

NIXDORF
COMPUTER

Nixdorf 8810

**Benutzerhandbuch
für das Betriebssystem
MS-DOS**

Bitte abreißen und in
die Tasche im Handbuch-
rücken einstecken.

Einführung	1
Die Inbetriebnahme Ihres Systems	2
Eingehendere Informationen der Dateien	3
Wissenswertes über die MS-DOS-Befehle	4
Verzeichnis der MS-DOS-Befehle	5
Sonderbearbeitungstasten und Steuerzeichenfunktionen in MS-DOS	6
Der Zeilen-Editor (EDLIN)	7
Das Dienstprogramm FC für den Dateienvergleich	8
Das Bindeprogramm MS-LINK	9
DEBUG	10
Anhang	11

Benutzerhandbuch für das Betriebssystem MS-DOS

Organisationsblatt

Dieses Blatt gibt eine Übersicht über alle Änderungen, die seit der ersten Auflage an diesem Modul durchgeführt wurden. Es wird bei jeder Änderungsmitteilung mitgeliefert und ist jeweils auszutauschen.

Erstauflage:

03/85

Rel. 2.11

Änderung 1 1.12.1985

Benutzerhandbuch für das Betriebssystem MS-DOS

Änderungswünsche / Fehler

Sollten Ihnen bei der Benutzung dieses Teils der Systemliteratur Fehler aufgefallen sein, oder sollten Sie Vorschläge zur Verbesserung des Moduls haben, so bitten wir Sie, diese schriftlich zu formulieren und an folgende Anschrift zu schicken:

Nixdorf Computer AG
Abt. ZSI
Fürstenallee 7
4790 Paderborn

Wiederherstellung des Urheberrechts. Verwenden und A. Mitschulung
ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugelassen, zu wider-
handlungen verpflichten zu Schadensersatz. Alle Rechte für den Fall der Patent-
erteilung oder Gebrauchsmusteranmeldung vorbehalten.

INHALTSVERZEICHNIS

KAPITEL 3

EINGEHENDERE INFORMATIONEN DER DATEIEN

3.1	Wie man Dateinamen festlegt.....	3 - 2
3.2	Die Stellvertreterzeichen.....	3 - 4
3.2.1	Das Stellvertreterzeichen ?.....	3 - 4
3.2.2	Das Stellvertreterzeichen *.....	3 - 5
3.3	Unzulässige Dateinamen.....	3 - 6
3.4	Wie man Dateien kopiert.....	3 - 7
3.5	Wie man Dateien sichert.....	3 - 9
3.6	Arbeitsbereiche und Inhaltsverzeichnisse.....	3 - 10
3.7	Dateinamen und Suchwege.....	3 - 13
3.7.1	Suchwegbezeichnungen.....	3 - 13
3.7.2	Suchwege und externe Befehle.....	3 - 15
3.7.3	Suchwege und interne Befehle.....	3 - 16
3.7.4	Die Anzeige Ihres momentanen Arbeitsbereichs..... am Bildschirm	3 - 17
3.7.5	Das Anlegen eines Arbeitsbereichs.....	3 - 18
3.7.6	Wie man den Arbeitsbereich wechselt.....	3 - 19

KAPITEL 4

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.1	Einführung.....	4 - 2
4.2	Die verschiedenen Befehlsarten.....	4 - 3
4.3	Die Optionen der Befehle.....	4 - 4
4.4	Gemeinsame Eigenschaften aller MS-DOS-Befehle.....	4 - 6
4.5	Die Stapelprogrammbausteine.....	4 - 8

INHALTSVERZEICHNIS

4.6	Der Stapelprogrammbaustein AUTOEXEC.BAT.....	4 - 11
4.6.1	Wie man den Stapelprogrammbaustein..... AUTOEXEC.BAT erstellt	4 - 13
4.7	Stapelprogrammbausteine mit Blindparametern.....	4 - 14
4.7.1	Die Ausführung eines Stapelprogrammbausteins.....	4 - 15
4.8	Eingabe und Ausgabe.....	4 - 16
4.8.1	Neubestimmung des Ausgabeziels.....	4 - 17
4.8.2	Verarbeitungsbefehle.....	4 - 18
4.8.3	Die Befehlsverkettung.....	4 - 18

KAPITEL 5

VERZEICHNIS DER MS-DOS-BEFEHLE

5.1	Die Befehlsformate.....	5 - 2
5.2	Die MS-DOS-Befehle.....	5 - 3
5.3	Stapelprogrammbausteinbefehle.....	5 - 45

KAPITEL 6

SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

6.1	Sonderbearbeitungstasten in MS-DOS.....	6 - 2
6.2	Steuerzeichenfunktionen in MS-DOS.....	6 - 8

INHALTSVERZEICHNIS

KAPITEL 7

DER ZEILEN-EDITOR (EDLIN)

7.1	Einführung.....	7 - 2
7.2	Wie man den Zeilen-Editor EDLIN startet.....	7 - 3
7.3	Sonderbearbeitungstasten in EDLIN.....	7 - 5
7.4	Erläuterung der EDLIN-Befehle.....	7 - 15
7.4.1	Die Befehlsoptionen.....	7 - 19
7.5	Die EDLIN-Befehle.....	7 - 21
7.6	Fehlermeldungen in EDLIN.....	7 - 51

KAPITEL 8

DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8.1	Einführung.....	8 - 2
8.1.1	Einschränkungen beim Vergleich von Binärdateien....	8 - 3
8.2	Syntax und Datenbezeichnungen.....	8 - 3
8.3	Wie man das Dienstprogramm FC aufruft.....	8 - 4
8.4	Die FC-Befehlszusätze.....	8 - 4
8.5	Wie erkannte Unterschiede angezeigt werden.....	8 - 7
8.6	Neubestimmung des Ausgabeziels für FC.....	8 - 8
8.7	Beispiele.....	8 - 9
8.8	Fehlermeldungen in FC.....	8 - 13

INHALTSVERZEICHNIS

KAPITEL 9

DAS BINDEPROGRAMM MS-LINK

9.1	Einführung.....	9	- 2
9.2	Kurzbeschreibung von MS-LINK.....	9	- 2
9.3	Einige Begriffe, die Sie kennen sollten.....	9	- 4
9.4	Die Dateien, mit denen MS-LINK arbeitet.....	9	- 6
9.4.1	Die Suffixe der Eingabedateien.....	9	- 6
9.4.2	Die Suffixe der Ausgabedateien.....	9	- 6
9.4.3	Die vorläufige Arbeitsdatei VM.TMP.....	9	- 7
9.5	Wie man MS-LINK startet.....	9	- 8
9.5.1	Erstes Verfahren: Beantwortung der Eingabeaufforderungen	9	- 8
9.5.2	Zweites Verfahren: die lange Befehlszeile.....	9	- 10
9.5.3	Drittes Verfahren: die Antwortdateien.....	9	- 11
9.6	Hilfs- und Steuerzeichen in MS-LINK.....	9	- 13
9.7	Die MS-LINK-Eingabeaufforderungen.....	9	- 15
9.8	MS-LINK-Befehlszusätze.....	9	- 17
9.9	BEispiele eines MS-LINK-Bindelaufs.....	9	- 21
9.10	Fehlermeldungen in MS-LINK.....	9	- 24

KAPITEL 10

DEBUG

10.1	EINLEITUNG.....	10	- 2
10.2	STARTEN DES DEBUG-PROGRAMMES.....	10	- 3
10.3	BEFEHLSPARAMETER.....	10	- 4
10.4	DEBUG-BEFEHLE.....	10	- 7

INHALTSVERZEICHNIS

A N H A N G

ANHANG TEIL A

HINWEISE FÜR BENUTZER VON SYSTEMEN MIT.....A - 1
DISKETTEN-EINZELLAUFWERK

ANHANG TEIL B

DISKETTENFEHLER.....A - 3

ANHANG TEIL C

ANSI-ESCAPE-ZEICHENFOLGEN.....A - 5

ANHANG TEIL D

WIE SIE IHR SYSTEM KONFIGURIEREN.....A - 12

EINFÜHRUNG

1.1 WAS IST MS-DOS?

Microsoft MS-DOS ist ein Betriebssystem für Computer, die auf den 8088-, 8086-, 80186- und 80286-Mikroprozessoren aufgebaut sind. MS-DOS ermöglicht Ihnen den Betrieb sowie die optimale Nutzung Ihres Computers, der Disketten- und Plattenlaufwerke und des Druckers.

1.2 WAS IST EIN BETRIEBSSYSTEM?

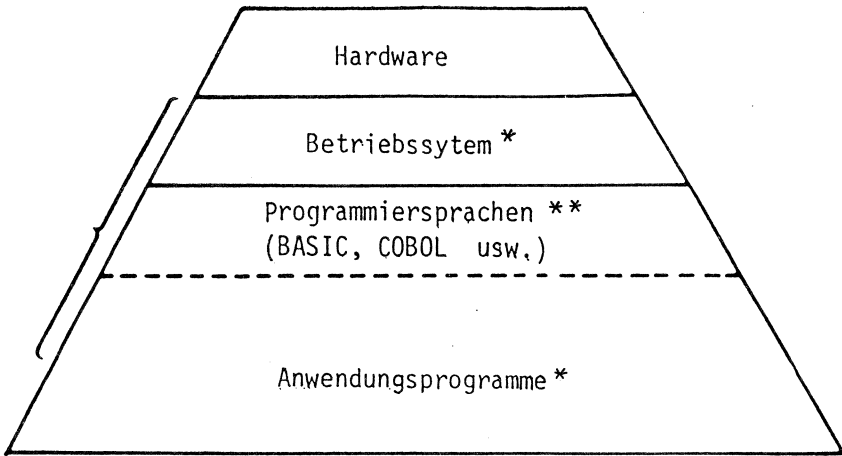
Man könnte das Betriebssystem als Ihren "stillen Helfer" bei der Arbeit mit dem Computer bezeichnen. Es stellt die Verbindung zwischen der Hardware auf der einen und Ihnen (dem Benutzer) und der übrigen Systemsoftware auf der anderen Seite her. Mit dem Betriebssystem ist es ähnlich wie mit dem elektrischen Strom in einem Haus, - obgleich dieser für das Funktionieren der elektrischen Geräte in Ihrem Haus unerlässlich ist, sind Sie sich seines Vorhandenseins im allgemeinen nicht bewußt.

Von allen Bestandteilen der Systemsoftware steht das Betriebssystem (BS) der Hardware am nächsten; es ist grundsätzlich nur für den Einsatz auf einem bestimmten Mikroprozessor (Computer) ausgelegt und geeignet. MS-DOS zum Beispiel ist ausschließlich für die Mikroprozessorfamilie 8086, 8088, 80186 und 80286 vorgesehen; ein Einsatz auf anderen Mikroprozessoren (wie z.B. dem Z8000) ist nur möglich, wenn wichtige Teile des Betriebssystems umgeschrieben werden.



EINFÜHRUNG

Die Abbildung 1 veranschaulicht die Beziehung zwischen Hardware, Systemsoftware und Anwendungssoftware.



Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden, ist insbesondere die Weitergabe, die Vervielfältigung sowie die Verwertung der Inhalte dieser Unterlage ohne schriftliche Genehmigung vorbehalten.

- * Müßte an andere Hardware angepaßt werden.
- ** Falls diese bereits auf das Betriebssystem abgestimmt sind, bleiben Sie unverändert.

Abbildung 1. Beziehungen zwischen Hardware und Software

MS-DOS ist ein Platten-/Diskettenbetriebssystem, das Ihnen die Erstellung und Verwaltung von Dateien, die Ausführung und die Bindung von Programmen sowie den Zugriff auf die Peripheriegeräte Ihres Computers (z.B. Drucker und Platten-/Diskettenlaufwerke) ermöglicht. Es stellt eine entscheidende Weiterentwicklung auf dem Gebiet der Betriebssysteme für Mikroprozessoren dar.

EINFÜHRUNG

1.3 WARUM IST MS-DOS SO WICHTIG?

Alle Microsoft-Programmiersprachen (BASIC-Interpreter, BASIC-Compiler, FORTRAN, COBOL, C, Pascal) können unter MS-DOS eingesetzt werden. Benutzer von MS-DOS haben die Gewähr, daß ihr Betriebssystem, im Falle der Ankündigung neuer Produkte oder wichtiger Neuerscheinungen, von Microsoft als erstes unterstützt wird.

1.4 DER INHALT DIESES HANDBUCHES

Das vorliegende Handbuch dient zur Beschreibung von MS-DOS und seiner Benutzung. Dieses Kapitel stellt einige Grundlagen von MS-DOS vor; das Kapitel 2 erläutert, wie man MS-DOS in Betrieb nimmt und damit Disketten formatiert und sichert.

In Kapitel 3 erfahren Sie, was Dateien sind und wie man mit ihnen umgeht. Die Kapitel 4 bis 6 führen Sie in die MS-DOS-Befehle ein, und Kapitel 7 beschreibt den Zeilen-Editor EDLIN. Lesen Sie diese Kapitel aufmerksam durch - sie enthalten Informationen über die Sicherung Ihrer Daten, über Systembefehle sowie die MS-DOS-Bearbeitungsbefehle.

Kapitel 8 erklärt Ihnen, wie man das MS-DOS-Dateienvergleichs-dienstprogramm FC benutzt. Mit Hilfe dieses Dienstprogramms können Sie die Inhalte zweier Ursprungs- oder Dateien in Binärform miteinander vergleichen.

Kapitel 9 beschreibt das praktische MS-DOS-Dienstprogramm MS-LINK, das Ihnen beim Schreiben von Programmen das Binden getrennt voneinander erstellter Programmmodule sowie die Erstellung verschiebbarer Programmmodule ermöglicht.

Kapitel 10 beschreibt das MS-DOS-Dienstprogramm DEBUG, das Ihnen beim Testen und Ändern binärer Programme hilft.

Der Anhang des Handbuchs enthält Anleitungen zur Benutzung von Systemen mit nur einem Diskettenlaufwerk und zur Behandlung von Diskettenfehlermeldungen.

EINFÜHRUNG

1.5 ZEICHENVERWENDUNG BEI DER SYNTAXBESCHREIBUNG

In allen Beschreibungen der Syntax von Befehlen und Anweisungen dieses Handbuchs werden die folgenden Zeichen verwendet:

[] Eckige Klammern weisen darauf hin, daß die von ihnen umschlossene Eingabe wahlfrei ist.

< > Spitze Klammern kennzeichnen Daten, die unbedingt eingegeben werden müssen. Wenn der Text innerhalb dieser Klammern in Groß- und Kleinbuchstaben dargestellt ist, beschreibt er jeweils die Eingabe, die Sie vornehmen müssen. Beispiel: <Dateiname>. Wenn der von diesen Klammern umschlossene Text nur in Großbuchstaben dargestellt ist, beschreibt er eine Taste, die Sie drücken müssen. Beispiel: <RETURN>.

{ } Geschweifte Klammern weisen darauf hin, daß Sie zwischen zwei oder mehreren Eingaben wählen können. Mindestens eine der von diesen Klammern umschlossenen Eingaben muß gewählt werden, außer wenn diese Eingaben zusätzlich zwischen eckigen Klammern stehen.

... Auslassungspunkte weisen darauf hin, daß eine Eingabe so oft wie erforderlich oder gewünscht wiederholt werden kann.

| Ein senkrechter Strich weist auf eine ODER-Anweisung in einem Befehl hin. Wenn der senkrechte Strich in Zusammenhang mit einem MS-DOS-Verarbeitungsbefehl benutzt wird, weist er auf eine Befehlskette hin.

GROSSBUCHSTABEN

Großbuchstaben weisen auf Teile von Befehlen oder Anweisungen hin, die genau in der angegebenen Form eingegeben werden müssen.

Alle übrigen Interpunktionszeichen wie Kommas, Doppelpunkte, Schrägstriche und ähnliche müssen ebenfalls genau in der angegebenen Form eingegeben werden.

„Weitergabe sowie Verwältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nach Gesetzen, die vom Verleger aus dem Schrifttum die Rechte für den Fall der Publikation oder Gebrauchsmustereintragung vorbehalten.“

EINFÜHRUNG

1.6 DIE MS-DOS-DATEIEN

Die MS-DOS-System-Diskette enthält die folgenden Dateien:

Dateiname	Zweck der Datei
COMMAND.COM	MS-DOS-Befehlsprozessor
* MSDOS.SYS	MS-DOS-Betriebssystem
* IO.SYS	E/A-Schnittstelle
	Hardware-Betriebssystem
EDLIN.COM	Zeilen-Editor
DEBUG.COM	Prüf- und Korrekturprogramm
LINK.EXE	Bindeprogramm
CHKDSK.COM	Platten-/Diskettenüberprüfungsprogramm
FORMAT.COM	Diskettenformatierungsprogramm
SYS.COM	Systemübertragungsprogramm
DISKCOPY.COM	Sicherungsdienstprogramm
RECOVER.COM	Dateienwiederherstellungsprogramm
PRINT.COM	Druckwarteschlangezwischen- speicherungsprogramm
MORE.COM	Textausgabefilterprogramm
SORT.EXE	Textsortierfilterprogramm
FIND.EXE	Suchfilterprogramm zum Auffinden einer Zeichenfolge in einer Liste von Dateien oder in einer Standardeingabe
EXE2BIN.EXE	Konvertiert .EXE-Dateien
CONFIG.SYS	Systemkonfigurierungsdatei
FC.EXE	Dateienvergleichsprogramm

Sie werden diese Liste von Dateien am Bildschirm wiedererkennen, wenn Sie den im nächsten Kapitel beschriebenen Befehl DIR (= Show DIRectory = Zeige Inhaltsverzeichnis) erlernt haben und benutzen. Die beiden Dateien mit vorangestelltem Sternchen (*) sind sogenannte unsichtbare Dateien, die bei der Eingabe des Befehls DIR nicht am Bildschirm erscheinen.

Im nächsten Kapitel lernen Sie, wie man das MS-DOS-System startet und wie man Disketten formatiert und sichert.

DIE INBETRIEBNAHME IHRES SYSTEMS



2 DIE INBETRIEBNAHME IHRES SYSTEMS

2.1	Was passiert, wenn Sie MS-DOS zum ersten Mal laden....	2 - 2
2.2	Wie man Datum und Uhrzeit eingibt.....	2 - 3
2.3	Wie man das Standardlaufwerk ändert.....	2 - 5
2.4	Wie man Disketten formatiert.....	2 - 6
2.4.1	Der Befehl FORMAT.....	2 - 6
2.5	Wie man Disketten sichert.....	2 - 8
2.5.1	Der Befehl DISKCOPY.....	2 - 9
2.6	Der automatische Programmaufruf.....	2 - 11
2.7	Die Dateien.....	2 - 11
2.7.1	Was ist eine Datei?.....	2 - 11
2.7.2	Wie MS-DOS Ihre Dateien verwaltet.....	2 - 12
2.7.3	Der Befehl DIR.....	2 - 13
2.7.4	Der Befehl CHKDSK.....	2 - 14
2.8	Wie man das System abschaltet.....	2 - 15

„Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung derselben ist ohne schriftliche Genehmigung des Nixdorf Computer AG. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustereintragung vorbehalten.“

DIE INBETRIEBNAHME IHRES SYSTEMS

2.1 WAS PASSIERT, WENN SIE MS-DOS ZUM ERSTEN MAL LADEN?

Befolgen Sie beim Einlegen der MS-DOS-System-Diskette und beim Laden des Betriebssystems die Anweisungen Ihres Computerherstellers. Das Laden von MS-DOS nimmt, je nach der Arbeitsspeicherkapazität Ihres Computers, 3 bis 45 Sekunden in Anspruch.

Sobald der Ladevorgang abgeschlossen ist, sucht das System auf der MS-DOS-System-Diskette die Datei COMMAND.COM und lädt sie in den Arbeitsspeicher. Diese Datei ist ein Programm, das die von Ihnen eingegebenen Befehle verarbeitet und dann die Ausführung der entsprechenden Programme veranlaßt. Sie wird auch als Befehlsprozessor bezeichnet.

Nach dem Laden des Befehlsprozessors erscheint die folgende Anzeige an Ihrem Bildschirm (der Cursor ist die Schreibmarke):

```
Current date is Tue 01-01-1980
Enter new date:
```

Nun müssen Sie das heutige Datum und die genaue Uhrzeit eingeben.

DIE INBETRIEBNAHME IHRES SYSTEMS

2.2 WIE MAN DATUM UND UHRZEIT EINGIBT

Geben Sie das Datum in der Form tt-mm-jj ein, wobei:

- tt eine ein- oder zweistellige Zahl von 1 bis 31 ist (zur Darstellung des Tages)
- mm eine ein- oder zweistellige Zahl von 1 bis 12 ist (zur Darstellung des Monats)
- jj eine zweistellige Zahl von 80 bis 99 (die 19 wird stillschweigend vorausgesetzt) oder eine vierstellige Zahl von 1980 bis 2099 ist (zur Darstellung des Jahres).

Als Antwort auf die Aufforderung zur Eingabe eines neuen Datums kann jedes beliebige Datum eingegeben werden, solange dies entsprechend der oben beschriebenen Form erfolgt. Als Abgrenzungszeichen zwischen den Zahlen können Bindestriche (-) oder Schrägstriche (/) verwendet werden.

Zum Beispiel:

6-1-82 oder 06/01/82

sind beide als Antwort auf die Eingabeaufforderung "Enter new date:" zulässig.

Falls Sie ein ungültiges Datum oder ein Datum in ungültiger Form eingeben, erscheint am Bildschirm die Eingabeaufforderung "Invalid date".

Die Eingabe kann mit der RETURN-Taste übersprungen werden.

"Weitergabe sowie Vervielfältigung dieser Unterlagen, Verwertung und Mitteilung derselben ist ohne schriftliche Genehmigung der Nixdorf Computer AG. In sonstigen Fällen sind die gesetzlichen Bestimmungen zu beachten. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustererteilung vorbehalten."

DIE INBETRIEBNAHME IHRES SYSTEMS

Wenn Sie auf diese Eingabeaufforderung geantwortet haben und zum Schluß die RETURN-Taste drücken (bei einigen Computern die ENTER-Taste), erscheint am Bildschirm die Eingabeaufforderung:

```
Current time is 0:00:56.51
Enter new time:
```

Geben Sie die genaue Zeit in der Form hh:mm ein, wobei:

hh eine ein- oder zweistellige Zahl von 0 bis 23 ist (zur Darstellung der Stunden) mm eine ein- oder zweistellige Zahl von 0 bis 59 ist (zur Darstellung der Minuten).

MS-DOS hat eine interne Uhrfunktion, mit deren Hilfe jedesmal der genaue Zeitpunkt protokolliert wird, zu dem Sie eine Datei erstellen oder bearbeiten. Der Zeitpunkt der Erstellung und der der letzten Bearbeitung einer Datei werden angezeigt.

Beachten Sie bitte, daß MS-DOS die Uhrzeit im 24-Stunden-System verarbeitet. Wenn es also halb drei Uhr nachmittags ist, müssen Sie 14:30 eingeben.

Beispiel:

```
Current time is 0:03:14.32
Enter new time:
```

Benutzen Sie bitte ausschließlich den Doppelpunkt (:) zur Abgrenzung von Stunden und Minuten. Wenn Sie ein ungültiges Abgrenzungszeichen eingeben, wiederholt MS-DOS die Eingabeaufforderung.

HINWEIS

Wenn Ihnen bei der Eingabe ein Fehler unterläuft, drücken Sie die CONTROL-Taste Ihrer Tastatur und gleichzeitig die Taste C. Die Tastenkombination CONTROL-C ermöglicht es Ihnen, die laufende Eingabe abubrechen. Dann können Sie eine neue Antwort auf die Eingabeaufforderung oder einen neuen Befehl eingeben. Um eine Zeile Text zu korrigieren, löschen Sie vor der Betätigung der RETURN-Taste den fehlerhaften Text buchstabenweise durch Betätigen der Rücktaste.

Nun haben Sie alle Schritte vollzogen, die für das Starten von MS-DOS erforderlich sind.

DIE INBETRIEBNAHME IHRES SYSTEMS

2.3 WIE MAN DAS STANDARDLAUFWERK ÄNDERT

Wenn Sie die Aufforderung zur Eingabe der neuen Uhrzeit beantwortet haben, erscheint die folgende Meldung am Bildschirm:

A>

Die Meldung A> - die MS-DOS-Eingabeaufforderung des Befehlsprozessors - teilt Ihnen mit, daß MS-DOS zur Entgegennahme von Befehlen bereit ist.

Das A in der vorhergehenden Eingabeaufforderung steht für das Standard-Diskettenlaufwerk. Das heißt, daß MS-DOS nur im Laufwerk A nach eingegebenen Dateinamen sucht und Dateien nur auf die in dieses Laufwerk eingelegte Diskette schreibt, solange Sie nicht ein anderes Laufwerk als Standardlaufwerk vorgeben. Durch Änderung der Laufwerksangabe oder durch Angabe von B: in einem Befehl können Sie MS-DOS dazu veranlassen, standardmäßig mit der Diskette in Laufwerk B zu arbeiten. Um die Laufwerksangabe zu ändern, geben Sie den neuen Laufwerksbuchstaben mit nachfolgendem Doppelpunkt ein.

Zum Beispiel:

A> (MS-DOS-Eingabeaufforderung)

A>B: (Als Antwort auf die Eingabeaufforderung haben Sie B: eingegeben.)

B> (Das System antwortet mit B:; Laufwerk B ist nun das Standardlaufwerk.)

Nun erscheint die Eingabeaufforderung B> am Bildschirm, und MS-DOS arbeitet standardmäßig mit der in Laufwerk B eingelegten Diskette, solange, bis Sie ein anderes Laufwerk als Standardlaufwerk vorgeben.

Falls Ihr Computer nur mit einem einzigen Diskettenlaufwerk ausgestattet ist, finden Sie wichtige Informationen in Teil A des Anhangs, "Anweisungen für Benutzer von Systemen mit Einzeldiskettenlaufwerken". Möglicherweise haben Sie auch einen Computer mit mehr als zwei Diskettenlaufwerken; informieren Sie sich bitte in diesem Fall in der Anweisung Ihres Computerherstellers.

„Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden. Zuwiderhandeln wird strafrechtlich verfolgt.“ Für den Fall der Fälschung oder Gebrauchsmustererregung vorbehalten.

DIE INBETRIEBNAHME IHRES SYSTEMS

2.4 WIE MAN DISKETTEN FORMATIERT

Bevor das Betriebssystem MS-DOS eine neue Diskette benutzen kann, müssen Sie diese zunächst "formatieren".

Leere Disketten sind mit dem MS-DOS-Befehl FORMAT zu formatieren. Durch den Befehl FORMAT erhält die Diskette ein Format, mit dem MS-DOS arbeiten kann; ferner untersucht dieser Befehl die Diskette auf defekte Spuren. Wenn die zu formatierende Diskette nicht leer ist, löscht das Formatierungsprogramm alle bereits auf der Diskette gespeicherten Daten. Obgleich sich diese Vorgehensweise aufgrund ihrer problemlosen Anwendung zum Löschen einer Diskette anbietet, empfiehlt sich zu diesem Zweck jedoch die Benutzung des MS-DOS-Befehls DEL oder ERASE.

Nähere Informationen zum Befehl DELETE entnehmen Sie bitte Kapitel 5 "Verzeichnis der MS-DOS-Befehle".

2.4.1 Der Befehl FORMAT

Der Befehl FORMAT weist die folgende Syntax auf:
FORMAT [d:]

wobei:

d: die Laufwerksangabe ist (des Laufwerks, das die zu formatierende Diskette enthält).

Beachten Sie bitte, daß die eckigen Klammern wahlfreie Informationen kennzeichnen. Wenn Sie kein Diskettenlaufwerk angeben (z. B. A: oder B:), formatiert MS-DOS die in das Standardlaufwerk eingelegte Diskette.

DIE INBETRIEBNAHME IHRES SYSTEMS

Da die MS-DOS-System-Diskette bereits in Laufwerk A eingelegt ist, können Sie mit der Formatierung Ihrer neuen leeren Diskette beginnen. Mit Hilfe des folgenden Kommandos formatieren Sie die neue Diskette in Laufwerk B.

