub (z.B. Zigarettenasche) und keine Flüssigkeiten gelangen
- Disketten dürfen nicht in der Umgebung von magnetischen Feldern aufbewahrt werden
- Nicht mit Kugelschreiber beschriften (möglichst Filzstift)
- Mit Disketten nicht durch Röntgenanlagen gehen
- Keine Disketten im Laufwerk lassen, wenn Sie den Strom ein- oder ausschalten (ausgenommen beim Systemstart). Durch eventuelle "Stromstöße" kann der Schreib-Lese-Kopf auf die Scheibe "stürzen" (Head-Crash)
- Temperaturen von -10 bis +52 Grad
- Disketten niemals biegen oder stapeln

Durch Überkleben der Kerbe mit einem kleinen Streifen (werden bei neuen Disketten mitgeliefert) auf der linken Seite schützen Sie Ihre Daten vor dem Löschen und Überschreiben. Dadurch kann jedoch auch nicht mehr gespeichert werden! Die Diskette ist damit schreibgeschützt (Write protected). Wenn Sie trotzdem versuchen auf der Diskette zu schreiben, so erhalten Sie eine Fehlermeldung:

Write protect
Abort, Retry, Ignore?

Wenn Sie trotzdem speichern wollen, müssen Sie den kleinen Streifen von der Diskette entfernen und **r** betätigen. Im anderen Falle geben Sie den Buchstaben **a** ein, damit der Vorgang abgebrochen wird.

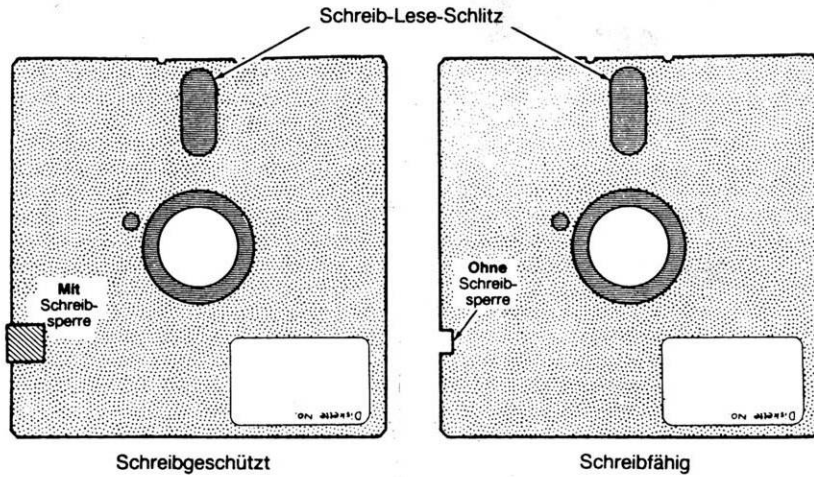


Bild 7.2 Schreibsperr der Diskette

Disketten mit dem Schreib-Lese-Schutz zuerst in das Laufwerk einschieben. Das Bild ist auf der Seite, an der sich die Laufwerk-Klappe befindet. Ist die Diskette ganz eingeschoben, können Sie das Laufwerk schließen.

7.4 Festplatten

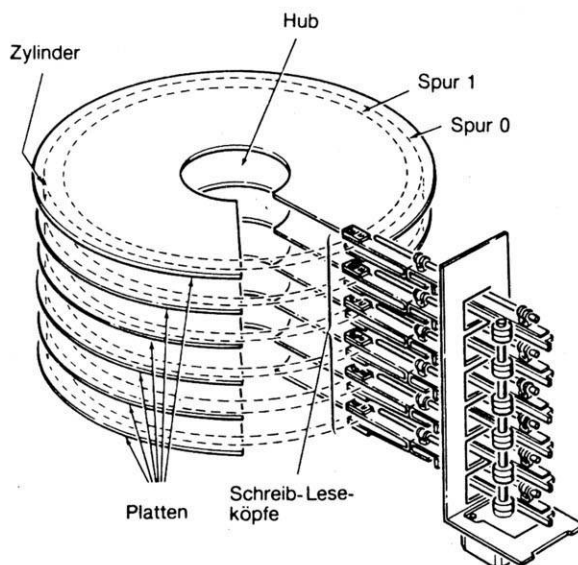


Bild 7.3 Festplattenlaufwerk

Ein Festplattenlaufwerk besteht aus mehreren, nicht biegsamen (hartes Aluminium), mit Metalloxid-Oberfläche versehenen Scheiben. Der externe Speicher ist in einem staubfreien, luftdicht abgeschlossenen Gehäuse untergebracht. Dadurch ist eine Festplatte wesentlich sicherer als eine Diskette.

Die Kapazität einer Festplatte ist unterschiedlich und beginnt bei PCs im allgemeinen bei 10 MB (PC/AT 20 MB), sie ist damit wesentlich höher als die einer Diskette.

Wegen der wesentlich höheren Umdrehungsgeschwindigkeit ist die Zugriffszeit sehr kurz (ca. Faktor 10 schneller als eine Diskette).

Mit MS-DOS können Sie maximal 32 MB pro Laufwerk verwalten. Es gibt Systeme, welche die Möglichkeit bieten, die Festplatte in mehrere Bereiche mit je maximal 32 MB einzuteilen.

Ein Nachteil der Festplatte ist, daß sie nicht wie eine Diskette, leicht transportiert werden kann. Ferner hat sie keinen Schreib-Schutz.

8 Das MS-DOS-Inhaltsverzeichnis

Seit der Version 2.0 von MS-DOS gibt es hierarchische Inhaltsverzeichnisse auf Diskette/Festplatte. Dies ist ein System mit mehreren Ebenen, welche wie ein "Baum" aufgeteilt sind. Eine solche Struktur kann wie folgt aussehen:

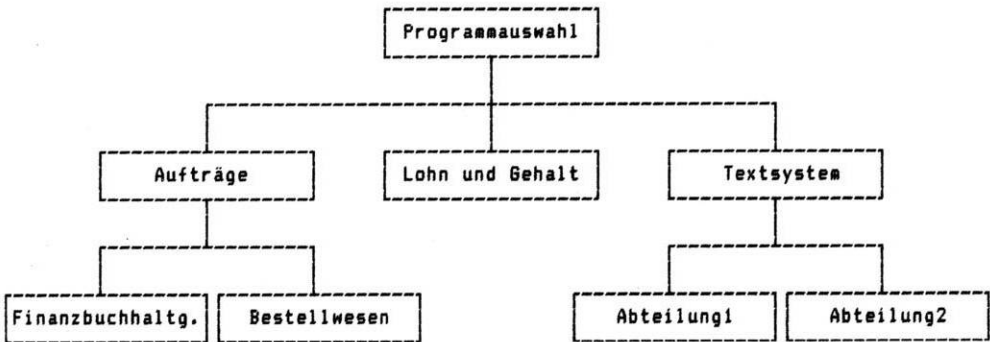


Bild 8.1 Inhaltsverzeichnis mit Sub-Directories

Die Spitze des Systems ist die "Wurzel" (Root-Directory). Dies ist das erste Verzeichnis, welches automatisch durch das Programm FORMAT angelegt wird. Die weiteren Verzeichnisse, welche sich in der Hierarchie darunter befinden, nennt man Sub-Directories. Diese können mit einem speziellen Befehl (MD) angelegt werden.

In der Hierarchie können Sie sich in jeder beliebigen Richtung bewegen.

Alle Dateien, welche mit MS-DOS erstellt werden, erhalten einen Namen. Diese werden zusammen mit weiteren Informationen über die Dateien in einem Directory (Verzeichnis) gespeichert. Von den Informationen sind die Dateigröße, Datum und Uhrzeit der Dateianlage (oder Änderung) sichtbar. Dafür verwenden Sie den Befehl DIR. Er zeigt Ihnen alle Dateien an, die in dem Directory gespeichert sind.

Mit den Sub-Directories erleichtert Ihnen MS-DOS die Verwaltung von verschiedenen Dateien. Sie haben die Möglichkeit, Ihre einzelnen Anwendungen in verschiedenen Sub-Directories anzulegen. Damit erreichen Sie eine erheblich bessere Übersicht. Die Sub-Directories können bei Bedarf so umfangreich wie nötig werden (Begrenzung stellt nur der externe Speicher dar).

Neben den Dateinamen erhalten auch die Sub-Directories Namen (nach Ihrer Wahl). Der Name des Root-Directory ist immer \. Die Sub-Directory-Namen werden wie Dateinamen behandelt. Bei der Anzeige des Inhaltsverzeichnisses sind diese am Eintrag in der Spalte mit der Dateigröße erkennbar (Eintrag: <DIR>). Für das vorhergehende Beispiel kann das Root-Directory wie folgt aussehen (Anzeige mit dem Befehl DIR):

Volume in drive B is FIRMA

Directory of B:\

COMMAND	COM	18128	01.01.85	22:50
VERTEIL	BAT	256	15.10.85	01:46
AUFTRAG	<DIR>		10.09.85	14:34
LUGA	<DIR>		10.09.85	14:35
TEXT	<DIR>		10.09.85	14:37
MSDOS	<DIR>		11.09.85	19:05
6 File(s)			46126 bytes free	

Wie Sie erkennen können, sind im Inhaltsverzeichnis nur die Sub-Directory-Namen der nächsten Ebene sichtbar. Mit dem Befehl TREE kann die gesamte Struktur angezeigt werden.

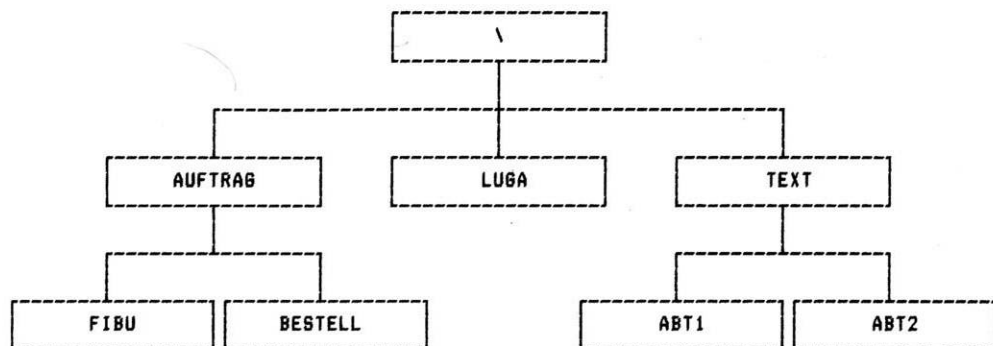


Bild 8.2 Inhaltsverzeichnis mit SUB-Directory-Namen

Nur die Dateien in **einem Directory** müssen unterschiedliche Namen haben, um sie auch als unterschiedliche Dateien zu erkennen.

So wie es ein aktuelles Laufwerk gibt, gibt es auch ein aktuelles Directory. Zwischen den Directories können Sie mit dem Befehl CHDIR (Kurzform CD - CHange DIRectory) beliebig wechseln.

Mit dem Anlegen eines neuen Directory werden automatisch zwei Dateien in diesem angelegt (nicht löscher). Diese beiden Einträge bezeichnen das neue Directory (Dateiname .) und das hierarchisch davor liegende (Dateiname ..). Bei Angabe eines Punktes ist das aktuelle, bei zwei Punkten das davor liegende Directory angesprochen. Damit speichert MS-DOS, in welcher Beziehung die einzelnen Directories zueinander stehen.

Benötigen Sie eine Datei, welche nicht im aktuellen Directory steht, so geben Sie zum Dateinamen den "Weg" (Path) an. Der Weg führt vom aktuellen Directory zu demjenigen, in welchem die gesuchte Datei gespeichert ist. Dazu müssen alle Directory-Namen angegeben werden (als Kennzeichnung steht hinter jedem Directory-Namen das Zeichen \).

Beispiel:

Das aktuelle Directory ist BESTELL. Das gesuchte Programm EDLIN.EXE befindet sich in TEXT. Der Befehl lautet:

```
..\..\text\edlin
```

oder

```
\text\edlin
```

Im ersten Fall geht der Weg rückwärts über AUFTRAG ins Root-Directory und von dort zu TEXT, in welchem sich das Programm EDLIN.EXE befindet.

Beginnen Sie den Weg mit \, so beginnt MS-DOS im Root-Directory, unabhängig davon, in welchem Directory Sie sich befinden (zweites Beispiel).

Die Länge des Weges darf 64 Stellen nicht überschreiten (ausgenommen dem Datei-Namen). Programme, welche unter älteren MS-DOS-Versionen erstellt wurden, können nicht mit den Weg-Angaben arbeiten.

8.1 Befehle zur Bearbeitung von Directories

Mit den folgenden vier Befehlen können Sie die Struktur der Directories verändern:

MKDIR oder MD	Anlegen eines Sub-Directory
CHDIR oder CD	Anzeigen oder Ändern des aktuellen Directory
RMDIR oder RD	Löschen eines Sub-Directory-Eintrags
PATH	bezeichnet den Weg durch die Verzeichnisse bis zu einem Programm (File-Typ COM, EXE oder BAT)

Außer diesen speziellen Befehlen zur Bearbeitung der Struktur sind folgende von wesentlicher Bedeutung:

COPY	Kopieren von Dateien von einem Directory zu einem anderen
DEL	Löschen von Dateien in einem Directory
DIR	Anzeigen des Inhaltsverzeichnisses eines Directory
SEARCH	Suchen von Dateien in den Directories

Die internen Befehle des MS-DOS sind jederzeit abrufbereit, da sie sich bereits im Speicher des Computers befinden. Extern sind Programme, welche beim Aufruf erst in den Speicher von der Diskette oder Winchester geladen werden müssen.

Es bietet sich an, alle Programme des MS-DOS in einem separaten Sub-Directory zu speichern (MS-DOS):

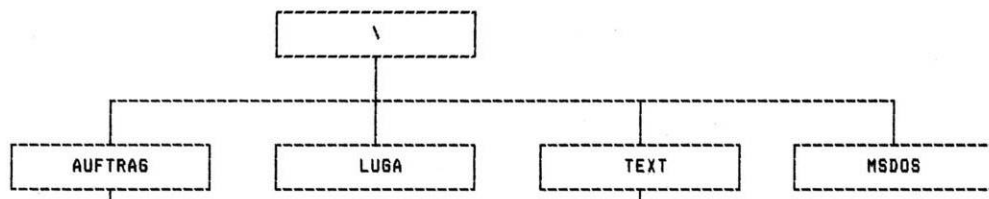


Bild 8.3 Directory-Struktur mit neuem Sub-Directory für MS-DOS

Geben Sie, während Sie sich in jedem beliebigen anderen Directory befinden, einen externen Befehl ein, so wird das aktuelle Directory nach dem Befehl durchsucht. Da dieser nicht gefunden wird, erhalten Sie die Fehlermeldung

Bad command or file name

Sie müssen vor dem Befehlsnamen den Weg angeben (\MSDOS\befehl). Sie können es jedoch vermeiden, bei jedem Aufruf eines externen Befehls den Weg anzugeben. Dazu verwenden Sie nach dem Start des Computers den Befehl

path=\system

Dieser Weg muß nur einmal von Ihnen eingegeben werden. Findet MS-DOS einen eingegebenen Befehl im aktuellen Directory nicht, so sieht es nach, ob mit PATH ein Weg definiert wurde, und verwendet diesen. In dieser Form können auch weitere Wege vorgegeben werden:

path=\system;\text\abt1;\text\abt2;\auftrag\bestell

Durch Eingabe von

path

können Sie sich die derzeit gültigen Weg(e) anzeigen lassen. Bitte beachten Sie, daß mit jeder Neueingabe von Wegen mit PATH die bisher gültigen gelöscht werden (also bei jeder Änderung immer alles notwendige angeben).

Die Vereinbarung mit PATH gilt nur für Programme (File-Typen COM, EXE und BAT). Alle anderen Dateien müssen sich im aktuellen Verzeichnis befinden (oder Sie geben den Weg an). Dies gilt vor allem für Programme, die in mehrere Teile gesplittet sind.

8.2 Anlegen eines Sub-Directory (MKDIR oder MD)

Um ein neues Sub-Directory anzulegen, verwenden Sie den Befehl **MKDIR** oder die Kurzform **MD**:

```
md tabelle
```

Das neue Sub-Directory tabelle wird "unterhalb" des aktuellen angelegt. Dabei werden automatisch die beiden Einträge **.** und **..** im neuen Sub-Directory erstellt:

```
Volume in drive A is MS-DOS 3-1
Directory of A:\TABELLE

.                <DIR>      12.09.85  23:56
..               <DIR>      12.09.85  23:56
2 File(s) 23450 bytes free
```

Ein neues Sub-Directory kann an jeder beliebigen Stelle unter Angabe des Weges erstellt werden:

```
md \auftrag\bestell\abt1
```

Vom Root-Directory aus führt der Weg über **AUFTRAG** nach **BESTELL**. Dort wird das neue Sub-Directory **ABT1** angelegt.

Ein Datei- oder Directory-Name darf in einem Directory nur **einmal** vorkommen, kann jedoch in der gesamten Struktur öfter erscheinen (siehe im vorliegenden Beispiel **ABT1**). Zur besseren Übersicht sollten Sie aber darauf verzichten.

Kann ein neues Sub-Directory nicht angelegt werden, weil der Name schon vorhanden ist, so erhalten Sie die Fehlermeldung

```
Unable to create directory
```

8.3 Anzeigen und Ändern des aktuellen Directory (CHDIR oder CD)

Jedes Directory hat einen Namen. Sie können den Namen des aktuellen Directory durch den Befehl

`cd`

anzeigen lassen. Das aktuelle Directory für ein bestimmtes Laufwerk erhalten Sie mit

`cd b:`

Sie geben also zusätzlich noch den Laufwerks-Namen an.

Beispiele für die Anzeige:

`A:\AUFTRAG\FIBU`

FIBU ist das aktuelle Directory in Laufwerk A

`B:\`

das Root-Directory ist in Laufwerk B aktuell

`A:\TABELLE`

TABELLE ist das aktuelle Directory in Laufwerk A

Das aktuelle Directory kann sehr leicht geändert werden. Sie geben zusätzlich zum Befehl CD noch den Weg an. Beispiele:

`cd \`

Rückkehr zum Root-Directory - von jedem beliebigen Sub-Directory aus

cd text

neues aktuelles Directory ist TEXT (\ darf nicht angegeben werden)

cd \auftrag\fibu

\	Weg vom Root-Directory aus beginnen
AUFTRAG	über AUFTRAG den Weg fortsetzen
FIBU	wird das neue aktuelle Directory

cd ..

vom derzeitigen Directory zum vorherigen wechseln (neue aktuelle ist damit im vorliegenden Beispiel AUFTRAG)

Wenn Sie den Weg nicht richtig eingeben bzw. der angegebene Directory-Name von MS-DOS nicht gefunden wird, erhalten Sie die Fehlermeldung

Invalid directory

8.4 Löschen eines Sub-Directory (RMDIR oder RD)

Das Root-Directory kann nicht gelöscht werden. Für das Löschen von Sub-Directory-Einträgen muß beachtet werden:

1. Im Sub-Directory müssen vorher alle Dateien gelöscht werden (Befehl DEL).
2. Es dürfen sich im zu löschenden Sub-Directory keine weiteren Directory-Einträge befinden.
3. Nur die Einträge . und .. bleiben bzw. werden durch den Befehl RD gelöscht.
4. Das aktuelle Directory kann nicht gelöscht werden.

Beispiele:

`rd luga`

Löschen des Sub-Directory LUGA

`rd \auftrag\bestell`

<code>\</code>	Weg von dem Root-Directory aus beginnen
<code>AUFTRAG</code>	über AUFTRAG den Weg fortsetzen
<code>BESTELL</code>	wird gelöscht

Sie erhalten die Fehlermeldung

`Invalid path, not directory,
or directory not empty`

wenn einer der folgenden Fehler festgestellt wurde:

1. Sie versuchen eine Datei und nicht ein Sub-Directory mit RD zu löschen
2. Das zu löschende Sub-Directory ist nicht "leer"
3. Sie versuchen das aktuelle Directory zu löschen
4. Ein angegebener Directory-Name ist falsch

8.5 Anzeigen des Inhaltsverzeichnisses eines Directory (DIR)

Den Befehl DIR haben Sie bereits im Kapitel 6.4.3 kennengelernt. Mit diesem Befehl können Sie sich das Inhaltsverzeichnis jedes Directory anzeigen lassen, unabhängig von Ihrer aktuellen Position. Beispiele:

`dir \*,*`

Anzeigen des gesamten Inhaltsverzeichnisses des Root-Directory

`dir \auftrag\*.*`

Weg über Root-Directory zu AUFTRAG und dessen gesamtes Inhaltsverzeichnis anzeigen lassen

`dir \text\abt1\*.txt /p`

Weg über Root-Directory und TEXT zu Sub-Directory ABT1 und alle Dateien mit dem File-Typ TXT anzeigen lassen (mit /p werden die Dateien seitenweise angezeigt)

`dir ..`

Anzeigen des Inhaltsverzeichnisse aus dem hierarchisch vorherliegenden Directory

`dir b:\sub\*.com`

Im Laufwerk B den Weg über das Root-Directory zu SUB und aus diesem Sub-Directory alle Dateien mit dem File-Typ COM anzeigen

`dir .`

oder

`dir`

oder

`dir *.*`

Anzeigen des Inhaltsverzeichnisses des aktuellen Directory

8.6 Löschen von Dateien in einem Directory (DEL oder ERASE)

Mit den Befehlen DEL oder ERASE können Sie Dateien in jedem vorhandenen Directory löschen (unabhängig vom aktuellen Directory). Bevor Sie ein Sub-Directory mit RD löschen können, müssen Sie mit DEL alle Dateien daraus löschen.

Beispiele:

```
del \auftrag\*.*
```

Weg über Root-Directory zu AUFTRAG und alle Dateien daraus löschen

```
del \text\abt1\*.txt
```

Weg über Root-Directory und TEXT zu Sub-Directory ABT1 und alle Dateien mit dem File-Typ TXT löschen

```
del ..
```

Löschen aller Dateien aus dem hierarchisch übergeordneten Directory

```
del .
```

oder

```
del
```

oder

```
del *.*
```

Löschen aller Dateien aus dem aktuellen Directory

Wenn Sie mit einem Befehl (DEL *.*) alle Dateien aus einem Directory löschen, so fragt MS-DOS

Are you sure (Y/N)?

Wenn Sie die Tasten y und Return betätigen, so wird der Befehl ausgeführt.

8.7 Kopieren von Dateien zwischen Directories (COPY)

Den Befehl COPY haben Sie bereits im Kapitel 6 kennengelernt. Mit ihm können Sie auch Dateien von einem zu einem beliebigen anderen Directory kopieren. Zu diesem Zweck geben Sie bei "woher" und "wohin" vor dem Dateinamen den Weg an. Beispiele:

```
copy *.txt \text\abt2
```

alle Dateien aus dem aktuellen Directory mit dem File-Type TXT in das Sub-Directory ABT2 kopieren

```
copy \msdos\command.com
```

die Datei COMMAND.COM aus Sub-Directory MSDOS in das aktuelle Directory mit dem gleichen Namen kopieren

```
copy \auftrag\fibu\op.dat \text\abt1\offen
```

die Datei OP.DAT aus dem Sub-Directory FIBU nach ABT1 kopieren und gleichzeitig den Dateinamen auf OFFEN ändern

```
copy a:\msdos\*.com b:
```

aus dem Sub-Directory MSDOS auf der Diskette in Laufwerk A alle Dateien mit dem File-Typ COM nach dem aktuellen Directory in Laufwerk B kopieren

8.8 Suchen von Dateien in den Directories (SEARCH)

Diesen Befehl liefern nur wenige Hersteller zu MS-DOS. Er dient dazu, Dateien in den Directories zu finden (wenn Sie nicht mehr wissen, wo diese gespeichert sind).

Der Suchvorgang beginnt im aktuellen Directory und setzt sich in den hierarchisch darunterliegenden fort. Wenn das aktuelle Directory AUFTRAG ist, so werden AUFTRAG, FIBU und BESTELL durchsucht.

SEARCH zeigt Ihnen als Ergebnis den Weg und den gefundenen Dateinamen an. Sollte der Dateiname mehr als einmal in der durchsuchten Struktur vorkommen, so wird er mit den Weg-Bezeichnungen auch mehrmals angezeigt.

Beispiele:

```
search *.txt /d/t
```

Suchen aller Dateien mit dem File-Typ TXT und Anzeige aller durchsuchten Directories (/D) in versetzter Form (/T), welche der hierarchischen Struktur entspricht.

```
search *.* /d/t
```

Suchen aller Dateien und Anzeige aller durchsuchten Directories in versetzter Form, welche der hierarchischen Struktur entspricht. Damit bekommen Sie ein Verzeichnis Ihrer gesamten Diskette/Winchester angezeigt.

```
search *.* /d/t > prn
```

wie vorhergehendes Beispiel, jedoch erfolgt keine Anzeige am Bildschirm, sondern es wird alles am Drucker ausgegeben (siehe nächstes Kapitel).

8.9 Zusammenfassendes Beispiel

Befehl	Funktion
cd \	das Root-Directory wird aktuelles Directory
md oa	Anlegen des Sub-Directory OA
cd oa	neues aktuelles Directory wird OA
copy con text.txt	Erstellen der Textdatei TEXT.TXT (Kopieren von Bildschirm/Tastatur - CON - in die Datei TEXT. TXT)
Dies ist ein File in einem Sub-Directory und mit dem Namen TEXT.TXT.	
	Text zur Datei TEXT.TXT eingeben
Tasten F6 und Return betätigen	einggegebenen Text speichern (TEXT.TXT)
dir	Anzeigen des Inhaltsverzeichnisses des aktuellen Directory OA
cd ..	Wechseln zum vorhergehenden Root-Directory
dir /p	Anzeigen des Inhaltsverzeichnisses des aktuellen Directory (Root-Directory) "seitenweise"
rd oa	Löschen des Sub-Directory OA Sie erhalten eine Fehlermeldung, da sich in OA noch die Datei TEXT.TXT befindet:
	Invalid path, not directory, or directory not empty
del oa*.*	alle Dateien im Sub-Directory OA löschen (Sie werden nach dem Befehl von MS-DOS noch befragt, ob Sie sicher sind, alles löschen zu wollen)
rd oa	Löschen des Sub-Directory OA

9 Datenumleitung und Filter

Alle MS-DOS-Befehle lesen eingegebene Daten vom Standardeingabegerät und geben Daten zum Standardausgabegerät weiter. Den sogenannten Treiber dazu nennt man CON (Konsole). Er liest Daten von der Tastatur und gibt Daten zum Bildschirm. Diese Funktion ist im MS-DOS vorgegeben.

Diese Standardfunktion kann auf zwei verschiedene Arten beeinflusst werden:

1. Datenumleitung
2. Befehlsverkettung (mit Filtern)

Bei der Datenumleitung kann ein anderes Eingabegerät als die Tastatur bestimmt werden (z.B. eine Datei, ein anderer Computer - kein Drucker). Ebenso kann ein anderes Ausgabegerät als der Bildschirm bestimmt werden (z.B. der Drucker, eine Datei - nicht die Tastatur).

Bei der Verkettung von Befehlen werden die auszugebenden Daten automatisch als Eingabe zu einem anderen Befehl verwendet.

9.1 Datenumleitung

Für die Datenumleitung gibt es besondere Zeichen für die Anweisung an das MS-DOS:

< woher	Das Zeichen < bestimmt, daß die Daten von "woher" (Datei oder Gerät) entgegengenommen werden.
> wohin	Das Zeichen > bestimmt, daß die Daten nach "wohin" (Datei oder Gerät) gesandt werden.
>> wohin	Die Zeichen >> bestimmen, daß die Daten an das Ende von "wohin" angefügt werden (normalerweise Datei).

Beispiele für die Datenumleitung:

```
sort < unsort.txt
```

Der Befehl SORT sortiert Daten zeilenweise in auf- oder absteigender Form (siehe Kapitel 11). Die Eingabe zu SORT erfolgt hier von der Datei UNSORT.TXT. Die in der Datei enthaltenen Daten werden aufsteigend sortiert am Bildschirm ausgegeben (ohne < unsort.txt müßte die Eingabe von der Tastatur aus erfolgen).

```
sort < unsort.txt > sort.txt
```

Die Eingabe zu SORT kommt von der Datei UNSORT.TXT. Die in der Datei enthaltenen Daten werden aufsteigend sortiert in der neuen Datei SORT.TXT auf der Diskette/Festplatte gespeichert.

```
sort < com1
```

Die Daten, welche über ein Modem kommen (COM1 ist der Geräte-name für die erste serielle Schnittstelle - siehe Kapitel 10), werden sortiert am Bildschirm angezeigt.

```
sort < \text\abt1\buch.txt > \auftrag\fibu\buch.txt
```

Die Eingabe zu SORT kommt von der Datei BUCH.TXT aus dem Sub-Directory ABT1. Die Daten werden sortiert und in das Sub-Directory FIBU mit dem Dateinamen BUCH.TXT gespeichert (der Dateiname darf hier für Ein- und Ausgabe gleich sein, da zwei verschiedene Directories angesprochen werden).

```
dir > prn
```

Mit DIR wird das Inhaltsverzeichnis vom aktuellen Laufwerk/Directory angezeigt. Hier erfolgt die Ausgabe nicht am Bildschirm. Sie wird auf den Drucker (PRN - siehe Kapitel 10) umgeleitet.

```
dir b: > a:inhalt.dir
```

Das Inhaltsverzeichnis des aktuellen Directory auf der Diskette in Laufwerk B wird auf die Diskette in Laufwerk A mit dem Namen INHALT.DIR gespeichert.

```
dir a: >> a:inhalt.dir
```

Das Inhaltsverzeichnis des aktuellen Directory auf der Diskette in Laufwerk A wird auf die Diskette in Laufwerk A an das Ende der Datei INHALT.DIR gespeichert.