FORMAT B:

MS-DOS zeigt die folgende Meldung am Bildschirm:

```
Insert new disk for drive b:  
and strike any key when ready
```

Nachdem Sie die neue Diskette in Laufwerk B eingelegt und eine beliebige Taste auf der Tastatur gedrückt haben, antwortet das System:

```
Formatting...
```

Wenn Sie den Befehl `FORMAT` mit dem Zusatz `/V` eingeben, fragt MS-DOS nach der Diskettenbezeichnung (siehe weiter unten). Der Zusatz `/S` weist MS-DOS an, zuzüglich zu der Formatierung auch seine Systemdateien auf die neue Diskette zu kopieren.

Nach der Beendigung der Formatierung mit `"/S"-Option` zeigt MS-DOS in etwa die folgende Meldung am Bildschirm an:

```
Formatting...Format complete  
System transferred
```

Disketten-/Plattenkennsatz (11 Zeichen, ENTER falls keiner gewünscht):

Diskettenbezeichnungen sind sehr nützlich zur Kennzeichnung Ihrer Disketten. Sie sind wie für das Auge unsichtbare Kennzeichnungsetiketten.

DIE INBETRIEBNAHME IHRES SYSTEMS

Wenn Sie jeder Ihrer Disketten eine eigene Diskettenbezeichnung zugewiesen haben, wissen Sie immer, mit welcher Diskette Sie gerade arbeiten. Die Diskettenbezeichnung einer Diskette können Sie mit Hilfe des MS-DOS-Befehls VOL am Bildschirm prüfen (weitere Informationen über den Befehl VOL entnehmen Sie bitte Kapitel 5 "Verzeichnis der MS-DOS-Befehle"). Wenn Sie die eingelegte Diskette kennzeichnen wollen, geben Sie als Antwort auf die oben angeführte Eingabeaufforderung eine Diskettenbezeichnung ein und drücken Sie die RETURN-Taste. Eine mögliche Diskettenbezeichnung wäre z.B. PROGRAMME. Falls Sie jedoch der eingelegten Diskette keine Diskettenbezeichnung zuordnen wollen, drücken Sie nur die RETURN-Taste. Dann sehen Sie am Bildschirm eine Meldung, die in etwa wie folgt aussieht:

```
362406 bytes total disk space
 38912 bytes used by system
323584 bytes available on disk
```

```
Format another(Y/N)?
```

Geben Sie Y ein, wenn Sie eine weitere Diskette formatieren wollen. Geben Sie N ein, wenn Sie das Programm FORMAT beenden möchten.

2.5 WIE MAN DISKETTEN SICHERT

Es ist sehr empfehlenswert, Sicherungskopien von allen Disketten dazu erstellen, mit denen man arbeitet. Denn wenn eine Diskette beschädigt oder eine Datei versehentlich gelöscht wird, stehen Ihnen dann alle - sonst verlorenen - Informationen noch auf Ihrer Sicherungsdiskette zur Verfügung. Auch von Ihrer MS-DOS-System-Diskette sollten Sie eine Sicherungskopie erstellen. Aus Sicherheitsgründen sollten Sie nicht mit der Original-Systemdiskette arbeiten, sondern mit der Sicherungskopie. Die Erstellung von Sicherungskopien erfolgt über den Befehl MS-DOS DISKCOPY, der im folgenden beschrieben wird.

DIE INBETRIEBNAHME IHRES SYSTEMS

2.5.1 DER BEFEHL DISKCOPY

Mit Hilfe des Befehls DISKCOPY können Sie den Inhalt einer Diskette auf eine andere Diskette kopieren. Dieser Befehl ist gleichermaßen zum Kopieren der MS-DOS-System-Diskette und der Disketten geeignet, die Ihre eigenen Dateien enthalten. Der Befehl DISKCOPY stellt die schnellste Möglichkeit zum Kopieren einer Diskette dar, da Sie mit seiner Hilfe die gesamte Diskette einschließlich MS-DOS-Systemdateien, falls vorhanden, in einem Arbeitsgang kopieren können.

Der Befehl DISKCOPY hat das folgende Format:

```
DISKCOPY [Laufwerk1:] [Laufwerk2:]
```

Laufwerk1 ist das Diskettenlaufwerk, das die zu kopierende oder "Ausgangs"-Diskette enthält; Laufwerk 2 enthält die leere oder "Ziel"-Diskette. Vor der Benutzung des Befehls DISKCOPY ist die leere Diskette zu formatieren.

Wenn Sie zum Beispiel eine Kopie Ihrer MS-DOS-System-Diskette erstellen wollen, die sich in Laufwerk A befindet, geben Sie ein:

```
DISKCOPY A: B:
```

MS-DOS antwortet mit den folgenden Meldungen:

```
Insert source disk into drive A:  
Insert target disk into drive B:  
Press any key when ready
```

Vergewissern Sie sich, daß die MS-DOS-System-Diskette in Laufwerk A eingelegt ist, und legen Sie eine leere, formatierte Diskette in Laufwerk B. Betätigen Sie dann eine beliebige Zeichentaste, worauf MS-DOS mit dem Kopieren der MS-DOS-Diskette beginnt. Wenn MS-DOS mit dem Kopieren der Diskette fertig ist, zeigt es am Bildschirm an:

```
Copy complete  
Copy another(Y/N)?
```

Geben Sie Y (für Yes) ein, wenn Sie mit dem Befehl DISKCOPY eine weitere Diskette kopieren wollen. Wenn Sie N (für Nein) eingeben, wird die Standardlaufwerk-Eingabeaufforderung am Bildschirm angezeigt.

Wiedergabe sowie Vervielfältigung durch jedwede Art von Medien, ohne schriftliche Genehmigung der Nixdorf Computer AG, sind ausdrücklich untersagt. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustererteilung vorbehalten.

DIE INBETRIEBNAHME IHRES SYSTEMS

In Laufwerk B befindet sich nun eine Kopie Ihrer MS-DOS-System-Diskette, die Sie als Sicherungskopie aufbewahren können.

Die Abbildung 2 veranschaulicht die Funktionsweise des Befehls DISKCOPY:

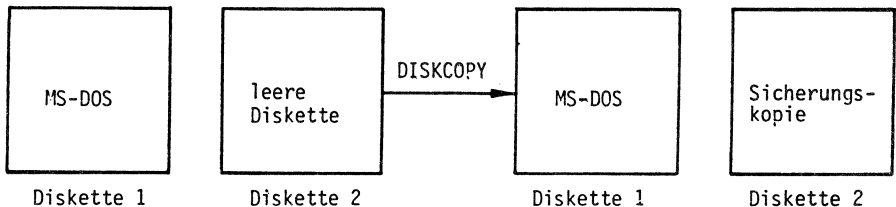


Abbildung 2. Der Befehl DISKCOPY

Bei Benutzung des Befehls DISKCOPY müssen die Ausgangs- und die Zieldiskette dasselbe Format und dieselbe Aufzeichnungsdichte aufweisen. Weitere Informationen über den Befehl DISKCOPY entnehmen Sie bitte Kapitel 5 "Verzeichnis der MS-DOS-Befehle".

HINWEIS

Wenn eine Diskette defekte Spuren enthält, kann DISKCOPY nicht arbeiten. Sichern Sie in einem solchen Fall Ihre Disketten mit Hilfe des Befehls COPY. (Der Befehl COPY überspringt defekte Spuren.) Informationen über die Benutzung des Befehls COPY zur Sicherung Ihrer Disketten entnehmen Sie bitte Kapitel 5 "Verzeichnis der MS-DOS-Befehle".

DIE INBETRIEBNAHME IHRES SYSTEMS

2.6 DER AUTOMATISCHE PROGRAMMAUFRUF

Wenn Sie bei jedem Start von MS-DOS ein bestimmtes Programm automatisch zur Ausführung bringen wollen, können Sie dazu den automatischen Programmaufruf benutzen. Zum Beispiel kann es für Sie nützlich sein, daß MS-DOS jedesmal, wenn es geladen wird, automatisch die Namen Ihrer Dateien am Bildschirm anzeigt.

Wenn Sie MS-DOS starten, sucht der Befehlsprozessor auf der MS-DOS-System-Diskette nach einer Datei mit dem Dateinamen AUTOEXEC.BAT (Datei für den automatischen Programmaufruf). Das in dieser Datei enthaltene Programm wird von MS-DOS jedesmal automatisch ausgeführt, wenn MS-DOS gestartet wird. Informieren Sie sich bitte in Kapitel 4 "Wissenswertes über die MS-DOS- Befehle", wie man eine AUTOEXEC.BAT-Datei erstellt.

2.7 DIE DATEIEN

2.7.1 WAS IST EINE DATEI?

Eine Datei ist eine Zusammenstellung von Informationen, die zueinander in unmittelbarer Beziehung stehen. Eine auf Diskette gespeicherte Datei ist mit einer Kartei in einer Schreibtischschublade vergleichbar. Wenn eine Kartei zum Beispiel die Namen und Adressen aller Angestellten in einem Büro enthält, dann kann man ihr den Namen "Personalkartei" geben. Entsprechend könnte man einer auf einer Diskette gespeicherten Datei mit demselben Inhalt den Namen "Personaldatei" geben.

Alle auf Ihren Disketten gespeicherten Programme, Texte und Daten sind in Dateien angeordnet, von denen jede einen nur einmal vorkommenden Namen trägt. Diesen Namen benutzen Sie, um eine bestimmte Datei anzusprechen. Wie man Dateien benennt, erfahren Sie in Kapitel 3 "Eingehendere Informationen über Dateien".

Jedesmal, wenn Sie Daten oder Text eingeben und speichern, erstellen Sie eine Datei. Dies ist auch der Fall, wenn Sie Programme schreiben, benennen und auf Disketten speichern.

DIE INBETRIEBNAHME IHRES SYSTEMS

2.7.2 WIE MS-DOS IHRE DATEIEN VERWALTET?

Die Dateien werden auf der Diskette in "Arbeitsbereichen" gespeichert und die Dateinamen in "Inhaltsverzeichnissen" angeordnet, die zusätzliche Informationen über den Umfang der Dateien, deren Speicherplatz auf der Diskette sowie deren Erstellungs- und letztes Bearbeitungsdatum enthalten. Jedem Arbeitsbereich ist ein Inhaltsverzeichnis zugeordnet. Wenn Sie auf Ihrer Diskette nur einen Arbeitsbereich angelegt haben, ist das Inhaltsverzeichnis dieses Arbeitsbereichs gleichzeitig auch das Inhaltsverzeichnis Ihrer Diskette. Den Arbeitsbereich, mit dem Sie gerade arbeiten, nennen wir Ihren momentanen Arbeitsbereich.

Ein weiterer Bereich des Systems heißt "Dateispeicherplatz-tabelle". Diese Tabelle hält die Speicherplätze Ihrer Dateien auf der Diskette fest und weist auch die freien Speicherplätze auf der Diskette aus, so daß Sie neue Dateien erstellen können.

Diese beiden Systembereiche, die Arbeitsbereiche mit ihren Inhaltsverzeichnissen und die Dateispeicherplatz-tabelle, ermöglichen es MS-DOS, die auf Ihren Disketten gespeicherten Dateien zu erkennen und sie zuzuordnen. Wenn Sie eine neue Diskette mit dem MS-DOS-Befehl FORMAT formatieren, wird die Dateispeicherplatz-tabelle auf diese Diskette kopiert und ein leerer Arbeitsbereich mit Inhaltsverzeichnis, der Grundarbeitsbereich, erstellt.

DIE INBETRIEBNAHME IHRES SYSTEMS

2.7.3 DER BEFEHL DIR (ZEIGE INHALTSVERZEICHNIS)

Um festzustellen, welche Dateien auf Ihrer Diskette gespeichert sind, können Sie den Befehl DIR benutzen. Mit diesem Befehl weisen Sie MS-DOS an, die Namen aller Dateien im momentanen Arbeitsbereich der benannten Diskette am Bildschirm anzuzeigen. Wenn zum Beispiel Ihre MS-DOS-System-Diskette in Laufwerk A eingelegt ist und Sie sehen wollen, welche Dateien sich im momentanen Arbeitsbereich dieser Diskette befinden, geben Sie ein:

DIR A:

MS-DOS antwortet darauf mit der Auflistung aller Dateien, die im Inhaltsverzeichnis des momentanen Arbeitsbereichs auf Ihrer MS-DOS-System-Diskette angeführt sind. Die Anzeige dürfte in etwa wie folgt aussehen:

```
Volume in drive A has no label
Directory of A:\
COMMAND  COM      16276   10-29-81   11:48a
DEBUG    COM      11534   10-28-82    9:21a
CHKDSK   COM       6272   10-26-82   12:12p
SYS      COM       1400   10-29-82    6:30p
EDLIN    COM       4419    1-01-80   12:41a
RECOVER  COM       2281   10-29-82    5:37p
PRINT    COM      3899   10-27-82   12:19p
LINK     EXE     41856    8-31-82    1:14p
FORMAT   COM      5605   10-28-82    9:55a
EXEFIX   COM      1350   10-06-82    2:57p
SORT     EXE      1280   10-27-82    3:18p
MORE     COM       291   10-27-82    3:20p
FIND     EXE     5888   01-01-80   12:57a
CONFIG   SYS        33   10-18-82    5:02p
LOCATE   EXE     5888   10-27-82   12:53p
FC       EXE    10624   10-27-82    7:00p
LOGIN    COM       299   10-18-82    6:30p

17 File(s)                23040 Bytes free
```

HINWEIS

Zwei MS-DOS-Systemdateien, IO.SYS und MS-DOS.SYS, sind unsichtbare Dateien und erscheinen nicht am Bildschirm, wenn Sie den Befehl DIR eingeben.

"Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung hinsichtlich dieser Unterlage ist ohne schriftliche Genehmigung des Nixdorf Computer AG. Nachdruck, Verbreitung, Übersetzung, Vervielfältigung, Verbreitung, Weitergabe oder Gebrauchsmusterentziehung vorbehalten."

DIE INBETRIEBNAHME IHRES SYSTEMS

Durch Eingabe von DIR und einem Dateinamen können Sie sich ferner Informationen über jede beliebige Datei auf Ihrer Diskette am Bildschirm anzeigen lassen. Wenn Sie zum Beispiel eine Datei mit dem Namen TOPDATEI.TXT erstellt haben, erhalten Sie durch den Befehl

```
DIR TOPDATEI.TXT
```

die Bildschirmanzeige aller Informationen aus dem Eintrag der Datei TOPDATEI.TXT im Inhaltsverzeichnis des momentanen Arbeitsbereichs.

Weitere Informationen über den Befehl DIR entnehmen Sie bitte Kapitel 5 "Verzeichnis der MS-DOS-Befehle".

2.7.4 DER BEFEHL CHKDSK (=CHECKDiSK = DISKETTE/PLATTE PRÜFEN)

Der MS-DOS-Befehl CHKDSK dient zur Überprüfung Ihrer Disketten auf organisatorische Folgerichtigkeit und Fehler und ähnelt in seiner Funktion deshalb einer Sekretärin, die ein Schreiben korrektur liest. Dieser Befehl analysiert die Inhaltsverzeichnisse der Arbeitsbereiche und die Dateispeicherplatztafel auf der von Ihnen angegebenen Diskette und erstellt dann einen Prüfbericht, der alle aufgefundenen Fehler enthält, wie zum Beispiel Dateien, in deren Verzeichnis sich eine von Null verschiedene Größe befindet, obwohl sie keine Daten enthalten.

Um die Diskette in Laufwerk A zu überprüfen, geben Sie ein:

```
CHKDSK A: und RETURN
```

MS-DOS zeigt einen Prüfbericht und alle gefundenen Fehler der Diskette in Laufwerk A am Bildschirm an. Ein Beispiel für diese Anzeige und weitere Informationen über den Befehl CHKDSK entnehmen Sie bitte der Beschreibung dieses Befehls in Kapitel 5. Jede Diskette sollte gelegentlich mit Hilfe des Befehls CHKDSK geprüft werden, um den einwandfreien Zustand Ihrer Dateien sicherzustellen.

DIE INBETRIEBNAHME IHRES SYSTEMS

2.8 WIE MAN DAS SYSTEM ABSCHALTET

MS-DOS benötigt keinen "Abschalt"-Befehl. Um Ihre Arbeitssitzung zu beenden, öffnen Sie einfach die Diskettenlaufwerkschächte und nehmen die Disketten heraus. Wenn die Standardlaufwerk-Eingabeaufforderung am Bildschirm erscheint, schalten Sie Ihren Computer ab.

HINWEIS

Entfernen Sie Ihre Disketten grundsätzlich immer aus den Laufwerken, bevor Sie Ihren Computer abschalten.

Zusammenfassung der in diesem Kapitel beschriebenen Befehle

BEFEHL	VERWENDUNGSZWECK	SYNTAX
FORMAT	Formatierung der Disketten für MS-DOS	FORMAT [d:]
DISKCOPY	Kopieren von Disketten	DISKCOPY [Laufwerk1:] [Laufwerk2:]
DIR	Anzeigen der Informationen aus dem Inhaltsverzeichnis	DIR [d:] [Dateiname]
CHKDSK	Fehlersuche auf Diskette/Platte	CHKDSK [d:]

Im nächsten Kapitel werden Sie mehr über MS-DOS-Dateien erfahren.

Weitergabe sowie Verwendung ohne schriftliche Genehmigung ist untersagt. Verantwortung für Missbrauch, Schäden oder Verlust von Daten liegt ausschließlich bei dem Benutzer. Die Haftung für Schäden ist ausgeschlossen. Alle Rechte für den Fall der Patentierung oder Gebrauchsmusteranmeldung vorbehalten.

EINGEHENDERE INFORMATIONEN DER DATEIEN

3 EINGEHENDERE INFORMATIONEN DER DATEIEN

3.1	Wie man Dateinamen festlegt.....	3 - 2
3.2	Die Stellvertreterzeichen.....	3 - 4
3.2.1	Das Stellvertreterzeichen ?.....	3 - 4
3.2.2	Das Stellvertreterzeichen *.....	3 - 5
3.3	Unzulässige Dateinamen.....	3 - 6
3.4	Wie man Dateien kopiert.....	3 - 7
3.5	Wie man Dateien sichert.....	3 - 9
3.6	Arbeitsbereiche und Inhaltsverzeichnisse.....	3 - 10
3.7	Dateinamen und Suchwege.....	3 - 13
3.7.1	Suchwegbezeichnungen.....	3 - 13
3.7.2	Suchwege und externe Befehle.....	3 - 15
3.7.3	Suchwege und interne Befehle.....	3 - 16
3.7.4	Die Anzeige Ihres momentanen Arbeitsbereichs..... am Bildschirm.....	3 - 17
3.7.5	Das Anlegen eines Arbeitsbereichs.....	3 - 18
3.7.6	Wie man den Arbeitsbereich wechselt.....	3 - 19



In Kapitel 2 haben Sie gelernt, daß die Inhaltsverzeichnisse der Arbeitsbereiche den Namen Ihrer Dateien enthalten. In diesem Kapitel werden Sie sich mit der Festlegung des Namens und dem Kopieren Ihrer Dateien vertraut machen. Sie werden auch einiges mehr über den hierarchischen Aufbau der MS-DOS-Arbeitsbereiche und -Inhaltsverzeichnisse erfahren, was Ihnen sowohl Zuordnung als auch Suche Ihrer Dateien erleichtert.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung des Inhalts, der Zeichnungen und der Tabellen, sowie die Weitergabe von Kopien, Nachbildungen, Reproduktionen, Vervielfältigungen, Übersetzungen, Mikrofilm- und Mikroformausgaben, sowie die Verbreitung und die öffentliche oder geschäftliche Verbreitung, ist ohne schriftliche Genehmigung der Nixdorf Computer AG. Alle Rechte für den Fall der Patent-, Marken- und Gebrauchsmustereintragung vorbehalten.

EINGEHENDERE INFORMATIONEN DER DATEIEN

3.1 WIE MAN DATEINAMEN FESTLEGT

Der Name einer typischen MS-DOS-Datei sieht folgendermaßen aus:

NEUDATEI.EXE

Die komplette Bezeichnung einer Datei besteht aus zwei Teilen: dem Dateinamen NEUDATEI und dem Dateinamensuffix EXE.

Ein Dateiname kann bis zu 8 Zeichen lang sein. Das Dateinamensuffix kann eine Länge von maximal 3 Zeichen haben. Sie können jeden Dateinamen nach Belieben in Klein- oder Großbuchstaben eingeben, da MS-DOS jeden eingegebenen Kleinbuchstaben automatisch in einen Großbuchstaben umwandelt.

Zusätzlich zum Dateinamen und zum Dateinamensuffix kann die Dateibezeichnung eine Laufwerksangabe enthalten. Die Laufwerksangabe zeigt MS-DOS, auf welcher Diskette bzw. in welchem Laufwerk nach dem eingegebenen Dateinamen zu suchen ist. Um zum Beispiel im Inhaltsverzeichnis gespeicherte Informationen über die Datei NEUDATEI.EXE zu erhalten, die sich auf der Diskette in Laufwerk A befindet (wobei Laufwerk A nicht das Standardlaufwerk ist), geben Sie den folgenden Befehl ein:

DIR A:NEUDATEI.EXE

Nun werden die im Inhaltsverzeichnis gespeicherten Informationen über die Datei NEUDATEI.EXE am Bildschirm angezeigt.

Wenn Laufwerk A das Standardlaufwerk ist, sucht MS-DOS nur auf der Diskette in Laufwerk A nach dem Dateinamen NEUDATEI.EXE, weshalb die Laufwerksangabe in diesem Fall nicht erforderlich ist. Sie wird nur dann benötigt, wenn Sie MS-DOS die Anweisung geben wollen, auf der Diskette im anderen Laufwerk nach einer Datei zu suchen.

EINGEHENDERE INFORMATIONEN DER DATEIEN

Ihre Dateinamen bestehen wahrscheinlich nur aus Buchstaben und Zahlen, aber es sind außerdem auch noch andere Zeichen zulässig. Für den Aufbau der Dateinamensuffixe sind dieselben Zeichen zulässig wie für den der Dateinamen. Im folgenden finden Sie ein vollständiges Verzeichnis der Zeichen, die Sie zur Gestaltung von Dateinamen und Dateinamensuffixen verwenden können:

A-Z 0-9 \$ & #
% ' () - ©
^ { } ~ \ !

Die Gesamtheit aller Angaben, die dazu dienen, eine Datei zu bezeichnen, nennen wir "Dateibezeichnung". Die Dateibezeichnung, im Sinne dieses Handbuchs, umfaßt also grundsätzlich:

[<Laufwerksangabe:>]<Dateiname>[<.Dateinamensuffix>]

Denken Sie bitte daran, daß eckige Klammern auf wahlfreie Eingaben hinweisen. Spitze Klammern (<>) bedeuten, daß Sie den Text für den jeweiligen Bestandteil der Dateibezeichnung selbst einsetzen müssen. Beachten Sie bitte, daß die Angabe der Laufwerksangabe nicht erforderlich ist, solange Sie nicht MS-DOS darauf hinweisen müssen, auf welcher Diskette in welchem Laufwerk es nach einer bestimmten Datei suchen soll. Ferner ist es nicht unbedingt erforderlich, Ihrem Dateinamen ein Dateinamensuffix hinzuzufügen.

Beispiele für Dateibezeichnungen:

B:MEINPROG.COB
A:IHRPROG.SUF
B:NEUDATEI.SUF

EINGEHENDERE INFORMATIONEN DER DATEIEN

3.2 DIE STELLVERTRETERZEICHEN

Zwei Sonderzeichen (Stellvertreterzeichen) können in Dateinamen und Suffixen benutzt werden: das Sternchen (*) und das Fragezeichen (?). Diese Sonderzeichen erhöhen Ihre Flexibilität bei der Benutzung von Dateinamen in MS-DOS-Befehlen.

3.2.1 Das Stellvertreterzeichen ?

Ein Fragezeichen (?) in einem Dateinamen oder Dateinamensuffix bedeutet, daß jedes beliebige Zeichen an der betreffenden Position stehen kann. Der MS-DOS-Befehl

```
DIR W?RT.EXE
```

zum Beispiel bewirkt, daß am Bildschirm alle Eintragungen des Inhaltsverzeichnisses im Standardlaufwerk angezeigt werden, die aus vier Buchstaben bestehen, mit W beginnen, ein beliebiges folgendes Zeichen enthalten, mit den Buchstaben RT enden und das Dateinamensuffix .EXE besitzen. Im folgenden sind als Beispiele einige Dateien aufgeführt, deren Ausgabe durch den Befehl DIR bewirkt werden könnte:

```
T.EXE  
WERT.EXE  
WIRT.EXE  
WORT.EXE
```

EINGEHENDERE INFORMATIONEN DER DATEIEN

3.2.2 Das Stellvertreterzeichen *

Ein Sternchen (*) als letztes Zeichen in einem Dateinamen oder Dateinamensuffix weist darauf hin, daß an dieser Stelle und in den restlichen Schreibstellen des Dateinamens oder des Dateinamensuffixes beliebige Zeichen stehen können.

Zum Beispiel:

DIR TEST*.EXE

bewirkt, daß alle Eintragungen des Inhaltsverzeichnisses im Standardlaufwerk am Bildschirm angezeigt werden, deren Dateinamen mit der Zeichenfolge TEST beginnen und das Suffix .EXE besitzen. Im folgenden sind als Beispiele einige Dateien aufgeführt, auf die sich der Befehl DIR in diesem Fall auswirken würde:

TEST.EXE
TEST1.EXE
TEST27.EXE
TEST674.EXE
TESTLAUF.EXE

Die nur aus Stellvertreterzeichen bestehende Dateibezeichnung *.* bezieht sich daher auf alle Dateien, die auf der Diskette gespeichert sind. Beachten Sie bitte, daß diese Datei- bezeichnungsmöglichkeit, wenn sie in MS-DOS-Befehlen benutzt wird, einerseits sehr leistungsfähig und andererseits sehr gefährlich sein kann. Der Befehl DEL *.* zum Beispiel bewirkt, daß alle Dateien im Standardlaufwerk ungeachtet ihrer Dateinamen oder Suffixe gelöscht werden!

Beispiele:

Um alle Dateien des Inhaltsverzeichnisses in Laufwerk A mit dem Dateinamen NEUDATEI, ungeachtet ihrer Dateinamensuffixe, am Bildschirm zu prüfen, geben Sie nur ein:

DIR A:NEUDATEI.*



Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden, Zwitterung der Rechte vorbehalten. Copyright © 1988 Nixdorf Computer AG. Alle Rechte vorbehalten.

INGEHENDERE INFORMATIONEN DER DATEIEN

Um alle Dateien des Inhaltsverzeichnisses in Laufwerk B mit den Dateinamensuffixen .TXT, ungeachtet ihrer Dateinamen, am Bildschirm zu prüfen, geben Sie ein:

```
DIR B:?????????.TXT oder  
DIR B: *.TXT
```

Die Benutzung dieses Befehls ist sinnvoll, wenn Sie zum Beispiel allen Ihren Textprogrammen das Dateinamensuffix .TXT gegeben haben. Wenn Sie dann den Befehl DIR mit acht Stellvertreterzeichen benutzen, können Sie alle Ihre Textdateien am Bildschirm prüfen, ohne sämtliche Textdateien im Gedächtnis haben zu müssen.

3.3 UNZULÄSSIGE DATEINAMEN

MS-DOS weist einigen Gerätenamen eine besondere Funktion zu. Für diese Gerätenamen sind daher bestimmte Zeichenkombinationen vorgesehen, die aus drei Buchstaben bestehen und nicht als Dateinamen oder Dateinamensuffixe benutzt werden dürfen. Die folgenden Namen sollten Sie daher nicht als Dateinamen benutzen:

AUX Dient zur Bezugnahme auf Ein- oder Ausgaben eines Zusatzgeräts (wie z.B. eines Druckers oder Diskettenlaufwerks).

CON Dient zur Bezugnahme auf Eingaben über die Tastatur bzw. auf Ausgaben über den Bildschirm.

LST oder

PRN Dient zur Bezugnahme auf den Drucker.

NUL Diesen Namen können Sie benutzen, wenn Sie keine Datei erstellen wollen, aber ein Befehl die Festlegung eines Dateinamens für eine Ein- oder Ausgabe erforderlich macht.

Auch wenn diese Dateinamen zusammen mit Laufwerksangaben oder Dateinamensuffixen verwendet werden, bleibt ihre Zuordnung zu den oben angeführten Geräten bestehen. Die Bezeichnung A:CON.XXX bezieht sich, trotz Laufwerksangabe und Suffix, grundsätzlich auf Tastatur und Bildschirm und ist nicht der Name einer Datei auf der Diskette.

EINGEHENDERE INFORMATIONEN DER DATEIEN

3.4 WIE MAN DATEIEN KOPIERT

Wie von einer Kartei auf Papier benötigen Sie oft auch von einer Datei auf Diskette mehr als ein Exemplar. Der Befehl COPY ermöglicht es Ihnen, eine oder mehrere Dateien auf eine andere Diskette zu kopieren. Sie können ferner Ihrer Kopie einen neuen Dateinamen geben, wenn Sie diesen Namen im Befehl COPY festlegen.

Mit Hilfe des Befehls COPY können Sie auch Kopien einer Datei auf derselben Diskette erstellen. In einem solchen Fall müssen Sie MS-DOS einen anderen Dateinamen angeben, da Sie sonst die ursprüngliche Datei überschreiben würden. Es ist daher in MS-DOS unmöglich, eine Datei auf dieselbe Diskette zu kopieren, wenn Sie nicht für die neue Kopie einen anderen Dateinamen festlegen.

Der Befehl COPY hat das folgende Format:

Quelldatei
Zielfile
 COPY Dateibezeichnung [Dateibezeichnung]

Zum Beispiel:

COPY A:TOPDATEI.TXT B:TOPDATEI.TXT

bewirkt, daß die Datei TOPDATEI.TXT von der Diskette in Laufwerk A in eine Datei mit dem(selben) Namen TOPDATEI.TXT auf der Diskette in Laufwerk B kopiert wird. Nun verfügen Sie zusätzlich zum Original über eine Kopie von TOPDATEI.TXT vorhanden.

Abbildung 3 veranschaulicht, wie eine Datei auf eine andere Diskette kopiert wird:

COPY

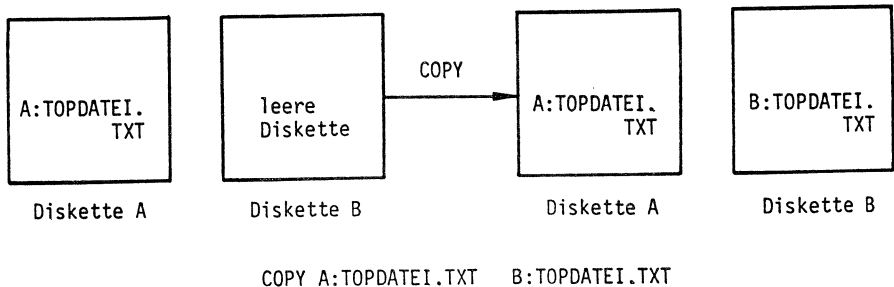


Abbildung 3. Das Kopieren einer Datei auf eine andere Diskette



EINGEHENDERE INFORMATIONEN DER DATEIEN

Wenn Sie die Datei mit dem Namen TOPDATEI.TXT auf dieselbe Diskette kopieren wollen, geben Sie ein:

COPY A:TOPDATEI.TXT A:NEUNAME.TXT

Nun befinden sich zwei Exemplare Ihrer Datei auf Diskette A - eine mit dem Namen TOPDATEI.TXT und die andere mit dem Namen NEUNAME.TXT. Die folgende Abbildung veranschaulicht dieses Beispiel.

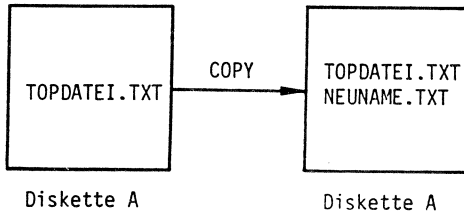


Abbildung 4. Das Kopieren einer Datei auf dieselbe Diskette

Mit Hilfe des Befehls COPY können Sie ferner alle Dateien einer Diskette auf eine andere Diskette kopieren (um zum Beispiel eine Sicherungskopie zu erstellen). Weitere Informationen über diesen Vorgang finden Sie im Kapitel 5 "Verzeichnis der MS-DOS-Befehle".

EINGEHENDERE INFORMATIONEN DER DATEIEN

3.5 WIE MAN DATEIEN SICHERT

MS-DOS ist ein leistungsfähiges und zuverlässiges Werkzeug für die Verarbeitung Ihrer persönlichen und geschäftlichen Informationen. Wie bei jedem anderen Informationssystem können jedoch auch hier Fehler auftreten und Informationen fehlgeleitet werden. Wenn Sie Informationen verarbeiten, die unersetzlich sind oder einen besonderen Schutz erfordern, sollten Sie daher geeignete Maßnahmen ergreifen, um Ihre Daten und Programme vor unbefugter Benutzung und versehentlicher Änderung oder Zerstörung zu schützen. Relativ einfache Vorkehrungen - wie zum Beispiel das Entfernen Ihrer unbenutzten Disketten aus dem Laufwerk, die Archivierung von Sicherungskopien von wertvollen Informationen und die Installation Ihrer Geräte an einem sicheren Ort - können dazu beitragen, die Sicherheit der Informationen in Ihren Dateien zu gewährleisten.



"Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung handlungen verpflichten zu Schadensersatz. Alle Rechte, für den Fall der Patenterteilung oder Gebrauchsmustereingtragung vorbehalten."

EINGEHENDERE INFORMATIONEN DER DATEIEN

3.6 ARBEITSBEREICHE UND INHALTSVERZEICHNISSE

Wie Sie in Kapitel 2 gelernt haben, werden die Namen der Dateien jeder Diskette in ein Verzeichnis eingetragen, das ferner Informationen über den Umfang, den Speicherplatz sowie das Erstellungsdatum und das Datum der letzten Bearbeitung jeder Datei enthält.

Wenn mehrere Personen an Ihrem Computer arbeiten oder Sie mit einer größeren Zahl verschiedener Projekte zu tun haben, kann die Zahl der Dateien im Verzeichnis sehr schnell groß und unübersichtlich werden. In einem solchen Fall wird es erforderlich sein, Ihre Dateien getrennt von denen Ihrer Kollegen zu halten bzw. Ihre Programme in "Arbeitsbereiche" einzuteilen, die Ihnen die Arbeit erleichtern.

In einem Büro trennt man verschiedene Karteien voneinander, indem man sie in verschiedene Karteikästen einordnet; dadurch weist man verschiedenen Informationen verschiedene Organisationsbereiche zu. MS-DOS ermöglicht es Ihnen, die Dateien auf Ihren Disketten in verschiedene Arbeitsbereiche mit eigenen Inhaltsverzeichnissen zusammenzufassen, wodurch Ihr Datei-Organisation überschaubarer wird.

Es kann zum Beispiel sinnvoll sein, alle Buchführungsprogramme in einen Arbeitsbereich "Buchführung" und alle Textdateien in einen Arbeitsbereich "Text" einzuordnen. Jeder Arbeitsbereich kann in seinem Inhaltsverzeichnis, in einem vernünftigen Rahmen, eine beliebig große Zahl von Dateien aufnehmen und sogar weitere Arbeitsbereiche enthalten (die dann als "Unter-Arbeitsbereiche" bezeichnet werden). Diese Form der Organisation der Dateien und Arbeitsbereiche nennt man "hierarchisch".

Die hierarchische Dateienverwaltung kann mit einer "Baumstruktur" verglichen werden: Die Arbeitsbereiche mit Ihren Inhaltsverzeichnissen sind die Zweige des Baums, und die Dateien sind die Blätter, aber mit dem Unterschied, daß ein solcher "Baum" nach unten wächst; die "Wurzel" ist also dort, wo sich normalerweise die Krone befindet, und stellt die erste Ebene der Organisation dar.

EINGEHENDERE INFORMATIONEN DER DATEIEN

Ein Arbeitsbereich mit Inhaltsverzeichnis wird automatisch erstellt, wenn Sie eine Diskette formatieren und Dateien auf ihr zu speichern beginnen. Gemäß den Anweisungen in Kapitel 4 "Wissenswertes über die MS-DOS-Befehle" können Sie weitere Arbeitsbereiche und Unter-Arbeitsbereiche erstellen.

Die Baumstruktur wird entsprechend erweitert, wenn Sie zur Einordnung neuer Dateigruppen oder für weitere Benutzer des Systems neue Arbeitsbereiche erstellen. Innerhalb jedes neuen Arbeitsbereichs können wiederum Dateien hinzugefügt oder neue Unter-Arbeitsbereiche erstellt werden.

In dieser Baumstruktur kann man sich frei bewegen; man kann zum Beispiel jede beliebige Datei im System suchen, indem man bei der Wurzel beginnt und sich auf einem der Zweige nach unten bewegt, bis man bei der gewünschten Datei angelangt ist. Umgekehrt können Sie Ihre Reise auch da beginnen, wo Sie sich im Dateienverwaltungssystem gerade befinden und sich in Richtung Wurzel bewegen.

Wenn Sie Ihren Computer einschalten, befinden Sie sich automatisch in dem Arbeitsbereich, der der Wurzel unserer Baumstruktur entspricht und den wir den "Grundarbeitsbereich" nennen wollen. In diesem Grundarbeitsbereich wird auch automatisch jede neue Datei erstellt, wenn Sie nichts anderes veranlassen. Wie wir bereits gesehen haben, können Sie jedoch auf Ihren Disketten so viele Arbeitsbereiche anlegen, wie Sie wollen. Diese Arbeitsbereiche können Sie verschiedenen Benutzern zuordnen. Dadurch können verschiedene Benutzer ohne weiteres Dateien mit demselben Namen halten, da diese als Bestandteile verschiedener Arbeitsbereiche nicht miteinander in Beziehung gesetzt werden. Um vom Grundarbeitsbereich in die anderen Arbeitsbereiche zu gelangen, müssen Sie MS-DOS entsprechende Anweisungen geben.

Abbildung 5 veranschaulicht ein typisches Dateienverwaltungssystem mit hierarchischen Arbeitsbereichen.

```

GRUNDARBEITSBEREICH

SPIELE  BIN    BENUTZER  BUCHFOHRUNG  PROGRAMM

        MEIER  BACH     HUBER

                TEXT.TXT

        TEXT.TXT  FORMULAR
    
```



Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihrer Inhalte, Vervielfältigung, Verbreitung, auch auszugsweise, ist ohne schriftliche Genehmigung der Nixdorf Computer AG. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustereintragung vorbehalten.

EINGEHENDERE INFORMATIONEN DER DATEIEN

Der Grundarbeitsbereich stellt die erste Ebene in der Organisation dar. Mit Hilfe des Befehls MKDIR können Sie, vom Grundarbeitsbereich ausgehend, Unter-Arbeitsbereiche erstellen (nähere Informationen über MKDIR entnehmen Sie bitte Kapitel 5 "Verzeichnis der MS-DOS-Befehle"). In diesem Beispiel sind fünf Unter-Arbeitsbereiche des Grundarbeitsbereichs erstellt worden. Es handelt sich dabei um die folgenden:

1. Ein Spiele-Arbeitsbereich mit dem Namen SPIELE
2. Ein Arbeitsbereich für die Gesamtheit der externen Befehls mit dem Namen BIN (weitere Informationen über den Arbeitsbereich BIN entnehmen Sie bitte Kapitel 4 "Wissenswertes über die MS-DOS- Befehle").
3. Ein Arbeitsbereich BENUTZER mit verschiedenen Unter- Arbeitsbereichen für alle Benutzer des Computersystems
4. Ein Arbeitsbereich mit dem Namen BUCHFRNG, der die Buchführungsdateien enthält
5. Ein Arbeitsbereich für Programme mit dem Namen PROGRAMM.

Herr Meier, Frau Bach und Herr Huber besitzen jeweils ihren eigenen Arbeitsbereich, der ein Unter-Arbeitsbereich des Arbeitsbereichs BENUTZER ist. Frau Bach verfügt in dem Arbeitsbereich \BENUTZER \BACH über ein Unterverzeichnis mit dem Namen FORMULAR. Frau Bach und Herr Huber haben beide in ihren Verzeichnissen Dateien, die den Namen TEXT.TXT tragen, aber in keiner Weise miteinander in Beziehung stehen.

Diese Organisation von Dateien und Arbeitsbereichen ist für Sie von geringer Bedeutung, solange Sie sich bei Ihrer Arbeit auf die Dateien eines einzigen, nämlich die Ihres eigenen Arbeitsbereichs beschränken; wenn Sie jedoch noch mit jemand anders zusammenarbeiten oder wenn Sie sich gleichzeitig mehreren Projekten widmen müssen, ist die Arbeit mit hierarchischen Arbeitsbereichen sehr vorteilhaft. So können Sie sich zum Beispiel aus Frau Bachs Arbeitsbereich FORMULAR das komplette Inhaltsverzeichnis ausgeben lassen, indem Sie eingeben:

```
DIR \ BENUTZER \ BACH \ FORMULAR
```

Beachten Sie bitte, daß der verkehrte Schrägstrich (\) zur Abgrenzung der Arbeitsbereiche gegenüber den übrigen Arbeitsbereichen und Dateien dient.

Um herauszufinden, welche Dateien sich in Herrn Hubers Arbeitsbereich befinden, könnten Sie eingeben:

```
DIR \ BENUTZER \ HUBER
```

EINGEHENDERE INFORMATIONEN DER DATEIEN

3.7 DATEINAMEN UND SUCHWEGE

Wenn Sie mit einem hierarchischen Dateienverwaltungssystem arbeiten und eine Datei suchen wollen, müssen Sie MS-DOS mitteilen, an welcher Stelle dieses Systems sich die gesuchte Datei befindet. Zum Beispiel verfügen sowohl Frau Bach als auch Herr Huber über Dateien mit dem Namen TEXT.TXT. Wenn einer von ihnen auf seine Datei zugreifen möchte, muß er zunächst MS-DOS mitteilen, in welchem Arbeitsbereich sich seine Datei befindet. Dies erfolgt, indem man MS-DOS mittels einer Suchwegbezeichnung zu der gewünschten Datei führt.

3.7.1 Suchwegbezeichnungen

Eine Datei- bzw. Suchwegbezeichnung ist eine Folge von Angaben zu Aufbewahrungsort (Laufwerk), Erkennungsmerkmalen (Name) und Art (Suffix) einer Datei. Eine Suchwegbezeichnung ist eine Folge von beliebig vielen Arbeitsbereichsnamen und einem Dateinamen, die voneinander jeweils durch einen negativen Schrägstrich (\) abgegrenzt sind. Die Arbeitsbereichsnamen beschreiben den Weg, den man vom hierarchisch höchsten zum nächst niedrigeren Arbeitsbereich usw. in der Baumstruktur zurücklegen muß, um die Datei zu finden.

Suchwegbezeichnungen weisen die folgende Syntax auf:

[<d>:] [<Arbeitsbereich>] \ [<Arbeitsbereich...>] \ [<Dateiname>]

Wenn eine Suchwegbezeichnung mit einem negativen Schrägstrich beginnt, sucht MS-DOS die gewünschte Datei ab der Wurzel (d.h. dem oberen Teil) des Baumes bzw. im Grundarbeitsbereich. In allen anderen Fällen beginnt MS-DOS mit der Suche im momentanen Arbeitsbereich des Benutzers und setzt die Suche von dort aus nach unten in der Baumstruktur fort. Die Suchwegbezeichnung von Frau Bachs TEXT.TXT-Datei ist beispielsweise \BENUTZER \ BACH \ TEXT.TXT.



EINGEHENDERE INFORMATIONEN DER DATEIEN

Wenn Sie sich in Ihrem eigenen Arbeitsbereich befinden, können Sie anstatt der Suchwegbezeichnung auch den einfachen Dateinamen verwenden.

Beispiele:

\	Weist auf den Grundarbeitsbereich hin.
\PROGRAMM	Beispielarbeitsbereich mit Programmdateien, der innerhalb des Grundarbeitsbereichs angeordnet ist.
\BENUTZER\BACH\FORMULAR\1A	Eine typische vollständige Suchwegbezeichnung.

Diese weist auf eine Datei mit dem Namen 1A in einem Arbeitsbereich mit dem Namen FORMULAR hin, der dem BENUTZER mit dem Namen Bach gehört.

INGEHENDERE INFORMATIONEN DER DATEIEN

BENUTZER \ HUBER

Eine relative Suchwegbezeichnung; sie benennt die Datei oder den Arbeitsbereich HUBER im Unter-Arbeitsbereich BENUTZER des momentanen Arbeitsbereichs. Wenn der momentane Arbeitsbereich der Grundarbeitsbereich (\) ist, hat dieser Name die Form BENUTZER \ HUBER.

TEXT.TXT Name einer Datei oder eines Arbeitsbereichs (mit Suffix) im momentanen Arbeitsbereich.

MS-DOS stellt besondere Kürzel zur Bezeichnung des momentanen Arbeitsbereichs sowie des dem momentanen Arbeitsbereich unmittelbar übergeordneten Arbeitsbereichs zur Verfügung:

- . Unter diesem Kürzel versteht MS-DOS in allen hierarchischen Verzeichnissen den Namen des momentanen Arbeitsbereichs. Sobald ein neuer Arbeitsbereich angelegt wird, erstellt MS-DOS automatisch diese Eingabe.
- .. Ist das Kürzel für den dem momentanen Arbeitsbereich unmittelbar übergeordneten Arbeitsbereich.

Wenn Sie eingeben: DIR ..

gibt MS-DOS das Verzeichnis der Dateien im unmittelbar übergeordneten Arbeitsbereich des momentanen Arbeitsbereichs aus.

Wenn Sie eingeben: DIR .. \ ..

gibt MS-DOS die Dateien im ÜBERGEORDNETEN Arbeitsbereich des übergeordneten Arbeitsbereichs aus.

3.7.2 Suchwege und externe Befehle

Externe Befehle sind in Form von Programmdateien auf Diskette gespeichert und müssen vor ihrer Ausführung von der Diskette in den Arbeitsspeicher eingelesen werden. (Weitere Informationen über externe Befehle entnehmen Sie bitte Kapitel 4 "Wissenswertes über die MS-DOS-Befehle".)

Wenn Sie mit mehr als einem Arbeitsbereich arbeiten, empfiehlt es sich, alle externen MS-DOS-Befehle in einem getrennten Arbeitsbereich zusammenzufassen, so daß sie keine Unordnung in Ihre übrigen Arbeitsbereiche bringen. Wenn Sie MS-DOS einen externen Befehl geben, durchsucht MS-DOS sofort Ihren momentanen Arbeitsbereich nach diesem Befehl. Mit Hilfe des Befehls PATH können Sie MS-DOS dort mitteilen, in welchem anderen Arbeitsbereich die externen Befehle gespeichert sind, wenn dies der Fall ist.



Wahrheits- und Verifizierung durch Umfragen, Bewertung und Mithaltung ihrer Basis nicht gestattet. Sie sind nicht für den Verkauf von Patent-Handlungen verpflichtend zu Schadenersatz. Alle Rechte für den Fall der Patent-erteilung oder Gebrauchsmusteranmeldung vorbehalten.

EINGEHENDERE INFORMATIONEN DER DATEIEN

Wenn Sie sich zum Beispiel in einem Arbeitsbereich mit dem Namen \BIN \PROG befinden und alle externen MS-DOS-Befehle in \BIN gespeichert sind, müssen Sie MS-DOS mitteilen, daß es den Weg \BIN wählen soll, um den Befehl FORMAT zu finden. Der Befehl PATH \BIN weist MS-DOS an, nach sämtlichen Befehlen in Ihrem momentanen Arbeitsbereich und dem Arbeitsbereich \BIN zu suchen. Es genügt, MS-DOS ein einziges Mal während der Arbeitssitzung diesen Suchweg anzugeben, damit es die externen Befehle grundsätzlich im Arbeitsbereich \BIN sucht. Wenn Sie den eingegebenen momentanen Suchweg herausfinden wollen, geben Sie nur das Wort PATH ein: der momentane Suchweg wird dann angezeigt.

Weitere Informationen über den MS-DOS-Befehl PATH entnehmen Sie bitte Kapitel 5 "Verzeichnis der MS-DOS-Befehle".

3.7.3 Suchwege und interne Befehle

Interne Befehle sind die einfachsten und gebräuchlichsten Befehle. Sie werden unmittelbar nach ihrer Eingabe ausgeführt, da sie alle im Befehlsprozessor stehen. (Weitere Informationen über interne Befehle entnehmen Sie bitte Kapitel 4 "Wissenswertes über die MS-DOS-Befehle".)

Einige interne Befehle können mit Suchweg arbeiten. Die vier Befehle COPY, DIR, DEL und TYPE sind flexibler in der Ausführung, wenn Sie nach deren Eingabe einen Suchweg eingeben.

Die Syntax dieser vier Befehle ist wie folgt:

COPY <Suchwegbezeichnung> <Suchwegbezeichnung>

Wenn die zweite Suchwegbezeichnung im Befehl COPY ein Arbeitsbereich ist, werden alle Dateien in diesen Arbeitsbereich kopiert. Die erste Suchwegbezeichnung dient dazu, die zu kopierenden Dateien im momentanen Arbeitsbereich anzugeben.

DEL <Suchwegbezeichnung>

Wenn die Suchwegbezeichnung für einen Arbeitsbereich steht, werden alle Dateien dieses Arbeitsbereichs gelöscht.

EINGEHENDERE INFORMATIONEN DER DATEIEN



Hinweis:

Wenn Sie einen Suchweg zu löschen versuchen, wird die Eingabeaufforderung "Bitte betätigen (Y/N)?" am Bildschirm angezeigt. Geben Sie Y ein, wenn Sie den Befehl ausführen wollen, oder N, wenn Sie den Befehl zurücknehmen möchten.

DIR <Suchwegbezeichnung>

Dient zur Anzeige des einem bestimmten Suchweg entsprechenden Inhaltsverzeichnisses.

TYPE <Suchwegbezeichnung>

Um diesen Befehl auszuführen, müssen Sie im Suchweg eine Datei angeben. MS-DOS antwortet auf den Befehl TYPE mit Suchwegbezeichnung, indem es die gesuchte Datei am Bildschirm anzeigt.

3.7.4 Die Anzeige Ihres momentanen Arbeitsbereichs am Bildschirm

Alle Befehle werden in Ihrem momentanen Arbeitsbereich ausgeführt. Den Namen dieses Arbeitsbereichs können Sie herausfinden, indem Sie den MS-DOS-Befehl CHDIR (= CHange DIrectory = Arbeitsbereich wechseln) ohne Option eingeben. Wenn Ihr momentaner Arbeitsbereich zum Beispiel der Arbeitsbereich BENUTZER MEIER ist und Sie eingeben:

CHDIR <RETURN>

wird am Bildschirm angezeigt:

A: \ BENUTZER \ MEIER

Dies ist Ihre momentane Laufwerksangabe mit der Bezeichnung Ihres momentanen Arbeitsbereichs (\ BENUTZER \ MEIER).

Wenn Sie nun das Inhaltsverzeichnis des Arbeitsbereichs \BENUTZER \MEIER am Bildschirm sehen wollen, brauchen Sie nur den MS-DOS-Befehl DIR einzugeben. Im folgenden sehen Sie ein Beispiel dafür, was am Bildschirm erscheinen kann, wenn Sie durch Eingabe des Befehls DIR die Angabe des Inhaltsverzeichnisses eines Unter-Arbeitsbereichs anfordern:

Diskette/Platte in Laufwerk A hat keinen Kennsatz Inhaltsverzeichnis von A: \BENUTZER \MEIER

		<DIR>					
.		<DIR>	8-09-82	10:09a			
..		<DIR>	8-09-82	10:09a			
TEXT		<DIR>	8-09-82	10:09a			
DATEI1	COM		5243	8-04-82	9:30a		

4 File(s)

8376320 Bytes free

Weitergabe sowie Vervielfältigung dieses Untertages, Verwertung und Mitteilung desselben oder einzelner Teile ist ohne schriftliche Genehmigung des Nixdorf Computer AG. Es sind keine Gewährleistungen hinsichtlich der Richtigkeit der Angaben zu machen. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustererteilung vorbehalten.

EINGEHENDERE INFORMATIONEN DER DATEIEN

Der eingelegten Diskette wurde bei der Formatierung kein Kennsatz zugeordnet. Beachten Sie bitte, daß MS-DOS in diesem Inhaltsverzeichnis sowohl Dateien als auch Arbeitsbereiche anführt. Wie Sie sehen können, hat Herr Meier in dieser Baumstruktur noch einen weiteren Arbeitsbereich mit dem Namen TEXT angelegt. Das Kürzel "." weist auf den momentanen Arbeitsbereich \BENUTZER\MEIER hin, und das Kürzel ".." steht für den übergeordneten Arbeitsbereich BENUTZER. DATEI1.COM ist eine Datei des Arbeitsbereichs \BENUTZER\MEIER. Alle aufgeführten Arbeitsbereiche und ihre Dateien sind auf der Diskette in Laufwerk A gespeichert.

Da Dateien und Arbeitsbereiche zusammen verwaltet werden (siehe obiges Beispiel), läßt es MS-DOS nicht zu, daß ein Unter-Arbeitsbereich denselben Namen wie eine seiner Dateien erhält. Wenn zum Beispiel ein Suchweg \BIN\BENUTZER\MEIER mit MEIER als Unter-Arbeitsbereich angegeben ist, können Sie im Arbeitsbereich BENUTZER nicht auch gleichzeitig eine Datei mit dem Namen MEIER erstellen.

3.7.5 Das Anlegen eines Arbeitsbereichs

Um in Ihrem momentanen Arbeitsbereich einen Unter-Arbeitsbereich anzulegen, wählen Sie den Befehl MKDIR (= Make DIRectory = Arbeitsbereich anlegen). Um zum Beispiel in Ihrem momentanen Arbeitsbereich einen neuen Arbeitsbereich mit dem Namen NEUBEREC zu erstellen, geben Sie ein:

MKDIR NEUBEREC

Nachdem MS-DOS diesen Befehl ausgeführt hat, enthält Ihre Baumstruktur innerhalb Ihres momentanen Arbeitsbereichs einen neuen Arbeitsbereich (Unter-Arbeitsbereich). Ferner können Sie Arbeitsbereiche an jeder beliebigen Stelle in der Baumstruktur einfügen, indem Sie MKDIR und anschließend eine Suchwegbezeichnung eingeben. MS-DOS erstellt dann automatisch die Eingaben . und .. im Inhaltsverzeichnis des neuen Arbeitsbereichs.