9.2 Befehlsverkettung und Filter

Filter sind externe Befehle, welche Daten vom Standardeingabegerät lesen, diese verarbeiten und an das Standardausgabegerät ausgeben. Die Filter des MS-DOS können alleine oder in Verbindung mit anderen Befehlen verwendet werden. Folgende Filter gibt es:

FIND	sucht nach von Ihnen vorgegebenen Zeichen im Standardeingabegerät und gibt die gefundenen Daten am Standardausgabegerät aus.
MORE	gibt Daten, welche vom Standardeingabegerät kommen, seitenweise am Standardausgabegerät aus.

SORT	sortiert auf- oder absteigend die Daten, welche vom Standardeingabegerät kommen, und gibt diese am Standardausgabegerät aus.
CIPHER	ver- und entschlüsselt Daten, welche vom Standardeingabegerät kommen, und gibt diese am Standardausgabegerät aus (nur von wenigen Herstellern).
APPLY	führt einen Befehl mehrere Male mit verschiedenen Dateien bzw. Variablen aus.

Den Filter SORT haben Sie bereits in Verbindung mit der Datenumleitung kennengelernt (Kapitel 9.1). Beachten Sie, daß Filter nur in Verbindung mit Programmen verwendet werden können, welche die Standardein- und -ausgabe des MS-DOS benutzen.

Bei einer Befehlsverkettung wird die Standardausgabe eines Befehls zur Standardeingabe des zweiten Befehls. Die Befehle werden mit dem Zeichen | verkettet. Die Begrenzung liegt lediglich bei der maximalen Länge einer Befehlszeile im MS-DOS (127 Zeichen).

Während der Abarbeitung einer Befehlskette legt MS-DOS auf der Diskette/Festplatte Dateien an, in welchen die Ein- und Ausgaben der Befehle zwischengespeichert werden. Für diese Dateien wird ab MS-DOS-Version 3 die aktuelle Uhrzeit (inklusive der hundertstel Sekunden) verwendet. Die Dateien haben die Bezeichnung:

hhmmsshh.### (z.B.: 10063455.###)

Dies ist in einem Netzwerk notwendig, da es vorkommen kann, daß mehrere Benutzer "gleichzeitig" diese sogenannten "PIPEs" verwenden und damit eine Unterscheidung möglich ist. Diese Dateien werden bei Abschluß der Befehlsverkettung automatisch wieder gelöscht.

Beispiele für die Befehlsverkettung:

```
dir | sort > prn
```

Das Inhaltsverzeichnis der aktuellen Directory im aktuellen Laufwerk dient als Eingabe für das Sortierprogramm. Nachdem SORT das Inhaltsverzeichnis aufsteigend sortiert hat, erfolgt die Ausgabe auf den Drucker (PRN - siehe Kapitel 10).


```
dir | sort | more
```

Das Inhaltsverzeichnis der aktuellen Directory im aktuellen Laufwerk dient als Eingabe für das Sortierprogramm SORT. Das sortierte Inhaltsverzeichnis dient als Eingabe für den Befehl MORE, welcher die Anzeige am Bildschirm "seitenweise" vornimmt.

```
search b:\*.* | sort | apply "copy % a:"
```

Der Befehl SEARCH erstellt eine Liste mit den Dateinamen des Laufwerks B (aus allen Directories). Diese Liste wird dem Filter SORT zum Sortieren übergeben. Die aufwärts sortierten Dateinamen werden an das Programm APPLY weitergegeben. Von APPLY werden die Dateien auf die Diskette in Laufwerk A kopiert (die Beschreibung zu APPLY finden Sie in Kapitel 11).

```
dir | find "DAT" | sort | more
```

Es werden nur Dateien angezeigt, welche den File-Typ DAT besitzen. Die gefundenen Dateien werden aufsteigend sortiert und seitenweise am Bildschirm angezeigt.

```
type \text\abt2\text1.txt | more
```

Der Text TEXT1.TXT in der Sub-Directory ABT2 wird am Bildschirm seitenweise (MORE) angezeigt (TYPE).

10 Gerätenamen

Zur ordnungsgemäßen Abarbeitung der Ein- und Ausgaben zu den Peripheriegeräten (Tastatur, Bildschirm, Laufwerke, Drucker, Modems usw.) benötigt MS-DOS Gerätetreiber. Diese beinhalten alle Informationen, die das Betriebssystem über ein Peripheriegerät wissen muß.

Für den Datentransport zu und von den Laufwerken (Diskette/Festplatte) werden sogenannte **Blocktreiber** benötigt. Diese verwenden die Befehle/Programme unter MS-DOS.

Für den Transport der Daten in serieller Form (für Tastatur, Bildschirm, Drucker, usw.) werden die **Charaktertreiber** benötigt. Diese werden durch den Namen des Peripheriegerätes benannt. Diese Namen sind für MS-DOS reserviert und dürfen nicht zur Bezeichnung von Dateien oder Sub-Directories verwendet werden.

Tabelle 10.1 Gerätetreiber

Treiber-Name	Peripheriegerät bzw. Schnittstelle
AUX	asynchrone Schnittstelle
CLOCK\$	Treiber für Echtzeit-Uhr
COM1	erste serielle Schnittstelle (V24)
COM2	zweite serielle Schnittstelle (V24)
CON	Console (Tastatur/Bildschirm)
LPT1	erste parallele Schnittstelle (normalerweise Drucker)
LPT2	zweite parallele Schnittstelle
LPT3	dritte parallele Schnittstelle
NUL	nicht vorhandener Ausgang
PRN	Drucker

Die Namen der Gerätetreiber können Sie in Befehlen verwenden.

Beispiele:

`copy con datei.dat`

Von dem Gerät CON (also hier die Eingabeeinheit Tastatur) werden die Daten in die Datei DATEI.DAT gespeichert. Alle Daten, die Sie eintippen (diese werden gleichzeitig auch am Bildschirm dargestellt), werden erst in DATEI.DAT gespeichert, wenn Sie die Tasten F6 und Return betätigen.

`copy datei.dat prn`

Die Daten der Datei DATEI.DAT werden auf das mit PRN bezeichnete Peripheriegerät (Drucker) übertragen.

`copy com1 modem.dat`

Alle Daten, die von der ersten seriellen Schnittstelle (COM1) kommen, werden in der Datei MODEM.DAT gespeichert (z.B. Daten von einem Modem, welches an COM1 angeschlossen ist).

Bitte beachten Sie zu den Gerätetreibern auch das Kapitel 13.

11 Die MS-DOS-Befehle mit Übungen

Auf den folgenden Seiten werden sämtliche MS-DOS-Befehle vorgestellt und beschrieben. Die Befehle sind anhand vieler Beispiele erklärt. Wiederholen Sie die angegebenen Beispiele zu jedem Befehl an Ihrem eigenen PC.

Befehle, welche nicht von allen PC-Herstellern geliefert werden, sind gekennzeichnet. Die Befehle sind alphabetisch geordnet.

Beachten Sie bitte zur Beschreibung der Befehle die folgende Darstellungsform beim Kommandoaufbau:

d:	Laufwerksbezeichnung
p	Teil einer Festplattelaufwerkes
s:	Ursprungslaufwerk
u:	Nummer des Laufwerkes
/x	Zusatzeingabe (Parameter)
volumelabel	Name einer Diskette bzw. Festplattenlaufwerk
mm	Monat oder Minute
dd	Tag
jj	Jahr (zwischen 1980 und 1999)
jjjj	Jahr (zwischen 1980 und 1999)
hh	Stunde
ss	Sekunde
cc	Hundertstelsekunde
n	Zahl ohne Kommastellen
dev	Gerätename
cttydev	Gerät, welches Ein- und Ausgaben erlaubt
()	(Variable/Zusatz) muß nicht eingegeben werden
...	Wiederholungsdarstellung für Variable/Zusatz

Zur Einteilung der Laufwerke beachten Sie bitte das Kapitel 6.4.1. Für die in diesem Kapitel angegebenen Beispiele werden folgende Laufwerksbezeichnungen verwendet:

- A:** erstes bzw. oberes oder linkes Laufwerk
- B:** zweites bzw. unteres oder rechtes Laufwerk
- C:** erstes Festplattenlaufwerk
- G:** RAM-DISK (wenn eingerichtet - diese kann auch eine andere Laufwerksbezeichnung haben, da immer der nächste bei Ihrem System freie Laufwerksname verwendet wird).

APPLY

externer Befehl
(nur von Zenith)

Zweck

APPLY führt einen Befehl mehrere Male mit verschiedenen Dateien aus. Es ist nicht mehr notwendig, einen Befehl immer wieder einzugeben. COMMAND.COM muß in dem Root-Directory vorhanden sein.

Kommandoaufbau

```
(d:)(weg)apply (d:) dateiname "BEFEHL %" (/x)
(d:)(weg)apply (-) "BEFEHL %" (/x)
(d:)(weg)apply "BEFEHL %" (d:) dateiname (/x)
(d:)(weg)apply "BEFEHL %" (-) (/x)
```

/x /s zusammengesetzte Befehlszeile wird nicht angezeigt

Beispiele

```
apply allbat.asc "TYPE %"
```

In der Datei ALLBAT.ASC sind folgende Zeilen gespeichert:

```
allbat.asc
msdos.txt
autoexec.bat
```

Jede Zeile der Datei ALLBAT.ASC wird jetzt für das %-Zeichen eingesetzt, d.h., es werden jetzt der Reihe nach folgende Befehle ohne weitere Eingaben durch Sie ausgeführt:

```
type allbat.asc
type msdos.txt
type autoexec.bat
```

Dabei wird jeweils vor Ausführung des Befehls die von APPLY zusammengesetzte neue Befehlszeile angezeigt (wenn /s beim APPLY-Befehl mit angegeben wurde, erfolgt dies nicht).

Eine sehr gute Anwendungsmöglichkeit besteht mit folgender Befehlskette:

```
search c:*. * | sort | apply "COPY % A:"
```

Das Programm SEARCH erstellt eine Liste mit den Dateinamen des Laufwerkes C:. Diese Liste wird dem Filter SORT zum sortieren übergeben, der sie dann dem Programm APPLY weiterreicht. Von APPLY werden die Dateien dann auf das Laufwerk A: kopiert. Auf dem Bildschirm wird laufend der jeweilige Arbeitsgang angezeigt.

```
apply "type %"
```

Jeder mit type anzuzeigende Text muß vorher (nur den Namen) am Bildschirm eingegeben werden (mit der zusätzlichen Angabe - wird der gleiche Effekt erreicht). Der Abbruch erfolgt durch die Taste F6 und Return.

ASGNPART

externer Befehl
(nur von Zenith)

Zweck

ASGNPART verbindet einen physikalischen Teil einer Festplatte mit einem logischen Laufwerksnamen.

Sie können Ihr Festplattenlaufwerk in maximal 4 Bereiche aufteilen. Die Teile werden mit den Ziffern 1 bis 4 bezeichnet. Jeder Teil kann MS-DOS oder ein anderes Betriebssystem enthalten. Damit Sie auf die einzelnen Bereiche unter MS-DOS zugreifen können, müssen diese mit Laufwerksnamen bezeichnet werden. Diese Verbindung zwischen Festplattenteil und Betriebssystem wird durch das Dienstprogramm ASGNPART hergestellt.

Kommandoaufbau

(d:)(weg)asgnpart (?)	Programmbeschreibung anzeigen
(d:)(weg)asgnpart u:	
(d:)(weg)asgnpart u:p d:	

Beispiele

asgnpart 0:

Es erscheint eine Tabelle, die die Größe der einzelnen Teilbereiche anzeigt.

```
dir d:
asgnpart 0:2 d:
dir d:
```

Wenn das Laufwerk C: (üblich) als Boot-Laufwerk eingetragen wurde, muß mit ASGNPART jeder weitere Bereich zugewiesen werden, bevor damit gearbeitet werden kann (kommt dem Einlegen einer Diskette gleich).

Wenn Sie von der Festplatte gebootet haben, verwenden Sie den Befehl asgnpart 0:2 d:. Der Bereich 2 kann anschließend als Laufwerk D: angesprochen werden.

Die Mindestgröße eines jeden Bereichs beträgt 32 kBytes.

ASSIGN

externer Befehl

Zweck

Mit ASSIGN kann einem Laufwerk eine andere logische Laufwerksbezeichnung zugeordnet werden (Laufwerksumleitung). Bei Benutzung folgender Befehle niemals verwenden:

ASGNPART
BACKUP (mit formatieren)
DISKCOMP
DISKCOPY
FORMAT
PRINT
RDCPM

Ferner darf ASSIGN nicht für ein Netzwerk-Laufwerk verwendet werden.

Kommandoaufbau

(d:)(weg)assign
(d:)(weg)assign (x=y (...))

Beispiele

assign a=c

Wenn anschließend das Laufwerk **a:** verwendet wird, benutzt das System das Festplattenlaufwerk **c:**.

assign a=c b=c

Wenn nach Ausführung des Befehls das Laufwerk **a:** oder **b:** verwendet wird, benutzt das System das Festplattenlaufwerk **c:**.

assign

Laufwerksumbenennungen wieder löschen bzw. den Ursprungszustand wieder herstellen.

Bemerkungen

Leider gibt es Anwendungssoftware, welche standardmäßig die Laufwerke A und B benutzen, ohne daß dies von Ihnen geändert werden könnte. Wenn Sie ein Festplatten-Computersystem haben, gibt es kein zweites Laufwerk, und normalerweise wurden auch die Programme auf die Platte kopiert. In diesem Falle kann vor dem Start eines Anwenderprogramms mit ASSIGN die Laufwerkszuweisung geändert werden (siehe 2. Beispiel).

ATTRIB

externer Befehl

Zweck

Eine Datei vor dem Löschen oder Verändern schützen (auf Read-Only = "Nur-Lesen" setzen). Mit dem Programm kann der Vorgang auch wieder zurückgesetzt werden (auf Read-Write setzen).

Kommandoaufbau

(d:)(weg)attrib dateibez
(d:)(weg)attrib +r dateibez
(d:)(weg)attrib -r dateibez

+r	auf Read-Only setzen
-r	auf Read-Write setzen

Beispiele

`attrib +r *.txt`

Alle Dateien mit dem Filetyp TXT werden auf Read-Only gesetzt und können damit nur noch gelesen werden (Löschen und Ändern ist nicht mehr möglich). Wenn Sie anschließend versuchen, die Dateien mit dem Befehl `del *.txt` zu löschen, erhalten Sie eine Fehlermeldung - der Löschbefehl wird nicht ausgeführt.

attrib *.*

Anzeigen der Attribute (auf Read-Only oder nicht) aller Dateien in einem Verzeichnis. Nur bei Read-Only-gesetzten Dateien wird am Anfang einer Zeile am Bildschirm ein R angezeigt. Beispiel für Bildschirmdarstellung:

```
      C:\MSDOS.KRS\MSDOS.7
      C:\MSDOS.KRS\CONFIGUR.TXT
R     C:\MSDOS.KRS\MSDOS.1
      C:\MSDOS.KRS\MSDOS4.TXT
R     C:\MSDOS.KRS\MSDOS5.TXT
R     C:\MSDOS.KRS\MSDOS6.TXT
R     C:\MSDOS.KRS\MSDOS.FUS
      C:\MSDOS.KRS\WSMSG5.OVR
      C:\MSDOS.KRS\WSOVL1.OVR
      C:\MSDOS.KRS\WS.COM
      C:\MSDOS.KRS\MAILMRGE.OVR
      C:\MSDOS.KRS\APPLY.TXT
R     C:\MSDOS.KRS\MSDOS2.TXT
R     C:\MSDOS.KRS\MSDOS.2
      C:\MSDOS.KRS\MSDOS3.TXT
```

attrib -r *.txt

Löschen des Attributes Read-Only bei allen Dateien mit dem File-Typ TXT. Damit können die Dateien wieder gelöscht oder verändert werden.

Bemerkungen

Wenn Sie mit **-r** das Read-Only-Attribut bei Dateien wieder löschen wollen, welche dieses nicht besitzen, so wird dies nur ignoriert, und zwar ohne Anzeige am Bildschirm, daß kein entsprechendes Read-Only-Attribut gefunden worden sei.

BACKUP

externer Befehl

Zweck

BACKUP sichert Dateien von Diskette oder Festplatte. Dabei können einzelne Dateien größer sein, als die Empfängerdisk Platz frei hat (dann werden Sie aufgefordert, eine neue Diskette einzulegen und das Programm setzt das Sichern fort).

Wenn das File-Sharing (Netzwerkbetrieb) aktiv ist, können nur Dateien gesichert werden, auf welche Sie Zugriff haben (siehe auch Befehl SHARE) - im anderen Falle erhalten Sie eine Fehlermeldung.

Kommandoaufbau

backup

backup ? Programmbeschreibung anzeigen

backup (dateibez (+dateibez ...)) d: (/x ...)

dateibez = Namen der zu sichernden Dateien (*. * für alle)

/x ... /a Hinzufügen von neuen zu sichernden Dateien
zu bereits gesicherten (Sie werden aufgefordert,
die letzte Sicherungsdiskette einzulegen).

/b(:datum) Nur Dateien mit dem angegebenen Datum und vor dem eingegebenen werden gesichert (wird das Datum nicht angegeben, so wird dafür das eingegebene Tagesdatum verwendet). Ohne Datumsangabe muß /B am Ende der Befehlszeile stehen (nur von Zenith).

/d(:datum) Nur Dateien mit dem angegebenen Datum und nach dem eingegebenen werden gesichert (wird das Datum nicht angegeben, so wird dafür das eingegebene Tagesdatum verwendet). Ohne Datumsangabe muß /D am Ende der Befehlszeile stehen.

/e:dateiname(+dateiname...)

Alle angegebenen Dateien werden nicht gesichert. Es darf zwischen den einzelnen Angaben kein Leerzeichen vorkommen (nur von Zenith).

/f

Die Sicherungsdiskette wird automatisch vor dem Sichern bzw. Kopieren formatiert. Sie werden nur noch befragt, ob die Empfangsdiskette ein- oder zweiseitig formatiert werden soll (nur von Zenith).

/s oder /g

Es werden alle Dateien des aktuellen Directory (Verzeichnis) und die weiter vorhandenen Dateien den Sub-Directories darunter gesichert. Dateien in darüberliegenden Directories werden nicht gesichert (wenn eine ganze Platte gesichert werden soll, muß das aktuelle Directory das ROOT-Directory sein).

/l

Auf der 1. Sicherungsdiskette wird ein "Logbuch" angelegt, in welchem alle gesicherten Dateinamen eingetragen werden (nur von Northern Telecom).

/m oder /w

Kopiert nur die Dateien, welche seit dem letzten Sicherungsvorgang verändert wurden (wird durch das Betriebssystem überwacht).

/n

Sicherungsdisketten nicht formatieren (wenn nicht angegeben, fragt das Programm vor jeder neuen Diskette, ob formatiert werden soll - nur von Zenith).

/q

Vor dem Sichern/Kopieren zeigt das Programm die Datei an und fragt Sie, ob diese gesichert werden soll (nur von Zenith).

/r

Ein Signal (Bell) ertönt bei jeder notwendigen Aktion (z.B., wenn eine neue Diskette eingelegt werden soll - nur von Zenith).

/t	Kopiert nur Dateien, welche das Tagesdatum besitzen (nur von Zenith).
/t:mm-tt-jj	Kopiert nur Dateien mit einer Uhrzeit im Directory, welche ab der angegebenen liegt (nur von Northern Telecom).
/v	Nach dem Kopieren wird überprüft, ob die kopierte Datei fehlerfrei übertragen wurde. Wenn ein Fehler gefunden wird, werden Sie befragt, ob die Datei noch einmal kopiert werden soll (nur von Zenith).

Bemerkungen

Die Parameter /a, /d(:datum), /s und /m sind üblich - alle anderen nur bei einzelnen Herstellern. Ferner müssen außer bei Zenith die Disketten vorher formatiert werden, da dies nicht vom BACKUP-Programm durchgeführt wird.

Beachten Sie bitte, daß außer bei Verwendung von wenn /a die Empfangsdiskette immer vor dem Kopieren gelöscht wird!

Beispiele

```
backup c:*,* a: /e:*.sys+*.com/f/g
```

Es werden alle Dateien/Programme im Laufwerk c: mit Ausnahme derer, welche den File-Typ SYS und COM haben, nach Laufwerk a: gesichert. Dabei wird jede neu eingelegte Diskette vor dem Kopiervorgang formatiert (nur bei Zenith). Ferner werden alle Dateien aus dem aktuellen Directory und den hierarchisch darunterliegenden Sub-Directories berücksichtigt.

```
backup *.* a: /q /s
```

Alle Dateien/Programme aus dem aktuellen Laufwerk nach Laufwerk a: sichern und vor dem Kopieren die einzelnen Dateien anzeigen und am Bildschirm abfragen, ob diese gesichert werden sollen (das aktuelle und dessen Sub-Directories werden berücksichtigt).

```
backup c:*. * a: /n
```

Alle Dateien, welche im Laufwerk c: vorhanden sind und seit dem letzten Backup-Vorgang verändert wurden, werden nach Laufwerk a: gesichert.

Das Verzeichnis der Sicherungsdisketten kann mit dem Befehl dir angezeigt werden (zusätzlich ist die Datei **BACKUPID.@@@** auf der Diskette).

Das "Zurückholen" von mit BACKUP gesicherten Dateien darf nur mit RESTORE erfolgen (nicht mit COPY).

Verwendung von BACKUP in einem Batch-File

Beispiel:

Erstellen eines BATCH-Files:

```
copy con genback.bat .
backup %1*.txt %2
backup %1*.dat %2 /a
```

F6-Taste

Der Aufruf erfolgt durch

```
genback c: a:
```

Damit werden vom aktuellen Verzeichnis in Laufwerk c: alle Dateien mit den File-Typen TXT und DAT, welche das Tagesdatum besitzen, gesichert. Nach dem Übertragen der Dateien mit dem File-Typ TXT werden die Dateien mit dem Typ DAT kopiert (/a bedeutet "hinten anhängen").

Wenn Sie vor der F6-Taste noch den BATCH-Befehl IF ERROR LEVEL verwenden, können folgende Codes abgefragt werden:

- 0 BACKUP ordnungsgemäß beendet
- 1 keine Dateien gefunden
- 2 File-Sharing Konflikte (bei Netzwerken)
- 3 Sie haben BACKUP vor dem Bearbeitungsende abgebrochen
- 4 BACKUP durch MS-DOS mit Fehler abgebrochen

Näheres hierzu finden Sie im Kapitel 12 (BATCH-Befehle).

BOOTF

externer Befehl
(nur von Zenith)

Zweck

BOOTF erlaubt die Ausführung von Programmen, welche nicht korrekt auf Computern mit einer eingebauten Festplatte laufen. Die Festplatte wird mit diesem Programm "abgeschaltet".

Kommandoaufbau

bootf

Bemerkungen

Bevor Sie den Befehl **bootf** ausführen, muß die entsprechende Programm-Diskette in Laufwerk A eingelegt werden (die Diskette **muß** bootbar sein). Das Programm **bootf** schaltet anschließend die Festplatte ab und startet den Computer erneut mit dem System auf der Diskette.

Bevor die Festplatte wieder verwendet werden kann, muß der Computer erneut gestartet werden (entweder mit **Ctrl-Alt-Del** oder durch Ab-/Anschalten des Computers).

BREAK

interner Befehl

Zweck

BREAK aktiviert eine Programmunterbrechung durch **Ctrl-Break** (oder **Ctrl-C**). Verträgt sich nicht immer mit allen Anwendungsprogrammen (dann bitte auf **OFF** setzen). Mit den Tastenkombinationen Ctrl-Break oder Ctrl-C können Programme unterbrochen werden, wenn auf **ON** gestellt wird.

Kommandoaufbau

break on	mit Ctrl-Break kann ein Programm unterbrochen werden
break of	mit Ctrl-Break kann ein Programm nicht unterbrochen werden (ausgenommen Ein/Ausgaben zu Peripheriegeräten)
break	Anzeigen, ob BREAK auf "off" oder "on" gesetzt ist

Bemerkungen

Ctrl-Break ist immer gültig bei: Standardein- und -ausgaben, Standard-Druckausgabe und Standard-AUX-Operationen.

Break wirkt nicht bei allen Anwendungsprogrammen!

BREAK ist standardmäßig immer auf **OFF** (nach dem Systemstart). Soll dieser Standardwert auf **ON** sein, fügen Sie in die Datei **CONFIG.SYS** den Befehl **BREAK=ON** ein (siehe Kapitel 13 "Die Konfiguration von MS-DOS").

CHDIR oder CD

interner Befehl

Zweck

CHDIR ändert das aktuelle Directory (Root- oder Sub-Directory), bzw. zeigt den Directory-Namen an.

Beachten Sie das Kapitel 8 "Das MS-DOS-Inhaltsverzeichnis" mit einer detaillierten Beschreibung.

Kommandoaufbau

```
chdir (d:)(weg)  
cd (d:)(weg)
```

Beispiele

```
cd \
```

Direkte Rückkehr in das Root-Directory (unabhängig davon, in welchem Directory Sie sich aktuell befinden).

```
cd text
```

Zum Sub-Directory TEXT verzweigen. Dies muß ein Sub-Directory des aktuellen sein.

```
cd text\programm
```

Zum Sub-Directory PROGRAMM verzweigen (dies ist ein Sub-Directory von TEXT). Sie müssen immer den gesamten hierarchischen Weg angeben (der Befehl kann also nicht `cd programm` lauten).

`cd ..`

Rückkehr zum hierarchisch übergeordneten Directory.

`cd a:\text\programm`

Im Laufwerk a: über TEXT zum Sub-Directory PROGRAMM verzweigen.

`cd`

Es wird Ihnen der Name des aktuellen Directory angezeigt (\, wenn es sich um das Root-Directory handelt).

CHKDSK

externer Befehl

Zweck

CHKDSK überprüft eine Festplatte oder Diskette. Es wird das gesamte Directory mit den Dateieinträgen überprüft. Vorhandene Fehler können mit dem Programm korrigiert werden. Es wird bei jedem Programmaufruf auch eine Statusmeldung angezeigt. Das Programm sollte immer angewendet werden, wenn der Verdacht besteht, daß eine Datei oder Festplatte/Diskette fehlerhaft ist.

Das Programm (Check Disk = überprüfe Platte) liest das Inhaltsverzeichnis und überprüft es auf Vollständigkeit und Fehler.

Fehlerhafte Sektoren kann das Programm nicht korrigieren. CHKDSK darf bei einem Netzwerk-Laufwerk nicht verwendet werden.

Kommandoaufbau

(d:)(weg)chkdsk (d:)(dateiname)(/x)

dateiname	Name der zu überprüfenden Datei
-----------	---------------------------------

/x	/f	korrigiert gefundene Fehler
	/v	zeitlaufend die gerade überprüfte Datei an

Beispiele

chkdsk

Aktuelles Laufwerk überprüfen und Statusmeldungen anzeigen.

chkdsk c:command.com

Programm COMMAND.COM in Festplattenlaufwerk c: überprüfen.

chkdsk c: /v /f

Festplattenlaufwerk c: überprüfen und dabei die Verzeichnis-Namen und Datei-/Programm-Namen anzeigen (am Ende werden eventuelle Fehlermeldungen mit ausgegeben). Korrektur von fehlerhaften Dateien oder Programm-Namen anzeigen (am Ende werden eventuelle Fehlermeldungen angezeigt). Korrektur von fehlerhaften Dateien oder Daten, welche einem Dateinamen nicht zugeordnet werden können.

chkdsk c: > druck

Festplattenlaufwerk c: überprüfen und die Ausgaben in der Datei DRUCK im aktuellen Laufwerk abspeichern (diese Datei kann später am Bildschirm angezeigt oder gedruckt werden).