Um Dateien in den neuen Arbeitsbereich einzufügen und in seinem Inhaltsverzeichnis einzutragen, benutzen Sie den Zeilen-Editor von MS-DOS, EDLIN. Wie man EDLIN zur Erstellung und Editierung von Dateien benutzt, wird in Kapitel 7 "Der Zeilen-Editor (EDLIN)" beschrieben.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4 WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.1	Einführung.....	4 - 2
4.2	Die verschiedenen Befehlsarten.....	4 - 3
4.3	Die Optionen der Befehle.....	4 - 4
4.4	Gemeinsame Eigenschaften aller MS-DOS-Befehle.....	4 - 6
4.5	Die Stapelprogrammbausteine.....	4 - 8
4.6	Der Stapelprogrammbaustein AUTOEXEC.BAT.....	4 - 11
4.6.1	Wie man den Stapelprogrammbaustein..... AUTOEXEC.BAT erstellt	4 - 13
4.7	Stapelprogrammbausteine mit Blindparametern.....	4 - 14
4.7.1	Die Ausführung eines Stapelprogrammbausteins.....	4 - 15
4.8.	Eingabe und Ausgabe.....	4 - 16
4.8.1	Neubestimmung des Ausgabeziels.....	4 - 17
4.8.2	Verarbeitungsbefehle.....	4 - 18
4.8.3	Die Befehlsverkettung.....	4 - 18

4

„Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung
ihres Inhalts ist ohne schriftliche Genehmigung des Nixdorf Computer AG
hinsichtlich der Weitergabe und Verwertung dieser Unterlage verpflichtet zu Schadensersatz. Alle Rechte für den Fall der Patent-
erteilung oder Gebrauchsmusterertragung vorbehalten.“

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.1 EINFÜHRUNG

Befehle stellen für den Benutzer eine Möglichkeit dar, mit dem Computer zu kommunizieren. Durch die Eingabe von MS-DOS-Befehlen über Ihre Tastatur können Sie das System dazu veranlassen, nützliche Funktionen auszuführen. MS-DOS-Befehle ermöglichen folgende Funktionen:

Vergleichen, Kopieren, Anzeigen, Löschen und Umbenennen von Dateien

Kopieren und Formatieren von Disketten

Die Ausführung von Systemprogrammen, wie zum Beispiel EDLIN, sowie die Ihrer eigenen Programme

Prüfung und Anzeige von Arbeitsbereichs-Inhaltsverzeichnissen

Eingabe von Datum, Uhrzeit und Anmerkungen

Einstellung verschiedener Drucker- und Bildschirmoptionen

Kopieren von MS-DOS-Systemdateien auf andere Disketten

MS-DOS für eine bestimmte Zeitdauer in Wartestellung bringen.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.2 DIE VERSCHIEDENEN BEFEHLSARTEN

Es gibt zwei Arten von MS-DOS-Befehlen:

- interne Befehle und
- externe Befehle

Interne Befehle sind die einfachsten und gebräuchlichsten Befehle. Sie können diese Befehle nicht sehen, wenn Sie sich das Inhaltsverzeichnis des Arbeitsbereichs auf Ihrer MS-DOS-System-Diskette am Bildschirm anzeigen lassen, da sie Teil des Befehlsprozessors sind. Wenn Sie diese Befehle eingeben, werden sie daher unverzüglich ausgeführt. Die folgenden internen Befehle stehen Ihnen zur Verfügung; sie werden in Kapitel 5 im einzelnen beschrieben:

BREAK	DEL (ERASE)	MKDIR (MD)	SET
CHDIR (CD)	DIR	PATH	SHIFT
CLS	ECHO	PAUSE	TIME
COPY	EXIT	PROMPT	TYPE
CTTY	FOR	REM	VER
DATE	GOTO	REN (RENAME)	VERIFY
IF	RMDIR (RD)	VOL	

Externe Befehle sind als Programmdateien auf Diskette gespeichert. Vor ihrer Ausführung müssen sie von der Diskette in den Arbeitsspeicher eingelesen werden. Wenn die Diskette, die den Befehl enthält, nicht in das Laufwerk eingelegt ist, kann MS-DOS den Befehl nicht finden und daher auch nicht ausführen.

Jede Datei, deren Dateibezeichnung mit dem Dateinamensuffix .COM, .EXE oder .BAT versehen ist, gilt als externer Befehl. So sind zum Beispiel Programme wie FORMAT.COM und COMP.COM externe Befehle. Da alle externen Befehle auf Diskette gespeichert sind, können Sie jederzeit neue externe Befehle erstellen und sie in das System einfügen. Programme, die Sie in den meisten der gebräuchlichen Programmiersprachen (einschließlich Assembler-sprache) erstellen können, sind als Dateien mit dem Suffix .EXE (executable = ausführbar) zu speichern.

Fügen Sie bitte nicht das Dateinamensuffix hinzu, wenn Sie einen externen Befehl eingeben!

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

Die folgenden externen Befehle werden in Kapitel 5 beschrieben:

CHKDSK	MORE
DISKCOPY	PRINT
FIND	RECOVER
FORMAT	SORT
EXE2BIN	SYS

4.3 DIE OPTIONEN DER BEFEHLE

Viele MS-DOS-Befehle arbeiten mit Optionen, die es Ihnen ermöglichen, zusätzliche Informationen in das System einzugeben. Wenn Sie bei diesen Befehlen keine Option eingeben, benutzt MS-DOS entsprechende Standardvorgaben. Nähere Informationen über Standardvorgaben finden Sie in den einzelnen Beschreibungen der Befehle in Kapitel 5.

Für alle MS-DOS-Befehle gilt das folgende Format:

Befehl [Optionen...]

wobei:

d: das Diskettenlaufwerk bezeichnet.

Dateiname

für einen beliebigen gültigen Namen oder für eine Diskettendatei steht, einschließlich eines wahlfreien Dateinamensuffixes. Der Dateiname steht nicht für eine Geräte- oder Diskettenlaufwerksangabe.

.SUF

für ein wahlfreies Dateinamensuffix steht, das aus einem Punkt und 1 bis 3 Zeichen besteht. Falls verwendet, folgen Dateinamensuffixe unmittelbar auf die Dateinamen selbst.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

Dateibezeichnung

für eine wahlfreie Laufwerksangabe, einen Dateinamen und ein wahlfreies, aus drei Buchstaben bestehendes Dateinamensuffix steht.

Format: [<d:>] <Dateiname> [<.suf>]

Suchwegbezeichnung für eine Suchwegbezeichnung oder einen Dateinamen mit dem folgenden Format steht:

[<Arbeitsbereich>] / [<Arbeitsbereich...>] / [<Dateiname>]

Zusatz

Optionen dienen zur Feinsteuerung der MS-DOS-Befehle.

Ihnen ist ein verkehrter Schrägstrich vorangestellt (zum Beispiel /P).

Argument

Argumente stellen zusätzliche Informationen zur Ausführung der MS-DOS-Befehle zur Verfügung. In der Regel erfolgt eine Wahl zwischen verschiedenen möglichen Argumenten, so zum Beispiel zwischen ON oder OFF.



Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden, Zuwiderhandlung strafbar. Alle Rechte vorbehalten. Für den Fall der Patenterteilung oder Gebrauchsmustererteilung vorbehalten.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.4 GEMEINSAME EIGENSCHAFTEN ALLER MS-DOS-BEFEHLE

Die nachfolgend beschriebenen Eigenschaften und Besonderheiten gelten grundsätzlich für alle MS-DOS-Befehle:

1. Zu den meisten Befehlen gehört eine oder mehrere Optionen.
2. Sowohl die Befehle als auch die Optionen können in Großbuchstaben oder in Kleinbuchstaben oder in einer Kombination von beiden eingegeben werden.
3. Befehle und Optionen müssen bei der Eingabe durch Abgrenzungszeichen getrennt werden. Dafür können mehrere Zeichen verwendet werden. Als leicht zu merkende Zeichen empfehlen wir die Verwendung des Leerzeichens und des Kommas als Abgrenzungszeichen.

Zum Beispiel:

```
DEL NEUDATEI.ALT TOPDATEI.TXT  
RENAME,DIESEDAT JENEDAT
```

Sie können jedoch außer Leerzeichen und Komma auch den Strichpunkt (;), das Gleichheitszeichen (=) oder einen Tabspung in Ihren MS-DOS-Befehlen als Abgrenzungszeichen verwenden.

In diesem Handbuch werden wir grundsätzlich ein Leerzeichen als Abgrenzungszeichen in unseren Befehlen benutzen.

4. Eine Dateibezeichnung dürfen Sie nicht mit Abgrenzungszeichen trennen, da der Doppelpunkt nach der Laufwerksangabe und der Punkt vor dem Suffix bereits diese Funktion erfüllen.
5. Wenn die Eingabeaufforderung "Press any key" am Bildschirm erscheint, können Sie jede beliebige Buchstabentaste von A bis Z oder jede numerische Taste von 0 bis 9 betätigen.
6. Wenn ein Dateiname mit einem Suffix versehen ist, müssen Sie dieses Suffix immer angeben, wenn Sie auf diese Datei Bezug nehmen, beispielsweise um sie aufzurufen.
7. Einen bereits ablaufenden Befehl können Sie jederzeit abbrechen, indem Sie die CONTROL-Taste und die C-Taste gleichzeitig drücken. Zur Darstellung dieses Vorgangs werden wir im weiteren Verlauf dieses Handbuchs die Schreibweise <CONTROL-C> verwenden.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

8. Ein Befehl, den Sie eingegeben haben, wird grundsätzlich erst ausgeführt, wenn Sie danach die RETURN-Taste drücken.
9. Im Befehl selbst dürfen weder Stellvertreterzeichen noch Gerätenamen (wie zum Beispiel PRN oder CON) vorkommen.
10. Wenn ein Befehl die Ausgabe einer umfangreichen Datei am Bildschirm bewirkt, d.h. wenn der Bildschirm schnell vollgeschrieben wird, bewirkt MS-DOS einen automatischen Bildlauf. Sie können diesen Bildlauf jederzeit unterbrechen, indem Sie <CONTROL-S> drücken. Wenn Sie die Anzeige wieder aufnehmen wollen, genügt es, eine beliebige Taste zu betätigen.
11. Zur Eingabe von Befehlen können sowohl die Bearbeitungs- als auch die Funktionstasten von MS-DOS benutzt werden. Nähere Einzelheiten hierzu entnehmen Sie bitte Kapitel 6 "Sonderbearbeitungstasten und Steuerzeichenfunktionen in MS-DOS", das eine komplette Beschreibung dieser Tasten enthält.
12. Die Eingabeaufforderung des Befehlsprozessors besteht aus der Angabe des Standarddiskettenlaufwerks und einem Größer-als-Zeichen, also zum Beispiel: A>.
13. Um die beiden Diskettenlaufwerke unterscheiden zu können, werden wir sie jeweils als Ausgangs- oder Ziellaufwerk ansprechen. Das Ausgangslaufwerk ist das Laufwerk, von dem die Informationen stammen, die Sie übertragen. Das Ziellaufwerk dagegen ist das Laufwerk, für das die Informationen bestimmt sind, die Sie übertragen.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.5 DIE STAPELPROGRAMMBAUSTEINE

Lange Befehlsfolgen sind manchmal recht umständlich zu schreiben und damit auch relativ fehlerträchtig. Muß man solche Befehlsfolgen häufig eingeben, dann kann die Sache schon ärgerlich werden. MS-DOS bietet Ihnen daher die Möglichkeit, eine solche Befehlsfolge als Stapelprogrammbausteindatei abzuspeichern und die gesamte Befehlsfolge zur Ausführung zu bringen, indem Sie nur einfach den Namen dieser Stapelprogrammbausteindatei eingeben. Der Stapelprogrammbaustein, d.h. die ganze Befehlsfolge, wird dann genauso abgearbeitet, als ob Sie sie Zeichen für Zeichen über die Tastatur eingegeben hätten. Eine Stapelprogrammbausteindatei muß durch ein geeignetes Suffix (.BAT für "BATch" = Stapel) gekennzeichnet werden; wenn Sie die Datei aufrufen wollen, um den Stapelprogrammbaustein auszuführen, brauchen Sie dieses Suffix nicht eigens einzugeben.

Erstellen können Sie eine Stapelprogrammbausteindatei entweder mit Hilfe des Zeilen-Editors (EDLIN) oder durch Eingabe des Befehls COPY. Nähere Einzelheiten zur Verwendung des Befehls COPY zur Erstellung eines Stapelprogrammbausteins entnehmen Sie bitte dem Abschnitt "Wie man den Stapelprogrammbaustein AUTOEXEC.BAT erstellt" weiter unten in diesem Kapitel.

Zwei MS-DOS-Befehle wurden ausdrücklich für die Verwendung in Stapelprogrammbausteindateien geschaffen; es handelt sich um die Befehle REM und PAUSE. Der Befehl REM ermöglicht es Ihnen, in Ihre Stapelprogrammbausteine Bemerkungen (REM = REMark = Bemerkung) einzufügen, die jedoch vom Befehlsprozessor nicht als Befehle angesehen und daher auch nicht als solche ausgeführt werden. Der Befehl PAUSE ermöglicht es Ihnen, in Ihrem Stapelprogrammbaustein einen Haltepunkt vorzusehen, wo MS-DOS eine Eingabeaufforderung ausgibt, deren Inhalt Sie selbst bestimmen können; Sie können dann die Ausführung des Stapelprogrammbausteins je nach Antwort auf die Eingabeaufforderung fortsetzen oder an dem Haltepunkt abbrechen. Die Befehle REM und PAUSE werden in Kapitel 5 im einzelnen beschrieben.

Stapelprogrammbausteine sind vor allen Dingen dann sehr nützlich, wenn Sie mehrere MS-DOS-Befehle mit einem einzigen Befehlsaufruf ausführen möchten, zum Beispiel, wenn Sie eine neue Diskette sowohl formatieren als auch prüfen möchten. Ein Stapelprogrammbaustein mit dem Namen NEUDISK, der dies ermöglicht, könnte beispielsweise wie folgt aussehen:

```
1:  REM Stapelprogrammbaustein zur Prüfung neuer Disketten
2:  REM Der Name des Stapelprogrammbausteins ist
    NEUDISK.BAT
3:  PAUSE Legen Sie eine neue Diskette in Laufwerk B ein.a
4:  FORMAT B:
5:  DIR B:
6:  CHKDSK B:
```

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

Wenn Sie diese Stapelprogrammbausteindatei ausführen möchten, geben Sie ganz einfach den Dateinamen ohne das Suffix .BAT ein:

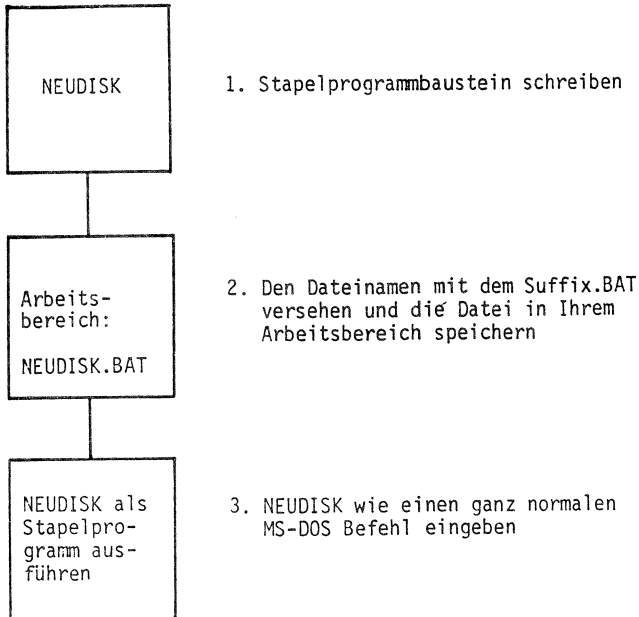
NEUDISK

Das Ergebnis wird genau dasselbe sein, als ob Sie die einzelnen Zeilen des Stapelprogrammbausteins als Einzelbefehle über die Tastatur eingegeben hätten.

Die drei Schritte, die erforderlich sind, um eine MS-DOS- Stapelprogrammbausteindatei zu schreiben, zu speichern und auszuführen, sind in Abbildung 6 dargestellt.

1. Stapelprogrammbaustein schreiben.
2. Den Dateinamen mit dem Suffix .BAT versehen und die Datei in Ihrem Arbeitsbereich speichern.
3. NEUDISK wie einen ganz normalen MS-DOS-Befehl eingeben.

Abbildung 6. Schritte zur Erstellung einer MS-DOS-Stapelprogrammbausteindatei



Wiedergabe sowie Vervielfältigung dieser Unterrichts-Veranstaltung und Mitteilung über die Inhalte ist ohne schriftliche Genehmigung der Nixdorf Computer AG. Alle Rechte vorbehalten. Die Haftung für Schäden, die durch die Benutzung dieser Unterrichts-Veranstaltung entstehen, ist ausgeschlossen.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

Im folgenden möchten wir Sie mit einigen Besonderheiten vertraut machen, die Sie unbedingt kennen sollten, bevor Sie eine Stapelprogrammbausteindatei mit MS-DOS erstellen und ausführen:

1. Zur Ausführung des Stapelprogrammbausteins bitte nur den Namen der Stapelprogrammbausteindatei, also ohne Suffix, eingeben.
2. Alle in der Datei mit dem Namen Dateiname .BAT enthaltenen Befehle werden ausgeführt.
3. Wenn Sie während des Ablaufs eines Stapelprogrammbausteins die Tasten <CONTROL-C> betätigen, erscheint folgende Eingabeaufforderung am Bildschirm:

Bearbeitung der Befehlsdatei abbrechen (Y/N)?

Wenn Sie jetzt Y eingeben, wird die Ausführung des Stapelprogrammbausteins an dieser Stelle abgebrochen, und die MS-DOS-Eingabeaufforderung erscheint am Bildschirm. Geben Sie jedoch N ein, dann wird der gerade in Ausführung befindliche Befehl abgebrochen und die Ausführung des Stapelprogrammbausteins mit dem nächsten Befehl in der Datei fortgesetzt.

4. Entfernen Sie die Diskette, auf der ein Stapelprogrammbaustein gespeichert ist, während dieser ausgeführt wird, dann fordert Sie MS-DOS auf, die Diskette wieder einzulegen, damit der nächste Befehl gelesen werden kann.
5. Der letzte Befehl in einem Stapelprogrammbaustein kann der Name einer anderen Stapelprogrammbausteindatei sein. Damit haben Sie die Möglichkeit, Stapelprogrammbausteine zu verketteten, d.h. am Ende eines abgearbeiteten Stapelprogrammbausteins jeweils einen weiteren zur Ausführung aufzurufen.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.6 DER STAPELPROGRAMMBAUSTEIN AUTOEXEC.BAT

Wie bereits in Kapitel 2 besprochen, haben wir einen Stapelprogrammbaustein mit dem Namen AUTOEXEC.BAT vorgesehen, der es Ihnen ermöglicht, automatisch bestimmte Programme auszuführen, wenn MS-DOS gestartet wird. Diese automatische Programmausführung ist sehr nützlich, wenn Sie ein bestimmtes Anwendungsprogramm (z.B. Microsoft Multiplan) unter MS-DOS ausführen möchten, oder wenn Sie möchten, daß MS-DOS jedesmal automatisch einen bestimmten Stapelprogrammbaustein ausführt, wenn Sie das System starten. Sie können sich dann die Notwendigkeit ersparen, zwei verschiedene Disketten einlegen zu müssen, indem Sie den Stapelprogrammbaustein AUTOEXEC.BAT verwenden.

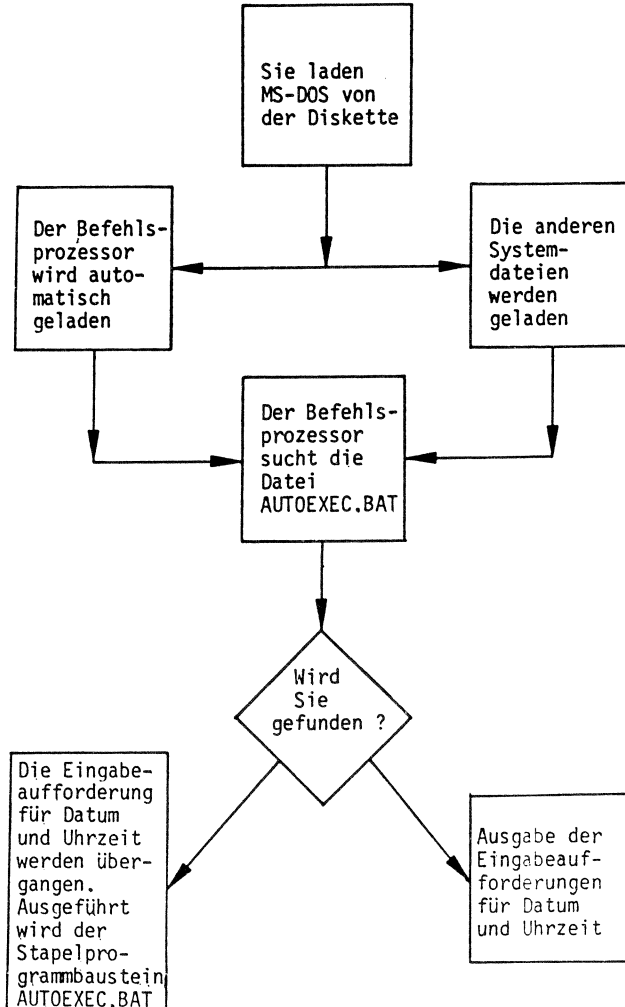
Wenn Sie MS-DOS starten, sucht der Befehlsprozessor automatisch die MS-DOS-System-Diskette nach einer Datei mit dem Namen AUTOEXEC.BAT ab. Die Datei AUTOEXEC.BAT ist ein Stapelprogrammbaustein, der jedesmal automatisch ausgeführt wird, wenn Sie das System starten.

Sobald MS-DOS die Datei AUTOEXEC.BAT findet, wird der Stapelprogrammbaustein durch den Befehlsprozessor ausgeführt und die Eingabeaufforderungen für Datum und Uhrzeit werden umgangen.

Findet MS-DOS jedoch keine Datei mit dem Namen AUTOEXEC.BAT, wenn Sie die System-Diskette einlegen und MS-DOS zum ersten Mal starten, dann werden die Eingabeaufforderungen für Datum und Uhrzeit ausgegeben. Abbildung 7 stellt dar, wie MS-DOS den Stapelprogrammbaustein AUTOEXEC.BAT verwendet.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

Abbildung 7. Wie MS-DOS den Stapelprogrammbaustein AUTOEXEC.BAT verwendet.



WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.6.1 Wie man den Stapelprogrammbaustein AUTOEXEC.BAT erstellt

Wenn Sie zum Beispiel jedesmal, wenn Sie MS-DOS starten, automatisch BASIC geladen haben möchten und ein Programm mit dem Namen MENU ausführen möchten, könnten Sie zu diesem Zweck den folgenden AUTOEXEC.BAT-Stapelprogrammbaustein erstellen:

1. Geben Sie ein:

```
COPY CON: AUTOEXEC.BAT
```

Mit dieser Anweisung wird MS-DOS veranlaßt, alle über die Tastatur (CON = console = Tastatur und Bildschirm) eingegebenen Informationen in die Datei AUTOEXEC.BAT zu kopieren. Beachten Sie bitte, daß der Stapelprogrammbaustein AUTOEXEC.BAT grundsätzlich im Grundarbeitsbereich Ihrer MS-DOS-System-Diskette stehen muß.

2. Nun geben Sie bitte ein:

```
BASIC MENU
```

Diese Anweisung wird in den Stapelprogrammbaustein AUTOEXEC.BAT geschrieben. Mit ihr wird MS-DOS angewiesen, jedesmal, wenn das System geladen wird, BASIC zu laden und das Programm MENU auszuführen.

3. Betätigen Sie die Tasten <CONTROL-Z> und dann die RETURN-Taste, womit Sie den Befehl BASICMENU in den Stapelprogrammbaustein AUTOEXEC.BAT übertragen.
4. Das Programm MENU wird nun jedesmal automatisch ausgeführt, sobald Sie MS-DOS starten.

Wenn Sie ein anderes BASIC-Programm ausführen möchten, das Sie erstellt haben, geben Sie ganz einfach den Namen dieses Programms anstatt MENU in der zweiten Zeile des Beispiels ein. In die Stapelprogrammbausteindatei AUTOEXEC.BAT können Sie jeden MS-DOS-Befehl, aber auch eine beliebige Folge von MS-DOS-Befehlen, eingeben.

HINWEIS

Beachten Sie bitte, daß MS-DOS die Eingabeaufforderungen für Datum und Uhrzeit nicht ausgibt, wenn Sie mit dem Stapelprogrammbaustein AUTOEXEC.BAT arbeiten, es sei denn, Sie fügen die Befehle DATE und TIME ausdrücklich in die Stapelprogrammbausteindatei AUTOEXEC.BAT ein. Wir empfehlen Ihnen dringend, dies zu tun, da MS-DOS diese Informationen benötigt, um Ihre Inhaltsverzeichnisse auf dem neuesten Stand zu halten.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.7 STAPELPROGRAMMBAUSTEINE MIT BLINDPARAMETERN

Es kann vorkommen, daß Sie ein Anwendungsprogramm erstellen möchten, das Sie öfter mit verschiedenen Daten ausführen wollen. Diese Daten können in verschiedenen MS-DOS-Dateien gespeichert sein.

In einem MS-DOS-Befehl ist ein Parameter eine Option, die Sie selbst bestimmen können. Um Ihr Problem zu lösen, bietet Ihnen MS-DOS die Möglichkeit, Stapelprogrammbausteine mit Blindparametern zu erstellen, d.h. mit Parametern, die bei jeder Ausführung des Stapelprogrammbausteins automatisch durch die jeweils benötigten ersetzt werden. MS-DOS stellt Ihnen für diese Blindparameter die Codes %0 bis %9 zur Verfügung.

Schauen wir uns einmal an, wie das in der Praxis funktioniert. Geben Sie die Zeile COPY CON:NEUDATEI.BAT ein, wodurch die nächsten eingegebenen Zeilen vom Bildschirm in die Datei NEUDATEI.BAT auf der Diskette im Standardlaufwerk übertragen werden:

```
A>COPY CON:NEUDATEI.BAT
COPY %1.MAC %2.MAC
TYPE %2.PRN
TYPE %0.BAT
```

Nun betätigen Sie die Taste CONTROL-Z und dann die RETURN-Taste. MS-DOS antwortet mit der Meldung:

```
1 File(s) copied
A>
```

Die Datei NEUDATEI.BAT, die aus drei Befehlen besteht, steht nun auf der Diskette im Standardlaufwerk.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

Die Blindparameter %1 und %2 werden bei der Ausführung des Stapelprogrammbausteins automatisch und nacheinander durch die Parameter ersetzt, die Sie angeben. Der Blindparameter %0 wird immer durch die Laufwerksangabe, falls angegeben, und den Dateinamen des Stapelprogrammbausteins (in unserem Beispiel NEUDATEI) ersetzt.

HINWEIS:

1. Sie können bis zu 10 Blindparameter (%0 bis %9) angeben. Wenn Sie mehr als 10 Blindparameter benötigen, informieren Sie sich bitte über den MS-DOS-Befehl SHIFT in Kapitel 5.
2. Wenn Sie innerhalb eines Stapelprogrammbausteins das Prozentzeichen als Teil eines Dateinamens verwenden wollen, müssen Sie es zweimal nacheinander eingeben. So müssen Sie zum Beispiel, um die Datei ABC%.EXE anzugeben, im Stapelprogrammbaustein ABC%%.EXE schreiben.

4.7.1 Die Ausführung eines Stapelprogrammbausteins

Wenn Sie nun den Stapelprogrammbaustein NEUDATEI.BAT ausführen wollen, müssen Sie die Parameter angeben, die anstatt der Blindparameter verwendet werden sollen. Dafür geben Sie den Dateinamen des Stapelprogrammbausteins (immer ohne Suffix!) ein und anschließend die Parameter, die MS-DOS anstelle von %1, %2 usw. einsetzen soll.

Beachten Sie bitte, daß die Datei NEUDATEI.BAT 3 Zeilen umfaßt:

```
COPY %1.MAC %2.MAC      TYPE %2.PRN
TYPE %0.BAT
```

Um nun den Stapelprogrammbaustein NEUDATEI auszuführen, geben Sie ein:

```
NEUDATEI A:PROG1 B:PROG2
```

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

Nun wird NEUDATEI anstelle von %0, A:PROG1 anstelle von %1 und B:PROG2 anstelle von %2 eingesetzt. Das Ergebnis ist genau dasselbe, als ob Sie jeden einzelnen der in der Datei NEUDATEI befindlichen Befehle mit den jeweiligen Parametern wie folgt eingegeben hätten:

```
COPY A:PROG1.MAC B:PROG2.MAC
TYPE B:PROG2.PRN
TYPE NEUDATEI.BAT
```

In der folgenden Tabelle ist beschrieben, wie MS-DOS beim Ersetzen der verschiedenen Parameter vorgeht:

NAME	PARAMETER1 (%0)	PARAMETER2 (%1)	PARAMETER3 (%2)
PROGRAMM- BAUSTEIN	(NEUDATEI)	(PROG1)	(PROG2)
NEUDATEI	NEUDATEI.BAT	PROG1.MAC	PROG2.MAC PROG2.PRN

Wie wir bereits gesehen haben, wird der Blindparameter %0 grundsätzlich durch den Laufwerksnamen (falls angegeben) und den Dateinamen des Stapelprogrammbausteins ersetzt.