Folgende Angaben enthält der Status-Report am Bildschirm:

- Disk-Name
- Anzahl Bytes der Disk
- Anzahl und Gesamtgröße der MS-DOS-System-Dateien (ohne COMMAND.COM)
- Anzahl und Gesamtgröße der Sub-Directories
- Anzahl und Gesamtgröße der Dateien/Programme
- Anzahl der Zeichen in fehlerhaften Sektoren (nur wenn in diesen Daten gespeichert sind)
- Angabe des noch freien Speichers auf der Disk
- Gesamtgröße des RAM-Speichers
- Angabe des noch freien Speichers im RAM für Programmausführung

Beispiel einer CHKDSK-Ausgabe:

Volume NIEDER created 20 May 1985 11.11

10547200 bytes total disk space
40960 bytes in 3 hidden files
36864 bytes in 9 directories
5435392 bytes in 500 user files
49152 bytes in bad sectors
4984832 bytes available on disk

655360 bytes total memory
609952 bytes free

Wenn nicht zuzuordnende Bereiche von Daten auf der Diskette/Festplatte gefunden werden (nur wenn beim Befehl /f angegeben wurde), erhalten diese von CHKDSK einen eigenen Dateinamen (FILE0000.CHK, FILE0001.CHK, usw.). Sie können sich anschließend die Dateien ansehen und gegebenenfalls löschen.

CHMOD

externer Befehl

(nur von Northern Telecom und Commodore)

Zweck

Anzeigen und Ändern von Datei- und Directory-Attributen (Systemdatei, Read-Only, Hiddendatei, Directory-Name, usw.).

Kommandoaufbau

(d:)(weg)chmod (d:)weg dateiname (+/-x)

+/-x	+s	Datei erhält System-Attribut
	-s	System-Attribut löschen
	+r	Datei auf Read-Only setzen
	-r	Datei auf Read-Write setzen
	+h	Datei-Attribut "Hidden" setzen
	-h	Datei-Attribut "Hidden" löschen
	+a	Archiv-Attribut setzen (automatisch, wenn Sie eine Datei erstellen oder ändern) - wird von BACKUP ausgewertet (/M)
	-a	Archiv-Attribut löschen (automatisch durch BACKUP, wenn gesichert wird)

Beispiele

```
chmod command.com +s+r+h
```

COMMAND.COM erhält die neuen Attribute System, Read-Only und Hidden. Die Datei läßt sich damit weder löschen, noch ist sie mit dem DIR-Befehl anzeigbar.

```
\chmod a:*,*
```

Der Befehl CHMOD wird aus dem Root-Directory zur Ausführung in den Speicher geladen. Sie erhalten am Bildschirm folgende Anzeige (Beispiel):

```
a:      io.sys      A S R H .
a:      msdos.sys   A S R H .
a:      command.com A . . . .
a:      sc3 (dir)    S R H V
```

23 files.

CIPHER

externer Befehl
(nur von weigen Herstellern)

Zweck

CIPHER ver- und entschlüsselt Dateien mit einem eingegebenen Code-Wort zu Sicherheitszwecken.

Verschlüsselte Dateien/Programme können nicht mehr gelesen/abgearbeitet werden (Umwandlung in nicht lesbare Zeichen). Nur mit demselben Code-Wort können Sie die Originaldateien wieder zurückbekommen. Bei dem Wort muß Groß-/Kleinschreibung beachtet werden.

Bei CIPHER handelt es sich um einen Filter.

Kommandoaufbau

```
(d:)(weg)cipher codewort > dateiname  
(d:)(weg)cipher codewort > dateiname  
(d:)(weg)cipher codewort > dateiname  
(d:)(weg)cipher codewort < dateiname  
(d:)(weg)cipher codewort < dateiname1 > dateiname2
```

Beispiele

```
cipher code < test.txt > sicher.dat
```

Die zu verschlüsselnde Datei TEST.TXT wird mit dem Wort code verschlüsselt, und in die neue Datei SICHER.DAT werden die verschlüsselten Daten geschrieben. Die alte Datei TEST.TXT sollte anschließend gelöscht werden (`del test.txt`).

```
cipher code < sicher.dat > test.txt
```

Die Datei SICHER.DAT wird mit code entschlüsselt und in TEST.TXT wieder ausgegeben.

`cipher Name > sicher.dat`

Anschließend kann am Bildschirm Text eingegeben werden, welcher in die Datei SICHER.DAT verschlüsselt (mit dem Codewort **Name**) gespeichert wird. Beenden können Sie die Eingabe mit der Tastenkombination **Ctrl-Z** und Return.

`cipher Name < sicher.dat`

Entschlüsselte Anzeige der mit dem Codewort **Name** gesicherten Datei SICHER.DAT.

CLS

interner Befehl

Zweck

CLS löscht den Bildschirm und setzt den Cursor in die linke obere Ecke des Monitors.

Kommandoaufbau

cls

COMMAND

externer Befehl

Zweck

COMMAND ruft das Programm COMMAND.COM als externen Befehl auf. Das Programm stellt die Verbindung zwischen Ihnen und dem Computer dar. Es wird automatisch beim Systemstart geladen und enthält unter anderem alle internen Befehle (COMMAND.COM stellt den Befehls-Interpreter des MS-DOS dar und wird beim Systemstart in den Speicher geladen).

Arbeiten Sie mit diesem Befehl, müssen Sie über den Aufbau von MS-DOS detaillierte Kenntnisse besitzen (daher wird hier nicht weiter auf diesen Befehl eingegangen). Beachten Sie für weitere Informationen die Kapitel 1 und 5.

Kommandoaufbau

(d:)(weg)command (d:)(weg)(cttydev)(/x)

cttydev Name des Gerätetreibers (normalerweise CON)

/x	/d	fragt beim Aufruf nicht nach Datum und Zeit (nur bei Zenith)
	/p	definiert COMMAND.COM im Speicher komplett als permanente Datei, die durch andere Programme nicht mehr überschrieben wird
	/c Befehl	definiert eine Zeichenkette bis zu maximalen Eingabelänge von 127 Zeichen (diese wird von COMMAND.COM als Befehl ausgeführt)

Beispiele

`command /d`

COMMAND.COM lädt sich damit selbst von der Diskette/Festplatte neu in den Speicher und fragt Sie nicht nach Datum und Uhrzeit.

`command /C dir`

Beachten Sie zu diesem Befehl auch das Kapitel 13.

COMP

externer Befehl

Zweck

COMP vergleicht den Inhalt von zwei Dateien/Programmen oder zwei Gruppen von Dateien auf Gleichheit. Kann nach einem Kopiervorgang (COPY-Befehl) verwendet werden, um festzustellen, ob Original und Kopie einer Datei identisch sind (wenn eine komplette Diskette mit DISKCOPY kopiert wurde, verwenden Sie besser den Befehl DISKCOMP). Wenn mehr als 10 Unterschiede festgestellt wurden, bricht das Programm den Vergleich automatisch ab. Es wird ebenfalls geprüft, ob das Dateiendekennzeichen gespeichert ist. Gefundene Fehler werden am Bildschirm angezeigt.

Kommandoaufbau

(d:)(weg)comp ? Programmbeschreibung anzeigen (nur Zenith)

(d:)(weg)comp (dateiname1 (dateiname2))

Beispiele

```
comp a:brief.txt c:brief.bak
```

Die Datei BRIEF.TXT im Laufwerk a: wird mit der Datei BRIEF.BAK im Laufwerk c: verglichen.

Sie erhalten das Ergebnis des Vergleichs am Bildschirm.

```
comp c:*. * c:\infos\*. *
```

Alle Dateien im aktuellen Directory werden mit denen im Sub-Directory INFOS verglichen (im Laufwerk c:) - Beispiel für Bildschirmanzeige:

C:\INFOS\WSMSG.S .OVR and C:WSMSG.S .OVR

Eof mark not found
Files compare ok

Ende-Markierung nicht gefunden - sonst
Gleichheit

C:\INFOS\WSOULY1 .OVR and C:WSOULY1 .OVR

Eof mark not found
Files compare ok

Ende-Markierung nicht gefunden - sonst
Gleichheit

C:\INFOS\VF25-85 and C:VF25-85

Files compare ok

Beide Dateien sind gleich

C:\INFOS\TEXT .TST and C:TEXT .TST

Files are different sizes

Die Größe der beiden Dateien ist unterschiedlich - der Vergleichsvorgang kann nicht durchgeführt werden

C:\INFOS\MAILMRGE.OVR and C:MAILMRGE.OVR

Eof mark not found
Files compare ok

Ende-Markierung nicht gefunden - sonst
Gleichheit

Compare more files (Y/N)?
comp ?

Die Programmbeschreibung zu COMP wird am Bildschirm angezeigt (nur bei Zenith).

CONFIGUR

externer Befehl
(nur von Zenith)

Zweck

CONFIGUR erstellt bzw. modifiziert das Übertragungsprotokoll für die seriellen und parallelen Ein-/Ausgänge des Computers (z.B. für Drucker-, Plotter- oder Modem-Anschluß). Ferner können die Bereiche der Festplatte auf manuelle oder automatische Zuweisung eingestellt werden (siehe auch ASGNPART).

CONFIGUR entspricht weitgehend dem Befehl MODE (CONFIGUR ist im Unterschied zu MODE "menügesteuert").

Kommandoaufbau

(d:)(weg)configur (das Programm arbeitet mit einer Menüauswahl)

Erläuterungen

Mit dem Programm können Sie Ihren Computer an verschiedene Peripheriegeräte anpassen (es wird ein spezielles Übertragungsprotokoll erstellt). Mit einem Übertragungsprotokoll wird die Art des Datenaustausches zwischen einem Computer und einem Peripheriegerät (Drucker, Plotter, Modem, usw.) bezeichnet. Für die Eingaben ist das Handbuch des anzuschließenden Gerätes notwendig. Die von Ihnen eingegebenen Daten werden im MS-DOS-Teil IO.SYS abgespeichert. Beachten Sie bitte, wenn Sie das System mit einer MS-DOS-Diskette starten, daß das veränderte IO.SYS auf allen Disketten gespeichert wird.

Es können zwei verschiedene Arten von "Ausgängen" an Ihrem Computer angepaßt werden:

parallele Ausgänge

PRN ---> LPT1, LPT2, LPT3

serielle Ausgänge

AUX ---> COM1, COM2

Die Angaben LPT1 und COM1 sind Standard-Ausgänge, welche im MS-DOS mit PRN bzw. AUX angesprochen werden können.

Für weitere Details beachten Sie bitte das MS-DOS-Handbuch zu Ihrem System oder den Befehl MODE.

COPY

interner Befehl

Zweck

Kopiert eine oder mehrere Dateien auf ein anderes oder das gleiche Laufwerk (wenn der Kopiervorgang auf dem gleichen Laufwerk im gleichen Directory erfolgt, müssen die Dateinamen unterschiedlich sein). COPY kann Dateien während des Kopiervorganges verketteten (Concatenation). Überträgt Daten zwischen verschiedenen Ein- und Ausgabekanälen des Computers (z.B. vom Bildschirm zum Drucker).

Kommando Aufbau

copy (d:)dateiname1 (d:)dateiname2 (/x)

copy (d:)dateiname1 d: (/x)

copy (d:)dateiname1 (d:)weg (/x)

copy (d:)weg (d:)(weg) (/x)

dateiname1 Name der zu kopierenden Datei

dateiname2 Name der neuen Datei

d: Name des Laufwerkes

/x /v nach dem Kopieren auf Fehler überprüfen

 /a Texte kopieren (ASCII-Files) – das Dateiendekennzeichen wird nach dem Kopieren hinzugefügt

 /b binäre Dateien kopieren (i.d.R. Programme) – alle Zeichen einschließlich dem Dateiendekennzeichen (Ctrl-Z bzw. Hexadezimal 1A) werden kopiert

Beispiele

copy test.txt a:

kopiert die Datei TEST.TXT vom aktuellen Laufwerk mit dem gleichen Namen auf die Diskette in Laufwerk a:.

```
copy test.txt test.alt
```

kopiert die Datei TEST.TXT auf dem aktuellen Laufwerk. Das Duplikat auf dem gleichen Laufwerk erhält den Namen TEST.ALT.

```
copy test.alt a:\sub\test
```

Kopieren der Datei TEST.ALT vom aktuellen Laufwerk in das Sub-Directory **sub** in Laufwerk **a:** mit dem neuen Namen TEST.

```
copy *.sys test.doc
```

Alle Dateien mit dem File-Typ SYS werden in der neuen Datei TEST.DOC zusammengefaßt (oder Befehl **copy config.sys+ansi.sys test.doc**, um die Dateien CONFIG.SYS und ANSI.SYS zu einer neuen Datei TEST.DOC zusammenzufügen).

```
copy autoexec.bat con
```

Dateiinhalt von AUTOEXEC.BAT auf den Bildschirm (**con**) "kopieren".

```
copy autoexec.bat prn
```

Dateiinhalt von AUTOEXEC.BAT auf den Drucker (**prn**) "kopieren".

```
copy con autoexec.neu
```

Alles was von der Tastatur nach dem Befehl eingegeben wird, wird in der Datei AUTOEXEC.NEU gespeichert - bzw. erst dann, wenn die Tasten F6 bzw. Ctrl-Z und Return gedrückt werden.

```
copy autoexec.bat autoexec.bat
```

Es erscheint daraufhin die Fehlermeldung **File cannot be copied onto itself**, (eine Datei kann nicht innerhalb des aktuellen Directory mit dem gleichen Namen kopiert werden).

```
copy test.alt a:\sub
```

Kopieren der Datei TEST.ALT aus dem aktuellen Directory in das Sub-Directory **sub** in Laufwerk **a:** mit dem gleichen Namen. Der Dateiname muß nur einmal angegeben werden (eine zweite Dateinamensangabe ist nur erforderlich, wenn der Name der zu kopierenden Datei in dem Ziel-Directory ein anderer sein soll).

```
copy autoexec.bat+config.sys start.doc
```

Aus den Dateien AUTOEXEC.BAT und CONFIG.SYS wird eine neue Datei mit dem Namen START.DOC gebildet ("Concatenation").

```
copy \nieder \nieder2
```

Kopieren aller Dateien aus dem Sub-Directory **nieder** in das bestehende Sub-Directory **nieder2** mit den gleichen Namen.

```
copy test.doc+,, a:test.bsp
```

Mit dem Zusatz **+,,** wird erreicht, daß die Datei TEST.DOC, welche nach Laufwerk **a:** mit dem neuen Namen TEST.BSP kopiert wird, das von Ihnen beim Start des Computers eingegebene Datum und die Uhrzeit erhält.

```
copy test.alt* *.txt
```

Es werden alle Dateien mit dem File-Typ TXT aus dem aktuellen Directory kopiert. Die Dateien erhalten den Namen TEST.ALT (Concatenation).

CTTY

interner Befehl

Zweck

CTTY ändert das Gerät, vom welchem aus Befehle eingegeben werden (z.B. können Ein-/Ausgaben anschließend über ein Modem erfolgen). Normalerweise ist dies die Tastatur (und Bildschirm).

Kommandoaufbau

ctty device

device	AUX	Hilfseingang (normalerweise COM1)
	CON	Tastatureingabe/Bildschirmausgabe (Standard)
	COM1	erste serielle Schnittstelle
	COM2	zweite serielle Schnittstelle (soweit vorhanden)

Beispiele

ctty aux

Jetzt können z.B. Ein-/Ausgaben über ein Terminal erfolgen, welches an AUX angeschlossen ist (in der Regel handelt es sich bei AUX um die erste V24-Schnittstelle = COM1).

ctty con

Standardzustand herstellen (Tastatureingabe und Bildschirmausgabe).

Bemerkungen

Mit vielen Programmen und Programmiersprachen (z.B. BASIC) funktioniert dieser Befehl nicht, da diese die Schnittstellen (z.B. Bildschirm) direkt ansprechen, also das MS-DOS "umgehen".

DATE

interner Befehl

Zweck

Anzeigen und/oder Ändern des Tagesdatums (Systemdatum). Ein einmal eingegebenes Datum bleibt bis zur nächsten Änderung gespeichert, auch wenn zwischenzeitlich der PC abgeschaltet worden ist.

Kommandoaufbau

date
date mm-tt-jj
date tt-mm-jj
date jj-mm-tt

mm	=	Monat	1 bis 12
tt	=	Tag	1 bis 31
jj	=	Jahr	80 bis 99 oder 1980 bis 2099

Beispiele

date 05-16-85

Das neue Systemdatum ist anschließend der 16. Mai 1985. Es erscheint keine weitere Anzeige auf dem Bildschirm, bzw. es kommt sofort wieder der Prompt.

date

Anschließend wird das gespeicherte Systemdatum angezeigt, und Sie können ein neues Datum eingeben oder durch Drücken der Taste Return das Datum unverändert belassen.

Bemerkungen

Wenn Sie das Datum in einem falschen Format eingeben, erhalten Sie die Fehlermeldung

```
Invalid date
Enter new date:
```

und Sie müssen das Datum neu eingeben.

Beachten Sie bitte hierzu auch den Befehl **SELECT**. Das Format des Datums ergibt sich aufgrund des jeweils geladenen Landes - siehe auch Kapitel 13 "Die Konfiguration von MS-DOS" (z.B. bei Deutschland: date 16.5.85).

DEL oder ERASE

interner Befehl

Zweck

Löschen von Dateien/Programmen. Es können wahlweise die Befehle DEL oder ERASE verwendet werden. Dateien/Programme mit dem Attribut "read-only" können ebenso wie Sub-Directory-Namen (diese nur mit dem Befehl RMDIR oder RD) nicht gelöscht werden (siehe Befehl ATTRIB).

Kommandoaufbau

```
del dateiname  
del (d:)weg(dateiname)  
erase dateiname  
erase (d:)weg(dateiname)
```

Beispiele

```
del test.alt
```

Löschen der Datei TEST.ALT im aktuellen Directory im aktuellen Laufwerk.

```
del a:\sub\test
```

Löschen der Datei TEST aus dem Sub-Directory **sub** auf der Diskette in Laufwerk **a:**.

```
del a:\sub
```

Mit diesem Befehl werden alle Dateien/Programme im Sub-Directory **sub** auf der Diskette in Laufwerk **a:** gelöscht.

`del *.*`

oder

`del`

Löschen aller Dateien/Programme im aktuellen Laufwerk/Directory. Vor dem Löschen werden Sie noch kurz befragt, ob Sie sich sicher sind, alle Dateien löschen zu wollen (**Are you sure (Y/N)?**). Die Frage erscheint nur dann, wenn Sie bei dem Befehl `DEL *.*` angeben.

DIR

interner Befehl

Zweck

Anzeigen von Inhaltsverzeichnissen mit den gespeicherten Informationen (File-Name, File-Typ, Größe der Datei in Bytes, Datum und Uhrzeit sowie Laufwerkbezeichnung, Name der Diskette/Festplatte, Directory-Name, Anzahl der Dateieinträge einschließlich dem Sub-Directory-Namen, freier Speicher). Datum und Uhrzeit werden abhängig vom Landescode (siehe Kapitel 13 "Die Konfiguration von MS-DOS") angezeigt. Datum und Uhrzeit werden jeweils bei Anlage oder Änderung einer Datei neu gespeichert.

Kommandoaufbau

dir (d:)(dateiname) (/x)

dir (d:)(weg) (/x)

..bi 01

..tb

/x	/p	Ausgabe mit Pause nach jeder Bildschirmseite
	/w	Die Ausgabe erfolgt nur mit den Dateinamen (je 5 Dateinamen in einer Zeile) - man nennt dies die Anzeige in "weiter" Form.

Beispiele

dir

oder

dir *.*

Es wird das Inhaltsverzeichnis des aktuellen Directory/Laufwerks angezeigt.

Beispiel:

```
Volume in drive C has no label
Directory of C:\

WSC      COM      25600    1-01-80    12:03a
WSOVLV1  OVR      46080    1-01-80    3:59a
WSMSGs   OVR      27392    1-01-80    3:57a
MAILMRGE OVR      12800    1-01-80    4:49a
WSTYpe   COM       1984     7-24-85    9:03p
1                0  10-26-85    2:24p
KAPITEL  BAK      84864    10-26-85    2:11p
KAPITEL  111     85248    10-26-85    2:24p
      8 File(s)      77824 bytes free
```

`dir *.sys`

Es werden nur Dateien mit dem File-Typ SYS aus dem aktuellen Directory angezeigt.

`dir /w`

Anzeige des Inhaltsverzeichnisses vom aktuellen Directory in "weiter" Form.

Beispiel:

```
Volume in drive C has no label
Directory of C:\

WSC      COM  WSOVLV1  OVR  WMSGs   OVR  MAILMRGE OVR  WSTYpe   COM
1                KAPITEL BAK KAPITEL 111 2
      9 File(s)      77312 bytes free
```

`dir > prn`

Drucken des Inhaltsverzeichnisses vom aktuellen Directory.

```
dir a:\sub
```

Anzeigen des Inhaltsverzeichnisses der Diskette im Laufwerk a: - nur aus dem Sub-Directory sub.

```
dir c: | sort > prn
```

Ausdrucken des sortierten Inhaltsverzeichnisses aus Laufwerk c:. Ohne > prn erfolgt die sortierte Ausgabe auf den Bildschirm.

```
dir c: | sort /+14 | more
```

Anzeige der sortierten Dateien aus Laufwerk c: (es wird ab der Spalte 14 aufsteigend sortiert - wenn noch /R hinter +14 angegeben wird, erfolgt der Sortiervorgang absteigend). Nach jeder Bildschirmseite stoppt der Ausgabevorgang (more). Beachten Sie hierzu auch das Kapitel 9.

Bemerkungen

Wenn statt einer Dateigrößenangabe <DIR> erscheint, so handelt es sich bei dem Namen um einen Sub-Directory-Eintrag.

Befehl	entspricht dem Befehl
DIR	DIR *.*
DIR dateiname	DIR dateiname.*
DIR .typ	DIR *.ext
DIR .	DIR oder DIR *.*
DIR ..	Es wird das Inhaltsverzeichnis des hierarchisch vorher liegenden Directory angezeigt.

Hinter dem Eintrag . in einem angezeigten Inhaltsverzeichnis verbirgt sich der Name des aktuellen Directory und hinter .. der Name des hierarchisch übergeordneten Directory.

Beispiele für einen "normalen" Directory-Ausdruck und einem "weiten" von
eines Sub-Directory auf einer Festplatte:

Volume in drive C is NIEDER

Directory of C:\GWBASIC

```

.          <DIR>      20.05.85  12.05
..         <DIR>      20.05.85  12.05
BASIC1    EXE      58720   7.02.84   0.10
FRANCE    BAS        512   9.11.84   0.10
BASICA    EXE     70784   5.11.84   0.14
ADR       BAS     3010   1.01.80   0.14
ADR2      BAS      930   1.01.80   0.15
ADR3      BAS     1094  10.04.85   0.15
WS        47941  10.04.85   0.15
ELITE     BAS       29   8.05.85   8.49
ETI       BAS      643   6.02.85   0.55
WINDOW    BAS      512   1.01.80   0.05
ANWEND     <DIR>      6.06.85  16.03
13 File(s) 4931584 bytes free

```

Volume in drive C is NIEDER

Directory of C:\GWBASIC

```

.          ..          BASIC1 EXE FRANCE  BAS BASICA  EXE
ADR       BAS  ADR2    BAS  ADR3    BAS  WS      ELITE  BAS
ETI-DOS   BAS  ETI-MP1 BAS  ETI-OA   BAS  ETI-PAIN BAS  ETI-EW1 BAS
ETI-EW2   BAS  ETI-MP2 BAS  ETI-SWL  BAS  ETI      BAS  ETI-GER  BAS
ETI-TKE   BAS  FREIB   BAS  WINDOW  BAS  ANWEND
24 File(s) 4927488 bytes free

```


DISKCOMP

externer Befehl

Zweck

Überprüft zwei Disketten auf absolute Gleichheit (der Vergleich erfolgt Track-by-Track). Die beiden Disketten müssen das gleiche Format besitzen und vom Inhalt her absolut identisch sein.

Wenn Sie eine neue Diskette formatiert haben und mit dem Programm COPY eine Diskette komplett (*.*) übertragen, so sind die beiden Disketten anschließend normalerweise nicht identisch.

Der Befehl darf nicht bei einem Netzwerk-Laufwerk verwendet werden.

Kommandoaufbau

(d:)(weg)diskcomp (s: (d:)) (/x)

/x	/1	vergleicht nur die erste Seite der Disketten
	/8	vergleicht nur die ersten 8 Sektoren (sonst 9)

/x gibt es nicht bei allen Herstellern!

Beispiel

diskcomp a: b:

Die Disketten in Laufwerk **a:** und **b:** werden auf Gleichheit überprüft. Die Laufwerksangaben dürfen nicht gleich sein (also nicht **diskcomp a: a:**). Bei Ungleichheit erscheint eine Fehlermeldung nach der Prüfung.

diskcomp

Diese spezielle Befehlsform gibt es nicht bei allen Herstellern (beachten Sie dazu auch die Kapitel 6 und 7).

Es wird dabei nur mit einem Laufwerk gearbeitet (wenn Sie z. B. ein Festplattensystem mit nur einem Diskettenlaufwerk haben, kann der Vergleichsvorgang trotzdem erfolgen). Sie werden in diesem Falle während der Programmausführung aufgefordert, das Original bzw. die Kopie abwechselnd einzulegen (Original in Speicher einlesen - Diskettenwechsel - Speicherinhalt mit Kopie vergleichen - Diskettenwechsel ...). Es wird hier immer mit dem aktuellen Laufwerk gearbeitet (am Prompt ersichtlich).

DISKCOPY

externer Befehl

Zweck

Der komplette Inhalt einer Diskette wird auf eine andere übertragen (Erstellung eines Disketten-Duplikates). Vor dem Kopiervorgang wird die neue Diskette formatiert (damit sind eventuell bisher auf der neuen Diskette vorhandene Daten gelöscht).

Es sind ein oder zwei Laufwerke notwendig. Die Originaldiskette sollte einen Schreibschutz haben, um Fehler zu vermeiden (Kopiervorgang in die verkehrte Richtung).

Mit dem Befehl kann nicht von oder nach einer Festplatte kopiert werden.

Der Befehl darf nicht bei einem Netzwerk-Laufwerk verwendet werden.

Kommandoaufbau

diskcopy (s: (d:)) (/x)

/x	/v	nach dem Kopieren Zieldiskette prüfen (entspricht DISK-COMP) (nur bei Zenith)
	/1	Es wird nur eine Seite einer Diskette kopiert (nicht bei Zenith)

Beispiele

diskcopy a: b:

Die Diskette in Laufwerk a: wird auf die Diskette in Laufwerk b: übertragen.

diskcopy

Es wird dabei nur mit einem Laufwerk gearbeitet. Wenn Sie ein Festplattensystem mit nur einem Diskettenlaufwerk haben, kann der Kopiervorgang trotzdem erfolgen (bei Zenith muß der Befehl **diskcopy a: b:** lauten). Sie werden in diesem Falle während der Programmausführung aufgefordert, das Original bzw. die Ziel-Diskette einzulegen (Original in Speicher einlesen - Diskettenwechsel - Speicherinhalt auf Ziel-Diskette speichern - Diskettenwechsel ...). Es wird hier immer mit dem aktuellen Laufwerk gearbeitet (am Prompt ersichtlich).

Wenn also am Bildschirm der Prompt **A>** steht, so erfolgt der Kopiervorgang von **a:** nach **a:**, wenn **B>**, dann von **b:** nach **b:**.

Bemerkung

Sollte sich mit DISKCOPY eine Diskette nicht kopieren lassen, so formatieren Sie mit dem Programm FORMAT eine neue und benutzen den Befehl COPY.

ECHO

interner Befehl

Zweck

ECHO kontrolliert die Ausgabe von Befehls- und Kommentarzeilen während der Ausführung einer Batchdatei (siehe Kapitel 12) bzw. zeigt einen Kommentar am Bildschirm an. Nach dem Einschalten des Computers ist ECHO standardmäßig angeschaltet.

Kommandoaufbau

echo
echo (on)
echo (off)
echo (text)

on	Einschalten der Anzeigefunktion
off	Ausschalten der Anzeigefunktion
text	Anzuzeigender Text

Beispiel

Erstellen Sie einen BATCH-File (siehe auch Kapitel 12) wie folgt:

```
copy con echotst.bat
```

Mit dem Befehl COPY können Sie die Batch-Datei ECHOTST.BAT eingeben

```
echo off
echo echo ist ausgeschaltet
echo Inhaltsverzeichnis Laufwerk A
dir a:
echo on
echo echo ist eingeschaltet
dir a:
```

Aufgrund der obigen Befehle wird am Bildschirm durch Aufruf von ECHOTST.BAT (nur den Befehl `echotst` und Return eingeben) folgendes angezeigt:

```
echo off
echo ist ausgeschaltet
Inhaltsverzeichnis Laufwerk A
```

Inhaltsverzeichnis des aktuellen Directory in Laufwerk A
wird angezeigt

```
echo echo ist eingeschaltet
echo ist eingeschaltet
```

Die erste Textzeile ist der Befehl, die zweite Zeile die vom
Befehl am Bildschirm angezeigte Zeile

```
dir a:
```

Inhaltsverzeichnis des aktuellen Directory in Laufwerk A
wird angezeigt

EDLIN

externer Befehl

Allgemeines und Zweck

EDLIN ist das Dienstprogramm zur Herstellung und Änderung von Textdateien (und damit natürlich auch von Quellenprogrammen). Es handelt sich dabei um einen zeilenorientierten Editor bzw. ein einfaches Textbearbeitungsprogramm (Maximallänge jeder Zeile ist 253 Zeichen). Die Zeilennummern werden vom Programm automatisch vergeben (1 bis 65529). Die Zeilennummern werden nicht auf der Diskette/Festplatte gespeichert.