4.8 EINGABE UND AUSGABE

MS-DOS geht grundsätzlich davon aus, daß Eingaben über die Tastatur erfolgen und Ausgaben auf dem Bildschirm. Selbstverständlich haben Sie jedoch die Möglichkeit, dies zu ändern. So kann beispielsweise eine Eingabe anstatt von der Tastatur auch von einer Datei kommen, und genauso kann eine Ausgabe entweder in eine Datei oder über einen Drucker erfolgen anstatt auf dem Bildschirm. Darüber hinaus besteht die Möglichkeit der Befehlsverkettung, womit Sie erreichen können, daß die Ausgabe eines Befehls von MS-DOS die Eingabe eines weiteren Befehls bewirkt. Die Neubestimmung des Ausgabeziels sowie die Befehlsverkettung werden in den folgenden Abschnitten dieses Kapitels eingehend behandelt.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.8.1 Neubestimmung des Ausgabeziels

Die meisten Befehle bewirken, daß die Ausgabe, die sie verursachen, zum Bildschirm Ihres Computers geleitet wird. MS-DOS ermöglicht es Ihnen, die Ausgabe anstatt auf den Bildschirm in eine Datei zu leiten, indem Sie in dem entsprechenden Befehl das Größer-als-Zeichen verwenden. Wie wir gesehen haben, bewirkt der Befehl

DIR

grundsätzlich die Anzeige am Bildschirm des Inhaltsverzeichnisses des momentanen Arbeitsbereichs der Diskette im Standardlaufwerk. Mit demselben Befehl können Sie jedoch die Ausgabe in eine Datei mit dem Namen NEUDATEI umleiten, indem Sie die Ausgabezielfeile im Befehl wie folgt angeben:

DIR>NEUDATEI

Wenn es eine Datei mit diesem Namen noch nicht gibt, dann erstellt MS-DOS Ihnen diese Datei automatisch und speichert das Arbeitsbereichs-Inhaltsverzeichnis dort ab. Liegt eine Datei NEUDATEI bereits vor, dann überschreibt MS-DOS den Inhalt dieser Datei mit den neuen Daten, was bedeutet, daß der alte Dateiinhalt gelöscht wird.

Wollen Sie jedoch den alten Dateiinhalt nicht löschen, sondern die neue Datei bzw. das auszugebende Inhaltsverzeichnis zuzüglich zu dem alten Dateiinhalt in die Datei NEUDATEI einbringen, dann müssen Sie zwei Größer-als-Zeichen (>>) eingeben, womit Sie MS-DOS anweisen, die durch den Befehl verursachte Ausgabe (in unserem Fall also Ihr Inhaltsverzeichnis) an das Ende des vorhandenen Dateiinhalts anzuhängen. So bewirkt das Kommando:

DIR>>NEUDATEI

daß Ihr Inhaltsverzeichnis an den bereits vorhandenen Inhalt der Datei NEUDATEI angefügt wird. Sollte die Datei NEUDATEI noch nicht existieren, wird sie von MS-DOS automatisch für Sie erstellt.

Es kann andererseits auch vorkommen, daß man die Eingabe für einen Befehl einer vorhandenen Datei entnehmen möchte, anstatt sie über die Tastatur einzugeben. Genau dies erreichen Sie in MS-DOS mit Hilfe eines Kleiner-als-Zeichens (<) in Ihrem Befehl. Der Befehl:

SORT<NAMEN>LISTE1

sortiert daher die Datei NAMEN und überträgt die sortierte Ausgabe in die Datei mit dem Namen LISTE1.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

4.8.2 Verarbeitungsbefehle

Ein Verarbeitungsbefehl liest Ihre Eingabe, verändert sie in einer bestimmten Art und Weise und gibt sie dann entweder am Bildschirm oder in eine Datei wieder aus, während ein Bearbeitungsbefehl die Daten, mit denen er arbeitet, nicht verändert. Da man Verarbeitungsbefehle in vielen verschiedenen Arten zusammenstellen kann, kann eine kleine Anzahl von Verarbeitungsbefehlen die Wirkung einer großen Anzahl von Einzelbefehlen haben.

Die Verarbeitungsbefehle von MS-DOS sind FIND, MORE und SORT.

Ihre Funktionen sind wie folgt:

FIND Sucht eine Datei nach einer konstanten Zeichenfolge ab.

MORE Bewirkt den Umbruch einer normalen Bildschirmausgabe in Bildschirmseiten, die eine nach der anderen angezeigt werden.

SORT Sortiert Text.

Wie diese Verarbeitungsbefehle angewendet werden, wird im nächsten Abschnitt beschrieben.

4.8.3 Die Befehlsverkettung

Wenn mehrere Befehle nacheinander auf dieselbe Datei angesetzt werden müssen, ist es oft nützlich, wenn man diese Befehle auf einmal eingeben kann. Die Besonderheit solcher Befehle ist, daß jeder von ihnen die Ausgabe seines Vorgängers als Eingabe verwendet. So könnte es zum Beispiel sinnvoll sein, an einen Befehl, der einen Text zur Anzeige bringt, einen weiteren Befehl anzuhängen, der eben diesen Text sortiert; damit würde erreicht, daß beide Aktionen in einem Zug durchgeführt werden.

Um Befehle zu verketteten, genügt es, sie durch ein Verkettungszeichen (|) verbunden hintereinander einzugeben. Um also zu erreichen, daß beispielsweise ein Inhaltsverzeichnis angezeigt und sortiert wird, brauchen Sie nur die beiden Befehle verbunden hintereinander einzugeben:

```
DIR | SORT
```

und Ihr Inhaltsverzeichnis wird alphabetisch sortiert am Bildschirm angezeigt. Der senkrechte Strich bewirkt, daß die von dem links von dem Strich stehenden Befehl kommenden Ausgaben dem rechts von ihm stehenden Befehl als Eingaben vorgegeben werden.

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

Selbstverständlich können Sie die Befehlsverkettung auch dann einsetzen, wenn die Ausgabe in eine Datei erfolgen soll. Wenn Sie beispielsweise Ihr Inhaltsverzeichnis in alphabetischer Reihenfolge in eine neue Datei eingeben möchten (zum Beispiel VERZ.FIL), brauchen Sie nur einzugeben:

```
DIR | SORT>VERZ.FIL
```

MS-DOS erstellt dann eine Datei mit dem Namen VERZ.FIL auf der Diskette in Ihrem Standardlaufwerk, da Sie kein anderes angegeben haben, und überträgt in diese Datei das alphabetisch geordnete Inhaltsverzeichnis des momentanen Arbeitsbereichs im Standardlaufwerk. Wenn Sie die Datei auf eine Diskette in einem anderen als dem Standardlaufwerk (z.B. Laufwerk B) gespeichert haben möchten, brauchen Sie nur einzugeben:

```
DIR | SORT>B:VERZ.FIL
```

womit Sie erreichen, daß das sortierte Verzeichnis in eine Datei mit dem Namen VERZ.FIL auf der Diskette im Laufwerk B gespeichert wird.

Eine Befehlskette kann auch mehr als zwei Befehle enthalten. So zum Beispiel:

```
DIR | SORT | MORE
```

womit Sie erreichen, daß Ihr Inhaltsverzeichnis sortiert und bildschirmseitenweise angezeigt wird, wobei am unteren Rand jeder Seite auf Ihrem Bildschirm der Hinweis -- more -- erscheint, solange Sie die letzte Bildschirmseite nicht gesehen haben.

Für die Befehlsverkettung und die MS-DOS-Verarbeitungsbefehle gibt es viele nützliche Anwendungen.

„Weitergabe sowie Vervielfältigung dieses Unterlage, Verwertung und Mitteilung ihres Inhalts ist ohne schriftliche Genehmigung der Nixdorf Computer AG. Nachdruck, Verbreitung, Übersetzung, Vervielfältigung, Verbreitung, auch auszugsweise, ist ohne schriftliche Genehmigung der Nixdorf Computer AG. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustererteilung vorbehalten.“

WISSENSWERTES ÜBER DIE MS-DOS-BEFEHLE

Zusammenfassung der in diesem Kapitel beschriebenen Befehle

BEFEHL	VERWENDUNGSZWECK	SYNTAX
REM	Fügt in Stapelprogramm- bausteinen Kommentarzeilen ein.	REM Kommentar
PAUSE	Setzt die Ausführung eines Stapelprogrammbausteins aus.	PAUSE Eingabeaufforderung
FIND	Sucht nach einer Zeichen- folge.	FIND Zeichenfolge Dateiname
MORE	Bewirkt die bildschirm- seitenweise Anzeige (je 23 Zeilen) einer Datei.	MORE
SORT	Sortiert Text.	SORT

Nähere Informationen bezüglich der Verwendung der Verarbeitungs-
befehle finden Sie im nächsten Kapitel, in dem die MS-DOS-Befehle
beschrieben werden.

VERZEICHNIS DER MS-DOS-BEFEHLE

5.1 DIE BEFEHLSFORMATE

Im folgenden wird beschrieben, wie MS-DOS-Befehle zu schreiben sind:

1. Alle in Großbuchstaben dargestellten Befehlswörter müssen zwingend eingegeben werden. Es handelt sich bei diesen Wörtern um sogenannte Schlüsselwörter, die genauso eingegeben werden müssen, wie sie dargestellt sind. Das einzige Zugeständnis ist, daß Sie die Schlüsselwörter beliebig in Groß- oder Kleinbuchstaben eingeben können, da MS-DOS grundsätzlich alle für Schlüsselwörter eingegebenen Zeichen in Großbuchstaben umwandelt.
2. Bestimmte Teile des Befehlstexts sind zwischen spitzen Klammern (<>) dargestellt. Die Wörter, die zwischen den spitzen Klammern stehen, stehen für Informationen, die Sie beisteuern müssen. So müssen Sie beispielsweise den Namen Ihrer Datei eingeben, wenn im Befehlstext das Wort Dateiname zwischen spitzen Klammern erscheint.
3. Bestimmte andere Wörter im Befehlstext sind zwischen eckigen Klammern ([]) gesetzt. Diese Wörter stehen für wahlfreie Informationen, die Sie eingeben oder aber auch weglassen können. Wenn Sie eine wahlfreie Information beisteuern wollen, geben Sie bitte nur die Information selbst und nicht die eckigen Klammern ein.
4. Auslassungspunkte (...) zeigen an, daß Sie den betreffenden Befehlstext so oft wiederholen können, wie Sie wollen.
5. Interpunktionszeichen wie Kommas, Gleichheitszeichen, Fragezeichen, Doppelpunkte oder verkehrte Schrägstriche müssen grundsätzlich (mit der einzigen Ausnahme der eckigen Klammern) eingegeben werden wie dargestellt.

VERZEICHNIS DER MS-DOS-BEFEHLE

5.2 DIE MS-DOS-BEFEHLE

HINWEIS

Wenn Sie mit einem Disketten-Einzellaufwerk arbeiten, informieren Sie sich bitte in Teil A des Anhangs über die zusätzlichen Maßnahmen.

ECHO	Ermöglicht es, die Funktion Kontrollanzeige ein- und auszuschalten.
FOR	Stapelprogrammbausteinbefehlserweiterung
GOTO	Stapelprogrammbausteinbefehlserweiterung
IF	Stapelprogrammbausteinbefehlserweiterung
PAUSE	Setzt die Ausführung eines Stapelprogrammbausteins aus.
SHIFT	Erhöht die Zahl der verfügbaren Blindparameter für Stapelprogrammbausteine.

„Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts ist nicht gestattet, soweit es ausdrücklich anders besprochen wurde. Bei Weitergabe oder Vervielfältigung ist der Name des Patent-erfinders oder Gebrauchsmustereintragungsbesitzers anzugeben.“



VERZEICHNIS DER MS-DOS-BEFEHLE

BREAK

BEFEHLSNAME: BREAK

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Ermöglicht das Ein- und Ausschalten der Funktion Befehlsabbruch über die Tasten CONTROL-C.

SYNTAX: BREAK ON
BREAK OFF

WIRKUNGSWEISE: Wir haben weiter oben gesehen, daß die Betätigung der Tastenkombination CONTROL-C zum Abbruch des gerade laufenden MS-DOS-Befehls führt. Nun kann es aber vorkommen, daß Sie ein Anwendungsprogramm ausführen möchten, in dem die Tastenkombination <CONTROL-C> für bestimmte andere Funktionen eingesetzt wird. Wenn Sie ein solches Anwendungsprogramm ausführen wollen, muß daher zunächst die MS-DOS-Funktion CONTROL-C ausgeschaltet werden, weil sonst bei jeder Benutzung der CONTROL-C-Funktion des Anwendungsprogramms der laufende MS-DOS-Befehl abgebrochen würde. Um die MS-DOS-Funktion CONTROL-C auszuschalten, geben Sie den Befehl BREAK OFF ein; um sie wieder einzuschalten, geben Sie BREAK ON ein, sobald die Ausführung Ihres Anwendungsprogramms beendet ist und Sie wieder direkt mit MS-DOS arbeiten wollen.

VERZEICHNIS DER MS-DOS-BEFEHLE

CHDIR

BEFEHLSNAME: CHDIR (CHANGE DIRECTORY)

BEFEHLSTYP: Interner Befehl

SYNONYM: CD

VERWENDUNGSZWECK: Bewirkt den Wechsel aus einem Arbeitsbereich oder Suchweg in einen anderen Arbeitsbereich oder Suchweg. Zeigt den momentanen Arbeitsbereich an.

SYNTAX: CHDIR [Suchwegbezeichnung]

WIRKUNGSWEISE: Nehmen wir an, Ihr momentaner Arbeitsbereich sei \BIN \BENUTZER \ MEIER und Sie möchten aus diesem Arbeitsbereich in den Arbeitsbereich (\BIN \BENUTZER \ MEIER \FORMULAR) wechseln. Geben Sie dann ein:

CHDIR \BIN \BENUTZER \ MEIER \ FORMULAR

und MS-DOS führt Sie in den neuen Arbeitsbereich.

Von diesem Befehl gibt es außerdem eine Kurzform:

CHDIR ...

Sie führt Sie in den Ihrem momentanen Arbeitsbereich übergeordneten Arbeitsbereich, ohne daß Sie dessen Suchwegbezeichnung voll einzugeben brauchen.

Wenn Sie CHDIR ohne Suchwegbezeichnung eingeben, wird Ihr momentaner Arbeitsbereich angezeigt. Lautet dieser beispielsweise \BIN \BENUTZER \ MEIER in Laufwerk B, und Sie geben CHDIR ein und schließen mit der RETURN-Taste ab, dann zeigt Ihnen MS-DOS an:

B: \BIN \BENUTZER \ MEIER

Diese Form des Befehls ist sehr nützlich, wenn Sie den Namen Ihres momentanen Arbeitsbereichs vergessen haben.



Wiedergabe sowie Vervielfältigung dieser Umrage, Verwertung und Mitteilung in irgendeiner Form ist ohne schriftliche Genehmigung des Nixdorf Computer AG. Nachdruck, Vervielfältigung und Verbreitung, auch auszugsweise, ist ohne schriftliche Genehmigung der Nixdorf Computer AG. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustererteilung vorbehalten.

VERZEICHNIS DER MS-DOS-BEFEHLE

CHKDSK

BEFEHLSNAME: CHKDSK (CHECK DISK)

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Überprüft das Inhaltsverzeichnis des Arbeitsbereichs in dem angegebenen Disketten-/Plattenlaufwerk und stellt dort gegebenenfalls vorhandene logische Fehler fest.

SYNTAX: CHKDSK [d:] <Dateibezeichnung> [/F] [/V]

WIRKUNGSWEISE: Es empfiehlt sich, gelegentlich mit dem Befehl CHKDSK über jede Diskette zu gehen, mit der Sie arbeiten, um die Fehlerfreiheit im Arbeitsbereichsverzeichnis sicherzustellen. Werden irgendwelche Fehler gefunden, dann zeigt sie CHKDSK in Form von Fehlermeldungen an, sofern welche festgelegt sind, und schließt die Anzeige mit einem Prüfbericht ab.

Der Prüfbericht hat in der Regel die folgende Form:

```
362496 bytes total disk space
 22528 bytes in 2 hidden files
335872 bytes in 55 user files
 4096 bytes free
```

```
262144 bytes total memory
234848 bytes free
```

Etwa in Ihrem Arbeitsverzeichnis gefundene Fehler wird CHKDSK nicht beseitigen, es sei denn, Sie geben den Befehl mit dem Zusatz /F (für "fix" = in Ordnung bringen) ein. Der Befehlszusatz /V bewirkt, daß CHKDSK seine Meldungen bereits während der Ausführung ausgibt.

VERZEICHNIS DER MS-DOS-BEFEHLE

Die Ausgabe von CHKDSK können Sie auch in eine Datei leiten. Sie brauchen dann nur einzugeben:

CHKDSK A:> Dateiname

Etwa gefundene Fehler wird CHKDSK in diesem Fall in der angegebenen Datei aufzeichnen. In diesem Fall dürfen Sie jedoch den Zusatz /F nicht benutzen.

Bei Verwendung des Befehlszusatzes /F werden folgende Fehler, falls sie festgestellt werden, automatisch berichtet:

Ungültige Laufwerksangabe

Ungültiger Parameter

Ungültiger Unterkatalogseintrag

Katalogwechsel zu Dateiname nicht möglich.
Dateibaum ab diesem Punkt nicht bearbeitet.

CHKDSK

Erste Blocknummer ungültig, Eintrag verkürzt.

Zuordnungsfehler, Größe verändert.

Hat ungültigen Block, Datei verkürzt.

Plattenfehler bei Lesevorgang von FAT

Plattenfehler bei Schreibvorgang auf FAT

<Dateiname> enthält nicht zusammenhängende Blöcke!

Alle angegebenen Dateien sind zusammenhängend.

Wichtig: Diese Vorschriften sind zu beachten. Verstöße sind Mängel, die in Haftung genommen werden können. Soweit nicht ausdrücklich zugestanden, übernehmen wir keine Haftung für Schäden. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustereintragung vorbehalten.

VERZEICHNIS DER MS-DOS-BEFEHLE

Folgende Fehler, die von CHKDSK angezeigt werden, müssen Sie selbst berichtigen, auch dann, wenn Sie den Befehlszusatz /F eingegeben haben!

Incorrect DOS Version

Sie können den Befehl CHKDSK in MS-DOS-Fassungen älter als 2.0 nicht ausführen!

Insufficient memory Processing cannot continue

Sie haben nicht mehr ausreichend Speicherkapazität für die Ausführung des Befehls CHKDSK auf dieser Diskette. Verschaffen Sie sich für die Ausführung von CHKDSK ausreichend Speicherkapazität.

Errors found, F parameter not specified Corrections will not be written to disk

Wenn Sie wünschen, daß CHKDSK die Fehler berichtigt, müssen Sie den Befehlszusatz /F eingeben.

Invalid current directory Processing cannot continue

Starten Sie das System erneut und führen Sie CHKDSK aus.

Cannot CHDIR to root Processing cannot continue

Die Diskette, die Sie prüfen wollen, ist fehlerhaft. Versuchen Sie, MS-DOS neu zu starten und die Diskette mit Hilfe des Befehls RECOVER wiederherzustellen.

<filename> is cross linked on cluster

Legen Sie sich eine Kopie der Datei an, die Sie aufbewahren möchten, und löschen Sie anschließend die beiden über dieselbe Speicherparzelle verketteten Dateien.

X lost clusters found in y chains Convert lost chains to files (Y/N)?

Wenn Sie diese Eingabeaufforderung mit Y beantworten, erstellt CHKDSK eine Datei für Sie und den entsprechenden Eintrag im Inhaltsverzeichnis des Arbeitsbereichs, um Ihnen die Lösung Ihres Problems zu ermöglichen (von CHKDSK erstellte Dateien erhalten den Namen FILEnnnnnnn).

VERZEICHNIS DER MS-DOS-BEFEHLE

CHKDSK

CHKDSK zeigt Ihnen dann am Bildschirm die folgende Meldung:

X bytes space freed

Wenn Sie die Eingabeaufforderung mit N beantworten und den Befehlszusatz /F nicht eingegeben haben, stellt CHKDSK die Speicherparzellen frei und gibt folgende Meldung aus:

X bytes disk space would be freed

Probable non-DOS disk
Continue (Y/N)?

Die Diskette, die Sie eingelegt haben, ist keine MS-DOS-Diskette. Sie müssen nun angeben, ob die Prüfung weitergeführt oder beendet werden soll.

Insufficient room in root directory
Erase files in root and repeat CHKDSK
CHKDSK kann aufgrund fehlenden Speicherraums nicht ausgeführt werden; die Ausführung wird erst möglich, wenn Sie im Grundarbeitsbereich Dateien löschen.

Unrecoverable error in directory?
Convert directory to file (Y/N)?

Wenn Sie diese Eingabeaufforderung mit Y beantworten, wandelt CHKDSK den fehlerhaften Arbeitsbereich in eine Datei um. Sie können dann den Fehler selbst beheben oder die Datei und mit ihr den Arbeitsbereich löschen.

CLS

BEFEHLSNAME: CLS

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Löscht den Bildschirm.

SYNTAX: CLS

WIRKUNGSWEISE: Der Befehl CLS veranlaßt MS-DOS, die ANSI-ESCAPE-Zeichenfolge ESC [2J auszugeben, die bewirkt, daß Ihr Bildschirm gelöscht wird.

5

Weitergabe sowie Vervielfältigung dieses Dokuments, Verweilung und Mitteilung an Dritte ist ohne schriftliche Genehmigung der Nixdorf Computer AG. In besonderen Umgebungen verpflichten wir zu Schadensersatz. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmusteranmeldung vorbehalten.

VERZEICHNIS DER MS-DOS-BEFEHLE

COPY

BEFEHLSNAME: COPY

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Bewirkt, daß eine oder mehrere Dateien auf eine Diskette kopiert wird bzw. werden. Wenn Sie dies wünschen, können Sie den verschiedenen Kopien verschiedene Namen geben. Der Befehl ermöglicht es Ihnen jedoch auch, Dateien auf dieselbe Diskette zu kopieren.

SYNTAX: COPY <Dateiname> [Dateibezeichnung]
[Suchwegbezeichnung] · [Suchwegbezeichnung]
[/V]

WIRKUNGSWEISE: Wenn die zweite Dateibezeichnung nicht eingegeben wird, erfolgt die Kopie auf die Diskette im Standardlaufwerk, und die Dateikopie erhält denselben Namen wie die Ausgangsdatei (nämlich den aus der ersten Dateibezeichnung). Bezieht sich die erste Dateibezeichnung auf das Standardlaufwerk und ist die zweite Dateibezeichnung nicht angegeben, dann wird der Befehl COPY abgebrochen. (Das Kopieren von Dateien in sich selbst ist nicht zulässig.) MS-DOS wird dann folgende Fehlermeldung anzeigen:

```
File cannot be copied on to itself
  0 File(s) copied
```

Die zweite Befehlsoption kann drei verschiedene Formen annehmen:

1. Wenn die zweite Befehlsoption nur eine Laufwerksangabe ist (d:), dann wird die Ausgangsdatei mit ihrem ursprünglichen Dateinamen auf die Diskette im angegebenen Laufwerk kopiert.
2. Ist die zweite Befehlsoption jedoch nur ein Dateiname, dann wird die ursprüngliche Datei als eine Datei mit dem angegebenen Dateinamen auf die Diskette im Standardlaufwerk kopiert.
3. Ist die zweite Befehlsoption jedoch eine volle Dateibezeichnung, dann wird die Ausgangsdatei als Datei mit dem angegebenen Dateinamen auf die Diskette in dem angegebenen Laufwerk kopiert.

VERZEICHNIS DER MS-DOS-BEFEHLE

COPY

Der Befehlszusatz /V bewirkt, daß MS-DOS prüft, ob die auf der Zieldiskette belegten Sektoren ordnungsgemäß beschrieben wurden. Obwohl bei der Verwendung des Befehls COPY selten Aufzeichnungsfehler vorkommen, haben Sie damit trotzdem die Möglichkeit, sich zu vergewissern, daß kritische Daten ordnungsgemäß übertragen wurden. Mit dieser Option läuft der Befehl COPY allerdings etwas langsamer ab, da MS-DOS jeden Eintrag auf der Diskette prüfen muß.

Der Befehl COPY ermöglicht es Ihnen auch, Dateien per Kopie zu verketteten bzw. zusammenzuführen. Diese Verkettung erzielen Sie, indem Sie einfach als Option für den Befehl COPY die Namen der zu verkettenden Dateien (Anzahl beliebig) jeweils durch ein Pluszeichen (+) verbunden eingeben.

Beispiel:

```
COPY A.XYZ + B.COM + B:C.TXT TOPDATEI.CRP
```

Dieser Befehl bewirkt die Verkettung von Dateien mit den Namen A.XYZ, B.COM und B:C.TXT und überträgt sie in die Datei mit dem Namen TOPDATEI.CRP auf der Diskette im Standardlaufwerk.

Bei der Zusammenfassung mehrerer Dateien zu einer einzigen können Sie auch Stellvertreterzeichen verwenden.

Beispiel:

```
COPY *.LST SAMMEL.PRN
```

Dieser Befehl führt alle Dateien mit dem Suffix .LST in einer Datei mit dem Namen SAMMEL.PRN zusammen.

Im folgenden Beispiel wird jede gefundene Datei, die dem Zeichensymbol *.LST entspricht, mit der entsprechenden .REF-Datei zusammengeführt und als .PRN-Datei abgelegt. Das Ergebnis ist eine Datei, die denselben Dateinamen aufweist, aber mit dem Suffix .PRN versehen ist. So würde beispielsweise DATEI1.LST mit DATEI1.REF zusammengeführt und als DATEI1.PRN abgelegt. Desgleichen wird XYZ.LST mit XYZ.REF zusammengeführt und als XYZ.PRN abgelegt usw.

```
COPY *.LST + *.REF *.PRN
```



„Weitergabe sowie Vervielfältigung dieser Unterlage, Verwendung und Mitteilung in anderen Zusammenhängen ist ohne schriftliche Genehmigung des Nixdorf Computer AG. Alle Rechte vorbehalten.“

VERZEICHNIS DER MS-DOS-BEFEHLE

Der folgende COPY-Befehl führt alle Dateien, die dem Zeichenmuster *.LST entsprechen, zusammen, dann alle Dateien, die dem Zeichenmuster .REF entsprechen und legt das Ergebnis in der Datei mit dem Namen SAMMEL.PRN ab:

```
COPY *.LST + *.REF SAMMEL.PRN
```

Wenn eine der Ausgangsdateien einen Namen mit demselben Suffix hat wie der Name der Zieldatei, darf kein COPY- Verkettungsbefehl eingegeben werden. So führt beispielsweise der nachstehend dargestellte Befehl zu einem Fehler, wenn die Datei GESAMT.LST bereits existiert:

```
COPY *.LST GESAMT.LST
```

Dieser Fehler würde jedoch erst entdeckt, wenn GESAMT.LST angefügt wird. Zu diesem Zeitpunkt ist die Datei aber möglicherweise bereits zerstört.

Der Befehl COPY vergleicht den Namen der Ausgangsdatei (Eingabe) mit dem der Zieldatei. Wenn beide Namen gleich sind, wird die betreffende Ausgangsdatei übersprungen und die Fehlermeldung "Content of destination lost before copy" ausgegeben. Die Verkettung weiterer Dateien läuft jedoch normal ab. Damit haben Sie die Möglichkeit, wie im folgenden Beispiel ersichtlich, Dateien "zusammenzufassen":

```
COPY GESAMT.LST +*.LST
```

Dieser Befehl bewirkt, daß alle dem Zeichenmuster *.LST entsprechenden Dateien mit der Ausnahme von GESAMT.LST an GESAMT.LST angehängt werden. Der Befehl führt zu keiner Fehlermeldung und stellt die korrekte Verwendungsweise des Befehls COPY dar, wenn mehrere Dateien zusammengeführt werden sollen.

VERZEICHNIS DER MS-DOS-BEFEHLE

CTTY

BEFEHLSNAME: CTTY

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Ermöglicht es Ihnen, von dem Systembedienungsplatz, von dem Sie gerade Ihre Befehle eingeben, auf einen anderen Systembedienungsplatz zu wechseln (TTY steht für den Systembedienungsplatz, d.h. für Tastatur und Bildschirm Ihres Computers).

SYNTAX: CTTY <Gerät>

WIRKUNGSWEISE: Mit Gerät ist hier der Systembedienungsplatz gemeint, über den Sie Ihre Befehle an MS-DOS eingeben. Der Befehl ist dann nützlich, wenn Sie diesen Systembedienungsplatz wechseln möchten.

CTTY /AUX

Dieser Befehl bewirkt die Übertragung aller Befehlsein-/ausgaben von dem derzeitigen "Gerät" (der Tastatur und dem Bildschirm) auf den AUX-Anschluß, d.h. zum Beispiel auf einen Drucker.

Der Befehl: CTTY /CON

holt die Befehlsein-/ausgaben zurück auf das ursprüngliche Gerät, also hier auf Tastatur und Bildschirm (CON = CONsole = Systembedienungsplatz). Ein Verzeichnis der zusammen mit dem Befehl CTTY zu verwendenden gültigen Gerätenamen finden Sie in dem Abschnitt "Unzulässige Dateinamen" in Kapitel 3 "Eingehendere Informationen über Dateien".

„Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts in irgendeiner Form ohne schriftliche Genehmigung der Nixdorf Computer AG ist ausdrücklich untersagt. Alle Rechte für den Fall des Patenterteilung oder Gebrauchsmusteranmeldung vorbehalten.“

VERZEICHNIS DER MS-DOS-BEFEHLE

DATE

BEFEHLSNAME: DATE

BEFEHLSSTYP: Interner Befehl

VERWENDUNGSZWECK: Ermöglicht die Eingabe oder die Änderung des vom System geführten Datums. Dieses Datum wird im Grundarbeitsbereich für alle Dateien, die Sie erstellen oder ändern, aufgezeichnet.

Ändern können Sie das Datum sowohl über die Tastatur als auch über einen Stapelprogrammbaustein. (Wenn Sie eine AUTOEXEC.BAT-Datei verwenden, wird MS-DOS keine Eingabeaufforderung für das Datum ausgeben; Sie müssen daher in diesem Fall in Ihre AUTOEXEC.BAT-Datei den Befehl DATE einfügen.)

SYNTAX: DATE [`<tt>` - `<mm>` - `<jj>`]

WIRKUNGSWEISE: Wenn Sie DATE eingeben, antwortet der Befehl DATE mit der Meldung:

```
Current date is <tt> . <mm> <jj>
Enter new date:
```

Wenn Sie das angezeigte Datum nicht verändern möchten, betätigen Sie einfach die RETURN-Taste.

Sie können auch unmittelbar nach dem Befehl ein bestimmtes Datum eingeben, wie zum Beispiel:

```
DATE 3-9-81
```

In diesem Fall brauchen Sie die Eingabeaufforderung "Enter new date" nicht zu beantworten.

Wenn Sie ein neues Datum eingeben wollen, verwenden Sie bitte dafür nur Ziffern; Buchstaben sind unzulässig. Die möglichen Eingaben sind:

```
<tt> = 1 bis 31
<mm> = 1 bis 12
<jj> = 80 bis 99 oder 1980 bis 2099
```

Die Angaben zu Tag, Monat und Jahr können entweder durch Bindestriche (-) oder Schrägstriche (/) voneinander abgegrenzt sein. MS-DOS ist so programmiert, daß es Monate und Jahre automatisch richtig wechselt, einerlei, ob ein Monat 31, 30, 29 oder 28 Tage hat. Genauso ist MS-DOS in der Lage, Schaltjahre zu berücksichtigen.

VERZEICHNIS DER MS-DOS-BEFEHLE

DATE

Bei ungültigen Eingaben oder Abgrenzungszeichen reagiert der Befehl DATE mit der Meldung:

Invalid date
Enter new date:

dann wartet der Befehl DATE, bis Sie ein gültiges Datum eingeben haben.

DEL

BEFEHLSNAME: DEL (DELETE)

BEFEHLSSTYP: Interner Befehl

SYNONYM: ERASE

VERWENDUNGSZWECK: Löscht alle Dateien mit der angegebenen Datei-
bezeichnung.

SYNTAX: DEL [Dateibezeichnung] [Suchwegbezeichnung]

WIRKUNGSWEISE: Wenn die Dateibezeichnung *.* eingegeben wird, erscheint die Eingabeaufforderung "Are you sure (Y/N)?" . Wenn Sie die Eingabeaufforderung mit Y oder y beantworten, werden alle Dateien, wie gewünscht, gelöscht. Zur Eingabe des Befehls DELETE können Sie auch ERASE schreiben.



VERZEICHNIS DER MS-DOS-BEFEHLE

DIR

BEFEHLSNAME: DIR (DIRECTORY)

BEFEHLSSTYP: Interner Befehl

SYNTAX: DIR [Dateibezeichnung] [Suchwegbezeichnung] [/P]/[W]

VERWENDUNGSZWECK: Anzeige des Inhaltsverzeichnisses eines Arbeitsbereichs.

WIRKUNGSWEISE: Wenn Sie einfach nur DIR eingeben, werden alle Einträge im Inhaltsverzeichnis des momentanen Arbeitsbereichs auf der Diskette im Standardlaufwerk am Bildschirm angezeigt. Wenn zuzüglich zum Befehlsnamen eine Laufwerksangabe erfolgt (DIR d:), werden sämtliche Einträge des Inhaltsverzeichnisses des momentanen Arbeitsbereichs auf der Diskette im angegebenen Laufwerk am Bildschirm angezeigt. Wenn nur ein Dateiname ohne Suffix eingegeben wird (DIR Dateiname), dann werden alle Dateien im momentanen Arbeitsbereich auf der Diskette im Standardlaufwerk, die dem angegebenen Dateinamen ohne Suffix entsprechen, am Bildschirm angezeigt. Geben Sie dagegen eine komplette Dateibezeichnung einschließlich Laufwerksangabe ein (zum Beispiel DIR d:Dateiname.SUF), dann werden alle Dateien im momentanen Arbeitsbereich der Diskette in dem angegebenen Laufwerk, die der angegebenen Dateibezeichnung entsprechen, aufgelistet. In allen Fällen werden die Dateien in den Inhaltsverzeichnissen mit ihrem Umfang in Byte und mit Datum und Uhrzeit ihrer letzten Bearbeitung angegeben.

VERZEICHNIS DER MS-DOS-BEFEHLE

Bei der Angabe des Dateinamens für diesen Befehl können Sie die Stellvertreterzeichen Fragezeichen (?) und Sternchen (*) verwenden. Beachten Sie bitte, daß MS-DOS die folgenden Eingabeformen des Befehls DIR, um Ihnen die Arbeit zu erleichtern, als äquivalent ansieht:

EINGABE	WIRKUNG WIE
DIR	DIR *.*
DIR DATEINAME	DIR DATEINAME.*
DIR .SUF	DIR *.EXT
DIR .	DIR *.*

Mit dem Befehl DIR können Sie zwei Befehlszusätze verwenden. Der Zusatz /P bewirkt die "seitenweise Anzeige" des Inhaltsverzeichnisses, d.h., daß der Computer zunächst einen Bildschirm bis zur letzten unteren Zeile auffüllt und diese Anzeige stehenläßt, bis Sie eine beliebige Taste drücken.

Der Befehlszusatz /W bewirkt die "breite Anzeige", d.h., es werden lediglich die Dateinamen ohne die anderen Dateiinformationen angezeigt, und von den Dateinamen werden fünf pro Zeile auf den Bildschirm gebracht.

VERZEICHNIS DER MS-DOS-BEFEHLE

DISKCOPY

BEFEHLSNAME: DISKCOPY

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Kopiert den kompletten Inhalt der Diskette im Ausgangslaufwerk auf die Diskette im Ziellaufwerk.

SYNTAX: DISKCOPY [d:] [d:]

WIRKUNGSWEISE: Die erste anzugebende Option ist das Ausgangslaufwerk, die zweite das Ziellaufwerk.

Als Ziellaufwerk können Sie dasselbe Laufwerk wie das Ausgangslaufwerk oder aber ein anderes angeben. Wenn die beiden Laufwerksangaben identisch sind, wird eine Einzellaufwerkskopie durchgeführt, d.h., es muß in das betreffende Einzellaufwerk zuerst die zu kopierende Diskette und dann die Diskette, auf die kopiert werden soll, eingelegt werden. Den richtigen Zeitpunkt für das Einlegen der verschiedenen Disketten zeigt Ihnen der Computer durch geeignete Meldungen an. Die weitere Ausführung des Befehls DISKCOPY erfolgt jeweils erst, wenn Sie dies durch die Betätigung einer beliebigen Taste veranlassen.

Nach Abschluß des Kopiervorgangs erhalten Sie von DISKCOPY die Meldung:

```
Copy complete
Copy another (Y/N)?_
```

Wenn Sie diese Frage mit Y beantworten, erfolgt der nächste Kopiervorgang zwischen denselben Laufwerken wie beim ersten. Selbstverständlich werden Sie auch hier vom System aufgefordert, zum richtigen Zeitpunkt die entsprechenden Disketten einzulegen.

Um den Befehl DISKCOPY zu verlassen, beantworten Sie die Frage einfach mit N.

VERZEICHNIS DER MS-DOS-BEFEHLE

Hinweise:

1. Wenn Sie keine der beiden Optionen angeben, wird eine Einzel-laufwerkskopie auf dem Standardlaufwerk ausgeführt.
2. Wenn Sie die zweite Option nicht angeben, wird das Standard-laufwerk automatisch als Ziellaufwerk genommen.
3. Beide Disketten müssen über dieselbe Anzahl physischer Sektoren gleicher Größe verfügen.
4. Auf Disketten, auf die oft zugegriffen wird, um Dateien zu erstellen und zu löschen, wird der Speicherraum in viele kleine Parzellen unterteilt, weil er nicht in einer bestimmten Reihenfolge zugewiesen wird: beim Schreiben auf die Diskette wird der nächstbeste leere Sektor genommen, völlig unabhängig von der Stelle, an der sich der Schreibkopf gerade befindet.

Aufgrund der Verzögerungen, die beim Suchen, Lesen und Schreiben auf einer Diskette mit stark zerstückeltem Inhalt entstehen, kann das Arbeiten mit einer solchen Diskette recht zeitraubend und somit unerfreulich werden. In diesem Fall empfiehlt es sich, anstatt des Befehls DISKCOPY für das Kopieren ganzer Disketten den Befehl COPY zu verwenden, da dieser Befehl die Zerstückelung der Diskette beim Kopieren beseitigt.

Beispiel:

`COPY A:*. * B:`

Dieser Befehl bewirkt, daß alle Dateien der Diskette in Laufwerk A auf die Diskette in Laufwerk B unter optimaler Ausnutzung des verfügbaren Speicherplatzes kopiert werden.

5. Der Befehl DISKCOPY ermittelt automatisch die Anzahl von Diskettenseiten, die kopiert werden müssen, ausgehend vom Ausgangslaufwerk und von der Ausgangsdiskette.
6. Werden während der Ausführung des Befehls DISKCOPY Diskettenfehler erkannt, dann gibt MS-DOS die folgende Meldung aus:

DISK error while reading drive A
Abort, Ignore, Retry?

Nähere Einzelheiten über diese Fehlermeldung finden Sie in Teil B "Diskettenfehler" des Anhangs.



VERZEICHNIS DER MS-DOS-BEFEHLE

EXE2BIN

BEFEHLSNAME: EXE2BIN

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Dieser Befehl konvertiert ausführbare Dateien (Dateien mit dem Suffix .EXE) in Binärdateien. Damit erreicht man eine Einsparung an Diskettenspeicherplatz und ein schnelleres Laden der Programme.

SYNTAX: EXE2BIN <Dateibezeichnung> [d:] [<Dateiname> [<.SUF>]]

WIRKUNGSWEISE: Diesen Befehl benötigen Sie nur, wenn Sie ausführbare Dateien mit dem Suffix .EXE in Binärdateien umwandeln wollen. Die mit der Dateibezeichnung benannte Datei ist die Eingabedatei. Wenn kein Suffix angegeben wird, erhält diese Datei standardmäßig das Suffix .EXE. Die Eingabedatei wird in das .COM-Format (Speicherbild des Programms) gebracht und in die Ausgabedatei geschrieben. Wenn Sie kein Laufwerk angeben, wird das Laufwerk benutzt, in dem sich die Diskette mit der Eingabedatei befindet. Wenn Sie keinen Ausgabe-Dateinamen angeben, wird der Eingabe-Dateiname verwendet. Wenn Sie keinen Namen für die Ausgabedatei angeben, wird der Name der Eingabedatei auch dafür verwendet. Wenn Sie im Namen der Ausgabedatei kein Suffix angeben, erhält die neue Datei automatisch das Suffix .BIN.

VERZEICHNIS DER MS-DOS-BEFEHLE

Die Eingabedatei muß das gültige .EXE-Format haben, das der Binder erzeugt. Der speicherresidente, d.h. der den Programmcode und die Daten enthaltende Teil der Datei, muß kleiner sein als 64 kByte. Ein STACK-Segment darf nicht darin enthalten sein.

Zwei Arten von Konvertierungen sind möglich, je nachdem, ob der anfängliche CS:IP (Code-Segment:Befehlszeiger) in der .EXE- Datei angegeben ist:

1. Wenn der CS:IP in der .EXE-Datei nicht angegeben ist, geht das System davon aus, daß eine reine Binärkonvertierung gewünscht wird. Wenn Segmentfestlegungen erforderlich sind (z.B. wenn das Programm Befehle enthält, die eine Segmentneuadressierung erfordern), erhalten Sie eine Eingabeaufforderung für die Benennung des Festwerts. Dieser Festwert ist die absolute Adresse des Segments, wo das Programm geladen werden muß. Das durch diese Konvertierung erzeugte Programm wird nur lauffähig sein, wenn es an der durch ein Benutzer-Anwendungsprogramm benannten absoluten Speicheradresse geladen wird. Der Befehlsprozessor allein ist nicht in der Lage, ein solches Programm ordnungsgemäß zu laden.
2. Wenn der CS:IP als 0000:100H angegeben wird, geht MS-DOS davon aus, daß die Datei als .COM-Datei mit dem durch die Assembler-Anweisung ORG auf 100H gesetzten Adressenzeiger ausgeführt werden soll; die ersten 100H Byte der Datei werden dabei gelöscht. Feste Segmentadressen sind nicht möglich, da .COM-Dateien segmentweise verschiebbar sein müssen, d.h., sie müssen den im Macro-Assembler-Handbuch beschriebenen Eingabebedingungen genügen. Sobald die Konvertierung abgeschlossen ist, können Sie die Ausgabedatei mit dem Suffix .COM versehen. Dann kann dieses Programm genauso wie die anderen auf Ihrer MS-DOS- System-Diskette befindlichen .COM-Programme geladen und ausgeführt werden.

Entspricht der CS:IP jedoch keiner dieser beiden Möglichkeiten oder entspricht er den Anforderungen für .COM-Dateien, beinhaltet aber festgeschriebene Segmente, dann wird die folgende Meldung ausgegeben:

File cannot be converted

Dieselbe Meldung erscheint auch am Bildschirm, wenn die Datei keine gültige ausführbare Datei ist.



VERZEICHNIS DER MS-DOS-BEFEHLE

Findet EXE2BIN einen Fehler, dann wird eine oder gegebenenfalls auch mehrere der nachstehenden Fehlermeldungen angezeigt:

File not found

Die Datei befindet sich nicht auf der angegebenen Diskette.

Insufficient memory

MS-DOS steht nicht genügend Speicherkapazität für die Ausführung von EXE2BIN zur Verfügung.

File creation error

EXE2BIN kann die Ausgabedatei nicht erstellen. Führen Sie den Befehl CHKDSK aus um festzustellen, ob der Arbeitsbereich voll ist oder ob andere Ursachen für den Fehler vorliegen.

Insufficient disk space

Auf der angegebenen Diskette ist nicht mehr genügend Platz für die Erstellung einer neuen Datei.

Fixups needed - base segment (hex):

Die Ausgangs(.EXE-)-Datei enthält Informationen, denen MS-DOS entnimmt, daß ein Ladesegment für die Datei benötigt wird.

Geben Sie die absolute Segmentadresse an, wo das fertige Programmmodul geladen werden soll.

File cannot be converted

Die Eingabedatei hat nicht das erforderliche Format.

ACHTUNG! Lesefehler in der .EXE-Datei

Weniger gelesen als im Vorsatz vorgesehen!
Diese Meldung ist lediglich als Warnung gedacht.

VERZEICHNIS DER MS-DOS-BEFEHLE

EXIT

BEFEHLSNAME: EXIT

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Bewirkt das Verlassen des Programms COMMAND.COM, d.h. des Befehlsprozessors, und die Rückkehr in den vorherigen Zustand.

SYNTAX: EXIT

WIRKUNGSWEISE: Dieser Befehl ermöglicht es Ihnen, während Sie ein Anwendungsprogramm ausführen, zu MS-DOS zurückzukehren und den Befehlsprozessor zu starten, um anschließend zu Ihrem Anwendungsprogramm zurückzukehren. Wenn Sie sich beispielsweise ein Inhaltsverzeichnis anschauen möchten, das auf der Diskette in Laufwerk B steht, während Sie ein Anwendungsprogramm ausführen, müssen Sie den Befehlsprozessor starten, indem Sie COMMAND als Antwort auf die Standardlaufwerk-Eingabeaufforderung eingeben:

A> COMMAND

Dann können Sie den Befehl DIR eingeben, und MS-DOS zeigt Ihnen das Inhaltsverzeichnis der Diskette im Standardlaufwerk. Wenn Sie dann den Befehl EXIT eingeben, kehren Sie damit in den vorherigen Zustand zurück, d.h. in Ihr Anwendungsprogramm.

FIND

BEFEHLSNAME: FIND

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Sucht die angegebene Textzeichenfolge in einer oder mehreren Dateien.

SYNTAX: FIND [/V /C /N] <Zeichenfolge>[<Dateiname...>]

WIRKUNGSWEISE: FIND ist ein Bearbeitungsbefehl, für den als Optionen eine Zeichenfolge und eine Reihe von Dateinamen vorgesehen sind. Der Befehl bewirkt die Anzeige aller Zeilen in den als Optionen angegebenen Dateien, die die angegebene Zeichenfolge enthalten.

5

Weitergabe sowie Vervielfältigung dieses Dokuments, Verbreitung mit jedem Mittel, ist ausdrücklich untersagt und ohne schriftliche Genehmigung nicht zulässig. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustererteilung vorbehalten.

VERZEICHNIS DER MS-DOS-BEFEHLE

Werden keine Dateinamen angegeben, dann geht FIND davon aus, daß die angegebene Zeichenfolge in dem gerade über dem Bildschirm angegebenen Text gesucht werden soll.

Befehlszusätze für FIND sind:

- /V Bewirkt, daß FIND alle Zeilen anzeigt, die die angegebene Zeichenfolge nicht enthalten.
- /C Bewirkt, daß FIND lediglich die Anzahl der Zeilen anzeigt, in denen die angegebene Zeichenfolge in jeder einzelnen der eingegebenen Dateien gefunden wurde.
- /N Bewirkt, daß FIND die Zeilen mit ihren auf die jeweilige Datei bezogenen relativen Zeilennummern anzeigt.

Die zu suchende Zeichenfolge muß zwischen Anführungszeichen eingegeben werden.

Beispiel:

```
FIND "Abrakadabra" BUCH1.TXT BUCH2.TXT
```

Dieser Befehl bewirkt die Anzeige aller Zeilen aus den Dateien BUCH1.TXT und BUCH2.TXT (in dieser Reihenfolge), die die Zeichenfolge "Abrakadabra" enthalten.

Der Befehl:

```
DIR B: | FIND /V "DAT"
```

bewirkt, daß MS-DOS alle Dateinamen der Diskette in Laufwerk B anzeigt, die die Zeichenfolge DAT nicht enthalten.

Wenn Sie eine Zeichenfolge suchen lassen wollen, die selbst zwischen Anführungszeichen steht, müssen Sie diese Zeichenfolge zwischen doppelte Anführungszeichen setzen.

VERZEICHNIS DER MS-DOS-BEFEHLE

Wird ein Fehler erkannt, dann reagiert der Befehl FIND mit einer der folgenden Meldungen:

Incorrect DOS-Version

Der Befehl FIND kann nur in MS-DOS-Fassungen ab 2.0 oder neuer ausgeführt werden.

FIND: Invalid number of parameters

Bei der Eingabe des Befehls FIND haben Sie keine Zeichenfolge angegeben.

FIND: Syntax error

Bei der Eingabe des Befehls FIND haben Sie eine unzulässige Zeichenfolge eingegeben.

FIND: File not found <filename>

Der von Ihnen angegebene Dateiname ist entweder nicht vorhanden oder kann von FIND nicht gefunden werden.

FIND: Read error in<filename>

Beim Lesen der im Befehl angegebenen Datei durch FIND ist ein Fehler aufgetreten.

FIND: Invalid parameter <option-name>

Sie haben eine Option angegeben, die nicht existiert.

VERZEICHNIS DER MS-DOS-BEFEHLE

FORMAT

BEFEHLSNAME: FORMAT

BEFEHLSSTYP: Externer Befehl

VERWENDUNGSZWECK: Formatiert die Diskette im angegebenen Laufwerk für die Aufnahme von MS-DOS-Dateien.

SYNTAX: FORMAT [d] : [/o] [/v] [/s]

WIRKUNGSWEISE: Dieser Befehl initialisiert den Arbeitsbereich mit seinem Verzeichnis und die Dateispeicherplatztabellen. Wenn kein Laufwerk angegeben ist, wird die Diskette im Standardlaufwerk formatiert.

HINWEIS

Die im folgenden beschriebenen Befehlszusätze müssen in der Reihenfolge angegeben werden, in der sie weiter oben im Syntaxbeispiel aufgeführt sind.

Der Befehlszusatz /O bewirkt, daß FORMAT die Diskette im IBM-PC-Format DOS Fassung 1.X formatiert. Dieser Befehlszusatz bewirkt außerdem, daß FORMAT den Arbeitsbereich mit einem OE5-Byte (hex) am Anfang jedes Eintrags neu konfiguriert, so daß die Diskette sowohl mit der Fassung 1.X von IBM-PC-DOS als auch mit MS-DOS 1.25/2.00 und IBM-PC-DOS 2.00 verwendet werden kann. Dieser Befehlszusatz sollte nur dann verwendet werden, wenn es unbedingt erforderlich ist, da einerseits die Konvertierung durch FORMAT dann ziemlich lange dauert und andererseits die Leistungsfähigkeit der MS-DOS-Fassungen 1.25 und 2.00 auf Disketten mit nur wenigen Einträgen im Inhaltsverzeichnis merklich nachläßt.

Die Eingabe des Befehlszusatzes /V bewirkt, daß FORMAT nach der Formatierung der Diskette eine Eingabeaufforderung nach der Diskettenbezeichnung ausgibt.

Der Befehlszusatz /S muß immer als letzter eingegeben werden. FORMAT kopiert dann die Betriebssystemdateien von der Diskette im Standardlaufwerk auf die soeben neu formatierte Diskette. Die Übertragung erfolgt in der Reihenfolge:

IO.SYS
MSDOS.SYS
COMMAND.COM

VERZEICHNIS DER MS-DOS-BEFEHLE

MKDIR

BEFEHLSNAME: MKDIR

BEFEHLSTYP: Interner Befehl

SYNONYM: MD

VERWENDUNGSZWECK: Bewirkt, daß ein neuer Arbeitsbereich angelegt wird.

SYNTAX: MKDIR <Suchwegbezeichnung>

WIRKUNGSWEISE: Mit diesem Befehl wird ein hierarchisch organisierter Arbeitsbereich angelegt und das entsprechende Verzeichnis erstellt. Wenn Sie sich in Ihrem Grundarbeitsbereich befinden, können Sie mit Hilfe des Befehls MKDIR Unter-Arbeitsbereiche einrichten.

Der Befehl:

MKDIR \ BENUTZER

bewirkt, daß in Ihrem Grundarbeitsbereich ein Unter-Arbeitsbereich \BENUTZER eingerichtet wird. Um innerhalb des Arbeitsbereichs \BENUTZER einen Unter-Unter-Arbeitsbereich MEIER anzulegen, geben Sie ein:

MKDIR \BENUTZER \MEIER



VERZEICHNIS DER MS-DOS-BEFEHLE

MORE

BEFEHLSNAME: MORE

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Bewirkt die seitenweise Ausgabe einer Datei am Bildschirm.

SYNTAX: MORE

WIRKUNGSWEISE: MORE ist ein Verarbeitungsbefehl, der Standardeingaben (wie beispielsweise einen über Ihre Tastatur eingegebenen Befehl) liest, und zunächst die erste Bildschirmseite darstellt. Der Befehl bewirkt dann eine Pause und zeigt am unteren Rand der Bildschirmseite die Meldung -- more -- an.

Wenn Sie nun die RETURN-Taste betätigen, wird MORE die nächste Bildschirmseite anzeigen usw., bis alle Eingabedaten gelesen bzw. angezeigt sind.

Der Befehl MORE ist nützlich, wenn Sie eine umfangreiche Datei seitenweise am Bildschirm prüfen wollen. Wenn Sie eingeben:

TYPE TOPDATEI.COM | MORE *1=0*

wird MS-DOS den Inhalt der Datei TOPDATEI.COM (die sich auf der Diskette im Standardlaufwerk befindet) seitenweise am Bildschirm darstellen.

VERZEICHNIS DER MS-DOS BEFEHLE

MODE

BEFEHLSNAME: MODE

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Bestimmt die Betriebsart eines Monitors oder Druckers. Die Druckerausgabe wird auf die serielle Schnittstelle umgeleitet.

SYNTAX: MODE LPT#[n][,m][,P]]
oder
MODE n
oder
MODE [n],m[,T]
oder
MODE COMn:Baud[,Parität[,Datenbits
[,Stopbits[,P]]]]
oder
MODE LPT#:=COMn

BEMERKUNGEN: Wenn ein n oder m Parameter ungültig ist oder fehlt, wird die Betriebsart für diesen Parameter nicht geändert. Es gibt vier Befehlsformate:

5

Wiederholungsrechte vorbehalten. Nachdruck, Vervielfältigung und Verbreitung, auch auszugsweise, ist ohne schriftliche Genehmigung der Nixdorf Computer AG. Die Nixdorf-Produkte sind in der Bundesrepublik Deutschland zum Patent angemeldet. Alle Rechte vorbehalten. Die Rechte an den Marken sind Eigentum der Nixdorf Computer AG. Die Rechte an den Marken sind Eigentum der Nixdorf Computer AG. Die Rechte an den Marken sind Eigentum der Nixdorf Computer AG.

VERZEICHNIS DER MS-DOS BEFEHLE

Option 1 ~ Druckersteuerung

MODE LPT#:[n][,m][,P]

wobei:

ist 1,2 oder 3 (die Druckernummer)
n ist 80 oder 132 (Zeichen pro Zeile)
m ist 6 oder 8 (Zeilen pro Zoll)
P spezifiziert Versuchswiederholungen
nach Time-out Fehlern

Zum Beispiel:

MODE LPT1:132,8

Der Drucker wird auf 80 bis 132 Zeichen pro Zeile und 8 Zeilen pro Zoll eingestellt. 80 Zeichen pro Zeile und 6 Zeilen pro Zoll sind die voreingestellten Werte für den Drucker.

"Ctrl" "Break" stoppt die Wiederholungsschleife. Falls Sie die Wiederholung von Versuchen bei Time-out-Fehlern stoppen wollen, Sie aber den Parameter P eingegeben haben, verwenden Sie die MODE-Option 1 ohne P.

Option 2

Verwenden Sie diese Option um die Bildschirmbetriebsart zu ändern.

MODE n

oder

MODE [n],m,[T]

VERZEICHNIS DER MS-DOS BEFEHLE

wobei:

n ist 40, 80, BW40, BW80, CO40, CO80 oder MONO

40 setzt die Breite der Anzeige auf 40 Zeichen pro Zeile

80 setzt die Breite der Anzeige auf 80 Zeichen pro Zeile

BW40 Der Anzeigemodus ist auf (CO40) Schwarz-Weiß mit 40 Zeichen pro Zeile gesetzt (Farbe ist ausgeschaltet).

BW80 Der Anzeigemodus ist auf (CO80) Schwarz-Weiß mit 80 Zeichen pro Zeile gesetzt (Farbe ist ausgeschaltet).

MONO ändert den aktiven Bildschirmadapter in den monochromen Bildschirmadapter. Dieses ergibt immer eine Anzeigebreite von 80 Zeichen pro Zeile.

m ist L oder R, um die Anzeige nach links oder rechts zu verschieben.