Wird bei den Befehlen eine Zeilennummer angegeben, welche größer ist als die größte vorhandene, so wird diese automatisch vom Programm in die Zeilennummer der Zeile nach der letzten vorhandenen geändert.

Mit EDLIN kann nur 64 kB des Speichers Ihres Computers genutzt werden. Davon werden ca. 75 % maximal mit Text gefüllt (beachten Sie die EDLIN-Befehle a und w).

Folgende Funktionen sind möglich:

- Erstellen und Speichern von neuen Texten
- Überarbeiten von bestehenden Texten (Dateien), alte und neue Fassung speichern
- Anzeigen, Ändern, Einfügen und Löschen von Zeilen
- Suchen, Ersetzen und Löschen von Zeichenketten in einer oder mehreren Zeilen

Kommandoaufbau

(d:)(weg)edlin dateiname (/b)

/b Wenn im Text das Zeichen Ctrl-Z vorkommt, muß dieser Parameter angegeben werden, da sonst nur bis zu diesem Zeichen Text bearbeitet werden kann (Ctrl-Z ist bei MS-DOS das Dateiende-Kennzeichen).

Bemerkungen

Wird ein Befehl während der Bearbeitung eines Textes falsch eingegeben, so erhalten Sie eine Fehlermeldung von EDLIN angezeigt.

Sie können die **Funktionstasten** Ihrer Tastatur wie folgt bei der Textbearbeitung benutzen:

TASTE	FUNKTION	BESCHREIBUNG
F1 oder-->	kopiert ein Zeichen	kopiert ein Zeichen aus dem Eingabespeicher in die neue Zeile
F2x	kopiert bis Zeichen x	kopiert alle Zeichen bis zum angegebenen Zeichen x aus dem Eingabespeicher in die neue Zeile
F3	alle Zeichen kopieren	kopiert alle Zeichen des Eingabespeichers in die neue Zeile
F4x	überspringen bis Zeichen x	ignoriert alle Zeichen aus dem Eingabespeicher bis zum Zeichen x
F5	neuer Eingabespeicher	schreibt eine neu eingegebene Zeile in den Eingabespeicher
F6	Eingabeende	erzeugt ein Ctrl-Z-Zeichen (damit wird das Zeichen für Dateiende erzeugt)
F7	Nullzeichen	das Zeichen wird mit dem Wert 00 Hexadezimal gespeichert

Backspace oder <--	ein Zeichen löschen	ein Zeichen im Speicher und in der Eingabezeile löschen
Ins	einfügen	umschalten auf Zeichen einfügen
Del	löschen	ein Zeichen im Eingabespeicher löschen (nicht jedoch in der sichtbaren Eingabezeile)
Esc	Eingabe löschen	löscht die gesamte Eingabezeile, nicht jedoch den Eingabespeicher

Befehlsübersicht zu EDLIN

#	Zeilennummer
#a	Nummer der ersten Zeile
#e	Nummer der letzten Zeile

Befehl: (#)a

Zweck: Laden von Zeilen der zu bearbeitenden Datei in den Speicher, bis dieser voll bzw. die Datei komplett geladen ist. Wenn schon ein Teil geladen ist, werden die hinzugeladenen Zeilen im Speicher an den bereits vorhandenen Text hinzugefügt. Wenn nur a angegeben wird, versucht das Programm den gesamten Text zu laden.

Befehl: (#a),(#e),#n,(#)c

Zweck: Kopieren von einer oder mehreren Zeilen an eine andere Stelle des Textes.

#a	=	erste zu kopierende Zeile
#e	=	letzte zu kopierende Zeile
#n	=	Zeilennummer, vor der der Text eingefügt werden soll
#	=	wie oft die Zeile(n) kopiert werden sollen
c	=	Kopierbefehl

Befehl: (#a)(, #e)d

Zweck: Es werden eine oder mehrere Zeilen gelöscht. Wird nur d angegeben, so wird nur die aktuelle Zeile gelöscht.

Befehl: #

Zweck: Eine Zeile neu eingeben bzw. die aktuelle Zeilennummer auf die Zeile # setzen. Wenn Sie, statt die Zeile neu oder verändert einzugeben, nur **Return** drücken, bleibt die Text-Zeile unverändert.

Befehl: e

Zweck: Das Programm beenden und den Text speichern (der alte Text bleibt erhalten und erhält den Namenszusatz .BAK). Wenn nicht mehr ausreichend Platz auf der Diskette/Festplatte ist, wird nur soviel wie möglich gespeichert - der Rest ist verloren.

Befehl: (#)i

Zweck: Es werden eine oder mehrere neue Zeile(n) vor die angegebene Zeile eingefügt. Der "Einfügemodus" bleibt so lange bestehen, bis **Ctrl-C** oder **Ctrl-Z** und **Return** eingegeben wird. Immer erforderlich, wenn ein neuer Text erstellt wird. Wird der Befehl i ohne Zahl eingegeben, so wird der neu einzugebende Text immer am Dateiende bzw. ab der aktuellen Zeile eingefügt.

Befehl: (#a)(, #e)l

Zweck: Anzeigen von Zeilen (nur die im Speicher vorhandenen können angezeigt werden) am Bildschirm. Mit der Tastenkombination **Ctrl-S** kann gestoppt oder weiter angezeigt werden. Die aktuelle Zeile ist mit einem Stern (*) gekennzeichnet.

Befehl: (#a),(#e),#m

Zweck: Verschieben von Zeile(n) innerhalb des Textes vor die angegebene Zeile. Wird keine Anfangs-/Ende-Zeilenummer angegeben, so wird nur die aktuelle Zeile verschoben. Anschließend werden die Zeilen automatisch neu nummeriert.

Befehl: (#a),(#e)p

Zweck: Zeilen werden seitenweise aufgelistet. Nach Auflistung der Zeilen auf dem Bildschirm wird die letzte angezeigte Zeile zur aktuellen (im Gegensatz zum Befehl l).

Befehl: q

Zweck: Programm beenden, ohne den Text zu speichern. Zur Sicherheit fragt Sie das Programm EDLIN nach Eingabe des Befehls, ob dieser ausgeführt werden soll.

Befehl: (#a),(#e)(?)Rzeichenkette1F6zeichenkette2

Zweck: In den bezeichneten Zeilen wird die zeichenkette1 durch die zeichenkette2 ersetzt. Wird das ? mit angegeben, so wird vor dem Austauschen des Textes die mit der zeichenkette1 gefundene Zeile angezeigt, und Sie werden befragt, ob Sie austauschen wollen.

Befehl: (#a),(#e)(?)Szeichenkette

Zweck: Suchen einer zeichenkette in den bezeichneten Zeilen. Wird ? mit angegeben, so wird die gefundene Zeile angezeigt, und Sie werden befragt, ob es sich um die gesuchte handelt (wenn Sie mit NEIN antworten, sucht das Programm im weiteren Text, ob die zeichenkette noch einmal vorkommt - der Vorgang kann so lange wiederholt werden, bis das Textende erreicht ist).

Befehl: (#)Tdateiname

Zweck: Kopiert eine Textdatei in den Text im Speicher ab der angegebenen Zeile. Nach dem Vorgang wird der Text im Speicher automatisch neu nummeriert.

Befehl: (#)w

Zweck: Schreibt die angegebene Zeilenzahl auf die Diskette/Festplatte zurück und löscht diese dann im Speicher. Die noch verbleibenden Zeilen werden neu nummeriert. Dies ist bei Dateien notwendig, welche zu groß sind, um komplett in den Speicher zu passen. Normalerweise werden anschließend mit Befehl **a** weitere Zeilen des Textes von der Diskette/Festplatte nachgeladen.

Wenn Sie statt einer Zeilennummer bei den Befehlen einen "." verwenden, so bezeichnet dieser die Nummer der aktuellen Zeile (die aktuelle Zeile im Text kann am "*" am Anfang einer Zeile erkannt werden). Ebenso kann für die Nummer der Zeile nach der letzten im Text vorhandenen, das Zeichen "#" verwendet werden.

Beispiel/Übung

Rufen Sie bitte das Programm EDLIN wie folgt auf:

```
edlin diskette
```

Daraufhin erscheint am Bildschirm

```
New File
```

EDLIN zeigt Ihnen damit an, daß der Text noch nicht auf der Diskette/Festplatte vorhanden ist (wenn die Datei DISKETTE schon vorhanden wäre, erhielten Sie die Meldung **End of input file**).

Tippen Sie den folgenden Text ein (Befehl **i** und Return und Text Zeile für Zeile eintippen):

Eine Diskette besteht aus einer 5.25 Inch großen runden Plastikscheibe, überzogen mit einer feinen magnetisierbaren Oxidschicht. Diese Scheibe wird in einer Hülle aufbewahrt, die sie fast vollständig bedeckt. Nur ein kleiner schmaler Schlitz für den Schreib-Lese-Kopf bleibt frei, der Schreib-Lese-Schlitz. Die Scheibe selbst und auch die darauf gespeicherten Informationen, sind sehr leicht zu zerstören. Sie sollten die Disketten immer sehr sorgfältig behandeln und folgendes beachten:

- Wenn Sie die Diskette aus ihrer Verpackung nehmen, berühren Sie nie die Oberfläche, vor allem nicht am Schreib-Lese-Schlitz.
- Ist die Diskette nicht im Laufwerk, so stecken Sie sie wieder in die Verpackung.
- Öffnen Sie die Klappe des Laufwerks erst, wenn die rote Kontroll-Lampe wieder erloschen ist.
- Achten Sie darauf, daß kein Staub oder irgendwelche Flüssigkeiten auf die Scheibe gelangen.
- Bewahren Sie die Diskette nicht in der Nähe magnetischer Felder auf. Ein Kugelschreiber kann schon Daten zerstören.
- Gehen Sie mit Disketten nicht durch Röntgenanlagen (z.B. sehr oft bei ausländischen Flughäfen).
- Lassen Sie keine Diskette im Laufwerk, wenn Sie den Strom des Computers ein- oder ausschalten. Durch eventuelle Stromstöße kann der Schreib-Lese-Kopf auf die Diskette "stürzen" (Head-Crash).
- Achten Sie darauf, daß die Disketten nicht zu heiß (über 52 Grad) oder zu kalt (unter 10 Grad) gelagert werden.
- Benutzen Sie zum Beschriften der Disketten niemals einen Kugelschreiber, sondern immer einen weichen Filzschreiber.
- Achten Sie darauf, daß Disketten niemals gebogen oder gestapelt werden.

Wollen Sie sich gegen unbeabsichtigtes Löschen von Dateien schützen, so verdecken Sie die Kerbe an der linken Seite der Diskette mit einem kleinen Streifen (werden bei neuen Disketten immer mitgeliefert). Die damit versehenen Disketten können nur noch gelesen werden.

Übungen zu den Befehlen:

30,#,1,c

Damit wird der Text ab Zeile 30 bis zum Textende (#) vor die Zeile 1 kopiert (die letzten Zeilen an den Textanfang kopieren).

1
und 2mal Return betätigen

Die aktuelle Zeile auf die Zeilennummer 1 setzen.

.,6d

oder

,6d

Von der aktuellen Zeile bis zur Zeile 5 den Text löschen (die ersten 5 Zeilen). Wenn die erste die aktuelle Zeile ist, kann statt 1 auch der "." verwendet werden.

Die aktuelle Zeile erkennen Sie am * zu Beginn einer Textzeile.

1

Es werden 23 Zeilen am Bildschirm angezeigt.

1,411

Es werden alle Zeilen ab der ersten am Bildschirm angezeigt (es kann mit Ctrl-S gestoppt werden - weiter mit Ctrl-S).

p

Am Bildschirm werden ab der aktuellen Zeilen 23 weitere angezeigt. Wenn Sie erneut **p** eingeben, werden die folgenden 23 Zeilen am Bildschirm gelistet (der Befehl kann bis Textende wiederholt werden).

30,41,1m

Die Zeilen 30 bis 41 einschließlich werden vor die Zeile 1 im Text verschoben.

1,#?RDisketteF6Disk

Ab der Zeile 1 bis zum Textende (#) wird der Text **Diskette** durch den Text **Disk** ersetzt (Vor jedem einzelnen Austausch wird die Zeile angezeigt, in welcher **Diskette** gefunden wird, und Sie werden befragt, ob der Text in der Zeile ausgetauscht werden soll - es wird dabei die bereits geänderte Zeile angezeigt, damit Sie erkennen können, ob Sie mit der Änderung einverstanden sind).

1,41RDiskF6Diskette

Ab der Zeile 1 bis zur Zeile 41 wird der Text **Disk** durch den Text **Diskette** automatisch ersetzt. Die geänderten Zeilen werden am Bildschirm angezeigt.

e

Textbearbeitung beenden und den Text speichern (hier unter dem Namen **DISKETTE**). Die alte Version des Textes (vor Eingabe des Befehls **edlin**) wird unter dem Namen **DISKETTE.BAK** gespeichert.

EXE2BIN

externer Befehl

Zweck

EXE2BIN formt Programme mit dem Format .EXE in das Format .COM um. Ein Programm vom Typ .COM hat die Vorteile, daß es weniger Platz auf der Diskette/Festplatte belegt und schneller abläuft.

Bitte beachten Sie, daß nicht jedes Programm mit dem File-Typ EXE in eines mit dem Typ COM umgewandelt werden kann.

Kommandoaufbau

(d:)(weg)exe2bin dateiname1 (d:)(dateiname2(.typ))

dateiname1	Name des .EXE-Programms
d:	Ziellaufwerk für das umgewandelte Programm
dateiname2	Name des .COM-Programms
typ	normalerweise .COM angeben (ohne = .BIN)

Beispiele

```
exe2bin programm c:programm.com
```

Das Programm **PROGRAMM.EXE** wird in **PROGRAMM.COM** auf Laufwerk c: umgewandelt.

```
exe2bin programm c:programm
```

Das Programm **PROGRAMM.EXE** wird in **PROGRAMM.BIN** auf Laufwerk c: umgewandelt.

```
exe2bin programm
```

Das Programm **PROGRAMM.EXE** wird in **PROGRAMM.BIN** auf dem aktuellen Laufwerk in dem aktuellen Directory umgewandelt.

Bemerkungen

Das Ursprungsprogramm muß in einem gültigen EXE-Format, erstellt mit dem LINKer, vorliegen. Der residente und Daten-Teil des Programms müssen kleiner als 64 kByte sein. Es darf kein sogenanntes STACK-Segment enthalten.

Wenn die Fehlermeldung **File cannot be converted** erscheint, liegt das Ursprungsprogramm nicht im gültigen Format vor.

Für weitere Angaben beachten Sie bitte die Handbücher Ihres Computerherstellers oder von MicroSoft. Für die Arbeit mit diesem Programm sind fundierte Systemkenntnisse erforderlich.

EXIT

interner Befehl

Zweck

Dieses Kommando ermöglicht das Verlassen des COMMAND.COM-Prozessors und die Rückkehr zu einem vorherigen Level (z.B. zu einer Applikation - Programm), soweit einer existiert.

Kommandoaufbau

`exit`

Beispiel

Laden Sie GWBASIC bzw. BASICA (Programmiersprache/Interpreter - ab Version 3 möglich):

`basica`

Wenn BASIC geladen ist, starten Sie den COMMAND-Prozessor:

`shell "COMMAND.COM"`

Es erscheint daraufhin der MS-DOS-Prompt (z.B. C>) am Bildschirm, und Sie können beliebige MS-DOS-Befehle, wie z.B.

`dir`

eingeben. Damit wird in diesem Falle das Inhaltsverzeichnis am Bildschirm angezeigt. Durch Eingabe von

`exit`

kommen Sie wieder in den vorherigen Level, also zu BASIC zurück.

FC

externer Befehl

Zweck

FC vergleicht zwei Dateien und zeigt eventuelle Unterschiede an (damit kann z.B. festgestellt werden, welche Datei die aktuellere ist). Ohne Zusatzeingaben vergleicht das Programm nach folgenden Kriterien:

1. Zeile für Zeile bei Textdateien
2. Zeichen für Zeichen bei Binärdateien (Programme)
3. Groß- und Kleinschreibung wird berücksichtigt
4. unterschiedliche Zwischenräume werden berücksichtigt
5. nach einem Unterschied müssen in Textdateien mindestens drei Zeilen wieder übereinstimmen

Kommandoaufbau

(d:)(weg)fc (/x...) dateiname1 dateiname2

- | | | |
|----|----|--|
| /x | /b | Dateien werden Zeichen für Zeichen verglichen |
| | /c | Alle Buchstaben einer Datei werden als Großbuchstaben angesehen |
| | /w | Alle Zwischenräume einer Textdatei werden als gleich groß angesehen. |
| | /# | Bestimmt die Zahl der Zeilen in Textdateien, die nach einer Differenz wieder gleich sein müssen (# = 1 bis 9). |

Testdaten:

Dateiname TEST

DATEIA ABCDEFGHIMNOPQRSTUVWXYZ

Dateiname TEST.NEU

DATEIB ABCGHIJ12PQRSTUVWXYZ45WXYZ

Dateiname TEST.ALT

DATEIB ABCGHIJPQRSTUVWXYZ1245

Beispiele

```
fc /b a:\sub\test.alt test.neu
```

Zeichen für Zeichen wird die Datei TEST.ALT, die in dem Sub-Directory sub des Laufwerks a: ist, mit der Datei TEST.NEU, die im aktuellen Laufwerk ist, verglichen. Die Unterschiede werden auf dem Bildschirm angezeigt:

```
0000001d: 50 31
00000020: 51 32
00000023: 52 50
00000026: 53 51
00000029: 54 52
0000002c: 55 53
0000002f: 56 54
00000032: 57 55
00000035: 58 56
00000038: 59 34
0000003b: 5a 35
0000003e: 31 57
00000041: 32 58
00000044: 34 59
00000047: 35 5a
```

Die erste Spalte bezeichnet die hexadezimale Adresse innerhalb der Daten, an welcher eine Differenz gefunden wurde. In der zweiten ist der Wert der ersten Datei und in der dritten der Wert der zweiten Datei angegeben (wiederum hexadezimal). Eine Tabelle für die Umrechnung von hexadezimalen in dezimale Zahlen finden Sie im Anhang.

```
fc /4 test test.neu > prn
```

Die Unterschiede werden auf den Drucker ausgegeben. Nach einem Unterschied in beiden Dateien müssen mindestens 4 Zeilen übereinstimmen. Folgendes erscheint am Drucker:

```
***** test
DATEIA
A
B
C
D
E
F
G
H
I
M
N
O
P
***** test.neu
DATEIB
A
B
C
G
H
I
J
1
2
P
*****
```

```
***** test
```

```
V
```

```
W
```

```
***** test.neu
```

```
V
```

```
4
```

```
5
```

```
W
```

```
*****
```

```
fc test test.neu > differenz
```

Die Unterschiede werden in der Datei DIFFERENZ auf der Diskette/Festplatte gespeichert. Die Datei enthält folgende Angaben:

```
***** test
```

```
DATEIA
```

```
A
```

```
***** test.neu
```

```
DATEIB
```

```
A
```

```
*****
```

```
***** test
```

```
C
```

```
D
```

```
E
```

```
F
```

```
G
```

```
***** test.neu
```

```
C
```

```
G
```

```
*****
```

```
***** test
```

```
I
```

```
M
```

```
N
```

```
O
```

```
P
```

***** test.neu

I

J

1

2

P

***** test

U

W

***** test.neu

U

4

5

W

FIND

externer Befehl

Zweck

FIND sucht in einer oder mehreren Dateien nach einer angegebenen Zeichenfolge und zeigt jede Zeile mit einer entsprechend gefundenen Zeichenkette am Bildschirm an. Bei FIND handelt sich um einen Filter.

Kommandoaufbau

(d:)(weg)find (/x) "zeichenkette" (dateiname ...)

/x	/c	zähle Zeilen (es wird nur angezeigt, wie oft die angegebene Zeichenkette in einer Datei gefunden wurde)
	/i	Groß- und Kleinschreibung nicht berücksichtigen (nur bei Zenith)
	/n	Anzeigen der Zeilennummer vor jeder gefundenen Zeile
	/v	Anzeigen aller Zeilen, in welchen die Zeichenkette nicht vorkommt

Testdaten

Dateiname: KAPITEL1

Der Kurs MS-DOS-Einführung wird auf Personal-Computer-Systemen durchgeführt. Dabei gelangen Computer der Baureihe Z-100 pc zum Einsatz.

Die Computer der Baureihe Z-100 PC sind voll kompatibel Modelle.

Dateiname: KAPITEL2

Der Hersteller bietet zum Markstandard kompatible Computer (Z-100 PC) und S-100-Computer (Baureihe Z-100 S) an.

Dateiname: KAPITEL3

Für Personal Computer der kompatiblen Klasse stehen am Weltmarkt eine sehr große Anzahl von Programmen zur Verfügung.

Der Hersteller bietet seinen Kunden hier ein System an (Z-100 PC).

Beispiele

```
find /n /i "Z-100 PC" kapitel1 kapitel2 kapitel3
```

Es werden die Zeilen mit der Zeichenkette "Z-100 PC" in den 3 Texten gesucht. Dabei soll Groß- und Kleinschreibung nicht berücksichtigt werden und bei der Ausgabe die Zeilennummer mit angezeigt werden (die Zeichenkette wird im dritten Text nicht gefunden, da diese auf 2 Zeilen verteilt ist). Am Bildschirm erscheint folgende Anzeige:

```
----- kapitel1
[3]BAUREIHE Z-100 PC ZUM EINSATZ.
[4]DIE COMPUTER DER BAUREIHE Z-100 PC SIND VOLL KOMPATIBLE
```

```
----- kapitel2
[2](Z-100 PC) UND S-100-COMPUTER (BAUREIHE Z-100 S) AN.
```

```
----- kapitel3
```

```
dir | find /n "DAT" | sort > prn
```

Es sollen nur Dateien angezeigt werden, welche im Namen die Zeichenkette "DAT" nicht haben. Dabei wird das Inhaltsverzeichnis nach Dateinamen aufsteigend sortiert ausgegeben. Die Ausgabe soll am Drucker erfolgen. Am Drucker erscheint:

```
[30]SORT    DAT    3228  18.07.85  11.32
[33]DATE    TXT    1024  16.07.85  22.43
```

```
find /n /i /v "Z-100 PC" kapitel1 kapitel2 kapitel3
```

Es werden die Zeilen mit der Zeichenkette "Z-100 PC" in den 3 Texten gesucht. Dabei soll Groß- und Kleinschreibung nicht berücksichtigt und bei der Ausgabe die Zeilennummer mit angezeigt werden (die Zeichenkette wird im dritten Text nicht gefunden, da diese auf 2 Zeilen verteilt ist). Es sollen nur die Zeilen angezeigt werden, in welchen die Zeichenkette nicht vorkommt. Am Bildschirm erfolgt folgende Anzeige:

----- kapitel1

[1]DER KURS MS-DOS-EINFÜHRUNG WIRD AUF PERSONAL-COMPUTER-
[2]SYSTEMEN DURCHGEFÜHRT. DABEI GELANGEN COMPUTER DER
[5]MODELLE.

----- kapitel2

[1]DER HERSTELLER BIETET ZUM MARKTSTANDARD KOMPATIBLE COMPUTER

----- kapitel3

[1]FÜR PERSONAL COMPUTER DER KOMPATIBLEN KLASSE STEHEN AM WELT-
[2]MARKT EINE SEHR GROßE ANZAHL VON PROGRAMMEN ZUR VERFÜGUNG.
[3]DER HERSTELLER BIETET SEINEN KUNDEN HIER EIN SYSTEM AN (Z-100
[4]PC).

FOR

interner Befehl

Zweck

Wiederholt mehrmals einen MS-DOS-Befehl.

Kommandoaufbau

Bei Eingabe am Bildschirm:

for %variable in (wert) do befehl

Bei Verwendung in Batch-Dateien:

for %%variable in (wert) do befehl

variable	Darf jedes Zeichen sein, außer 0 bis 9
wert	Dateinamen oder Zeichenketten
befehl	MS-DOS-Befehl

Beispiele

```
for %a in (*.bat) do type %a > prn
```

Im ersten Teil des Befehls wird %a definiert. Dann werden alle Dateien mit dem File-Typ BAT herausgesucht. Anschließend werden der Reihe nach alle BAT-Dateien mit **type %a** (%a wird durch die Namen der gefundenen Dateien ersetzt) auf den Drucker (> **prn**) ausgegeben (und zwar mit ihrem Inhalt).

```
for %z in (config.sys autoexec.bat) do type %z
```

Dabei werden die Dateien CONFIG.SYS und AUTOEXEC.BAT mit ihrem Inhalt der Reihe nach auf dem Bildschirm angezeigt.

Bemerkungen

Wenn der Befehl FOR nicht in einer Batch-Datei verwendet wird, darf nicht %, sondern nur % angegeben werden.

FORMAT

externer Befehl

Zweck

FORMAT bereitet eine Diskette/Festplatte zur Aufnahme von MS-DOS (bzw. Dateien) vor. Es werden das Inhaltsverzeichnis initialisiert, die sogenannte "File-Allocation-Table" angelegt und der "Boot-Loader" kopiert. Disketten werden vom Hersteller immer unformatiert zur Verfügung gestellt, d.h., der Computer kann diese Disketten nicht verarbeiten. FORMAT zerstört alle eventuell vorhandenen Daten unwiederbringlich auf einer Disk!!

Nach dem Formatieren kann einer Diskette oder Festplatte ein Name zugeordnet werden (ein sogenanntes Label).

Beachten Sie zu diesem Befehl auch die Kapitel 6 und 7 - hier werden die Aufteilung und das Inhaltsverzeichnis näher beschrieben.

Den Befehl nicht bei einem Netzwerk-Laufwerk verwenden.

Kommandoaufbau

(d:)(weg)format (d:) (/x...)

- | | | |
|----|----|--|
| /x | /b | erstellt Freiraum für das MS-DOS-Betriebssystem und (IO.SYS und MSDOS.SYS) den COMMAND-Processor (mit dem Befehl SYS kann später das System übertragen werden) |
| | /c | nicht formatieren, sondern nur das Inhaltsverzeichnis mit allen Dateien löschen (wenn mit /m ursprünglich formatiert wurde, muß /m hier mit angegeben werden) - (nur von Zenith) |
| | /4 | Diskette ein- (160-180 kB) oder zweiseitig (320-360 kB) in einem Laufwerk mit hoher Kapazität (1 MB) formatieren - die erstellten Disketten können jedoch nicht in Laufwerken mit einer niedrigeren Kapazität verwendet werden |
| | /8 | nur mit 8 Sektoren pro Spur formatieren (normal 9) - nur für MS-DOS-Version 2 |
| | /m | nur einseitig formatieren, statt normalerweise doppelseitig (nur von Zenith) |

- /l nur einseitig formatieren, statt normalerweise doppelseitig
- /n Alle Anfragen und Erklärungen des Programmes werden unterdrückt (sinnvoll, wenn FORMAT in einer BATCH-Datei verwendet wird) - nach Start des Programms beginnt sofort der Formatierungsvorgang (nur von Zenith)
- /s nach dem Formatieren wird automatisch MS-DOS kopiert (IO.SYS, MSDOS.SYS und COMMAND.COM) - damit kann der PC von der neuen Disk gestartet werden (bootbar)
- /v Die Diskette/Festplatte wird nach dem Formatieren auf Fehlerfreiheit überprüft - fehlerhafte Sektoren werden automatisch markiert und können nicht mehr benutzt werden (wird vom Programm angezeigt). Wenn Sie /V beim Formatieren einer Festplatte angeben, dauert der Vorgang erheblich länger.

Beispiele

`format`

Damit wird das Programm gestartet, und Sie werden am Bildschirm gefragt, welches Laufwerk formatiert werden soll (bzw. welche Diskette oder Festplatte-Partition).

`format a: /s`

Die Diskette in Laufwerk a: wird formatiert und MS-DOS übertragen (IO.SYS, MSDOS.SYS und COMMAND.COM).

`format a: /s /n /v`

Wie vorheriges Beispiel, nur wird hier der Formatiervorgang sofort gestartet (nur bei Zenith) und anschließend wird die Diskette auf Fehlerfreiheit geprüft.

```
format c: /s /v
```

Die Festplatte c: formatieren, MS-DOS (IO.SYS, MSDOS.SYS und COMMAND.COM) übertragen und auf Fehlerfreiheit überprüfen.

GOTO

externer Befehl

Zweck

GOTO führt einen unbedingten Sprungbefehl in einer BATCH-Datei aus. Kann nur innerhalb einer BATCH-Datei verwendet werden (siehe Kapitel 12).

Kommandoaufbau

goto marke

Beispiel

Eine BATCH-Datei wird mit dem Befehl

copy con da.bat

erstellt und enthält folgende Befehle:

```
if exist %1 goto xyz
goto yz
:xyz
echo Die Datei %1 ist vorhanden
goto ende
:yz
echo Die Datei %1 ist nicht vorhanden
:ende
```

Wenn Sie die BATCH-Datei mit

da command.com

ausführen, erhalten Sie folgende Meldungen am Bildschirm:

```
if exist command.com goto xyz
:xyz
echo Die Datei command.com ist vorhanden
goto ende
:ende
```

Wenn Sie die BATCH-Datei mit

da nicht.com

ausführen, erhalten Sie folgende Meldungen am Bildschirm:

```
if exist nicht.com goto xyz
goto yz
:yz
echo Die Datei nicht.com ist nicht vorhanden
:ende
```

Bemerkungen

Beachten Sie zu vorstehendem Beispiel auch die Befehle IF und ECHO.

Beachten Sie zu diesem Befehl auch das Kapitel 12 "BATCH-Befehle".

GRAFTABL

externer Befehl

Zweck

Lädt die Tabelle der ASCII-Zeichen (Werte 128 bis 255) in den Speicher für die Benutzung im Grafik-Mode (Farb-Grafik-Karte muß im PC installiert sein).

Kommandoaufbau

(d:)(weg)graffabl

Bemerkungen

Dieser Befehl ist vor allem notwendig, wenn Sie einen anderen als den Original-Zeichensatz (für USA) verwenden (z.B. für den deutschen KEY-BGR). Nur wenn dieser Befehl eingegeben wurde, werden z.B. die deutschen Umlaute im Grafik-Modus (MODE-Befehl beachten) richtig dargestellt. Der Befehl muß bei jedem Neustart des Computers erneut eingegeben werden.

Beachten Sie hierzu auch die Befehle **SELECT**, **KEYBxxxx** und **PSC** bzw. **GRAPHICS**.

GRAPHICS

externer Befehl

(nicht für Zenith - siehe PSC)

Zweck

GRAPHICS dient zur richtigen Ausgabe von Grafikzeichen und Sonderzeichen des Bildschirms auf den Drucker (Hardcopy-Funktion mit Shift-PrtSc).

Kommandoaufbau

(d:)(weg)graphics (printer-typ) (/x)

printer-typ	color1	Farbdrucker mit schwarzem Farbband
	color4	Farbdrucker mit vierfarbigem Farbband
	color8	Farbdrucker mit CMY vierfarbigem Farbband
	compact	Compact-Drucker
	graphics	Grafik-Drucker (Standarddrucker)

/x	/r	Es wird Schwarz als Schwarz und Weiß als Weiß gedruckt (wenn nicht angegeben, wird Schwarz als Weiß und Weiß als Schwarz gedruckt).
	/b	Die Hintergrundfarbe wird gedruckt (nur bei COLOR4 und COLOR8) - wenn nicht angegeben, wird die Hintergrundfarbe nicht gedruckt.

Der Printer-Typ und /x sind nicht bei allen Herstellern möglich bzw. unterschiedlich (die obigen Angaben gelten für IBM).

IF

interner Befehl

Zweck

IF macht eine Befehlsausführung von der Erfüllung einer Bedingung abhängig. Dieser Befehl wird vorallem in BATCH-Dateien (siehe auch Kapitel 12) verwendet.

Kommandoaufbau

if (not) bedingung befehl

bedingung ERROR LEVEL nummer
 zeichenkette1==zeichenkette2
 exist dateiname

Beispiele

Erstellt wird folgende BATCH-Datei (Befehl: copy con such.bat):

```
echo off
chkdsk %1:
if errorlevel 0 echo                      Der Ausgangscode des Programmes ist 0.
```

Aufruf der BATCH-Datei erfolgt z.B. mit

```
such c
```

Wenn der Fehlercode des Programmes **chkdsk** 0 ist, wird folgendes am Bildschirm angezeigt:

```
echo off                      Der Ausgangscode des Programmes ist 0.
```

Wenn der Fehlercode von **chkdsk** nicht 0 ist, wird nur der Text **echo off** am Bildschirm angezeigt.

Erstellt wird folgende BATCH-Datei (Befehl: **copy con loesche.bat**):

```
echo off
if "%1"==" " goto ende
del %1
echo
Die Datei %1 wurde gelöscht.
:ende
```

Wenn die BATCH-Datei mit

loesche

aufgerufen wird, erscheint folgende Anzeige am Bildschirm:

```
echo off
```

Wenn die BATCH-Datei mit

loesche da.bat

aufgerufen wird, erscheinen folgende Anzeigen am Bildschirm:

```
echo off
```

Die Datei da.bat wurde gelöscht.

Bemerkung

Die ERRORLEVEL-Nummern finden Sie im MS-DOS-Handbuch zu Ihrem System oder bei einzelnen Befehlen in diesem Buch.

JOIN

externer Befehl

Zweck

Temporär eine Laufwerks-Bezeichnung mit einem Sub-Directory vertauschen (wenn ein definiertes Sub-Directory angesprochen wird, verwendet das System die Daten im Laufwerk).

Der Befehl darf nicht mit Netzwerk-Laufwerken verwendet werden.

Die verwendete Laufwerks-Bezeichnung kann nicht verwendet werden, solange JOIN wirksam ist (Meldung: **invalid drive specification**).

Wenn JOIN wirksam ist, dürfen folgende Befehle nicht verwendet werden:

BACKUP
DISKCOMP
DISKCOPY
RESTORE

Bei JOIN dürfen Laufwerke, welche mit ASSIGN, ASGNPART oder SUBST umgeleitet wurden, nicht benutzt werden.

Kommandoaufbau

(d:)(weg)join
(d:)(weg)join d: (d:)weg
(d:)(weg)join d: /d

Beispiele

```
join a: c:\drivea
```

Der Laufwerksbezeichnung **a:** wurde der Sub-Directory-Name **drivea** zugewiesen. Wenn anschließend das Sub-Directory angesprochen wird, so erfolgt der Zugriff auf das Laufwerk **a:**. Das Laufwerk **a:** kann nicht mehr mit der Bezeichnung **a:** angesprochen werden. Der Zugriff auf das Laufwerk erfolgt z.B. mit

```
dir \drivea
```

Damit wird das Inhaltsverzeichnis von Laufwerk **a:** angezeigt.

```
join
```

Anzeigen der umgeleiteten Laufwerksnamen:

```
A: => C:\DRIVEA
```

```
join a:/d
```

Der ursprüngliche JOIN-Befehl wird zurückgesetzt und damit der Originalzustand wiederhergestellt, d.h., daß das Laufwerk **a:** wieder mit **a:** angesprochen werden kann. Der Sub-Directory-Eintrag **drivea** bleibt bestehen, wird aber wie jeder andere behandelt.

Bemerkung

Wenn Sie die Fehlermeldung **Not enough memory** erhalten, verringern Sie die Anzahl der BUFFER in CONFIG.SYS und starten Ihren PC erneut (siehe dazu auch das Kapitel 12). Sollte die Fehlermeldung immer noch erscheinen, so kann der Befehl nicht ausgeführt werden.

KEYBxxxxexterner Befehl**Zweck**

Laden einer anderen Tastaturbelegung (für verschiedene Länder/Sprachen)
 - nach dem Systemstart ist die Tastatur mit Standard-ASCII (USA) belegt.

Kommandoaufbau

(d:)(weg)keybxxxx

Folgende Tabelle gilt nur für Zenith:

Land	kompletter Zeichensatz (256 Zeichen)	ASCII Zeichensatz (128 Zeichen)
xxxx = Australien	US	AUS
Dänemark	DA	ADA
Niederlande	US	AUS
Großbritannien	UK	AUK
USA	US	AUS
Finnland	SW	ASW
Frankreich	FR	AFR
Kanada (Electronic)	CANE	nicht vorhanden
Kanada (Selectric)	CANS	nicht vorhanden
Deutschland	GR	AGR
Italien	IT	AIT
Norwegen	NO	ANO
Spanien	SP	ASP
Schweden	SW	ASW
Schweiz (französ.)	CHF	nicht vorhanden
Schweiz (deutsch)	CHG	nicht vorhanden

Für alle anderen Hersteller gilt (nur 256-Zeichen-Code):

KEYBUK	=	Großbritannien
KEYBGR	=	Deutschland
KEYBFR	=	Frankreich
KEYBIT	=	Italien
KEYBSP	=	Spanien

Bemerkungen

Wenn eine Tastaturbelegung geladen wird, so benötigt diese permanent 2 kB zusätzlich an Speicherplatz. Durch Laden eines Zeichensatzes für ein anderes Land wird der vorher geladene Zeichensatz im Speicher gelöscht.

Wenn Sie die Tasten **Ctrl-Alt-F1** betätigen, so wird auf den Zeichensatz für USA umgeschaltet, ohne daß dieser geladen werden muß. Zurückschalten auf den geladenen Zeichensatz **Ctrl-Alt-F2**.

Beachten Sie bitte auch den Befehl **SELECT**.

LABEL

externer Befehl

Zweck

Erstellen, Ändern oder Löschen eines Namens für eine Diskette oder Festplatte.

Kommandoaufbau

(d:)(weg)label

(d:)(weg)label (d:)(name)

Beispiele

Zum erstenmal werden Sie beim Formatieren nach dem Namen zu einer Disk befragt. Wenn Sie statt einen Namen einzugeben nur Return drücken, wird kein Name vergeben. Der Name einer Disk darf maximal 11 Zeichen lang sein.

label a:

Es wird am Bildschirm der alte Name z.B. wie folgt angezeigt:

Volume in drive A has no label

Ferner werden Sie anschließend nach dem neuen Namen der Disk befragt:

Enter desired volume label (11 characters, RETURN for none)?

Wenn die Disk bereits einen Namen hat (Anzeige z.B.: **Volume in drive A is NAME**), und Sie drücken bei obiger Abfrage nur Return (für Löschen des Namens), so erscheint noch folgende Frage am Bildschirm:

Delete current volume label (Y/N)?

Wenn Sie den Namen löschen wollen, so drücken Sie nur den Buchstaben y und Return.

label a:neu-name

Die Diskette in Laufwerk a: erhält den neuen Namen NEU-NAME.

Bemerkungen

Der Befehl darf nicht verwendet werden, wenn die Befehle ASSIGN oder SUBST angewendet werden. Ferner darf er nicht für ein Netzwerk verwendet werden.

MKDIR oder MD

interner Befehl

Zweck

Mit diesem Befehl kann ein neues Sub-Directory angelegt werden (siehe auch Kapitel 8).

Kommandoaufbau

mkdir (d:)weg
md (d:)weg

Beispiele

mkdir \text
oder
mkdir text

Es wird ein neues Sub-Directory mit dem Namen TEXT angelegt (der Eintrag ist in dem aktuellen Directory sichtbar). Dazu werden das eingegebene Datum und die Uhrzeit gespeichert. Ferner werden in dem neuen Sub-Directory die Einträge

- . dahinter verbirgt sich der Name dem neuen Directory
- .. dahinter verbirgt sich der Name des hirarchisch vorher liegenden Directorys

angelegt.

Mit dem Befehl befinden Sie sich nicht automatisch in dem neuen Sub-Directory (dazu ist der Befehl **CHDIR** bzw. **CD** notwendig).

```
md \text\programm
```

Es wird ein neues Sub-Directory mit dem Namen PROGRAMM in dem Sub-Directory TEXT angelegt (TEXT muß ein Sub-Directory-Eintrag in dem aktuellen Directory sein).

```
md .
```

und

```
md ..
```

Mit diesen beiden Befehlen erhalten Sie die Fehlermeldung **Unable to create directory**. Diese Namen können bei dem Befehl MKDIR nicht verwendet werden.

MODE

externer Befehl

Zweck

MODE bestimmt die Betriebsart für den Bildschirm, monochrom oder graphisch; bestimmt das Protokoll für den parallelen oder seriellen Ausgang; Ausgabe vom parallelen zum seriellen Ausgang umleiten.

Mit MODE kann das gleiche durchgeführt werden wie mit CONFIGUR (diesen Befehl gibt es nur bei Zenith). Nur können hier die Werte bei MODE nicht gespeichert werden, und CONFIGUR ist leichter bedienbar, da es mit der sogenannten Menütechnik arbeitet und Sie sehr übersichtlich am Bildschirm abgefragt werden.

Wenn Sie MODE nach jedem Start Ihres Computers benötigen, sollten Sie diesen Befehl in die Datei AUTOEXEC.BAT einfügen (siehe Kapitel 12).

Kommandoaufbau und Beispiele

(d:)(weg)mode ? zeigt eine Beschreibung zum Befehl am Bildschirm an (nur bei Zenith)

(d:)(weg)mode lpt#:(n) (,(m)) (,(p))

	parallelen Ausgang konfigurieren
#	Nummer des parallelen Ausgangs (1-3, falls vorhanden)
n	Zeichen pro Zeile (80 oder 132)
m	Zeilen pro Inch (6 oder 8)
p	Drucken wiederholt versuchen, d.h. gegebenenfalls warten, bis dieser drucken kann (kann nur mit Ctrl-Break beendet werden)

mode lpt2:132,,p

Der Drucker am 2. parallelen Ausgang soll bis 132 Zeichen pro Zeile und 6 Zeilen pro Inch drucken. Ferner soll das System gegebenenfalls warten, bis der Drucker bereit ist (keine Fehlermeldung oder Abbruch, wenn der Drucker z.B. nicht eingeschaltet ist).

mode lpt2:

Der Drucker am 2. parallelen Ausgang wird umgestellt auf 80 Zeichen pro Zeile und 6 Zeilen pro Inch. Ferner erfolgt Abbruch, wenn der Drucker nicht bereit ist (Ausgang auf Standardwerte zurückstellen).

mode (n)((,m))((,t))((,s))

Bildschirm konfigurieren

(die einzelnen Werte werden nicht von allen Herstellern geliefert)

n bezeichnet den Bildschirm-Modus:

40 40 Zeichen pro Zeile

80 80 Zeichen pro Zeile (Standardwert)

BW40 40 Zeichen pro Zeile und Schwarzweiß-Modus

BW80 80 Zeichen pro Zeile und Schwarzweiß-Modus

CO40 40 Zeichen pro Zeile und Farb-Modus

CO80 80 Zeichen pro Zeile und Farb-Modus

GR40 40 Zeichen pro Zeile und Grafik (320*200)

GR80 80 Zeichen pro Zeile und Grafik (640*200)

MONO 80 Zeichen pro Zeile und Monochrom-Modus

m R Bildschirm nach rechts verschieben

L Bildschirm nach links verschieben

t ruft eine Testzeile zur Bildschirmjustage auf - kann nur zusammen mit **m** verwendet werden. Sie können nach dem Aufruf die Anzeige am Bildschirm, je nach Eingabe, nach rechts bzw. links verschieben.

s bestimmt den Scroll-Modus

0 Software-Scroll-Mode (Standardwert)

1 Jump-Scroll-Mode

nur möglich bei **n** = 80, BW80, CO80, GR40, GR80

2 Smooth (Hardware) Scroll-Mode

nur möglich bei **n** = GR40, GR80

```
mode co40,r,t
```

Es wird eine Zeile mit 40 Zahlen am Bildschirm um 1 Spalte nach rechts versetzt angezeigt. Sie können die Anzeige jetzt noch weiter nach rechts verschieben.

```
mode COM#:baud-rate(,(parity)((daten-länge)((stop-bit)((p))))
```

	seriellen Ausgang konfigurieren
#	Nummer des seriellen Ausgangs (1 oder 2, falls vorhanden)
baud-rate	Übertragungsgeschwindigkeit (110, 150, 300, 600, 1200, 2400, 4800 oder 9600)
parity	bestimmt die Art der Parity (odd (O), <u>even (E)</u> oder no parity (N))
daten-länge	(7 oder 8 Bit)
stop-bit	(1 oder 2 - <u>Standardwert 2 bei 110 Baud, sonst 1</u>)
p	Ausgabe wiederholt versuchen, auch wenn keine Antwort kommt (kann dann nur mit Ctrl-Break beendet werden)

```
mode com2:9600,e,8,2
```

Der serielle Ausgang **COM2** wird auf eine Übertragungsgeschwindigkeit von 9600 Baud, even parity, 8 Bit und 2 Stop-Bit eingestellt.

```
mode com2:300,,,,p
```

Der serielle Ausgang **COM2** wird auf eine Übertragungsgeschwindigkeit von 300 Baud, even parity, 7 Bit und 1 Stop-Bit eingestellt. Ferner wird eine Übertragung nicht abgebrochen, wenn der Empfänger der Daten nicht empfangsbereit ist.

```
mode lpt#:=com#
```

leitet Daten vom parallelen zum seriellen Ausgang um (LPT1 bis LPT3 und COM1 bis COM2)

Bevor Sie diesen Befehl verwenden, konfigurieren Sie bitte den seriellen Ausgang (der Parameter `p` ist unbedingt mit anzugeben). Um diese Umleitung wieder aufzuheben, muß nur der parallele Ausgang neu konfiguriert werden.

MORE

externer Befehl

Zweck

Mit MORE können Daten seitenweise auf den Bildschirm ausgegeben werden. Bei dem Befehl handelt es sich um einen Filter (siehe Kapitel 9).

Kommandoaufbau

(d:)(weg)more
(d:)(weg)befehl | more

Beispiele

`type test | more`

Die Datei TEST wird am Bildschirm angezeigt - nach jeder vollen Bildschirmseite stoppt die Ausgabe, bis Sie eine beliebige Taste betätigt haben (in der letzten Zeile am Bildschirm erscheint -- More --).

`more < test`

Mit diesem Befehl erreichen Sie den gleichen Effekt wie vorher. Diese Form der Anwendung ist effektiver, da die Daten nicht wie im ersten Beispiel von MS-DOS zwischengespeichert werden (siehe Kapitel 9). Der Ablauf erfolgt schneller.

NODEBUG

externer Befehl
(nur von Zenith)

Zweck

Zenith-PC's haben im ROM ein Monitorprogramm gespeichert, das auch verschiedene Funktionen zur Fehlersuche beinhaltet (Debugging). Mit dem Programm NODEBUG kann dieses Programm bzw. diese Funktionen ausgeschaltet werden. Wenn ein von Ihnen aufgerufenes Programm laufend in dieses Programm im ROM "springt", so verwenden Sie bitte den Befehl NODEBUG nach dem Einschalten Ihres Zenith-PC's.

Kommandoaufbau

(d:)(weg)nodebug

Beispiel

`nodebug`

Mit Eingabe des Befehls wird beim Zenith-PCs das ROM-Programm "abgeschaltet".

PART

externer Befehl

(nur von Zenith - andere verwenden den Befehl FDISK/Kapitel 6)

Zweck

Beachten Sie vor allem auch die Kapitel 6 und 7 mit einer Einführung in diesen Befehl.

PART teilt eine oder mehrere Festplatten in einzelne logische Laufwerke auf (teilt die Festplatte ein). Ferner wird das Startlaufwerk festgelegt, dort-hin muß mit dem Befehl FORMAT auch das MS-DOS übertragen werden.

Bei Zenith kann eine Festplatte in bis zu 4 logische Laufwerke aufgeteilt werden, welche mit MS-DOS oder einem anderen Betriebssystem (z.B. Concurrent) benutzt werden können.

Kommandoaufbau

(d:)(weg)part

Beachten Sie bitte das Kapitel 6 und das Kapitel 17 im MS-DOS-Handbuch von Zenith für eine detaillierte Beschreibung.

PATH

interner Befehl

Zweck

PATH benennt die Wege, die zu einem Verzeichnis führen, in dem ein externer Befehl gesucht werden soll. Ein Befehl, welcher in einem anderen als dem gerade aktuellen Directory steht, wird bei Eingabe gefunden, wenn das Directory, in dem der Befehl steht, mit PATH zugewiesen wurde. Beachten Sie zu diesem Befehl auch das Kapitel 8.

Kommandoaufbau

path (d:)(weg(:(d:)weg)...))

Beispiele

path

Zeigt die aktiven Vereinbarungen mit PATH an. Beispiel:

```
PATH=\MSDOS;\AUFTRAG\FIBU;\TEXT
```

path ;

Löscht alle mit PATH getroffenen Vereinbarungen.

```
path c:\system:a:\sub\texte
```

Damit können die Befehle des Sub-Directory **system** in Laufwerk **c:** und **texte** (Sub-Directory von **sub**) in Laufwerk **a:** in jedem anderen Directory verwendet werden.

PAUSE

interner Befehl

Zweck

PAUSE unterbricht die Ausführung einer Batch-Datei (siehe Kapitel 12) und gibt einen Text am Bildschirm aus, in dem Sie aufgefordert werden, zur Fortsetzung der Arbeit eine beliebige Taste zu drücken. Die Abarbeitung einer Batch-Datei kann durch Betätigung von Ctrl-Break abgebrochen werden – Sie werden noch befragt, ob Sie wirklich abbrechen wollen:

Terminate batch job (Y/N)?

Kommandoaufbau

pause (text)

Beispiel

Folgende Befehle in einer Batch-Datei:

echo	Legen Sie in das Laufwerk A eine neue Diskette ein
pause	Durch Eingabe der Tastenkombination <Ctrl-Break> können Sie abbrechen

erzeugen am Bildschirm die Anzeigen (echo ist eingeschaltet):

ECHO Legen Sie in das Laufwerk A eine neue Diskette ein
PAUSE Durch Eingabe der Tastenkombination <Ctrl-Break> können Sie abbrechen

Strike a key when ready . . .

PREP

externer Befehl
(nur von Zenith)

Zweck

PREP initialisiert eine Festplatte und bereitet diese dadurch zur Aufnahme von Daten vor. Dieses Programm wird normalerweise nicht benötigt, da eine neue Festplatte vom Hersteller bereits in dieser Weise vorbereitet ist (ein derartiger Befehl wird von fast keinem Hersteller mitgeliefert).

Beachten Sie, daß PREP **alle** Daten auf Ihrer Festplatte unwiederbringlich zerstört! Beachten Sie dazu auch die Kapitel 6 und 7.

Das Programm sollte angewandt werden, wenn wiederholt Fehler beim Zugriff auf die Festplatte auftreten und mit dem Programm DETECT diese nicht mehr behoben werden können.

Kommandoaufbau

(d:)(weg)prep

PRINT

externer Befehl

Zweck

Mit PRINT kann eine Textdatei ausgedruckt werden, während ein anderes Programm bearbeitet wird.

Kommandoaufbau

(d:)(weg)print

(d:)(weg)print dateiname1 (/x) (dateiname2 (/x) ...)

/x	/a	Abbruch - löscht die bezeichnete(n) Datei(en) aus der Drucker-Warteschlange (alle bezeichneten Dateien bis /s). Diesen Parameter gibt es nur bei Zenith.
	/b:#	Definition des Puffers von PRINT. Der Standardwert (ohne Angabe dieses Parameters) ist 512 Bytes. Die maximale Angabe ist 16384 Bytes. Nur beim ersten Aufruf von PRINT angeben.
	/c:#	Die angegebene Datei #-mal (1 bis 255) drucken.
	/d:dev	Normalerweise wird alles am Drucker (Gerätename PRN) ausgegeben. Hier können Sie einen anderen Gerätenamen (dev) angeben (z.B. COM1 oder CON).
	/f	Seitenvorschub (fügt ein leeres Blatt zwischen die zu druckenden Dateien ein). Nur bei Zenith.
	/l:#	Setzt den linken Rand am Drucker auf die Spalte #. Nur bei Zenith.
	/m:#	Spezifiziert, wieviel Zeit (Takte) PRINT für den Druck verwenden darf, bis eine weitere von Ihnen gleichzeitig durchgeführte Tätigkeit kommt. Der Standardwert beträgt 2. Angaben von 1 bis 255 sind erlaubt. Nur beim ersten Aufruf von PRINT angeben.

<code>/n:#</code>	Spezifiziert, wieviel Zeit (Takte) während der Arbeit von PRINT für die andere von Ihnen gleichzeitig durchgeführte Tätigkeit zur Verfügung steht. Der Standardwert ist 8. Es kann ein Wert zwischen 1 und 255 angegeben werden. Nur beim ersten Aufruf von PRINT angeben. Nur von Zenith bei anderen Herstellern <code>/s:#</code> .
<code>/p:#</code>	Seitenlänge #-Zeilen
<code>/q:#</code>	Spezifiziert, wie viele Dateien Sie in die PRINT-Warteschlange schreiben können. Der Standardwert ist 10 (möglich ist ein Wert von 1 bis 32). Nur beim ersten Aufruf von PRINT angeben.
<code>/r:#</code>	Den rechten Rand am Drucker auf die Spalte # setzen.
<code>/s</code>	Siehe <code>/n:#</code> . Bei Zenith: Gibt die bezeichnete Datei und alle folgenden der Befehlszeile in die Warteschlange ein (bis <code>/a</code> oder Befehlszeilenende).
<code>/t</code>	Ende - löscht alle Dateien aus der Warteschlange von PRINT.
<code>/u:#</code>	Setzt die PRINT-Wartezeit. Spezifiziert die Anzahl der Takte, bis der Drucker frei ist. Währenddessen wird die von Ihnen gleichzeitig durchgeführte Tätigkeit bearbeitet. Der Standardwert ist 1 (Werte von 1 bis 255 sind erlaubt). Nur beim ersten Aufruf von PRINT angeben.

Beispiele

`print`

Es werden alle Dateinamen angezeigt, welche sich in der Warteschlange befinden (wenn keine vorhanden, wird angezeigt, daß die Warteschlange leer ist). Beispiel:

```
B:KAPITEL .1 is currently being printed
B:KAPITEL .2 is in queue
B:KAPITEL .3 is in queue
```



```
print a:test /c2 /f c:test.neu /f c:test.alt /f kapitel1 /f
```

Die Dateien TEST auf der Diskette in Laufwerk a:, TEST.NEU, TEST.ALT in Laufwerk c: und KAPITEL1 im aktuellen Laufwerk sollen ausgedruckt werden. Dabei wird nach jedem Text ein Seitenvorschub eingefügt. Vom Text TEST sollen 2 Exemplare gedruckt werden.

```
print c:test.alt /a
```

Die Datei TEST.ALT in Laufwerk c: soll nicht gedruckt werden (aus der Drucker-Warteschlange löschen).

```
print c:kapitel2 /s /f c:kapitel3 /f
```

Die Dateien KAPITEL2 und KAPITEL3 werden zur Drucker-Warteschlange hinzugefügt (jeweils einen Seitenvorschub zwischen den zu druckenden Dateien einfügen).

PROMPT

interner Befehl

Zweck

PROMPT ändert das Systemanforderungszeichen.

Kommandoaufbau

prompt \$(text)

— Zeilenvorschub
b das Zeichen |
d Systemdatum (Tagesdatum)
e Escape-Sequenz
g das Zeichen >
h Schritt zurück
l das Zeichen <
n das aktuelle Laufwerk (z.B. A oder B)
p Name des aktuellen Directory
s Leerschritt
t Uhrzeit
v Nummer der MS-DOS-Version

Beispiele

prompt \$n:

Der neue Prompt ist C:

prompt \$_Zeit = \$t\$_Datum = \$d\$_

Der neue Prompt ist die Anzeige der Uhrzeit und in einer neuen Zeile das Datum. Der Cursor steht in der nächsten Zeile. Beispiel:

Zeit = 11:54:34:56

Datum = 21-09-85

```
prompt $n$g
```

Damit wird der "normale" Prompt erzeugt.

Mit dem Befehl PROMPT können auch die Funktionstasten neu belegt und andere Tätigkeiten durchgeführt werden. Dazu muß in der Datei CONFIG.SYS die Zeile **DEVICE=ANSI.SYS** angegeben werden (siehe Kapitel 13).

```
prompt $e{l7m$n:$e{l
```

Der Prompt wird invers dargestellt und, anschließend wieder auf normale Darstellung umschalten.

```
prompt $e{l33;44m  
prompt $n$g
```

Wenn Sie einen Farbmonitor haben, ist anschließend der Hintergrund blau und die Schrift (Vordergrund) gelb.

```
prompt $e{l0;60;"DIR B: ";13p  
prompt
```

Die Funktionstaste F2 (60) erhält den Inhalt "DIR B: Return". Wenn Sie anschließend F2 betätigen, wird der Befehl DIR B: ausgeführt. Die Taste kann bis zum nächsten Systemstart nicht mehr umbelegt werden.