T fragt nach der Verwendung eines Testmusters, um die Anzeige auszurichten.



VERZEICHNIS DER MS-DOS BEFEHLE

Manchmal kann die Lesbarkeit des Monitors verbessert werden, indem die Anzeige um ein oder zwei Zeichen verschoben wird. Wenn Sie T in dem MODE-Befehl eingeben, fragt ein Aufforderungszeichen, ob der Bildschirm richtig ausgerichtet ist. Durch Eingabe von Y (ja) wird der Befehl beendet. Durch Eingabe von N (nein) wird nochmals verschoben und das Aufforderungszeichen wieder angezeigt. Zum Beispiel:

MODE 80,L,T

Dieses setzt den Anzeigemodus auf 80 Zeichen pro Zeile und rückt die Anzeige zwei Zeichen nach links. Da das Testmuster erneut gezeigt wird, haben Sie die Möglichkeit, die Anzeige wieder zu verschieben, ohne den Befehl erneut einzugeben.

Option 3

Wird mit der seriellen Schnittstelle (RS232C) verwendet.
(Asynchrone Kommunikationsschnittstelle)

MODE COMn:Baud[,Parität[,Datenbits [,Stopbits[,P]]]]

wobei:

n Wird entweder auf 1 oder 2 gesetzt (1: Interne serielle Schnittstelle; 2: Zusätzliche asynchrone Kommunikationsschnittstelle)

Baud 110, 150, 300, 600, 1200, 2400, 4800 oder 9600

Parität Entweder E (gerade), O (ungerade) oder N (keine). Die Voreinstellung ist E.

Datenbits Entweder 7 oder 8. Die Voreinstellung ist 7.

Stopbits Entweder 1 oder 2. Wenn die Baudrate 110 ist, ist die Vorgabe 2. Wenn die Baudrate nicht 110 ist, ist die Vorgabe 1.

VERZEICHNIS DER MS-DOS BEFEHLE

Um die Baudrate einzugeben, brauchen Sie nur die ersten zwei Zeichen eingeben; weitere Ziffern werden ignoriert.

Mit diesen Protokollparametern wird die serielle Schnittstelle initialisiert. Zumindest die Baudrate muß eingegeben werden. Um die Voreinstellung zu akzeptieren, können alle anderen Parameter fehlen und durch Eingabe von Kommata ersetzt werden.

Zum Beispiel:

```
MODE COM1:30,0,8,1,P
```

Dieser Befehl setzt die Baudrate auf 300, ungerade Parität, acht Datenbits und ein Stopbit. Unter Ausnutzung der voreingestellten Werte geben Sie folgendes ein:

```
MODE COM1:30,,,,P
```

Die Parität wird als gerade vorgegeben, die Datenbits auf sieben und die Stopbits auf eins.

Um die serielle Schnittstelle als serielle Druckerschnittstelle zu benutzen und den Zusatz der Wiederholung nach Time-out Fehlern zu haben, geben Sie den Zusatz P ein. Durch Drücken von Ctrl-Break wird die Wiederholungsschleife gestoppt. Durch neues Initialisieren der seriellen Schnittstelle, ohne die Eingabe von P, wird das fortlaufende Wiederholen wieder abgeschaltet.

Option 4

Wird verwendet, um den parallelen Druckerausgang auf eine serielle Schnittstelle umzulenken.

```
MODE LPT#:=COMn
```

wobei:

- # Entweder 1, 2 oder 3 (Druckernummer)
- n Entweder 1 oder 2 (1: Interne serielle Schnittstelle, 2: zusätzliche asynchrone Schnittstelle)



VERZEICHNIS DER MS-DOS BEFEHLE

Die Druckerausgabe auf LPT# wird damit auf den Kommunikationskanal n umgelenkt.

BEMERKUNG: Wenn Sie einen seriellen Drucker an die serielle Schnittstelle anschließen wollen, initialisieren Sie zuerst mit der Option 3 die serielle Schnittstelle mit dem Zusatz P; sodann lenken Sie die Ausgabe um.

Um die Neu-Adressierung für den durch LPT# bezeichneten Drucker zu stoppen, geben Sie folgendes ein:

MODE LPT#

VERZEICHNIS DER MS-DOS-BEFEHLE

PATH

BEFEHLSNAME: PATH

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Ermöglicht die Angabe eines Befehlssuchwegs.

SYNTAX:

PATH [**<Suchwegbezeichnung>**] [**<Suchwegbezeichnung>**] ...]

WIRKUNGSWEISE: Dieser Befehl ermöglicht es Ihnen, MS-DOS mitzuteilen, in welchen Arbeitsbereichen nach der Durchsuchung Ihres momentanen Arbeitsbereichs weitergesucht werden soll. Standardvorgabe ist "Kein Suchweg".

Wenn Sie beispielsweise veranlassen wollen, daß MS-DOS Ihren Arbeitsbereich \BIN \BENUTZER \MEIER nach externen Befehlen ab sucht, geben Sie bitte ein:

```
PATH \BIN \BENUTZER \MEIER
```

MS-DOS wird nun benötigte externe Befehle im Arbeitsbereich \BIN \BENUTZER \MEIER suchen, bis Sie einen anderen Suchweg vorgeben oder MS-DOS ausschalten.

Sie können MS-DOS auch anweisen, in mehreren Arbeitsbereichen zu suchen, indem Sie die entsprechenden Suchwegbezeichnungen jeweils durch Strichpunkte getrennt angeben.

Beispiel:

```
PATH \BIN \BENUTZER \MEIER; \BIN \BENUTZER \BACH; \BIN \ENTV
```

MS-DOS wird dann über die angegebenen Suchwege auf die entsprechenden Arbeitsbereiche zugreifen und sie nach externen Befehlen absuchen. Die Suche findet in der durch die Reihenfolge der Suchwegbezeichnungen in der Befehlszeile vorgegebenen Reihenfolge statt.

Wenn der Befehl PATH ohne Optionen eingegeben wird, bringt er den momentanen Suchweg zur Anzeige. Wenn Sie eingeben PATH ;, wird MS-DOS vom NUL-Suchweg ausgehen, was bedeutet, daß nur der momentane Arbeitsbereich nach externen Befehlen abgesucht wird.



VERZEICHNIS DER MS-DOS-BEFEHLE

PRINT

BEFEHLSNAME: PRINT

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Bewirkt, daß eine Textdatei auf dem Drucker ausgedruckt wird, während Sie mit anderen MS-DOS-Befehlen am Bildschirm weiter arbeiten. Diese Art des Drucks nennen wir "Drucken im Hintergrund".

SYNTAX: PRINT [[Dateibezeichnung] [/T] [/C] [/S]] ...

WIRKUNGSWEISE: Der Befehl PRINT kann nur dann verwendet werden, wenn an Ihrem Computer auch ein Drucker angeschlossen ist. Zu diesem Befehl verfügen Sie über folgende Zusätze:

/T Für TERMINATE: Dieser Befehlszusatz bewirkt, daß alle in der Warteschlange des Druckers eingetragenen Dateinamen (d.h. die Namen der Dateien, die darauf warten, ausgedruckt zu werden) gelöscht werden. Eine entsprechende Meldung wird auf dem Drucker ausgedruckt.

/C Für CANCEL: Dieser Befehlszusatz bewirkt, daß der Druck der Datei mit der vorhergehenden Dateibezeichnung sowie aller folgenden, in die Warteschlange des Druckers eingetragenen Dateien ausgesetzt wird, bis Sie den Befehlszusatz /P eingeben.

/P Für PRINT: Dieser Befehlszusatz bewirkt, daß die Datei mit der vorhergehenden Dateibezeichnung und alle folgenden in die Warteschlange des Druckers eingetragen werden, bis Sie den Befehlszusatz /C eingeben.

Die Eingabe des Befehls PRINT ohne Optionen bewirkt die Anzeige des Inhalts der Warteschlange des Druckers am Bildschirm, ohne daß dabei die Warteschlange verändert wird.

VERZEICHNIS DER MS-DOS-BEFEHLE

Drive not ready

Wenn diese Meldung am Bildschirm erscheint, während PRINT versucht, auf eine Diskette zuzugreifen, wird der Zugriffsversuch wiederholt, bis das Laufwerk betriebsbereit ist. Liegt ein anderer Fehler vor, dann wird die momentane Dateibezeichnung aus der Warteschlange gelöscht. In diesem Fall würde eine Fehlermeldung auf dem Drucker ausgedruckt.

Alle Druckaufträge aufgehoben

Wenn der Befehlszusatz /T (TERMINATE) eingegeben wird, wird die Fehlermeldung "Alle Druckaufträge durch Bediener aufgehoben" auf Ihrem Drucker ausgedruckt. Wird der Name der gerade im Druck befindlichen Datei durch die Eingabe des Zusatzes C aus der Warteschlange gelöscht, dann wird die Fehlermeldung "Druckauftrag für Dateiname aufgehoben" auf dem Drucker ausgedruckt.

PROMPT

BEFEHLSNAME: PROMPT

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Mit diesem Befehl können Sie die System-Eingabeaufforderung ändern.

SYNTAX: PROMPT [**<Text der Eingabeaufforderung>**]

WIRKUNGSWEISE: Dieser Befehl ermöglicht es Ihnen, die MS-DOS-System-Eingabeaufforderung (zum Beispiel A>) zu ändern. Wenn Sie die Option nicht eingeben, wird die System-Eingabeaufforderung auf ihre Standardform gestellt, nämlich die Angabe des Standardlaufwerks. Sie können mit Hilfe der weiter unten beschriebenen Zeichen aber auch eine andere Eingabeaufforderung, wie beispielsweise die genaue Uhrzeit, vorgeben.

VERZEICHNIS DER MS-DOS-BEFEHLE

Mit Hilfe der folgenden Zeichen können Sie erreichen, daß der Befehl PROMPT besondere Formen von Eingabeaufforderungen festlegt. Um diese Zeichen in die Befehlszeile einzugeben, muß ihnen grundsätzlich ein Dollarzeichen (\$) vorangestellt werden:

Wenn Sie dieses Zeichen eingeben:	Erhalten Sie diese Eingabeaufforderung
\$	Das Dollarzeichen
t	Die genaue Uhrzeit
d	Das heutige Datum
p	Den momentanen Arbeitsbereich auf der Diskette im Standardlaufwerk
v	Die Nummer der verwendeten MS-DOS-Fassung
n	Die Angabe des Standardlaufwerks
g	Das Zeichen >
l	Das Zeichen <
q	Das Zeichen =
l	Die Befehlsfolge
-	Wagenrücklauf-Zeilenvorschub
s	Ein Leerzeichen (nur führendes möglich)
h	Einen Rückschritt
e	Den ASCII-Code X 1B(ESCAPE)

5

PROMPT

Beispiele:

PROMPT \$n Legt die normale MS-DOS-System-Eingabeaufforderung fest (d.h. das Größer-als-Zeichen).

PROMPT Uhrzeit = \$t\$ Datum = \$d

Legt eine zweizeilige Eingabeaufforderung folgender Form fest:
 Uhrzeit = (die genaue Uhrzeit)
 Datum = (das heutige Datum)

Wenn Ihr Computer über ein Steuerprogramm für ANSI-ESCAPE-Zeichenfolgen verfügt, können Sie in Ihren Eingabeaufforderungen ESCAPE-Steuerzeichenfolgen verwenden.

Zum Beispiel:

PROMPT \$e [7m\$n:\$e[m

Bringt die Eingabeaufforderungen in Negativdarstellung und kehrt für die Ausgabeweiteren Texts zur Positivdarstellung zurück.

Weitergabe sowie Vermittlung dieser Unterlagen, Verwertung, nach Mitteilung
 handlungen verpflichtet zu Schadensersatz. Alle Rechte für den Fall der Patent-
 erteilung oder Gebrauchsmustereintragung vorbehalten.

VERZEICHNIS DER MS-DOS-BEFEHLE

RECOVER

BEFEHLSNAME: RECOVER

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Dieser Befehl ermöglicht es Ihnen, auf einer Diskette mit fehlerhaften Sektoren eine Datei oder aber auch den ganzen Disketteninhalt wiederherzustellen.

SYNTAX: RECOVER <Dateiname | d:>

WIRKUNGSWEISE: Wenn auf Ihrer Diskette ein Sektor fehlerhaft ist, können Sie entweder die Datei, in der dieser Sektor enthalten ist (allerdings ohne den fehlerhaften Sektor) wiederherstellen oder aber den kompletten Disketteninhalt (wenn der fehlerhafte Sektor innerhalb des Arbeitsbereichs liegt).

Um eine bestimmte Datei wiederherzustellen, geben Sie bitte ein:

```
RECOVER <Dateiname>
```

Die Eingabe dieses Befehls bewirkt, daß MS-DOS die Datei Sektor für Sektor liest und dabei den bzw. die fehlerhaften Sektor(en) überspringt. Wenn MS-DOS den bzw. die fehlerhaften Sektor(en) gefunden hat, wird dieser bzw. werden diese markiert, und MS-DOS wird in Zukunft keine Daten mehr in diesen Sektor schreiben.

Wenn Sie den Inhalt einer ganzen Diskette wiederherstellen wollen, geben Sie ein:

```
RECOVER <d:>
```

wobei d: die Angabe des Laufwerks ist, in dem die Diskette liegt, deren Inhalt wiederhergestellt werden sollte.

Wenn im Grundarbeitsbereich nicht mehr genügend Speicherkapazität frei ist, bewirkt RECOVER die Ausgabe einer Meldung und schreibt Informationen bezüglich der Dateien, die nicht mehr bearbeitet werden konnten, in die Dateispeicherplatztabelle. Sie können den Befehl RECOVER erneut ausführen, um auch diese Dateien zurückzugewinnen, sobald in Ihrem Grundarbeitsbereich mehr Speicherplatz frei ist.

VERZEICHNIS DER MS-DOS-BEFEHLE

REM

BEFEHLSNAME: REM

BEFEHLSSTYP: Interner Befehl

VERWENDUNGSZWECK: Bewirkt während der Ausführung eines Stapelprogrammbausteins die Anzeige der Kommentare, die in derselben Programmzeile wie der REM-Befehl stehen.

SYNTAX: REM [Kommentar]

WIRKUNGSWEISE: In einer Kommentarzeile dürfen Sie als Abgrenzungszeichen nur Leerzeichen, Tabsprünge und Kommas verwenden.

Beispiele:

- 1: REM Dieser Stapelprogrammbaustein überprüft neue Disketten
- 2: REM Sein Name ist NEUDATEI.BAT
- 3: PAUSE Legen Sie eine neue Diskette in Laufwerk B ein
- 4: FORMAT B:/S
- 5: DIR B:
- 6: CHKDSK B:

REN

BEFEHLSNAME: REN (RENAME)

BEFEHLSSTYP: Interner Befehl

SYNONYM: RENAME

VERWENDUNGSZWECK: Bewirkt, daß in der ersten angegebenen Datei- bezeichnung der Dateiname durch den zweiten angegebenen Dateinamen ersetzt wird.

SYNTAX: REN <Dateibezeichnung><Dateibezeichnung>

WIRKUNGSWEISE: In der ersten angegebenen Datei- bezeichnung muß die Laufwerksangabe erscheinen, wenn die Diskette in einem anderen Laufwerk als dem Standardlaufwerk liegt. Eine Laufwerksangabe für die zweite Option (Dateiname) wird nicht berücksichtigt. Die Datei bleibt auf der Diskette, auf der sie sich befindet.



Wiedergabe sowie Verbreitung dieses Dokuments ist ohne schriftliche Genehmigung von Nixdorf Computer AG. Die Nixdorf-Produkte sind durch Patente geschützt. In Vertretung der Nixdorf-Produkte sind die Nixdorf-Produkte durch Patente geschützt. In Vertretung der Nixdorf-Produkte sind die Nixdorf-Produkte durch Patente geschützt.

VERZEICHNIS DER MS-DOS-BEFEHLE

In jeder der beiden Optionen können Sie nach Belieben Stellvertreterzeichen verwenden. Es werden grundsätzlich alle Dateien, die der Dateibezeichnung in der ersten Option entsprechen, neu benannt. Wenn Sie Stellvertreterzeichen im Dateinamen der zweiten Option verwenden, dann werden die entsprechenden Zeichenpositionen in der ersten Option nicht verändert.

Der folgende Befehl bewirkt beispielsweise, daß alle Dateibezeichnungen mit dem Suffix .LST statt dessen das Suffix .PRN erhalten:

```
REN *.LST *.PRN
```

Im nächsten Beispiel bewirkt der Befehl REN die Umbenennung der Datei REGEN auf der Diskette im Laufwerk B in NEGER:

```
REN B:REGEN N?G?R
```

Die Datei wird dabei auf der Diskette belassen, auf der sie sich befindet, nämlich auf derjenigen im Laufwerk B.

Wenn Sie versuchen, eine Datei mit einem Namen umzubenennen, der im Inhaltsverzeichnis bereits vorhanden ist, erhalten Sie die Fehlermeldung: "File not found".

RMDIR

BEFEHLSNAME: RMDIR

BEFEHLSTYP: Interner Befehl

SYNONYM: RD

VERWENDUNGSZWECK: Dieser Befehl ermöglicht es Ihnen, einen Arbeitsbereich samt seinem Inhaltsverzeichnis aus der hierarchischen Dateienverwaltung zu löschen.

SYNTAX: RMDIR <Suchwegbezeichnung>

WIRKUNGSWEISE: Dieser Befehl bewirkt die Löschung des angegebenen Arbeitsbereichs einschließlich seines Inhaltsverzeichnisses vorausgesetzt, daß er, bis auf die Symbole . und .., leer ist.

VERZEICHNIS DER MS-DOS-BEFEHLE

Wenn Sie also beispielsweise den Arbeitsbereich \BIN \BENUTZER \MEIER löschen möchten, geben Sie zuerst einen Befehl DIR ein, um sich zu vergewissern, daß dieser Arbeitsbereich keine wichtigen Dateien mehr enthält, die Sie nicht gelöscht haben möchten.

RMDIR \BIN \BENUTZER \MEIER

Damit wird sowohl der Arbeitsbereich als auch sein Inhaltsverzeichnis aus der hierarchischen Dateienverwaltungsorganisation gelöscht.

SET

BEFEHLSNAME: SET

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Dieser Befehl ermöglicht es Ihnen, einer bestimmten Zeichenfolge einen bestimmten Wert zuzuordnen, der dann von weiteren Programmen in dieser Form übernommen werden kann.

SYNTAX: SET [**<Zeichenfolge=Zeichenfolge>**]

WIRKUNGSWEISE: Dieser Befehl ist nur nützlich, wenn Sie für bestimmte Zeichenfolgen, die in Programmen angesprochen werden, die Sie geschrieben haben, bestimmte Werte festlegen möchten, die in diesen Programmen verwendet werden sollen. Jedes Anwendungsprogramm kann alle Werte, die mit Hilfe des Befehls SET festgelegt wurden, nachschlagen, indem es den Befehl SET ohne Optionen aufruft. So können Sie beispielsweise mit SET TTY=VT52 erreichen, daß Ihr TTY-Wert, d.h. die Bezeichnung Ihrer Datenstation, grundsätzlich VT52 lautet, bis Sie sie mit einem anderen SET-Befehl wieder ändern.

VERZEICHNIS DER MS-DOS-BEFEHLE

Den Befehl SET können Sie auch in Stapelprogrammbausteinen verwenden. Dort ermöglicht er Ihnen, Ihre Blindparameter über Namen anstatt über Nummern aufzurufen. Wenn Ihr Stapelprogrammbaustein die Anweisung "LINK %DATEI%" enthält, können Sie den Namen des Werts, den MS-DOS dieser Variablen zuordnen soll, mit dem Befehl SET festlegen. So würde der Befehl SET DATEI=MIETE den Parameter %DATEI% durch den Dateiname MIETE ersetzen. Wenn Sie also die Namen Ihrer Blindparameter austauschen wollen, brauchen Sie dafür nicht jeden einzelnen Stapelprogrammbaustein eigens zu ändern. Beachten Sie bitte, daß der Name durch ein Prozentzeichen abgeschlossen werden muß, wenn Sie Text (anstatt Zahlen) als Blindparameter verwenden.

SORT

BEFEHLSNAME: SORT

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Der Befehl SORT bewirkt, daß die Daten von Ihrem Bildschirm gelesen, dann sortiert und schließlich in sortierter Form entweder wieder auf Ihren Bildschirm oder wahlweise in Dateien geschrieben werden.

SYNTAX: SORT [/R] [/+n]

WIRKUNGSWEISE: Der Befehl SORT kann beispielsweise dazu verwendet werden, eine Datei in eine bestimmte Spalte alphabetisch zu sortieren. Zwei Befehlszusätze stehen Ihnen zur Wahl der möglichen Optionen zur Verfügung:

/R Bewirkt eine Sortierung in absteigender Reihenfolge, d.h. von Z nach A.

/+n Bewirkt den Beginn der Sortierung in Spalte n, wobei n eine Zahl ist. Wenn Sie diesen Befehlszusatz nicht eingeben, beginnt SORT standardmäßig mit dem Sortieren in Spalte 1.

VERZEICHNIS DER MS-DOS-BEFEHLE

Beispiele:

Der folgende Befehl bewirkt, daß die Datei UNSORT.TXT gelesen, in absteigender Reihenfolge sortiert und dann in die Datei mit dem Namen SORT.TXT geschrieben wird:

```
SORT /R UNSORT.TXT SORT.TXT
```

Die nachstehende Befehlszeile bewirkt die Verkettung des Bearbeitungsbefehls DIR mit dem Verarbeitungsbefehl SORT. SORT bewirkt, daß das Inhaltsverzeichnis ab Spalte 14 (das ist die Spalte, in der die Angabe des Umfangs der Dateien im Verzeichnis beginnt) sortiert und das Ergebnis wie der auf den Bildschirm gebracht wird. Sie erreichen mit diesem Befehl die Anzeige eines nach Dateigrößen sortierten Inhaltsverzeichnisses:

```
DIR | SORT /+14
```

Der Befehl:

```
DIR | SORT/+14 | MORE
```

hat dieselbe Wirkung wie der Befehl im vorhergehenden Beispiel mit dem Unterschied, daß der Verarbeitungsbefehl MORE Ihnen die Möglichkeit verschafft, das am Bildschirm ausgegebene sortierte Inhaltsverzeichnis seitenweise zu lesen.

SYS

BEFEHLSNAME: SYS (SYSTEM)

BEFEHLSTYP: Externer Befehl

VERWENDUNGSZWECK: Dieser Befehl bewirkt die Übertragung der MS-DOS- Systemdateien von der Diskette im Standardlaufwerk auf die Diskette in dem mit d angegebenen Laufwerk.

SYNTAX: SYS <d> :

WIRKUNGSWEISE: Der Befehl SYS dient normalerweise dazu, das Betriebssystem auf den neuesten Stand zu bringen oder es auf eine formatierte leere Diskette zu übertragen. Es wird auf jeden Fall eine Angabe zu d: benötigt.

VERZEICHNIS DER MS-DOS-BEFEHLE

Auf der Zieldiskette werden die Dateien IO.SYS und MSDOS.SYS dieselbe Speicherkapazität beanspruchen wie das neue System. Das bedeutet, daß Sie Systemdateien von einer MS-DOS-2.0-Diskette nicht auf eine MS-DOS-1.1-Diskette übertragen können. Sie müssen in diesem Fall die MS-DOS-1.1-Diskette mit Hilfe des Befehls FORMAT neu formatieren, bevor Sie den Befehl SYS anwenden können.

Die Zieldiskette muß außerdem ganz leer sein oder bereits die Systemdateien IO.SYS und MSDOS.SYS enthalten. Bei der Übertragung werden die Dateien in dieser Reihenfolge kopiert:

IO.SYS
MSDOS.SYS

IO.SYS und MSDOS.SYS sind beide unsichtbare Dateien, die nicht angezeigt werden, wenn Sie den Befehl DIR ausführen. Die Datei COMMAND.COM, d.h. der Befehlsprozessor selbst, wird nicht übertragen. Um diese Datei zu übertragen, müssen Sie den Befehl COPY benutzen.

Entdeckt der Befehl SYS einen Fehler, wird eine der folgenden Meldungen angezeigt:

No room for system on destination disk
MS-DOS findet auf der Zieldiskette keinen ausreichenden Speicherplatz für die Dateien IO.SYS und MSDOS.SYS.

Incompatible system size

Die von den Systemdateien IO.SYS und MSDOS.SYS beanspruchte Speicherkapazität auf der Zieldiskette stimmt nicht mit der von den neuen Systemen benötigten überein.

VERZEICHNIS DER MS-DOS-BEFEHLE

TIME

BEFEHLSNAME: TIME

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Dieser Befehl ermöglicht es Ihnen, die genaue Uhrzeit einzustellen und anzuzeigen.

SYNTAX: TIME [<hh> [: <mm>]]

WIRKUNGSWEISE: Wenn Sie den Befehl TIME ohne Argumente eingeben, wird die folgende Meldung angezeigt:

Zeit ist <hh>:<mm>:<ss>.<cc>
Neue Zeit eingeben (hh:mm):

Wenn Sie die angezeigte Uhrzeit nicht ändern möchten, brauchen Sie nur die RETURN-Taste zu betätigen. Wenn Sie eine neue Uhrzeit eingeben möchten, dann schreiben Sie zum Beispiel:

TIME 8:20

Die Zeit darf nur mit Zahlen eingegeben werden; Buchstaben sind nicht zulässig. Eingegeben werden können folgende Zahlen:

<hh> = 00 bis 24n <mm> = 00 bis 59

Stunden- und Minutenangaben müssen durch Doppelpunkte gegeneinander abgegrenzt sein. Die Sekunden <ss> und Hundertstelsekunden <cc> können Sie nicht eingeben.

Wenn die eingegebenen Daten und Abgrenzungszeichen gültig sind, wird die neue Zeitangabe von MS-DOS ab dem Zeitpunkt der Eingabe verwendet. Sind die eingegebenen Daten oder Abgrenzungszeichen nicht gültig, dann gibt MS-DOS folgende Meldung aus:

Invalid time
Enter new time:

MS-DOS wartet dann darauf, daß Sie eine gültige neue Zeit eingeben.



VERZEICHNIS DER MS-DOS-BEFEHLE

TYPE

- BEFEHLSNAME:** TYPE
- BEFEHLSSTYP:** Interner Befehl
- VERWENDUNGSZWECK:** Bewirkt die Anzeige des Dateiinhalts am Bildschirm.
- SYNTAX:** TYPE<Dateibezeichnung>
- WIRKUNGSWEISE:** Dieser Befehl ermöglicht es Ihnen, den Inhalt einer Datei am Bildschirm zu prüfen, ohne diese Datei zu verändern. (Wenn Sie den Namen der Datei nicht wissen, können Sie ihn sich mit Hilfe des Befehls DIR im Inhaltsverzeichnis zeigen lassen. Wenn Sie den Inhalt der Datei ändern möchten, steht Ihnen dafür der Zeilen- Editor EDLIN zur Verfügung.) Der einzige Formatierungsvorgang, der bei der Ausführung von TYPE stattfindet, besteht darin, daß Tabsprünge mit den Standard-tabulatorabständen (je 8 Spalten) abgestimmt werden. Beachten Sie bitte, daß die Anzeige von Binärdateien bewirkt, daß Steuerzeichen (wie beispielsweise CONTROL-Z) wie Warnton, Papiervorschub und ESCAPE-Befehlszeichen folgen an Ihren Computer gesendet werden.

VERZEICHNIS DER MS-DOS-BEFEHLE

VER

BEFEHLSNAME: VER

BEFEHLSSTYP: Interner Befehl

VERWENDUNGSZWECK: Bewirkt die Ausgabe der Nummer der verwendeten MS-DOS- Fassung.

SYNTAX: VER

WIRKUNGSWEISE: Wenn Sie nicht mehr wissen, welche Fassung von MS-DOS Sie gerade benutzen, geben Sie VER ein. Die Nummer der verwendeten Fassung von MS-DOS erscheint dann an Ihrem Bildschirm.

VERIFY

BEFEHLSNAME: VERIFY

BEFEHLSSTYP: Interner Befehl

VERWENDUNGSZWECK: Schaltet die Prüffunktion beim Schreiben auf Diskette ein oder aus.