Bemerkungen

Im Rahmen dieses Buches konnte auf den Befehl PROMPT nur wenig eingegangen werden. Beachten Sie hierzu Ihr Handbuch zum Computer. Weitere interessante Möglichkeiten werden laufend in den einschlägigen Computerzeitschriften veröffentlicht.

PSC

externer Befehl

(nur von Zenith - bei anderen Herstellern GRAPHICS)

Zweck

PSC dient zur richtigen Ausgabe von Grafikzeichen und Sonderzeichen des Bildschirms auf den Drucker.

Kommandoaufbau

pscdruckername

druckername	ids	IDS Prism Drucker
	oki	Okidata Drucker
	ts315	Transtar 315 Color Drucker
	p920	Printek 920 Drucker
	mx80	Epson-Drucker mit Grafikfähigkeit
	mpi	MPI Drucker
	hptj	Hewlett Packard Thinkjet Drucker

Damit werden Grafik- und Sonderzeichen bei Betätigen der Tastenkombination **Shift-PrtSc** richtig auf den Drucker ausgegeben.

Beispiel

pscmx80

Bei Benutzung eines EPSON-Druckers mit Graphikoption kann eine Grafik vom Bildschirm anschließend durch Betätigung von **Ctrl-PrtSc** auf den Drucker "kopiert" werden (benötigt unter Umständen mehrere Minuten). In dieser Zeit können Sie am Computer nicht weiterarbeiten.

RDCPM

externer Befehl
(nur von Zenith)

Zweck

RDCPM liest CP/M-formatierte Disketten und kopiert CP/M-Dateien auf MS-DOS-Format um (CP/M ist ebenfalls ein Betriebssystem).

Kommandoaufbau

(d:)(weg)rdcpm
(d:)(weg)rdcpm ? Zeigt Beschreibung zum Programm am Bildschirm
(d:)(weg)rdcpm dir d:(dateiname)/(x)
(d:)(weg)rdcpm (s:)ursprungsdateiname (d:)(zieldateiname)/(x)

dir	zeigt Inhaltsverzeichnis der CP/M-Diskette
/x /z	für Zenith-CP/M-Disketten
	(ohne Angabe IBM-PC-CP/M-86)

Beispiele

```
rdcpm dir a: /z
```

Es werden alle Dateien der CP/M-Diskette (Zenith-Format) in Laufwerk a: am Bildschirm angezeigt.

```
rdcpm dir a:*.cmd /z
```

Es werden alle Dateien mit dem File-Typ CMD der CP/M-Diskette (Zenith-Format) in Laufwerk a: am Bildschirm angezeigt.

```
rdcpm a:*.dat
```

Es werden alle Dateien mit dem File-Typ DAT der CP/M-Diskette (IBM-PC-Format für CP/M-86) in das MS-DOS-Format umkopiert und mit dem gleichen Namen auf das aktuelle Laufwerk umkopiert.

Bemerkungen

Bitte beachten Sie, daß keine Programme umkopiert werden sollten, da diese nicht unter MS-DOS lauffähig sind!

Folgende CP/M-Formate können von RDCPM gelesen werden:

Zenith	CP/M-85	ss/dd	48 Tpi	5,25 Zoll
	CP/M-85	ds/dd	48 Tpi	5,25 Zoll
	CP/M-86	ss/dd	48 Tpi	5,25 Zoll
	CP/M-86	ds/dd	48 Tpi	5,25 Zoll
IBM	CP/M-86	ss/dd	48 Tpi	5,25 Zoll
	CP/M-86	ds/dd	48 Tpi	5,25 Zoll

RECOVER

externer Befehl

Zweck

RECOVER erneuert eine Datei oder Diskette mit fehlerhaften Sektoren.

RECOVER darf nicht bei einem Netzwerk-Laufwerk verwendet werden.

Kommando Aufbau

(d:)(weg)recover d:

(d:)(weg)recover (d:)(weg)dateiname

Beispiele

```
recover a:test.neu
```

Die Datei A:TEST.NEU wird auf fehlerhafte Sektoren überprüft. Fehlerhafte Sektoren werden ausgeschlossen (es wird immer die alte und neue Dateigröße angezeigt). Wenn Fehler festgestellt wurden, fehlen anschließend im Text diese Teile. Sollte es sich um ein Programm handeln, ist dieses i.d.R. anschließend nicht mehr brauchbar (daher immer eine Kopie aller Daten und Programme mit möglichst aktuellem Stand fertigen).

```
recover a:
```

Die Diskette in Laufwerk a: wird auf fehlerhafte Sektoren überprüft.

Bemerkungen

Beachten Sie bitte, daß das Programm alle bisherigen Dateinamen löscht und neue Namen vergibt (also vorher eine Kopie fertigen bzw. das Inhaltsverzeichnis ausdrucken, um ggfs. den ursprünglichen Zustand wieder herstellen zu können).

Die Dateien erhalten in der bisherigen Reihenfolge die neuen Namen:

FILE0001.REC
FILE0002.REC
FILE0003.REC

REM

interner Befehl

Zweck

Befehl für Kommentarzeilen in Batch-Dateien (siehe auch Kapitel 12).

Kommandoaufbau

rem (text)	
. (text)	Kurzform des Befehls

Beispiel

```
rem
rem Diese Batch-Datei formatiert und überprüft neue Disketten
rem Der Name dieser Datei ist NEU.BAT
rem
pause Legen Sie eine neue Diskette in Laufwerk A ein
rem Die neue Diskette in Laufwerk A wird formatiert
format a: /s
rem Jetzt wird das Inhaltsverzeichnis der neuen Diskette angezeigt
dir a:
rem Jetzt wird die neue Diskette auf Fehler geprüft
chkdsk a:
rem Ende der Batch-Datei NEU.BAT
```

Bemerkungen

In einer Kommentarzeile (Befehl REM) dürfen nur Komma, Tabulator oder Leerzeichen als Separator verwendet werden (z.B. kein Punkt).

RENAME oder REN

interner Befehl

Zweck

Vorhandenen Dateien/Programmen einen neuen Namen zuordnen.

Kommandoaufbau

```
ren (d:)(weg)dateiname-alt dateiname-neu  
rename (d:)(weg)dateiname-alt dateiname-neu
```

Beispiele

```
ren test.alt versuch.alt
```

Anschließend hat die Datei TEST.ALT den Namen VERSUCH.ALT.

```
ren a:\sub\test.neu versuch.neu
```

Die Datei TEST.NEU in der Sub-Directory sub auf der Diskette in Laufwerk a: wird in VERSUCH.NEU geändert. Dabei muß nur beim alten Namen der genaue Weg und die Laufwerksbezeichnung angegeben werden.

```
ren *.txt *.doc
```

Dabei erhalten alle Dateien mit dem File-Typ TXT den neuen Typ DOC zugewiesen.

```
ren b:text*.* ??s*.*
```

Mit diesem Befehl werden nur Teile von Dateinamen verändert. Bearbeitet werden alle Dateinamen in Laufwerk b:, welche mit TEXT beginnen. Die ersten beiden Zeichen (TE) bleiben unverändert. Der dritte Buchstabe wird in S umgewandelt. Der Rest der Dateinamen bleibt bestehen. Damit wurde also nur das dritte Zeichen in den Namen verändert.

Bemerkung

Benennen Sie niemals gekaufte Programme um, es sei denn, Sie werden extra dazu aufgefordert.

RESTORE

externer Befehl

Zweck

RESTORE holt die mit BACKUP gesicherten Dateien auf Diskette oder Winchester zurück. Siehe auch Beschreibung zu BACKUP in diesem Kapitel.

Kommandoaufbau

(d:)(weg)restore
(d:)(weg)restore ? Programmbeschreibung anzeigen (nur Zenith)
(d:)(weg)restore d: ((d:)(weg)dateibez+(d:)(weg)dateibez ...)) (/x ...)

dateibez Namen der zurückzukopierenden Datei(en) (*. * für alle)

/x	/b:(datum)	Dateien, welche auf der Sicherungsdisk das angegebene oder ein Datum davor besitzen, werden zurückkopiert (wird das Datum nicht angegeben, so wird dafür das System- bzw. Tagesdatum verwendet). Nur von Zenith.
	/d:(datum)	Dateien, welche auf der Sicherungsdisk das angegebene oder ein Datum danach besitzen, werden zurückkopiert (wird das Datum nicht angegeben, so wird dafür das System- bzw. Tagesdatum verwendet). Nur von Zenith.
	/e:dateiname(+dateiname...)	Alle hier angegebenen Dateien werden nicht zurückkopiert. Nur von Zenith.
	/f	Normalerweise werden beim Zurückkopieren auch Sub-Directories wieder angelegt, falls diese nicht vorhanden sind (die mit BACKUP und dem Parameter /G oder /S gesicherten Dateien und Directories). Mit dem Parameter /F werden alle Dateien von der Sicherungsdisk in die aktuelle Directory zurückkopiert (die gesicherte Directory-Struktur wird nicht beachtet). Nur von Zenith.

/p	Wenn eine Datei das Attribut Read-Only (ATTRIB) hat oder seit dem letzten BACKUP verändert wurde, werden Sie am Bildschirm befragt, ob eine Rückkopie erfolgen soll (gilt auch für IO.SYS und MSDOS.SYS bzw. bei IBM IBMBIO.COM und IBMDOS.COM).
/q	Vor dem Zurückkopieren wird jede Datei angezeigt und Sie werden befragt, ob diese zurückkopiert werden soll (nur bei Zenith).
/r	Immer wenn Sie eine Aktion durchführen müssen (z.B. Diskettenwechsel), ertönt ein akustisches Signal (nur bei Zenith).
/s	Es wird nur die spezifizierte Sub-Directory zurückkopiert.
/t	Es werden nur die Dateien kopiert, welche das System- bzw. Tagesdatum besitzen (nur von Zenith).
/v	Nach dem Kopieren wird überprüft, ob die Datei fehlerfrei übertragen wurde (nur von Zenith).

Beispiele

```
restore a: d:*.*/d:01.09.85
```

Es werden vom Laufwerk a: alle Dateien nach Laufwerk d: zurückkopiert, welche das Datum vom 01.09.85 oder später tragen.

```
backup c:*,* a: /e:*.sys*.com /f /s  
restore a: c:*,* /p
```

Es werden alle Dateien aus allen Directories im Laufwerk c: mit Ausnahme von *.sys und *.com auf die Diskette(n) in Laufwerk a: gesichert (Disketten werden vor dem kopieren automatisch formatiert).

Mit dem RESTORE-Befehl werden alle Dateien zurückkopiert. Zu Dateien, welche das Read-Only-Attribut besitzen, werden Sie vor dem Kopiervorgang befragt, ob dieser durchgeführt werden soll (gilt auch für Dateien, welche sich seit dem BACKUP-Befehl verändert haben).

```
restore a: c:*. * /q
```

Bevor alle Dateien von Laufwerk a: kopiert werden, zeigt Ihnen RESTORE die Dateinamen am Bildschirm einzeln an und fragt dazu, ob diese kopiert werden soll oder nicht.

```
backup a:*.bas+c:kap.*+d:bas???.com f: /r /s
restore f: a:\BASIC\*. * /r /s
```

Es werden von

Laufwerk a: die Dateien mit dem File-Typ BAS,

von c: alle mit dem File-Namen KAP und beliebigem
File-Typ,

von d: alle Dateien, welche mit BAS beginnen und
maximal 6-stellige File-Namen, sowie den Typ COM
besitzen

nach Laufwerk f: gesichert.

Bei jeder von Ihnen notwendigen Aktion (Diskettenwechsel oder Eingabe) ertönt ein akustisches Signal. Die Sub-Directory-Namen bzw. die Struktur wird ebenfalls gesichert.

Mit RESTORE werden alle Dateien wieder zurückgeholt (alles auf Laufwerk a:), welche aus der Sub-Directory BASIC von Laufwerk a: gesichert wurden (/s). Bei jeder von Ihnen notwendigen Aktion ertönt ein akustisches Signal.

Bemerkungen

Wenn Sie den Befehl RESTORE in einer Batch-Datei verwenden, kann mit IF ERROR LEVEL anschließend der Fehlercode abgefragt werden:

- | | | |
|---|---|--|
| 0 | = | RESTORE fehlerfrei beendet |
| 1 | = | keine Datei gefunden |
| 2 | = | File-Sharing-Konflikt (Netzwerk)
einige oder alle Dateien nicht zurückkopiert |
| 3 | = | Sie haben RESTORE abgebrochen (durch Ctrl-Break) |
| 4 | = | RESTORE durch Fehler abgebrochen |

RMDIR oder RD

interner Befehl

Zweck

Löschen von Sub-Directory-Einträgen (ohne die dort enthaltenen Dateien). Vorher müssen Sie alle in der Sub-Directory gespeicherten Dateien und eventuelle weitere Directories löschen (sonst wird der Befehl RD nicht ausgeführt). Beachten Sie hierzu das Kapitel 8.

Kommandoaufbau

rmdir (d:)weg
rd (d:)weg

Beispiele

md versuch	Sub-Directory VERSUCH anlegen
cd versuch	VERSUCH wird aktuelle Directory
md versuch2	Sub-Directory VERSUCH2 in VERSUCH anlegen
cd ..	aktuelle Directory wird die hierarchisch vor VERSUCH liegende
rd versuch	Löschen der Sub-Directory VERSUCH - nicht möglich, da darin noch VERSUCH2 vorhanden ist. Fehlermeldung: Invalid path, not directory, or directory not empty
rd versuch\versuch2	Löschen der Sub-Directory VERSUCH2
rd versuch	Sub-Directory VERSUCH kann jetzt gelöscht werden, da diese leer ist

SEARCH

externer Befehl
(nur von wenigen Herstellern)

Zweck

SEARCH sucht Dateien in den Directories (siehe Kapitel 8). Wenn alle durchsucht werden sollen, müssen Sie sich beim Start des Befehls in der Root-Directory befinden (es wird in der Hierarchie immer nur vorwärts gesucht).

Kommandoaufbau

(d:)(weg)search (dateiname) (/x)

/x	/c	Nur die aktuelle Directory durchsuchen
	/d	Die durchsuchten Directory-Namen anzeigen
	/t	Datei- und Directory-Namen versetzt anzeigen, damit die Struktur sichtbar wird

Beispiele

```
search *.sys /t
```

Es werden alle SYS-Dateien angezeigt (mit Sub-Directory-Namen) - die Namen werden versetzt angezeigt, damit die Struktur erkannt werden kann.

```
IO      SYS
MSDOS   SYS
MSDOS
<DIR>
  UDISK  SYS
  DISK2
<DIR>
TEXT
<DIR>
UDISK   SYS
```



```
search *.sys /t > prn
```

Wie vorher, jedoch erfolgt die Ausgabe auf den Drucker.

Bemerkung

Es sollte hier kein Filter verwendet werden, da dies zu keiner korrekten Abarbeitung durch diesen führen würde. Wenn SEARCH 32 Directory-Ebenen durchlaufen hat und noch weitere vorhanden sind, erhalten Sie die Fehlermeldung:

```
Tree structure too deep. Ignoring below this point
```

In diesem Falle müssen Sie mit SEARCH in einer tieferen Ebene beginnen und für einzelne "Zweige" separat ausführen.

SELECT

externer Befehl

Zweck

Herstellen eines Tastatur-Layout und des Formates für Datum/Uhrzeit für ein bestimmtes Land. SELECT führt das Programm DISKCOPY aus und erstellt die Dateien CONFIG.SYS (mit COUNTRY=#) und AUTO-EXEC.BAT (mit KEYBxxxx). Beachten Sie dazu die Kapitel 12 und 13. Der Befehl darf nicht bei einem Netzwerk-Laufwerk verwendet werden.

Kommandoaufbau

(d:)(weg)select xxxx yyy

Land	Landes-Code (yyy) 8-Bit	Tastatur-Code (xxxx) 7-Bit
Australien	061 US	AUS
Belgien	032 FR	AFR
Kanada, französisch (Electronic)	001 CANE	nein
Kanada, französisch (Selectric)	001 CANS	nein
Dänemark	045 DN	ADA
Finnland	358 SW	ASW
Frankreich	033 FR	AFR
Deutschland	049 GR	AGR
Israel	972 nein	nein
Italien	039 IT	AIT
Niederlande	031 US	AUS
Norwegen	047 NO	ANO
Spanien	034 SP	ASP
Schweden	046 SW	ASW
Schweiz (franz.)	041 CHF	nein
Schweiz (deutsch)	041 CHG	nein
Großbritannien	044 UK	AUK
USA	001 US	AUS

Der 7-Bit-Kode wird nur von Zenith geliefert. Obige Tabelle ist in diesem Umfang nur bei Zenith möglich. Die meisten Hersteller liefern nur für

USA	001	US
Frankreich	033	FR
Spanien	034	SP
Italien	039	IT
Großbritannien	044	UK
Deutschland	049	GR

Beispiele

```
select GR 049
```

Als erstes wird das Programm DISKCOPY von SELECT aufgerufen und eine Kopie der Diskette wird erstellt. Anschließend werden die beiden Dateien CONFIG.SYS und AUTOEXEC.BAT auf der neuen Diskette erstellt.

Bemerkungen

Das Programm DISKCOPY muß in der aktuellen Directory vorhanden sein. Der Vorgang ist nur mit Disketten möglich.

SET

interner Befehl

Zweck

SET setzt den Wert einer Zeichenkette gleich dem Wert einer anderen Zeichenkette zur Benutzung in Programmen (kommt vor allem in Zusammenhang mit BATCH-Dateien zur Anwendung).

Kommandoaufbau

```
set  
set (zeichenkette1=(zeichenkette2))
```

Beispiele

```
set
```

Es wird der im Environment-Bereich (Zwischenspeicher) gespeicherte Teil angezeigt (nach dem Systemstart befindet sich dort nur der Name des Befehlsinterpreters COMMAND.COM). Beispiel für Anzeige:

```
COMSPEC=C:\COMMAND.COM  
PATH=\MSDOS;\TEXT\ABT1  
VAR=TEXT
```

```
set var=text
```

Batch-Datei TEST.BAT:

```
rem Dies ist eine Beispieldatei  
echo off  
echo In der folgenden Zeile wird die Variable VAR gezeigt:  
echo %var%
```

Nach Aufruf von TEST.BAT wird angezeigt:

```
REM Dies ist eine Beispieldatei
```

```
ECHO OFF
```

```
In der folgenden Zeile wird die Variable VAR gezeigt:
```

```
TEXT
```

```
set TTY=VT100
```

Damit wurde der Name des Gerätetreibers TTY auf VT100 geändert.

```
set TTY=
```

Damit wird die Variable TTY wieder gelöscht

Bemerkungen

Sollten Sie die Fehlermeldung Out of Environment Space erhalten, so ist nicht mehr genügend Speicher für Variablen vorhanden! Dieser Speicher hat nur eine auf 127 Zeichen begrenzte Größe.

SHARE

externer Befehl

Zweck

Installiert den sogenannten File-Sharing-Support, wenn Sie ein Netzwerk einsetzen.

Kommandoaufbau

(d:)(weg)share

(d:)(weg)share /x

/x /f:#

Definition der Größe des Speichers für Informationen für Dateien, welche im Netzwerk verwendet werden. Für jede verwendete (offene) Datei wird Speicher für den vollen Dateinamen und 11 weitere Stellen benötigt. Der Standardwert ist 2.048.

/l:#

Der Standardwert ist 20. Es handelt sich hierbei um die Anzahl der sogenannten "Locks" (Dateizugriffsschutz). Wenn der Wert über 20 gesetzt wird, muß auch /F angegeben werden.

Bemerkungen

Weitere Informationen finden Sie im Handbuch zur Netzwerk-Software von MicroSoft "Networks Manager's Guide" und im Anhang zu diesem Buch.

SHIFT

interner Befehl

Zweck

SHIFT erhöht die Zahl der Variablen in einer Batch-Datei auf mehr als 10. Beachten Sie dazu auch das Kapitel 12.

Kommandoaufbau

shift

Beispiel

Batch-Datei LOESCHE.BAT:

```
rem Dies ist die Datei LOESCHE.BAT
:anfang
if "%1"==" " goto ende
del %1
shift
goto anfang
:ende
```

Beispiel für den Aufruf der Datei LOESCHE.BAT:

```
loesche datei1 datei2 datei3
```

Beispiel für die Anzeigen bzw. Aktionen:

```
rem Dies ist die Datei LOESCHE.BAT
:anfang
if "datei1"==" " goto ende
del datei1
shift
goto anfang
:anfang
if "datei2"==" " goto ende
del datei2
shift
goto anfang
:anfang
if "datei3"==" " goto ende
del datei3
shift
goto anfang
:anfang
if ""==" " goto ende
:ende
```

Die zum Befehl eingegebenen Dateinamen werden der Reihe nach bearbeitet und im vorliegenden Beispiel gelöscht. Der Befehl SHIFT "holt" also immer den nächsten Namen aus der von Ihnen eingegebenen Befehlszeile.

SHIP oder SHIPDISK

externer Befehl

(die Programmnamen sind bei den einzelnen Herstellern unterschiedlich)

Zweck

Vor dem Transportieren eines Computers mit einer Winchester muß dieser Befehl eingegeben werden. Dabei wird der Schreib/Lesekopf in eine sogenannte Parkposition für den Transport gebracht.

Dieser Befehl muß vor jedem Transport eingegeben werden, da sonst die Winchester beschädigt werden kann und damit nicht mehr brauchbar ist (keine Garantie)!

Kommandoaufbau

(d:)(weg)ship

Bemerkungen

Es gibt Computer mit Winchester, bei welchen der Befehl nicht notwendig ist, da beim ausschalten automatisch der Schreib/Lesekopf arretiert wird (erkundigen Sie sich bei Ihrem Lieferanten).

SORT

externer Befehl

Zweck

Sortieren von Daten. Es handelt sich bei SORT um einen Filter. Die Daten können auf- oder absteigend sortiert werden. Beachten Sie bitte das Kapitel 9.

Kommando Aufbau

(d:)(weg)sort (/x) (< dateiname1) (> dateiname2)

/x	/r	sortiert absteigend
	/+#	sortiert die Datenzeilen ab der Spalte #

Beispiele

```
sort < unsort.asc
```

Der Inhalt der Datei UNSORT.ASC wird aufsteigend sortiert am Bildschirm angezeigt.

```
sort < unsort.asc > sort.asc
```

Der Inhalt der Datei UNSORT.ASC wird aufsteigend sortiert in die Datei SORT.ASC übertragen.

```
dir c: | sort
```

Das Inhaltsverzeichnis des Laufwerks c: wird aufsteigend sortiert am Bildschirm angezeigt.

```
dir c: | sort > prn
```

Das Inhaltsverzeichnis des Laufwerks c: wird aufsteigend sortiert am Drucker ausgedruckt.

```
dir c: | sort /r /+14 | more
```

Das Inhaltsverzeichnis des Laufwerks c: wird absteigend nach Dateigrößen (die Anzahl der Bytes bzw. die Größe jeder Datei, steht im Inhaltsverzeichnis ab der 14. Spalte) sortiert auf den Bildschirm ausgegeben. Nach jeder vollgeschriebenen Bildschirmseite stoppt die Ausgabe, bis eine beliebige Taste gedrückt wurde.

SUBST

externer Befehl

Zweck

Einem Sub-Directory wird eine Laufwerksbezeichnung zugewiesen. Die Befehle

ASGNPART
ASSIGN
BACKUP
DISKCOMP
DISKCOPY
FORMAT
JOIN
LABEL
PART
RESTORE

dürfen nicht verwendet werden. SUBST darf nicht für ein Netzwerk-Laufwerk verwendet werden.

Kommandoaufbau

(d:)(weg)subst
(d:)(weg)subst d: (d:)weg
(d:)(weg)subst d: /x

/x /d Die mit SUBST erstellte Zuweisung löschen

Beispiele

```
subst f: a:\auftrag\fibu
```

Der Sub-Directory FIBU wird die Laufwerksbezeichnung f: zugewiesen. Wenn Sie z.B. den Befehl dir f: benutzen, erhalten Sie das Inhaltsverzeichnis der Sub-Directory FIBU.

Bemerkungen

Normalerweise sind nur die Laufwerksbezeichnungen bis e: möglich. Wenn Sie in der Datei CONFIG.SYS mit dem Befehl LASTDRIVE arbeiten, können Namen bis maximal z: vergeben werden (siehe Kapitel 13).

SYS

externer Befehl

Zweck

SYS kopiert die Systemdateien IO.SYS (bzw. IBMBIO.COM) und MS-DOS.SYS (bzw. IBMDOS.COM) auf eine Diskette/Winchester. Beachten Sie bitte, daß auf der Empfänger-Diskette/-Winchester auf den ersten Sektoren ausreichend Platz vorhanden sein muß, sonst wird zwar das System unter Umständen kopiert, es kann aber beim Booten nicht ordnungsgemäß geladen werden. Der erste Bereich (Sektoren) ist meist zu klein, wenn vorher eine ältere Version von MS-DOS dort gespeichert war.

Der Befehl gilt nicht für Netzwerk-Laufwerke.

Kommandoaufbau

(d:)(weg)sy s d:

Beispiel

sys c:

Das System wird vom aktuellen Laufwerk auf die Winchester (Laufwerk c:) übertragen.

TIME

interner Befehl

Zweck

Anzeigen bzw. ändern der Uhrzeit.

Kommandoaufbau

time

time hh(:mm(:ss(.cc)))

hh	Stunde	mögliche Werte von 00 bis 23
mm	Minute	mögliche Werte von 00 bis 59
ss	Sekunde	mögliche Werte von 00 bis 59
cc	hunderstel Sekunde	mögliche Werte von 00 bis 99

Beispiele

time

Es wird die aktuelle Uhrzeit angezeigt und Sie können eine neue Uhrzeit eingeben.

time 12:18

Die neue Uhrzeit ist 12 Uhr und 18 Minuten (0 Sekunden und 0 hundertstel Sekunden) - das System setzt mit dieser Uhrzeit fort.

time 0

Setzen der Zeit auf 0. Damit kann die Zeit für eine bestimmte Tätigkeit oder Ablauf festgestellt werden.

TREE

externer Befehl

Zweck

TREE zeigt das strukturierte Verzeichnis einer Diskette oder Winchester mit an.

Kommandoaufbau

(d:)(weg)tree ? Befehlsbeschreibung wird angezeigt (nur Zenith)
(d:)(weg)tree (d:)(/x)

/x /f Zu den Namen der einzelnen Sub-Directories werden auch die darin enthaltenen Dateinamen angezeigt.

Beispiele

tree

Es werden alle Sub-Directory-Namen angezeigt.

tree /f | more

Es werden alle Directories und die darin enthaltenen Dateinamen angezeigt. Nach jeder angezeigten Bildschirmseite wird die Anzeige gestoppt, bis Sie eine beliebige Taste betätigen.

tree /f > prn

Es werden alle Verzeichnisse und die darin enthaltenen Dateinamen ausgedruckt.

Beispiel für eine Ausgabe mit dem Parameter /f:

DIRECTORY PATH LISTING FOR VOLUME MS-DOS 3-1

Files: TREE.COM
 SELECT.COM
 MODE.COM
 SORT.EXE

Path: \MSDOS

Sub-directories: None

Files: ASSIGN.COM
 CHKDSK.COM

Path: \AUFTRAG

Sub-directories: FIBU

Files: None

Path: \AUFTRAG\FIBU

Sub-directories: None

Files: FIBU.EXE
 JOURNAL.DAT
 OP.DAT

TYPE

interner Befehl

Zweck

Den Inhalt einer Datei am Bildschirm anzeigen. Mit Ctrl-S kann mit der Anzeige jederzeit gestoppt bzw. fortgefahren werden.

Kommandoaufbau

type dateiname
type (d:)(weg)dateiname

Beispiele

type autoexec.bat

Der Inhalt der Datei AUTOEXEC.BAT wird am Bildschirm angezeigt.

type a:\sub\test.alt

Der Inhalt der Datei TEST.ALT in der Sub-Directory sub in Laufwerk a: wird angezeigt.

type text | more

Der Inhalt der Datei TEXT wird angezeigt (nach jeder vollen Bildschirmseite wartet die Anzeige, bis eine beliebige Taste gedrückt wird).

type autoexec.bat > prn

Der Inhalt der Datei AUTOEXEC.BAT wird ausgedruckt.

Bemerkungen

Statt dem Befehl **type autoexec.bat** kann alternativ **copy autoexec.bat con** verwendet werden (siehe Befehl COPY).

VER

interner Befehl

Zweck

Die Versionsnummer der Systemdateien IO.SYS (bzw. IBMBIO.COM) und MSDOS.SYS (IBMDOS.COM) werden angezeigt (also die Versionsnummer von MS-DOS).

Kommandoaufbau

ver

Beispiel

ver

Es erscheint z.B.:

IO.SYS Version 3.04

MS-DOS Version 3.1

oder

IBM Personal Computer DOS Version 3.10

VERIFY

interner Befehl

Zweck

Wenn mit diesem Befehl "Verify" eingeschaltet wird, erfolgt eine Überprüfung aller Disketten- und Winchester-Operationen auf korrekte Datenaufzeichnung bzw. -übertragung.

Standardmäßig nach dem Systemstart ausgeschaltet.

Kommandoaufbau

verify (on)
verify (off)

Beispiele

verify

Es wird der momentane Stand angezeigt (ob ein- oder ausgeschaltet):

VERIFY is off

verify on

Prüfen einschalten.

Bemerkungen

Wenn VERIFY eingeschaltet ist, benötigt z.B. das Kopieren von Dateien mehr Zeit.

VOL

interner Befehl

Zweck

Der Name einer Diskette oder Winchester wird angezeigt (Label). Der Name kann mit dem Befehl LABEL verändert oder gelöscht werden.

Kommandoaufbau

vol (d:)

Beispiel

vol a:

Der Name der Diskette in Laufwerk a: wird am Bildschirm angezeigt -

Beispiel:

Volume in drive A is MS-DOS 3-1

12 BATCH-Befehle

In einer Batchdatei werden nacheinander mehrere Befehle auf Betriebssystemebene abgearbeitet. Schreiben Sie diese Befehle vorher in eine Batchdatei, werden diese so abgearbeitet, als wenn sie direkt eingegeben werden. Damit ist es nicht erforderlich, wiederkehrende Routinen bei Bedarf neu zu schreiben.

MS-DOS erkennt eine Batchdatei am File-Typ BAT. Der Name muß eindeutig sein. Es darf nicht schon ein Programm mit dem gleichen File-Namen und dem File-Typ COM oder EXE vorhanden sein.

Beim Starten sieht das System nach, ob eine Datei mit dem Namen AUTO-EXEC.BAT vorhanden ist; wenn ja, so werden die darin enthaltenen Befehle automatisch abgearbeitet. Ist eine solche nicht im Root-Directory vorhanden, so werden Sie nur nach Datum und Uhrzeit befragt.

Der Ablauf einer Batch-Datei (bzw. Programm) kann jederzeit durch Betätigung von Ctrl-Break abgebrochen werden. Sie erhalten die Frage, ob Sie wirklich abbrechen wollen.

Terminate batch job (Y/N)?

Wenn Sie während des Ablaufs einer Batchdatei die Diskette wechseln und ein neuer Befehl ausgeführt werden soll, erhalten Sie die Meldung

Insert disk with batch file
and press any key when ready

Legen Sie dann die Diskette mit der Batchdatei ein und betätigen Sie eine beliebige Taste.

Kommandoaufbau

dateiname (variable1 (variable2 (...)))

dateiname	Batchdatei mit File-Typ BAT (kann innerhalb der Datei als %0 verwendet werden).
variable#	Dies können Dateinamen oder andere Angaben sein. In den Befehlen innerhalb der Datei als %1 bis %9 verwendbar .

Beispiele

Erstellen Sie bitte folgende Batch-Datei (mit dem Befehl COPY oder ED-LIN):

```
copy con autoexec.bat
```

```
oder
```

```
edlin autoexec.bat
```

```
echo off
```

```
date
```

```
time
```

```
keybgr
```

```
cls
```

(F6 und Return)

Starten Sie Ihren Computer neu, indem Sie die Tastenkombination (gleichzeitig drücken) Ctrl-Alt-Del betätigen. Anschließend erscheint am Bildschirm:

```
echo off
```

```
Current date is Fri 11-01-85
```

```
Enter new date:
```

```
Current time is 07:14:26:75
```

```
Enter new time:
```


Anschließend wird der deutsche Zeichensatz geladen und der Bildschirm gelöscht.

Erstellen Sie die folgende Batch-Datei mit dem Namen UEB.BAT:

```
copy %1 a:%2.neu
type %1
pause
type %2.neu
rem Dies ist die Batchdatei
type %0.bat
```

Geben Sie jetzt ein:

```
ueb diskette.asc diskette
```

Den Namen UEB können Sie in der Batch-Datei mit %0, DISKETTE.ASC mit %1 und DISKETTE mit %2 verwenden.

Im vorliegenden Beispiel erhalten Sie folgende Anzeigen:

```
A>copy diskette.asc a:diskette.neu
1 File(s) copied
```

```
A>type diskette.asc
```

Es wird der Inhalt der Datei DISKETTE.ASC am Bildschirm angezeigt.

```
A>pause
Strike a key when ready . . .
```

```
A>type DISKETTE.neu
```

Es wird der Inhalt der Datei DISKETTE.NEU am Bildschirm angezeigt.

A>rem dies ist die Batchdatei

A>type UEB.bat

Es wird der Inhalt der Datei UEB.BAT am Bildschirm angezeigt.

Die Batchdatei für die Überprüfung einer Datei, ob diese vorhanden ist, könnte wie folgt aussehen:

```
echo off
if exist %1 goto ja
echo Die Datei %1 ist nicht vorhanden
goto ende
:ja
echo Die Datei %1 ist vorhanden
:ende
```

Die Batchdatei AUTOEXEC.BAT könnte wie folgt aussehen:

```
echo off
keybgr
date
time
path=\msdos
cls
echo Programmauswahl:
echo -----
echo DIR                Anzeigen des Inhaltsverzeichnis
echo FIBU               Finanzbuchhaltung
echo LUGA               Lohn und Gehalt
echo TEXT              Textprogramm
echo -----
echo Programmaufruf durch Eingabe des Namens
rem
```

Beispiel für die Batchdatei zum Löschen von Dateien (LOE.BAT):

```
if "%1"==" " goto ende
:weiter
if "%1"==" " goto ausgang
if not exist %1 goto nicht
del %1
echo Datei %1 ist gelöscht
goto aus
:nicht
echo Datei %1 nicht vorhanden
goto weiter
:ende
echo Datei nicht angegeben
goto ausgang
:aus
shift
goto weiter
:ausgang
```

Drei Beispiele für den Aufruf von UEB.BAT:

ueb text

```
A>if "TEXT"==" " goto ende
A>:weiter
A>if "TEXT"==" " goto ausgang
A>if not exist TEXT goto nicht
A>del TEXT
A>echo Datei TEXT ist gelöscht
Datei TEXT ist gelöscht
A>goto aus
A>:aus
A>shift
A>goto weiter
A>if ""==" " goto ausgang
```

ueb

```
A>if ""==" goto ende
A>:ende
A>echo Datei nicht angegeben
Datei nicht angegeben
A>goto ausgang
```

loe niemals

```
A>if "NIEMALS"==" goto ende
A>:weiter
A>if "NIEMALS"==" goto ausgang
A>if not exist NIEMALS goto nicht
A>:nicht
A>echo Datei NIEMALS nicht vorhanden
Datei NIEMALS nicht vorhanden
A>goto weiter
A>:weiter
A>if ""==" goto ausgang
```

Normalerweise wird als erstes in der Batchdatei der Befehl `echo off` angegeben, damit nicht jeder Befehl angezeigt wird. Die Anzeige der Anspringpunkte (z.B. `:weiter`) erfolgt nicht.

Für die Programmauswahl beim Start des Computers kann ein einfaches Programm gewählt werden (AUTOEXEC.BAT):

```
:Author                Hans C. Nieder
:start
echo off
cls
echo
echo |
echo | 1 = MS-DOS
echo | 2 = Textprogramm
echo | 3 = Finanzbuchhaltung
echo |
echo | Wählen Sie 1, 2 oder 3
echo |
:schleife
eingabe
if errorlevel 49 goto eins
if errorlevel 50 goto zwei
if errorlevel 51 goto drei
goto schleife
:eins
cls
echo Programm beendet
goto ende
:zwei
if not exist TEXT.COM goto nicht
TEXT
goto start
:drei
if not exist FIBU.COM goto nicht
FIBU
goto start
:nicht
echo Das gewählte Programm ist nicht vorhanden
pause
goto start
:ende
```

Mit dem Programmende kann ein Wert welchen Sie an der Tastatur eintippen, in den Bereich geschrieben werden, welchen Sie anschließend mit errorlevel lesen und verarbeiten können. Das Programm EINGABE kann sehr einfach mit dem Befehl DEBUG erstellt werden:

```
A>debug
-neingabe.com                (Return)
-a                            (Return)
xxxx:0100 (Tab) mov (Tab) ax,0c07 (Return)
xxxx:0103 (Tab) int (Tab) 21    (Return)
xxxx:0105 (Tab) mov (Tab) ah,4c (Return)
xxxx:0107 (Tab) int (Tab) 21    (Return)
xxxx:0109                                (Return)
-rcx                          (Return)
:0000
9                              (Return)
-w                             (Return)
writing 0009 bytes
-q                             (Return)
A>
```

Weitere Beispiele finden Sie im Kapitel 11 bei den einzelnen Befehlen - z.B. bei

ECHO
FOR
GOTO
IF
PAUSE
REM
SHIFT

13 Die Konfiguration von MS-DOS 3.1

Die Datei MSDOS.SYS (bei IBM: IBMDOS.COM) ist der "Direktor" Ihres PC's. Sie erzeugt die Instruktionen für die zu erledigenden Arbeiten. Die Vorgänge laufen hardwareunabhängig ab (der hardwareabhängige Teil zur Bedienung der Peripherie ist IO.SYS bzw. IBMBIO.COM).

MSDOS.SYS ist ein intelligentes Interface für Anwendungsprogramme. Es verfügt über Programmteile zur Dateiverwaltung, Steuerung von Informationsblöcken bei Disketten-/Winchesterzugriff, usw.

Es können Aufrufe der Anwenderprogramme erkannt und bei Bedarf an IO.SYS zur Ausführung weitergeleitet werden.

MSDOS.SYS können Sie an Ihre Anforderungen anpassen (erweitern). Dazu wird die Datei CONFIG.SYS benötigt. Es handelt sich dabei um eine Textdatei (vergleichbar mit AUTOEXEC.BAT), die Sie mit EDLIN oder COPY erstellen können.

Während des Ladevorganges von MS-DOS wird nach der Datei CONFIG.SYS gesucht. Wenn diese gefunden wird, werden die darin enthaltenen Befehle ausgeführt.

Der Startvorgang in Einzelschritten:

- 1 Lesen des ersten Sektors von der Diskette/Winchester
- 2 Laden des Programmes BOOT-LOADER und starten
- 3 Der BOOT-LOADER lädt IO.SYS und übergibt die Kontrolle
- 4 IO.SYS initialisiert die Peripherie und lädt MSDOS.SYS
- 5 IO.SYS übergibt die Kontrolle an das Modul SYSINIT (stellt die Verbindung zwischen IO.SYS und MSDOS.SYS her), welches mit IO.SYS geladen wurde
- 6 SYSINIT sucht CONFIG.SYS und führt die darin enthaltenen Befehle aus
- 7 SYSINIT lädt COMMAND.COM (oder den in CONFIG.SYS angegebenen Befehls-Interpreter)
- 8 COMMAND.COM sucht AUTOEXEC.BAT und führt die darin enthaltenen Befehle aus (erst hier kann der Startvorgang durch Ctrl-Break abgebrochen werden)

13.1 Die CONFIG.SYS-Befehle

Die Befehle AVAILDEV und SWITCHAR des MS-DOS Version 2 gibt es nicht mehr, da sonst Konflikte im Netzwerk-Betrieb entstehen würden.

13.1.1 BREAK

Zweck

Damit kann die Programmunterbrechung modifiziert werden. Mit der Tastenkombination Ctrl-Break können Programme abgebrochen werden, wenn BREAK angeschaltet ist. Ein- und Ausgaben in Richtung Standardein- und -ausgang können unabhängig von BREAK mit Ctrl-Break unterbrochen werden.

Verschiedene Anwenderprogramme beachten die Einstellung bei BREAK nicht.

Kommandoaufbau

```
break=on  
break=off
```

Bemerkungen

Nach dem Start von MS-DOS kann mit dem internen Befehl abgefragt werden, ob BREAK ein- oder ausgeschaltet ist.

13.1.2 BUFFERS

Zweck

Legt die Anzahl der Zwischenspeicher im RAM Ihres Computers fest. Zwei werden generell angelegt (immer zum angegebenen Wert hinzuzurechnen). Es kann ein Wert zwischen 01 und 99 angegeben werden. Jeder Zwischenspeicher (BUFFER) belegt 528 Zeichen.

Der Speicher dient zur Zwischenspeicherung der Daten zwischen Diskette/Winchester und Programmen. Durch eine höhere Anzahl von BUFFER wird vielfach ein Programm schneller.

Kommandoaufbau

```
buffers=#
```

Beispiel

```
buffers=20
```

Es werden 20 zusätzliche Zwischenspeicher definiert. Dies ergibt mit den 2 Systemspeichern 22 je 528 Zeichen (damit wird 11.616 Bytes im RAM reserviert = ca. 11,34 kBytes).

13.1.3 COUNTRY

Zweck

Dieser Befehl setzt den Landeskodex. Damit wird das Format der Datumsdarstellung und Uhrzeit, sowie das Dezimalkomma festgelegt. Wenn der Befehl nicht angegeben wird, so benutzt das MS-DOS den Kodex für USA.

Kommandoaufbau

country=#

Land	Landeskodex
Australien	061
Belgien	032
Kanada	001
Dänemark	045
Deutschland	049
Finnland	358
Frankreich	033
Großbritannien	044
Israel	972
Italien	039
Niederlande	031
Norwegen	047
Schweden	046
Schweiz	041
Spanien	034
USA	001

13.1.4 FILES

Zweck

FILES definiert die Anzahl der maximal gleichzeitig geöffneten Dateien. Sollte diese Definition für ein Programm notwendig sein, so wird dies angegeben. MS-DOS benötigt selbst 5 Dateien. Daher wird eine Angabe von weniger als 5 nicht beachtet. Wird FILES nicht angegeben, so wird 8 automatisch definiert. Eine Angabe von 05 bis 99 ist möglich.

Kommandoaufbau

files=#

Bemerkungen

Die vom System benötigten 5 Standard-Dateien sind:

Nummer	Name	Funktion	Bemerkungen
0000	STDIN	Standardeingabe	kann umgeleitet werden
0001	STDOUT	Standardausgabe	kann umgeleitet werden
0002	STDERR	Fehlerausgabe	kann nicht umgeleitet werden
0003	STDAUX	Serielle Schnittstelle	kann nicht umgeleitet werden
0004	STDPRN	Ausgabe auf Drucker	kann nicht umgeleitet werden

13.1.5 FCBS**Zweck**

Dieser Befehl wird für den Netzbetrieb benötigt. Es sind für diesen Befehl detaillierte Kenntnisse in der Programmierung notwendig. Es werden mit FCBS Angaben zu offenbaren File-Control-Blocks gemacht.

Kommandoaufbau

fcbs=#1,#2

Bemerkungen

#1	Standard = 4	Möglich sind Werte von 1 bis 255
#2	Standard = 0	Möglich sind Werte von 0 bis 255

Der erste Wert muß größer oder gleich dem zweiten sein.

13.1.6 LASTDRIVE

Zweck

Mit diesem Befehl kann die maximal aktive Anzahl der logischen Laufwerke festgelegt werden. Wenn in Ihrem Computer mehr Laufwerke vorhanden sind als hier angegeben, wird der Befehl ignoriert. Standardwert ist Laufwerk E (= 5 Laufwerke). Laufwerkbezeichnungen von A bis Z sind möglich (Laufwerk A=1, B=2, C=3, usw.).

Beachten Sie hierzu auch den Befehl SUBST.

Kommando Aufbau

lastdrive=x

13.1.7 SHELL

Zweck

Hier kann eine Datei angegeben werden, die statt COMMAND.COM geladen werden soll. Wird dieser Befehl angegeben, so lädt SYSINIT die hier bezeichnete Datei statt COMMAND.COM (Befehlsinterpreter).

Kommando Aufbau

shell=dateiname

13.1.8 DEVICE

Zweck

Dieser Befehl kann einen oder mehrere zusätzliche Gerätetreiber als Erweiterung zu MSDOS.SYS laden. Damit wird der Datenfluß zwischen dem Computer und einem Gerät überwacht und bei Bedarf bearbeitet.

Zwei zusätzliche Gerätetreiber werden mit MS-DOS 3.1 mitgeliefert:

ANSI.SYS	erweiterte Bildschirm- und Tastaturfunktionen
VDISK.SYS	simuliert im Speicher ein Laufwerk (Virtuelle Disk)

Kommandoaufbau

device=gerätetreiber

13.1.8.1 ANSI.SYS

Damit kann eine auf eine erweiterte Bildschirmsteuerung gemäß dem ANSI-Standard zurückgegriffen werden (z.B. direkte Positionierung des Cursors an eine bestimmte Stellen am Bildschirm).

Beachten Sie dazu den Befehl PROMPT im Kapitel 11.

Die einzelnen Steuerkode finden Sie im Handbuch zu Ihrem Computer.

13.1.8.2 VDISK.SYS

Mit diesem Treiber kann ein Laufwerk im Speicher simuliert werden. Das Laufwerk wird wie jedes andere unter MS-DOS behandelt. Nach dem ausschalten des Computers ist dieses gelöscht!

Kommandoaufbau

device=(d:)(weg)vdisk.sys (b) (s) (d) (/e)

- | | |
|----|---|
| b | Größe des Laufwerks in kBytes (Standardwert ist 64 kB) |
| s | Sektorgröße (Standardwert ist 128 Bytes) nur 128, 256 oder 512 |
| d | Maximal mögliche Anzahl der Dateien (Standardwert ist 64) 2-512 |
| /e | Damit wird das Laufwerk im Speicherbereich über 1 MByte angelegt (nur möglich, wenn mehr als 1 MB Speicher im Computer vorhanden) |

13.1.9 Beispiel

Beispiel für eine Datei CONFIG.SYS:

```
BUFFERS=20  
FILES=12  
BREAK=ON  
COUNTRY=049  
LASTDRIVE=Z  
DEVICE=\MSDOS\ANSI.SYS  
DEVICE=\MSDOS\MOUSE.SYS  
DEVICE=VDISK.SYS 360 256 32
```

ANHANG I - MS-NET (Netzwerk)

Ein Netzwerk ist einfach ausgedrückt eine Punkt-zu-Punkt-Verbindung. Es werden mehrere Computer durch Kabel verbunden. Normalerweise benutzen die zusammengeschlossenen PCs eine gemeinsame Festplatte. Ein Computer im Verbund dient dabei als sogenannter Server (Zentrale).

MS-DOS 3.1 bietet die Betriebssystemgrundlage für den Einsatz eines Netzwerkes. Zum Betrieb ist jedoch z.B. noch MS-NET (Netzwerk-Betriebssoftware) notwendig. Jedoch ist auch andere Netzwerk-Software unter MS-DOS 3.1 lauffähig (z.B. Novell NetWare).

Zusätzlich ist noch eine entsprechende Hardware notwendig, welche die Hardware-Ebene der Kommunikation darstellt (z.B. von 3com).

Damit MS-DOS 3.1 mit seinen Netzwerk-Funktionen auch genutzt werden kann, muß die Netzwerk-Betriebssoftware diese unterstützen. Ferner können Programme, welche für Netzwerke erstellt wurden, die von MS-DOS zur Verfügung gestellten Funktionen nutzen.

Erstmals ist es mit MS-DOS 3.1 möglich, das sogenannte Record-Locking zu benutzen. Darunter versteht man, daß ein von einem Benutzer von einer Winchester gelesener Datensatz (z.B. Adresse) nicht von einem anderen benutzt werden kann. Hierbei unterscheidet man:

Nur Lesen möglich
Nur Schreiben möglich
Lesen und Schreiben möglich
usw.

Zu beachten ist, das die "Zentrale" (der Server) während des Betriebs von MS-NET nicht als Arbeitsplatz genutzt werden kann. Er verwaltet das gesamte Netzwerk.

Für das Verständnis von Netzwerken und deren Handhabung sind detaillierte Kenntnisse in MS-DOS und der Netzwerk-Architektur notwendig. Bitte beachten Sie hierzu die einschlägige Literatur.

ANHANG II – Begriffserklärungen

Im folgenden finden Sie einige wichtige Ausdrücke mit der Erläuterung.

Alphanumerisch

Buchstaben, Ziffern und Sonderzeichen mit denen Informationen bzw. Daten dargestellt werden.

ANSI

Abkürzung für AMERICAN NATIONAL STANDARD INSTITUTE. Dieses Institut setzt Industriesstandards. Am häufigsten tritt dieser Name in Verbindung mit Programmiersprachen auf. Beachten Sie auch den Bildschirmtreiber ANSI.SYS.

Anwenderprogramm

Diese stellen Programme für eine bestimmte Aufgabenstellung, wie Finanzbuchhaltung, Textverarbeitung, Auftragsverwaltung, usw., dar.

Ausgabe

Das sichtbarmachen oder speichern von Ergebnissen einer Computertätigkeit. Eine Ausgabe kann auf den Drucker, Bildschirm, die Diskette, usw., erfolgen.

Batch-Befehle

Sammlung von Befehlen in einer Datei mit dem File-Typ BAT. Wenn der File-Name als Befehl eingegeben wird, verarbeitet Ihr Computer Befehl für Befehl.

Booten

Das Laden und Starten des Betriebssystems.

Byte

Damit wird eine Speicherzelle eines Computers bezeichnet. Jedes Zeichen wird mit einem Byte dargestellt.

Centronics-Schnittstelle

Parallele Schnittstelle (meist Druckeranschluß), die sich als weitverbreiteter Standard eingeführt hat.

Cursor

Ein Zeichen (meißt ein blinkendes Unterstreichungszeichen), welches die aktuelle Position am Bildschirm anzeigt und die nächste Eingabe dargestellt wird.

Datei

Eine Sammlung von Daten (z.B. Text, Buchungen, Aufträge). Diese können erweitert oder geändert werden. Eine Datei ist mit einer Kartei vergleichbar.

Auch ein Programm kann als Datei bezeichnet werden.

Directory

Verzeichnis der Dateien. Es können mehrere auf einer Diskette/Festplatte vorhanden sein (Root-Directory/Sub-Directory). Das Directory befindet sich in einem festgelegten Bereich.

Diskette

Eine dünne, biegsame Plastikscheibe mit einer magnetisierten Oberfläche. Sie werden auch als Floppies oder Floppydisks bezeichnet.

Editor

Einfaches Textbearbeitungsprogramm für das erstellen oder verändern von Daten (z.B. Texten).

Festplatte

Die Bezeichnung bezieht sich auf eine bestimmte Technologie einer Festplatte. Es handelt sich um mehrere "harte Disketten", welche als Stapel in einem staubdichten Gehäuse untergebracht sind. Dieses System zeichnet sich durch eine höhere Schnelligkeit und Sicherheit aus.

Hard-Copy

Erstellen einer 1:1-Kopie vom Bildschirm auf den Drucker.

Hardware

Der "sichtbare" Teil Ihres Computers. Alle physischen Teile.

Peripherie

Alle an einen Computer angeschlossenen Geräte (auch Diskette, Winchester und Bildschirm).

RAM

Abkürzung für RANDOM ACCESS MEMORY. Ein flüchtiger Speicher. Nach dem Abschalten des Computers wird alles gelöscht. Beim Start wird das MS-DOS dorthin geladen (externe Befehle und Programme werden von der Diskette/Winchester in den RAM zur Ausführung geladen).

ROM

Abkürzung für READ ONLY MEMORY. Ein Speicher in welchem die darin enthaltenen Daten/Programme nur gelesen werden können. Die Informationen werden beim Ausschalten des Computers nicht gelöscht.

Software

Alle Programme, welche in einem Computer bearbeitet werden können. Erst durch die Software können Sie mit Ihrem PC arbeiten.

Speicher oder Arbeitsspeicher

Es handelt sich um den im Computer eingebauten Speicher, in welchem Informationen aufbewahrt werden (Programme und Daten). Siehe auch RAM und ROM.

ANHANG III – Umrechnungstabellen

HEX	DEZ	HEX	DEZ	HEX	DEZ	HEX	DEZ
0	0	0	0	0	0	0	0
1	4.096	1	256	1	16	1	1
2	8.192	2	512	2	32	2	2
3	12.288	3	768	3	48	3	3
4	16.384	4	1.024	4	64	4	4
5	20.480	5	1.280	5	80	5	5
6	24.576	6	1.536	6	96	6	6
7	28.672	7	1.792	7	112	7	7
8	32.768	8	2.048	8	128	8	8
9	36.864	9	2.304	9	144	9	9
A	40.960	A	2.560	A	160	A	10
B	45.056	B	2.816	B	176	B	11
C	49.152	C	3.072	C	192	C	12
D	53.248	D	3.328	D	208	D	13
E	57.344	E	3.584	E	224	E	14
F	61.440	F	3.840	F	240	F	15

Bild III.1 Umrechnungstabelle Hexadezimal/Dezimal

Beispiele für die Umrechnung von Zahlen:

HEX	HEX/DEZ	DEZ	DEZ/HEX
EB	E = 224 B = 11 ----- 235	235	224 = E 11 = B ----- EB
F3	F = 240 3 = 3 ----- 243	243	240 = F 3 = 3 ----- F3
7E	7 = 112 E = 14 ----- 126	126	112 = 7 14 = E ----- 7E

Hexadezimale Zahlen werden aufgeteilt. Wenn eine solche Zahl zwei Stellen hat, suchen Sie den ersten Wert in der vorletzten und den zweiten in der letzten Spalte und addieren diese, um den dezimalen Wert zu erhalten. Bei 4-stelligen Zahlen beginnen Sie in der viertletzten Spalte, usw.

Sie rechnen eine dezimale Zahl um, indem Sie die nächst niedrigere Zahl in der Tabelle suchen (und merken das erste Zeichen vor), diese subtrahieren und die nächstniedrige Zahl (bzw. Gleichheit) aus der Tabelle suchen.

Beispiel:

$$\text{Zahl } 2764 = 2764 - 2560(A) = 204 - 192(C) = 12 - 12(C) = 0 \Rightarrow \text{ACC}$$

$$\text{Zahl } 2050 = 2050 - 2048(8) = 2 - 0(0) = 2 - 2(2) = 0 \Rightarrow 802$$

ANHANG IV - Zeichendarstellungen/Code-Tabelle

N	U	L	U	00000000	@
S	O	00000000	A	00100000	1
H	00000000	00000000	00000000	00000000	2
S	00000000	00000000	00000000	00000000	3
T	00000000	00000000	00000000	00000000	4
X	00000000	00000000	00000000	00000000	5
E	00000000	00000000	00000000	00000000	6
F	00000000	00000000	00000000	00000000	7
O	00000000	00000000	00000000	00000000	8
T	00000000	00000000	00000000	00000000	9
E	00000000	00000000	00000000	00000000	10
N	00000000	00000000	00000000	00000000	11
X	00000000	00000000	00000000	00000000	12
F	00000000	00000000	00000000	00000000	13
O	00000000	00000000	00000000	00000000	14
T	00000000	00000000	00000000	00000000	15
E	00000000	00000000	00000000	00000000	16
N	00000000	00000000	00000000	00000000	17
X	00000000	00000000	00000000	00000000	18
F	00000000	00000000	00000000	00000000	19
O	00000000	00000000	00000000	00000000	20
T	00000000	00000000	00000000	00000000	21
E	00000000	00000000	00000000	00000000	22
N	00000000	00000000	00000000	00000000	23
X	00000000	00000000	00000000	00000000	24
F	00000000	00000000	00000000	00000000	25
O	00000000	00000000	00000000	00000000	26
T	00000000	00000000	00000000	00000000	27
E	00000000	00000000	00000000	00000000	28
N	00000000	00000000	00000000	00000000	29
X	00000000	00000000	00000000	00000000	30
F	00000000	00000000	00000000	00000000	31
O	00000000	00000000	00000000	00000000	32
T	00000000	00000000	00000000	00000000	33
E	00000000	00000000	00000000	00000000	34
N	00000000	00000000	00000000	00000000	35
X	00000000	00000000	00000000	00000000	36
F	00000000	00000000	00000000	00000000	37
O	00000000	00000000	00000000	00000000	38
T	00000000	00000000	00000000	00000000	39
E	00000000	00000000	00000000	00000000	40
N	00000000	00000000	00000000	00000000	41
X	00000000	00000000	00000000	00000000	42
F	00000000	00000000	00000000	00000000	43
O	00000000	00000000	00000000	00000000	44
T	00000000	00000000	00000000	00000000	45
E	00000000	00000000	00000000	00000000	46
N	00000000	00000000	00000000	00000000	47
X	00000000	00000000	00000000	00000000	48
F	00000000	00000000	00000000	00000000	49
O	00000000	00000000	00000000	00000000	50
T	00000000	00000000	00000000	00000000	51
E	00000000	00000000	00000000	00000000	52
N	00000000	00000000	00000000	00000000	53
X	00000000	00000000	00000000	00000000	54
F	00000000	00000000	00000000	00000000	55
O	00000000	00000000	00000000	00000000	56
T	00000000	00000000	00000000	00000000	57
E	00000000	00000000	00000000	00000000	58
N	00000000	00000000	00000000	00000000	59

D	016 020, 10 _A 00010000 ₁		04 _A 060, 30 _A 00110000 ₁	0	112 _A 'p' 160 _A 70 _A 01110000 ₁	P	144 _A 'p' 220 _A 90 _A 10010000 ₁		176 _A 'o' 260 _A 80 _A 01110000 ₁		208 _A 'p' 320 _A D0 _A 11010000 ₁		240 _A 'p' 360 _A F0 _A 11110000 ₁
C	017 021, 11 _A 00010001 ₁		04 _A 061, 31 _A 00110001 ₁	1	113 _A 'q' 161 _A 71 _A 01110001 ₁	Q	145 _A 'q' 221 _A 91 _A 10010001 ₁		177 _A '1' 261 _A B1 _A 01110001 ₁		209 _A 'q' 321 _A D1 _A 11010001 ₁		241 _A 'q' 361 _A F1 _A 11110001 ₁
E	018 022, 12 _A 00010010 ₁		05 _A 062, 32 _A 00110010 ₁	2	114 _A 'r' 162 _A 72 _A 01110010 ₁	R	146 _A 'r' 222 _A 92 _A 10010010 ₁		178 _A '2' 262 _A B2 _A 01110010 ₁		210 _A 'r' 322 _A D2 _A 11010010 ₁		242 _A 'r' 362 _A F2 _A 11110010 ₁
A	019 023, 13 _A 00010011 ₁		05 _A 063, 33 _A 00110011 ₁	3	115 _A 's' 163 _A 73 _A 01110011 ₁	S	147 _A 's' 223 _A 93 _A 10010011 ₁		179 _A '3' 263 _A B3 _A 01110011 ₁		211 _A 's' 323 _A D3 _A 11010011 ₁		243 _A 's' 363 _A F3 _A 11110011 ₁
K	020 024, 14 _A 00010100 ₁		05 _A 064, 34 _A 00110100 ₁	4	116 _A 't' 164 _A 74 _A 01110100 ₁	T	148 _A 't' 224 _A 94 _A 10010100 ₁		180 _A '4' 264 _A B4 _A 01110100 ₁		212 _A 't' 324 _A D4 _A 11010100 ₁		244 _A 't' 364 _A F4 _A 11110100 ₁
N	021 025, 15 _A 00010101 ₁		05 _A 065, 35 _A 00110101 ₁	5	117 _A 'u' 165 _A 75 _A 01110101 ₁	U	149 _A 'u' 225 _A 95 _A 10010101 ₁		181 _A '5' 265 _A B5 _A 01110101 ₁		213 _A 'u' 325 _A D5 _A 11010101 ₁		245 _A 'u' 365 _A F5 _A 11110101 ₁
S	022 026, 16 _A 00010110 ₁		05 _A 066, 36 _A 00110110 ₁	6	118 _A 'v' 166 _A 76 _A 01110110 ₁	V	150 _A 'v' 226 _A 96 _A 10010110 ₁		182 _A '6' 266 _A B6 _A 01110110 ₁		214 _A 'v' 326 _A D6 _A 11010110 ₁		246 _A 'v' 366 _A F6 _A 11110110 ₁
T	023 027, 17 _A 00010111 ₁		05 _A 067, 37 _A 00110111 ₁	7	119 _A 'w' 167 _A 77 _A 01110111 ₁	W	151 _A 'w' 227 _A 97 _A 10010111 ₁		183 _A '7' 267 _A B7 _A 01110111 ₁		215 _A 'w' 327 _A D7 _A 11010111 ₁		247 _A 'w' 367 _A F7 _A 11110111 ₁
A	024 030, 18 _A 00011000 ₁		05 _A 070 _A 38 _A 00111000 ₁	8	120 _A 'x' 170 _A 78 _A 01111000 ₁	X	152 _A 'x' 230 _A 98 _A 10011000 ₁		184 _A '8' 270 _A B8 _A 01111000 ₁		216 _A 'x' 330 _A D8 _A 11011000 ₁		248 _A 'x' 370 _A F8 _A 11111000 ₁
N	025 031, 19 _A 00011001 ₁		05 _A 071 _A 39 _A 00111001 ₁	9	121 _A 'y' 171 _A 79 _A 01111001 ₁	Y	153 _A 'y' 231 _A 99 _A 10011001 ₁		185 _A '9' 271 _A B9 _A 01111001 ₁		217 _A 'y' 331 _A D9 _A 11011001 ₁		249 _A 'y' 371 _A F9 _A 11111001 ₁
B	026 032, 1A _A 00011010 ₁		05 _A 072 _A 3A _A 00111010 ₁	:	122 _A ':' 172 _A 7A _A 01111010 ₁	Z	154 _A 'z' 232 _A 9A _A 10011010 ₁		186 _A ':' 272 _A BA _A 01111010 ₁		218 _A 'z' 332 _A DA _A 11011010 ₁		250 _A 'z' 372 _A FA _A 11111010 ₁
S	027 033, 1B _A 00011011 ₁		05 _A 073 _A 3B _A 00111011 ₁	{	123 _A '{' 173 _A 7B _A 01111011 ₁	{	155 _A '{' 233 _A 9B _A 10011011 ₁		187 _A '{' 273 _A BB _A 01111011 ₁		219 _A 'A' 333 _A DB _A 11011011 ₁		251 _A 'g' 373 _A FB _A 11111011 ₁
C	028 034, 1C _A 00011100 ₁		06 _A 074 _A 3C _A 00111100 ₁	!	124 _A '!' 174 _A 7C _A 01111100 ₁	!	156 _A '<' 234 _A 9C _A 10011100 ₁		188 _A '<' 274 _A BC _A 01111100 ₁		220 _A 'O' 334 _A DC _A 11011100 ₁		252 _A 'o' 374 _A FC _A 11111100 ₁
E	029 035, 1D _A 00011101 ₁		06 _A 075 _A 3D _A 00111101 ₁	~	125 _A '~' 175 _A 7D _A 01111101 ₁	~	157 _A '~' 235 _A 9D _A 10011101 ₁		189 _A '~' 275 _A BD _A 01111101 ₁		221 _A 'O' 335 _A DD _A 11011101 ₁		253 _A 'o' 375 _A FD _A 11111101 ₁
A	030 036, 1E _A 00011110 ₁		06 _A 076 _A 3E _A 00111110 ₁	?	126 _A '?' 176 _A 7E _A 01111110 ₁	?	158 _A '>' 236 _A 9E _A 10011110 ₁		190 _A '>' 276 _A BE _A 01111110 ₁		222 _A 'I' 336 _A DE _A 11011110 ₁		254 _A 'O' 376 _A FE _A 11111110 ₁
B	031 037, 1F _A 00011111 ₁		06 _A 077 _A 3F _A 00111111 ₁		127 _A ' ' 177 _A 7F _A 01111111 ₁		159 _A ' ' 237 _A 9F _A 10011111 ₁		191 _A ' ' 277 _A BF _A 01111111 ₁		223 _A 'I' 337 _A DF _A 11011111 ₁		255 _A ' ' 377 _A FF _A 11111111 ₁

Bild IV.1 PC-Code-Tabelle mit Zeichendarstellung

In der Tabelle finden Sie alle mit PCs darstellbaren Zeichen. Sie können sie aufgrund der Tabelle in ANHANG III dargestellten Umrechnungstabelle benutzen.

Jedes Zeichen können Sie am Bildschirm mit der Taste ALT darstellen. Dazu muß erst der dezimale Wert eines Zeichens errechnet werden.

Beispiele:

Zeichen	HEX-Wert	Umrechnung	Dezimal-Wert	Zeichenerstellung
£	9C	9= 144 C= 12	156	ALT+156
@	40	4= 64 0= 0	64	ALT+ 64
p	9E	9= 144 E= 14	158	ALT+158
\	5C	5= 80 C= 12	92	ALT+ 92
\$	15	1= 16 5= 5	21	ALT+ 21

Um das Zeichen in der ersten Spalte zu erhalten, drücken Sie die Taste Alt und halten diese fest. Dazu tippen Sie die drei Ziffern auf der rechten Seite Ihrer Tastatur und nach loslassen der Alt-Taste wird das Zeichen am Bildschirm angezeigt.

Bitte beachten Sie, daß nicht jeder Drucker alle Zeichen darstellen kann.

Nicht in jedem Anwendungsprogramm können mit der Alt-Taste alle möglichen Zeichen dargestellt werden.

Wenn der deutsche Zeichensatz (deutsche Tastatur) geladen ist, können einige Zeichen in Ihrer Position in der Tabelle vertauscht sein (z.B. erscheint statt | das Zeichen >).

Erläuterung zur Tabelle

Dezimal-Wert

Zeichen-Darstellung, wenn deutscher
Zeichensatz geladen ist (KEYBGR)

Binäre Darstellung

Oktale Darstellung

ASCII-Zeichen

**Die Markt & Technik-
Junior-Serie!**

Version 3.1

MicroPro

Junior-WordStar

mit MailMerge – Textverarbeitung für IBM-PCs und Kompatible

Ein leistungsfähiges, vielseitiges, komfortables und schnelles Textverarbeitungsprogramm, mit dem Sie allen Anforderungen des Schreiballtags gerecht werden. Die Zusatzfunktion MailMerge hilft Ihnen, häufig anfallende Routinearbeiten, wie Bausteinverarbeitung, Serienbriefe und einiges mehr zu automatisieren.

- Ausführliche Hilfsinformationen
- Fettdruck, Unterstreichen, Hoch- und Tiefstellen usw.
- Blocksatz, Flatterrand, zentriert Texte, Zahlenkolonnen, Textbausteine
- Korrektur durchgehender Fehler mit Suchen und Ersetzen

- auch als Programmeditor hervorragend geeignet

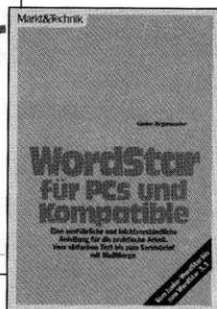
Hardware-Anforderung:

IBM-PC oder kompatibler Computer unter PC-/MS-DOS ab Version 2.0, mindestens ein Diskettenlaufwerk, mindestens 256 Kbyte RAM-Speicher.

Best.-Nr. MP 104

DM 399,-*

* inkl. MwSt. Unverbindliche Preisempfehlung



Und dazu die richtige Literatur:

G. Jürgensmeier
WordStar für PCs und Kompatible

Best.-Nr. MT 90224,
ISBN 3-89090-224-3

DM 59,-

(sFr 54,30/öS 460,20)

Markt & Technik
Zeitschriften · Bücher
Software · Schulung

Markt & Technik-Produkte erhalten Sie in den Fachabteilungen der Warenhäuser, im Versandhandel, in Computer-Fachgeschäften oder bei Ihrem Buchhändler.

Markt & Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0
Bestellungen im Ausland bitte an: SCHWEIZ: Markt & Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 41 56 56 · ÖSTERREICH: Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526 · Ueberreuter Media Verlagsges. mbH, Alser Straße 24, A-1091 Wien, Telefon (0222) 481538-0.

**Die Markt & Technik-
Junior-Serie!**

MULTIPLAN JUNIOR

MICROSOFT

Elektronisches Planungssystem

Multiplan Junior gehört zu der Serie der Microsoft-Tabellenkalkulationsprogramme, die schon millionenfach verkauft wurden. Multiplan Junior können Sie für Ihre Kostenrechnung, Finanz- und Budgetplanung einsetzen. Multiplan berechnet alle von einem geänderten Wert abhängigen Zahlen automatisch neu. Leistungsfähige mathematische und statistische Funktionen vereinfachen die Lösung Ihrer Aufgaben. Reihen und Spalten können Sie bequem kopieren, umstellen und löschen.

- Ein äußerst vielseitiges und benutzerfreundliches Tabellenkalkulationsprogramm

- Tabellengröße von 255 Zeilen mal 63 Spalten
- leistungsfähige mathematische und statistische Funktionen

Hardware-Anforderung:

IBM-PC oder kompatibler Computer, mindestens 128 Kbyte RAM-Speicher, ein Diskettenlaufwerk, MS-DOS 1.1 oder höhere Version.

Best.-Nr. MP 102

DM 299,-*

* inkl. MwSt. Unverbindliche Preisempfehlung



Und dazu die richtige Literatur:

Dr. P. Albrecht
Multiplan deutsch
Best.-Nr. MT 90380,
ISBN 3-89090-380-0
DM 58,-
(sFr 53,40/öS 452,40)



Markt & Technik-Produkte erhalten Sie in den Fachabteilungen der Warenhäuser, im Versandhandel, in Computer-Fachgeschäften oder bei Ihrem Buchhändler.

Markt & Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0
Bestellungen im Ausland bitte an: SCHWEIZ: Markt & Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 41 56 56 · ÖSTERREICH: Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526 · Ueberreuter Media Verlagsges. mbH, Alser Straße 24, A-1091 Wien, Telefon (0222) 481538-0.

**Die Markt & Technik-
Junior-Serie!**

MICROSOFT **Word-Junior** **Textverarbeitungsprogramm**

Microsoft Word Junior erleichtert Ihnen nicht nur Ihre Schreibarbeit, sondern ist auch besonders einfach und schnell zu erlernen. Texte redigieren, trennen und archivieren, Serienbriefe schreiben, Dokumentationen erstellen und vieles mehr wird mit Microsoft Word Junior zeitsparend erledigt.

- Eingebaute Hilfsbildschirme mit Windowtechnik
- Fußnotenverwaltung
- Mausunterstützung
- Fettdruck, Unterstreichung
- Korrektur durchgehen der Fehler mit
 - * Suchen und Ersetzen

- Blocksatz, zentrierte Texte

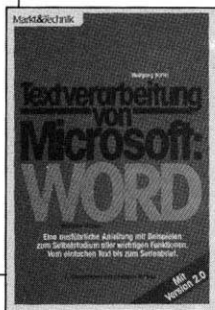
Hardware-Anforderung:

IBM-PC oder kompatibler Computer unter PC-/MS-DOS ab Version 2.0, mindestens ein Diskettenlaufwerk, mindestens 112 Kbyte RAM-Speicher.

Best.-Nr. MP 111

* inkl. MwSt. Unverbindliche Preisempfehlung

DM 399,-*



Und dazu die richtige Literatur:

W. Bartel

Textverarbeitung von Microsoft WORD

Best.-Nr. MT 90177,
ISBN 3-89090-177-8

DM 48,-

(sFr 44,20/öS 374,40)


Markt & Technik
Zeitschriften · Bücher
Software · Schulung

Markt & Technik-Produkte erhalten Sie in den Fachabteilungen der Warenhäuser, im Versandhandel, in Computer-fachgeschäften oder bei Ihrem Buchhändler.

Markt & Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 46 13-0
Bestellungen im Ausland bitte an: SCHWEIZ: Markt & Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 41 56 56 · ÖSTERREICH: Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526 · Ueberreuter Media Verlagsges. mbH, Alser Straße 24, A-1091 Wien, Telefon (0222) 481538-0.

Weitere Fachbücher aus unserem Verlagsprogramm

Integrierte Software

M. Kolberg

Integrierte Datenverarbeitung mit Framework II

1986, 397 Seiten

• KENNENLERNEN: Im ersten Abschnitt wird gezeigt, was FRAMEWORK leisten kann.

• AUSPROBIEREN: Vor der Darstellung der einzelnen Feinheiten des Programms ist eine Form der Einführung in die Arbeit mit FRAMEWORK sinnvoll.

• BEHERRSCHEN: Die Darstellung orientiert an den jeweiligen Details bei der Verwendung von Konzepten, Textverarbeitung, Kalkulationsprogramm, Grafik, Datenbank und Telekommunikation.

Best.-Nr. MT 90164

ISBN 3-89090-164-6

DM 68,-

W. Maaß

Software-Schnellkurs: Symphony/deutsch

1985, 220 Seiten

Symphony ist ein kombiniertes Kalkulations-, Textverarbeitungs-, Graphik- und Datenbankprogramm, das mit einer Fenster-Technik versehen ist – wie bei allen Software-Schnellkursen werden dabei die Einzelbefehle nicht isoliert beschrieben, sondern in praxisgerechten Funktionsgruppen und Abhängigkeiten dargestellt – praktisch erprobte Tipps, Hinweise und Warnungen vervollständigen die Darstellungen – Software-Schnellkurse bieten, was der Name verspricht: den kürzesten Weg in die Praxis!

Best.-Nr. MT 818, ISBN 3-89090-102-6

(sFr. 44,50/öS 374,40)

DM 48,-

J. Steiner

Symphony für Praktiker

1985, 532 Seiten

Das vorliegende Buch ist ein Nachschlagewerk für den Praktiker. Detailliert werden die Programmteile beschrieben, wobei die einzelnen Programmfunktionen mit praxisgerechten Beispielen anschaulich erläutert werden. Der Leser lernt unter anderem den Aufbau einer einfachen Tabelle, die Veranschaulichung von Zahlenkolonnen in grafischer Form, die Anwendung der Textverarbeitung, die Erstellung einer Datenbank mit Eingabefeldern, die Kopplung der Datenbank mit Texten zu Serienbriefen sowie die Anwendungsmöglichkeiten der Kommunikation.

Best.-Nr. MT 667, ISBN 3-89090-148-4

(sFr. 62,60/öS 530,40)

DM 68,-

M. Kolberg

Unternehmensanalyse mit Symphony

1986, 499 Seiten inkl. Diskette

Das Buch beschreibt die Anwendung von Symphony an konkreten Beispielen einer Unternehmensanalyse. Dabei werden folgende Punkte behandelt: Aufnahme des IST-Zustandes des Unternehmens, Darstellung der vergangenen Entwicklung, Analyse der Schwachstellen, Prognose zur zukünftigen Entwicklung sowie Entwicklung von Verbesserungsansätzen.

Best.-Nr. MT 90157, ISBN 3-89090-157-3

(sFr. 71,80/öS 608,40)

DM 78,-

G. T. LeBlond/D. F. Cobb

Das Lotus 1-2-3 Kompendium

1984, 482 Seiten

Wie man 1-2-3 mit PC/MS-DOS startet • die Erstellung von Tabellen • die Befehlsmenüs • Zellenformate • mathematische, statistische und finanzmathematische Funktionen • Dateiverwaltung • Datenbankverbindungen • die Makro-Sprache • das ausführliche Anwenderhandbuch für Laien und Profis.

Best.-Nr. MT 90362, ISBN 3-89090-362-2

(sFr. 53,40/öS 452,40)

DM 58,-

D. Andersen/D. F. Cobb

Lotus 1-2-3 für Praktiker

1985, 291 Seiten

Eine praktische Anleitung, wie Sie mit 1-2-3 problembewußt und anspruchsvoll Kalkulationen, Grafiken, Datenbanken und Makros effizienter programmieren können. Aufeinander aufbauende Beispiele erleichtern dem Anfänger den Einstieg in 1-2-3, fortgeschrittene Anwender finden detaillierte Hinweise und Problemdiskussionen.

Best.-Nr. MT 799, ISBN 3-89090-084-4

(sFr. 62,60/öS 530,40)

DM 68,-

J. Steiner

Lotus 1-2-3 Version 2

Kompendium

1986, ca. 300 Seiten

Lotus 1-2-3 ist eines der erfolgreichsten und interessantesten Softwarepakete auf dem PC-Markt. Die integrierten Programme Tabellenkalkulation, Präsentationsgrafik und die Datenbankfunktionen haben einen Standard in diesem Bereich gesetzt. Mit der eingebauten verbesserten Programmiersprache wird auch die Makro-Programmierung noch leistungsfähiger.

Ein unentbehrliches Hilfsmittel zur Einarbeitung in Lotus 1-2-3 und ideales Nachschlagewerk für den geübten Anwender.

Best.-Nr. MT 90384, ISBN 3-89090-384-3

(sFr. 53,40/öS 452,40)

DM 58,-

Programmiersprachen

E. Huckert

C-Toolbox Textverarbeitung

1986, 273 Seiten inkl. Disketten

Vollständige Textverarbeitung inklusive Spellchecker und Silbentrennung in C-Quellcode für IBM-PC und Kompatibel. Das umfassendste Grundlagenbuch zur Textverarbeitung allgemein für Hobby- und Profiprogrammierer mit Vorkenntnissen in C.

Best.-Nr. MT 90364, ISBN 3-89090-364-9

(sFr. 63,50/öS 538,20)

DM 69,-

Die angegebenen Preise sind Ladenpreise

Sie erhalten Markt & Technik-Bücher bei Ihrem Buchhändler

Markt & Technik Verlag AG Unternehmensbereich Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München

Weitere Fachbücher aus unserem Verlagsprogramm

W. Kassera

Programmieren mit TURBO-PASCAL 3.0 1986, 350 Seiten

Dieses Buch ist eine Einweisung zur Programmerstellung mit Turbo-Pascal auf MS-DOS-Rechnern. Der Befehlssatz, der auch auf Grundversionen der üblichen PCs läuft, wird an ausführlichen Beispielen erläutert. Ausgehend von Einführungsbeispielen gibt es u. a. Aufschluß über: Lösung von Rechenaufgaben mit Turbo-Pascal, Erstellung von Bildschirmmasken, Menüaufbau, Dateiverwaltung, Einsatz von Maschinenprogrammen und vieles mehr. Pascal-eigene Sprachbegriffe wie Funktionen und Prozeduren werden eingehend behandelt und in Anwendungsbeispielen eingesetzt. Eine Diskette mit über 100 Prozeduren im MS-DOS-Format ist dem Buch beigelegt.

Best.-Nr. MT 90159, ISBN 3-89090-159-X
(sFr. 54,30/6S 460,20)

DM 59,-

K. Knecht

Microsoft BASIC 1984, 204 Seiten

Eine Übersicht der Version 5.0 von Microsoft BASIC · umfangreiche Beispiele für CP/M-Systeme und TRS-80 · Programmieren mit Sprüngen und Schleifen · Umgang mit Zeichenketten und Matrizen · die Arbeitsweise des Editors · Aufbau verschiedener Dateitypen · für Anfänger.

Best.-Nr. MT 650, ISBN 3-922120-87-3
(sFr. 44,20/6S 374,40)

DM 48,-

J. Purdum/T. Leslie

Die C-Programmbibliothek 1986, 361 Seiten

Dieses Buch erspart dem C-Programmierer Stunden mühsamer Kleinarbeit und hilft, effizientere Programme zu schreiben. Der erste Teil zeigt, wie man zu universellen Bibliotheksfunktionen kommt und gibt Tips, wie C noch wirkungsvoller eingesetzt werden kann. Der zweite Teil enthält eine Reihe ausführlich erklärter C-Funktionen als wertvolle Ergänzung Ihrer Programmbibliothek. Dazu gehören unter anderem ein Terminalinstallationsprogramm, mehrere Sortier-Algorithmen und ein Satz ISAM-Funktionen.

Best.-Nr. MT 90133, ISBN 3-89090-133-6
(sFr. 63,50/6S 538,20)

DM 69,-

J. N. P. Hume/R. C. Holt

UCSD-Pascal 1984, 492 Seiten

Eine Unterweisung in das weit verbreitete und auf allen gängigen Mikrocomputern verfügbare UCSD-Pascal · überschaubare Lerneinheiten · eingehende Behandlung der Datenstrukturen, Records und Dateien · numerische Verfahren für Statistiken · Sortieralgorithmen · die Maschinensprache · für Apple-Besitzer: die »Turtle-Grafik«.

Best.-Nr. MT 715, ISBN 3-89090-058-5
(sFr. 58,90/6S 499,20)

DM 64,-

I. Lücke/P. Lücke

Turbo-Pascal 1985, 290 Seiten

Eine logisch aufgebaute Einführung in die Benutzung und Anwendung von Turbo-Pascal (einschließlich Version 3.0): die Programmstruktur · Syntaxdiagramme · anwenderdefinierte Funktionen und Prozeduren · die Turbo-Toolbox · Turbo-Lader · die Implementation von MS-DOS, CP/M-86, CP/M · BCD-Arithmetik · Grafik, Farbe, Sound und Window-Technik.

Best.-Nr. MT 90150, ISBN 3-89090-150-6
(sFr. 45,10/6S 382,10)

DM 49,-

P. M. Chirlan

Der Einstieg in C 1985, 290 Seiten

C ist der neue Star unter den Programmiersprachen. Zum Erlernen dieser vielseitigen und mächtigen Programmiersprache, die gerade ihren Siegeszug durch Europa antritt, finden Sie in diesem Buch alles, was Sie wissen müssen. Gleich von Anfang an schreibt der Leser seine eigenen Programme, und anhand von über 70 genauestens erläuterten Beispielen kann er seine praktischen und theoretischen Fertigkeiten vertiefen. Besonders Augenmerk legt der Autor auf die wichtige Methode des strukturierten Programmierens, für das sich C bestens eignet. Auch das wichtige Thema Datentypen kommt nicht zu kurz. Das Buch erörtert den Prozeß der Fehlersuche und gibt dem Benutzer Hinweise, wie durch klaren Programmierstil und Dokumentation Fehler zu vermeiden sind. Die im Anhang aufgeführte Syntaxübersicht macht es zu einem wertvollen Hilfsmittel.

Best.-Nr. MT 90086, ISBN 3-89090-086-0
(sFr. 55,20/6S 468,-)

DM 60,-

Z. Stanka/S. Lösch

Die C-Sprache 1984, 288 Seiten

Ein Buch zur praktischen Konzipierung von C-Programmen · Charakteristik der C-Sprache: kleiner Wortschatz, aber unbegrenzte Kombinationsmöglichkeiten aufgrund ihrer ungewöhnlichen Logik · Beispiele von Laufdiagrammen · Beschreibung der wichtigsten Systemfunktionen · ein praktischer Führer durch die C-Sprache nicht nur für Profis!

Best.-Nr. PW 80328, ISBN 3-921803-28-4
(sFr. 54,30/6S 460,20)

DM 59,-

Dr. Dobb's Journal

C-Tools

1986, 692 Seiten

Eine Sammlung kommentierter C-Quellprogramme: Compiler, Assembler, Linker, Funktion für Textverarbeitung und Treiber.

Während der letzten Jahre hat Dr. Dobb's Journal viele dieser Programme abgedruckt. Die nützlichsten werden im vorliegenden Band wiedergegeben. Dazu zählen ein vollständiger C-Compiler mit allen Unterstützungs-routinen sowie Versionen verschiedener Utilityprogramme wie z.B. Grep und ein C-Crossreferenz-Programm.

Best.-Nr. MT 90190, ISBN 3-89090-190-5
(sFr. 71,80/6S 608,40)

DM 78,-

Die angegebenen Preise sind Ladenpreise

Sie erhalten Markt & Technik-Bücher bei Ihrem Buchhändler

Markt & Technik Verlag AG Unternehmensbereich Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München



HANS C. NIEDER

wurde 1954 in Prien am Chiemsee geboren. Nach Abschluß einer Handelsschule arbeitete er acht Jahre bei einem größeren Kreditinstitut. Der Einstieg in die EDV begann bereits Mitte der 70er Jahre bei der Bundeswehr. Nach zweijähriger Weiterbildung folgten Jahre der Selbständigkeit und Führung eines Softwarehauses (Beratung und Software-Erstellung). Seit 1980 werden von Hans C. Nieder Seminare an verschiedenen Akademien durchgeführt.

Heute ist er als Software Marketing Manager bei einem namhaften PC-Hersteller beschäftigt.

MS-DOS Version 3

Zielsetzung dieses Buches zur neuesten Version des Betriebssystems MS-DOS ist es, dem Leser eine leicht verständliche Einführung zu bieten und in kompakter Form die wichtigsten Bestandteile übersichtlich mit vielen Übungsbeispielen darzustellen. Dabei wurden die Implementationen möglichst vieler Hersteller von PCs berücksichtigt. Die Version 3.1 ist bereits mit einbezogen.

Das Buch ist im wesentlichen in drei Bereiche gegliedert:

Eine leicht verständliche Einführung für den unerfahrenen Anwender, in welchem die grundlegenden Kenntnisse vermittelt werden. Der zweite Teil beinhaltet im wesentlichen eine Übersicht der alphabetisch geordneten Befehle mit vielen Übungen. Im dritten Teil wird in die Konfiguration bzw. Anpassung des MS-DOS 3 eingeführt (vor allem CONFIG.SYS).

Dieses Buch wendet sich nicht nur an den Anfänger, sondern gerade mit seinem zweiten und dritten Teil auch an den Fortgeschrittenen und Profi. Damit wird es zum ständigen Begleiter eines MS-DOS-Anwenders.

ISBN N 3-89090-201-4



DM 58,-
sFr. 53,40
öS 452,40