SYNTAX: VERIFY [ON | OFF]

WIRKUNGSWEISE: Dieser Befehl dient demselben Zweck wie der Befehlszusatz /V in dem Befehl COPY. Wenn Sie sich vergewissern wollen, daß Ihre Dateien ordnungsgemäß auf die Diskette geschrieben wurden, können Sie mit Hilfe des Befehls VERIFY veranlassen, daß MS-DOS bei jedem Vorgang prüft, ob Ihre Dateien fehlerlos übertragen wurden (zum Beispiel, daß keine fehlerhaften Sektoren vorliegen). Die Prüffunktion wird von MS-DOS jedesmal aktiviert, wenn Sie Daten auf eine Diskette schreiben. Eine Fehlermeldung erhalten Sie nur, wenn es MS-DOS nicht gelungen ist, eine fehlerfreie Übertragung auf die Diskette durchzuführen.

Die Prüffunktion bleibt eingeschaltet, bis Sie sie über ein Programm (mit Hilfe eines SET-VERIFY-Systemaufrufs) wieder ausschalten oder den Befehl VERIFY OFF eingeben.

Wenn Sie den momentanen Zustand der Funktion in Erfahrung bringen möchten, geben Sie VERIFY ohne Optionen ein.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden, Zuvorherige Genehmigung erforderlich. Gebrauchsmusteremittlung vorbehalten.



VERZEICHNIS DER MS-DOS-BEFEHLE

VOL

BEFEHLSNAME: VOL (VOLUME)

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Bewirkt die Anzeige der Diskettenbezeichnung, falls eine solche festgelegt wurde.

SYNTAX: VOL [d:]

WIRKUNGSWEISE: Dieser Befehl bewirkt die Ausgabe der Diskettenbezeichnung der Diskette in Laufwerk d. Wenn Sie kein Laufwerk angeben, wird die Diskettenbezeichnung der Diskette im Standardlaufwerk angezeigt.

Wenn die betreffende Diskette keine Diskettenbezeichnung besitzt, bewirkt der Befehl VOL die Anzeige der folgenden Meldung:

Volume in drive x has no label .

VERZEICHNIS DER MS-DOS-BEFEHLE

5.3 STAPELPROGRAMMBAUSTEINBEFEHLE

Die folgenden Befehle wollen wir Stapelprogrammbausteinbefehle nennen. Sie ermöglichen es Ihnen, Ihre Stapelprogrammbausteine vielseitiger und leistungsfähiger zu gestalten. Wir werden im folgenden die Befehle ECHO, FOR, GOTO, IF und SHIFT beschreiben.

Wenn Sie nicht die Absicht haben, Stapelprogrammbausteine zu schreiben bzw. zu verwenden, brauchen Sie dieses Unterkapitel nicht zu lesen.

ECHO

BEFEHLSNAME: ECHO

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Ermöglicht es Ihnen, die Funktion Kontrollanzeige ein- und auszuschalten.

SYNTAX: ECHO [ON | OFF | Meldung]

WIRKUNGSWEISE: Normalerweise erfolgt die Kontrollanzeige aller Befehle eines Stapelprogrammbausteins am Bildschirm, sobald der Befehlsprozessor sie liest. Wenn Sie ECHO OFF eingeben, wird diese Kontrollanzeige ausgeschaltet. Mit ECHO ON können Sie sie wieder einschalten.

Wenn Sie nur den Befehl ECHO ohne die Optionen ON oder OFF eingeben, wird der derzeitige Zustand der Funktion am Bildschirm gezeigt.



VERZEICHNIS DER MS-DOS-BEFEHLE

FOR

BEFEHLSNAME: FOR

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Befehlserweiterung, die bei der Datei-
verarbeitung sowohl im Dialogverfahren als
auch über Stapelprogrammbausteine verwendet
werden kann.

SYNTAX: In Stapelprogrammbausteinen:
FOR %%<c> IN <Menge> DO <Befehl>
Im Dialogbetrieb:
FOR %<c> IN <Menge> DO <Befehl>

WIRKUNGSWEISE: <c> kann ein beliebiges Zeichen mit Ausnahme
der Ziffern von 0 bis 9 sein. Diese
Einschränkung ist erforderlich, um
Verwechslungen mit den Stapelprogramm-
bausteinparametern %0 bis %9 zu vermeiden.

<Menge> ist (%<Posten>*)

Der Variablen %%<c> wird nacheinander jeder in <Menge> enthaltene
Wert zugeordnet und danach wird jeweils <Befehl> berechnet. Ist
einer der Werte von <Menge> ein Ausdruck mit einem der
Stellvertreterzeichen * und/oder ?, dann wird der Variablen
nacheinander jedes passende Muster von der Diskette zugeordnet.
In diesem Fall darf Menge nur einen einzigen <Posten> enthalten,
da weitere Posten grundsätzlich unberücksichtigt bleiben.

HINWEIS:

Die Wörter IN, FOR und DO müssen in Großbuchstaben geschrieben
werden.

Beispiele:

```
FOR %%f IN ( *.ASM) DO MASM %%f;
```

```
FOR %%f IN (BLAU ROT GRÜN) DO REM %%f
```

VERZEICHNIS DER MS-DOS-BEFEHLE

Die Eingabe eines Doppelprozentzeichens (%%) ist erforderlich, damit nach der Verarbeitung der Stapelprogrammbausteinparameter (%0 bis %9) ein Prozentzeichen (%) übrigbleibt. Wenn nur ein einziges Prozentzeichen (%) eingegeben würde (%f), würde der Stapelprogrammbaustein-Parameterprozessor das Prozentzeichen erkennen, das danach stehende Zeichen (f) prüfen, feststellen, daß f an dieser Stelle ein ungültiges Zeichen ist und die Variable folgerichtig aussondern, so daß der Befehl FOR sie niemals erkennen könnte. Steht FOR jedoch nicht in einem Stapelprogrammbaustein, dann darf nur ein Prozentzeichen (%) verwendet werden.



GOTO

BEFEHLSNAME: GOTO

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Befehlserweiterung, die bei der Dateiverarbeitung sowohl im Dialogverfahren als auch über Stapelprogrammbausteine verwendet wird.

SYNTAX: GOTO <Marke>

WIRKUNGSWEISE: GOTO bewirkt, daß in dem Stapelprogrammbaustein die Befehle ab der Zeile nach der Marke zur Ausführung gebracht werden. Wurde keine Marke festgelegt, dann wird die Ausführung des Stapelprogrammbausteins beendet.

Beispiel:

```
: Blau
REM Schleife ...
GOTO Blau
```

Der im Beispiel dargestellte Befehl wird eine unendliche Folge von Meldungen mit dem Inhalt "Schleife ..." erzeugen.

Wenn Sie in einem Stapelprogrammbaustein eine Zeile mit einem Doppelpunkt (:) beginnen, wird diese Zeile bei der Ausführung des Stapelprogrammbausteins übergangen. Die auf GOTO folgenden Zeichen stellen grundsätzlich eine Marke dar. Dieses Verfahren kann auch dazu verwendet werden, Kommentarzeilen einzusetzen.

Wiedergabe sowie Verweigerung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich angegeben, ist diese Unterlage den Besondereigentümern vorbehalten.

VERZEICHNIS DER MS-DOS-BEFEHLE

IF

BEFEHLSNAME: IF

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Befehlsweiterung, die bei der Dateiverarbeitung sowohl im Dialogverfahren als auch über Stapelprogrammabauusteine verwendet wird.

SYNTAX: IF<Bedingung><Befehl>

WIRKUNGSWEISE: Der Parameter <Bedingung> kann eine der folgenden Formen annehmen:

ERRORLEVEL<Zahl>

Diese Bedingung ist nur dann wahr, wenn das zuvor über COMMAND ausgeführte Programm einen Ausgangscode Zahl oder höher hatte.

<Zeichenfolge1>==<Zeichenfolge2>

Diese Bedingung ist nur dann wahr, wenn <Zeichenfolge1> und <Zeichenfolge2> nach der Parametervertauschung gleich sind. In den Zeichenfolgen sind keine Abgrenzungszeichen zulässig.

EXIST <Dateiname>

Diese Bedingung ist nur dann wahr, wenn <Dateiname> existiert.

NOT<Bedingung>

Diese Bedingung ist nur dann wahr, wenn <Bedingung> falsch ist.

Die Anweisung IF ermöglicht die bedingte Ausführung von Befehlen. Wenn die <Bedingung> wahr ist, wird der <Befehl> ausgeführt. Wenn nicht, bleibt der <Befehl> unberücksichtigt.

HINWEIS:

Die Worte ERRORLEVEL, EXIST und NOT müssen in Großbuchstaben eingegeben werden.

Beispiele:

```
IF NOT EXIST \TMP\FOO ECHO Ich kann die Datei nicht finden!
```

```
IF NOT ERRORLEVEL 3LINK $1,,;
```

VERZEICHNIS DER MS-DOS-BEFEHLE

PAUSE

BEFEHLSNAME: PAUSE

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Setzt die Ausführung eines Stapelprogramm-
bausteins aus.

SYNTAX: PAUSE [Kommentar]

WIRKUNGSWEISE: Während der Ausführung eines Stapelprogramm-
bausteins kann es notwendig werden, Disketten
zu wechseln oder andere Maßnahmen zu ergreifen,
die eine Unterbrechung in der
Programmausführung erfordern. Der Befehl
PAUSE setzt die Ausführung des Programms aus,
bis Sie eine beliebige Taste mit Ausnahme von
<CONTROL-C> betätigen.

Sobald der Befehlsprozessor auf PAUSE stößt, gibt er die Meldung
aus:

Strike any key when ready

Betätigen Sie die Taste <CONTROL-C,> wird eine weitere Meldung
ausgegeben

Abort batch job (Y/N)?

Wenn Sie diese Eingabeaufforderung mit J beantworten, wird die
Ausführung des restlichen Teils des Stapelprogramm-
bausteins abgebrochen und auf die normale Betriebssystemebene zurückver-
zweigt. Daraus ergibt sich die Möglichkeit, PAUSE für das Zerlegen einer
Stapelprogramm-
bausteindatei in mehrere Teile zu verwenden. Des
weiteren kann die Ausführung des Stapelprogramm-
bausteins an jeder beliebigen Stelle beendet werden.

Der Kommentar zu dem Befehl ist wahlfrei und kann in die Befehls-
zeile von PAUSE eingegeben werden. Mit Hilfe dieses Kommentars
können Sie jedoch auch den Benutzer der Stapelprogramm-
bausteindatei über geeignete Mitteilungen ansprechen, wenn die
Ausführung des Stapelprogramm-
bausteins ausgesetzt ist. So können
Sie zum Beispiel eine Aufforderung zum Wechseln von Disketten in
den Befehl einbauen. In solchen Fällen verwendet man eine wahl-
freie Eingabeaufforderung. Diese wird grundsätzlich vor der
Meldung "Wenn bereit, eine beliebige Taste betätigen" ausgegeben.

VERZEICHNIS DER MS-DOS-BEFEHLE

SHIFT

BEFEHLSNAME: SHIFT

BEFEHLSTYP: Interner Befehl

VERWENDUNGSZWECK: Ermöglicht den Zugriff auf mehr als 10 Blindparameter bei der Ausführung von Stapelprogrammbausteindateien.

SYNTAX: SHIFT

WIRKUNGSWEISE: Normalerweise kann in Befehlsdateien nur über die 10 Blindparameter %0 bis %9 verfügt werden. Wenn Sie mehr als zehn Blindparameter benötigen, können Sie mit Hilfe des Befehls SHIFT die Befehlszeilenparameter ändern.

Zum Beispiel:

wenn %0 = "Blume"
%1 = "Baum"
%2 = "Name"
%3 bis %9 leer sind,

bewirkt ein Befehl SHIFT die folgende Veränderung:

%0 = "Baum"
%1 = "Name"
%2 bis %9 sind leer.

Wenn eine Befehlszeile mehr als 10 Parameter enthält, werden die nach dem zehnten (%9) auftretenden Parameter durch aufeinanderfolgende SHIFT-Befehle nacheinander in die Position %9 verschoben.

SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

**6 SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN
IN MS-DOS**

6.1 Sonderbearbeitungstasten in MS-DOS.....6 - 2
6.2 Steuerzeichenfunktionen in MS-DOS.....6 - 8

„Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung
ihres Inhalts, Vervielfältigung oder Verbreitung, auch auszugsweise, ist ohne
schriftliche Genehmigung ausdrücklich untersagt. Alle Rechte für den Fall der Patent-
erteilung oder Gebrauchsmustererteilung vorbehalten.“



SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

6.1 SONDERBEARBEITUNGSTASTEN IN MS-DOS

Die MS-DOS-Sonderbearbeitungstasten müssen im einzelnen beschrieben werden, da die Funktionen, die sie ermöglichen, sich von denjenigen konventioneller Betriebssysteme weitestgehend unterscheiden. So brauchen Sie zum Beispiel komplizierte Zeichenfolgen nicht so oft zu wiederholen, da in MS-DOS die zuletzt eingegebene Befehlszeile automatisch in einen speziellen Zeilenspeicher gestellt wird.

Dank diesem Zeilenspeicher und den MS-DOS-Sonderbearbeitungstasten verfügen Sie über folgende Möglichkeiten:

1. Jede Zeile kann mühelos durch Betätigung von nur zwei Tasten wiederholt werden.
2. Wenn Sie mitten in Ihrer Befehlszeile einen Fehler gemacht haben, können Sie Ihre Befehlszeile überarbeiten bzw. berichtigen, ohne die ganze Befehlszeile neu eingeben zu müssen.
3. Wenn eine einzugebende Befehlszeile einer bereits zuvor erstellten ähnlich ist, brauchen Sie sie nicht ganz neu zu schreiben, sondern können die zuvor eingegebene Zeile mit Hilfe einer Sonderbearbeitungstaste abändern und übernehmen.

Der Zusammenhang zwischen einer Befehlszeile und dem Zeilenspeicher ergibt sich aus Abbildung 8.

Abbildung 8. Befehlszeile und Zeilenspeicher



SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

Wie Sie in Abbildung 8 sehen können, gibt der Benutzer einen Befehl an MS-DOS in die Befehlszeile ein. Wenn Sie die <RETURN> Taste betätigen, wird der Befehl automatisch zur Ausführung an den Befehlsprozessor (COMMAND.COM) weitergeleitet. Gleichzeitig wird eine Kopie dieses Befehls in den Zeilenspeicher gestellt. Sie können daher Ihren Befehl beliebig zurücknehmen oder mit Hilfe der Sonderbearbeitungstasten ändern.

In Tabelle 6.1 sind alle Sonderbearbeitungstasten komplett aufgelistet. Die Arbeitsweise dieser Tasten ist in Kapitel 7 "Der Zeilen-Editor (EDLIN)" näher erläutert, insbesondere bezüglich ihrer Verwendungsmöglichkeiten bei der Bearbeitung Ihrer Textdateien.

HINWEIS

Die Tasten auf Ihrer Tastatur entsprechen möglicherweise nicht den in den folgenden Beispielen angeführten. Deswegen werden wir bei der Beschreibung diese Tasten nicht über ihre Namen, sondern über ihre FUNKTION ansprechen. Wenn also in einem Beispiel gesagt wird, Sie sollen die <SKIP>Taste betätigen, müssen Sie auf Ihrer Tastatur die Taste suchen, die der Bearbeitungsfunktion "1 Zeichen überspringen" entspricht und sie betätigen. Für bestimmte Funktionen müssen Sie unter Umständen zwei Tasten drücken. Schlagen Sie bitte in der Betriebsanleitung Ihres Computers nach, welche Tasten welchen der hier beschriebenen MS-DOS-Bearbeitungsfunktionen entsprechen.

In der folgenden Tabelle haben wir eine leere Spalte vorgesehen, in die Sie die Tastenbezeichnungen Ihrer Tastatur eintragen können, so daß Sie die Sonderbearbeitungstasten leichter finden können.

SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

Tabelle 6.1. Übersicht über die Sonderfunktionstasten

MS-DOS-Taste	Bearbeitungsfunktion	Ihre Taste
<COPY1>	Kopiert ein Zeichen aus dem Zeilenspeicher in die Befehlszeile.	
<COPYUP>	Kopiert Zeichen aus dem Zeilenspeicher in die Befehlszeile bis zu dem angegebenen Zeichen.	
<COPYALL>	Kopiert alle restlichen Zeichen aus dem Zeilenspeicher in die Befehlszeile.	
<SKIP1>	Überspringt ein Zeichen im Zeilenspeicher, ohne es zu kopieren.	
<SKIPUP>	Überspringt die angegebenen Zeichen im Zeilenspeicher, ohne sie zu kopieren, bis zu dem angegebenen Zeichen.	
<VOID>	Löscht den bereits eingegebenen Teil der Befehlszeile, ohne jedoch den Zeilenspeicher anzutasten.	
<INSERT>	Schaltet die Betriebsart Einfügen ein/aus.	
<NEWLINE>	Ersetzt den Inhalt des Zeilenspeichers durch die erstellte Befehlszeile.	
<CONTROL-Z>	Setzt ein Dateiendezeichen in den neuen Zeilenspeicherinhalt.	

SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

Wenn Sie den folgenden Befehl eingeben:

DIR PROG.COM

zeigt MS-DOS auf Ihrem Bildschirm Informationen über die Datei PROG.COM an. Gleichzeitig wird die Befehlszeile in den Zeilenspeicher übertragen. Wollen Sie die Befehlszeile wiederholen, brauchen Sie nur zwei Tasten zu betätigen: **<COPYALL>** und die **<RETURN>** Taste.

Sobald Sie die Bearbeitungstasten betätigen, wird die wiederholte Befehlszeile wie folgt am Bildschirm angezeigt:

<COPYALL> DIR PROG.COM <RETURN>

Beachten Sie bitte, daß die Bearbeitungstaste **<COPYALL>** den Inhalt des Zeilenspeichers in Ihre Befehlszeile überträgt.

Die Betätigung der RETURN-Taste bewirkt, daß die Befehlszeile dem Befehlsprozessor zur Ausführung zugeführt wird.

Wenn Sie Informationen über eine Datei mit dem Namen PROG.ASM auf den Bildschirm holen möchten, können Sie den Inhalt des Zeilenspeichers verwenden und eingeben:

<COPYUP> C

<COPYUP> C bewirkt, daß alle Zeichen aus dem Zeilenspeicher in die Befehlszeile übertragen werden, und zwar bis zu dem **C**, aber ohne dieses. MS-DOS bringt folgende Anzeige:

DIR PRO_

Beachten Sie bitte, daß der Unterstrich Ihre Schreibmarke darstellt. Nun geben Sie ein:

.ASM

SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

Das Ergebnis ist:

DIR PROG.ASM

Die Befehlszeile DIR PROG.ASM steht nun im Zeilenspeicher und ist bereit, dem Befehlsprozessor zur Ausführung zugeführt zu werden. Wenn Sie dies wünschen, betätigen Sie die RETURN-Taste.

Nehmen wir einmal an, Sie möchten den folgenden Befehl ausführen:

TYPE PROG.ASM

Um dies zu erreichen, geben Sie ein:

TYPE <INSERT> <COPYALL> <RETURN-Taste>

Beachten Sie, daß die Zeichen, die Sie schreiben, unmittelbar in die Befehlszeile eingegeben werden und dort die entsprechenden Zeichen des Zeilenspeichers überschreiben. Dieses automatische Ersetzen wird ausgeschaltet, wenn Sie die Bearbeitungstaste INSERT drücken. Dadurch nehmen die Zeichen "TYPE" den Platz der Zeichen "DIR" im Zeilenspeicher ein. Um nun zwischen "TYPE" und "PROG.ASM" ein Leerzeichen einzufügen, betätigen Sie die Bearbeitungstaste <INSERT> und anschließend die Leertaste. Um den restlichen Inhalt des Zeilenspeichers in die Befehlszeile zu übertragen, drücken Sie die Tasten <COPYALL> und anschließend die <RETURN> Taste. Der Befehl TYPE PROG.ASM wird durch MS-DOS ausgeführt. Der Zeilenspeicher enthält nun TYPE PROG.ASM.

SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

Nehmen wir an, es wäre Ihnen ein Schreibfehler unterlaufen und Sie hätten BYTE anstatt TYPE geschrieben. Das Ergebnis wäre zunächst einmal ein Befehlsfehler gewesen. Als nächstes müssen Sie den Schreibfehler berichtigen. Wenn Sie die Befehlszeile mit dem Schreibfehler vor der Betätigung der RETURN-Taste durch Drücken der Bearbeitungstaste <NEWLINE> in den Zeilenspeicher gestellt haben:

BYTE PROG.ASM <NEWLINE>

können Sie jetzt den fehlerhaften Befehl berichtigen, indem Sie eingeben:

T <COPY1> P <COPYALL>

Die Bearbeitungstaste <COPY1> überträgt ein einzelnes Zeichen aus dem Zeilenspeicher in die Befehlszeile. Die Taste COPYALL überträgt dann den restlichen Inhalt des Zeilenspeichers in die Befehlszeile. Daraus ergibt sich als neue Befehlszeile die von Ihnen gewünschte:

TYPE PROG.ASM

Dasselbe Ergebnis können Sie auch erzielen, wenn Sie den Inhalt des Zeilenspeichers mit den Tasten <SKIP1> und <INSERT> wie folgt verändern:

<SKIP1> <SKIP1> <COPY1> <INSERT> YP <COPYALL>

Die Wirkung der Betätigung der angegebenen Tasten ist wie folgt:

<SKIP1>	-	Überspringt das erste Zeichen im Zeilenspeicher.
<SKIP1>	-	Überspringt das zweite Zeichen im Zeilenspeicher.
<COPY1>	T	Kopiert das dritte Zeichen im Zeilenspeicher.
<INSERT> YP	TYPE	Fügt zwei Zeichen ein.
<COPYALL>	TYPE PROG.ASM	Überträgt den Rest des Zeilenspeicherinhalts.



SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

Beachten Sie bitte, daß <SKIP1> die Befehlszeile nicht verändert. Diese Bearbeitungstaste verändert lediglich den Inhalt des Zeilenspeichers, wo sie den ersten Buchstaben löscht. Genauso löscht die Taste <SKIPUP> alle Zeichen im Zeilenspeicher bis zu dem angegebenen, dieses jedoch ausgeschlossen.

Wie Sie sehen, erleichtern Ihnen diese MS-DOS- Sonderbearbeitungstasten Ihre Arbeit am Bildschirm und auf der Tastatur erheblich. Im nächsten Abschnitt beschreiben wir eine Reihe von Steuerzeichenfunktionen, die Sie in ähnlicher Weise bei der Eingabe von Befehlen unterstützen.

6.2 STEUERZEICHENFUNKTIONEN IN MS-DOS

Eine Steuerzeichenfunktion ist eine Funktion, die sich in der Befehlszeile auswirkt. Wir haben bereits Steuerzeichenfunktionen kennengelernt, und zwar <CONTROL-C> und <CONTROL-S>. Weitere Steuerzeichenfunktionen beschreiben wir im folgenden.

Wir erinnern daran, daß Sie beim Schreiben der Steuerzeichen wie beispielsweise <CONTROL-C> die CONTROL-Taste drücken müssen, während Sie das entsprechende Zeichen, wie hier beispielsweise das C, schreiben.

SONDERBEARBEITUNGSTASTEN UND STEUERZEICHENFUNKTIONEN IN MS-DOS

Tabelle 6.2. Übersicht über die Steuerzeichenfunktionen

Steuerzeichen	Funktion
<CONTROL-N>	Schaltet die Kontrollanzeige um auf Drucker Ausgabe.
<CONTROL-C>	Bewirkt den Abbruch des momentanen Befehls.
<CONTROL-H>	Entfernt das letzte Zeichen aus der Befehlszeile und löscht dieses Zeichen vom Bildschirm.
<CONTROL-J>	Bewirkt eine physische Zeilenschaltung, ohne jedoch die Befehlszeile zu löschen. Betätigen Sie die Zeilenschaltungstaste (<LINE FEED >), um die momentane logische Zeile über die physischen Grenzen der Bildschirmzeile hinaus zu verlängern.
<CONTROL-P>	Schaltet um von Bildschirm Ausgabe auf Drucker Ausgabe.
<CONTROL-S>	Stoppt den Bildlauf bei Bildschirmausgabe. Drücken Sie eine beliebige Taste, um den Bildlauf wiederherzustellen.
<CONTROL-X>	Hebt die momentane Zeile auf, löscht die Befehlszeile und bewirkt zunächst die Ausgabe eines verkehrten Schrägstrichs (\), und dann einen Wagenrücklauf und eine Zeilenschaltung. Der Zeilenpeicher, mit dem die Sonderbearbeitungstasten arbeiten, wird davon nicht berührt.



Wenn Sie diese Veröffentlichung ohne Erlaubnis des Verlegers ververvielfältigen, übertragen oder in irgendeiner Weise öffentlich zugänglich machen, sind Sie rechtlich verpflichtet, die Rechte des Verlegers zu schützen. Alle Rechte für den Fall der Patentierung oder Gebrauchsmusteranmeldung vorbehalten.

DER ZEILEN-EDITOR EDLIN

7 DER ZEILEN-EDITOR (EDLIN)

7.1 Einführung.....7 - 2

7.2 Wie man den Zeilen-Editor EDLIN startet.....7 - 3

7.3 Sonderbearbeitungstasten in EDLIN.....7 - 5

7.4 Erläuterung der EDLIN-Befehle.....7 - 15

7.4.1 Die Befehloptionen.....7 - 19

7.5 Die EDLIN-Befehle.....7 - 21

7.6 Fehlermeldungen in EDLIN.....7 - 51

Wiedergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden, Zuwiderhandlung ist strafbar. Alle Rechte vorbehalten. Für den Fall der Patenterteilung oder Gebrauchsmustereintragung vorbehalten.

DER ZEILEN-EDITOR EDLIN

7.1 EINFÜHRUNG

In diesem Kapitel möchten wir Ihnen erklären, wie man das Zeilenbearbeitungsprogramm EDLIN, auch Zeilen-Editor genannt, benutzt. EDLIN hilft Ihnen, Dateien zu erstellen, zu ändern und anzuzeigen, ganz gleich ob es sich um (hochsprachliche) Programm- oder Textdateien handelt.

Mit dem Zeilen-Editor EDLIN können Sie:

1. Neue (hochsprachliche) Programmdateien erstellen und speichern;
2. vorhandene Dateien bearbeiten und sowohl die bearbeitete als auch die ursprüngliche Fassung speichern;
3. Zeilen löschen, bearbeiten, einfügen und anzeigen;
4. Textelemente innerhalb einer oder mehrerer Zeilen suchen, löschen oder ersetzen.

In Dateien, die mit EDLIN erstellt oder bearbeitet wurden, wird der Text in Zeilen eingeteilt, die jeweils bis zu 253 Zeichen lang sein können. Während der Bearbeitung werden den Textzeilen entsprechende Zeilennummern von EDLIN zugeordnet und angezeigt, ohne daß diese Nummern in der gespeicherten Datei in Erscheinung treten.

Wenn Sie neue Zeilen in einen Text einfügen, werden alle darauf folgenden Zeilen im Text automatisch neu nummeriert. Wenn Sie umgekehrt Zeilen aus einer Datei löschen, werden alle auf den gelöschten Text folgenden Zeilen ebenfalls automatisch neu nummeriert. Daraus ergibt sich, daß die Zeilen Ihrer Datei immer lückenlos und ordnungsgemäß nummeriert sind.

DER ZEILEN-EDITOR EDLIN

7.2 WIE MAN DEN ZEILEN-EDITOR EDLIN STARTET

Um EDLIN zu starten, geben Sie bitte ein:

EDLIN <Dateibezeichnung>

Wenn Sie eine neue Datei erstellen wollen, geben Sie als <Dateibezeichnung> den Namen der zu erstellenden Datei ein. Wenn EDLIN diese Dateibezeichnung auf der Diskette in dem entsprechenden Laufwerk nicht findet, erstellt der Zeilen-Editor eine neue Datei mit dem von Ihnen angegebenen Namen und bewirkt die Ausgabe der folgenden Meldung mit der nachstehenden Eingabeaufforderung:

New File

*

Beachten Sie bitte, daß die Eingabeaufforderung von EDLIN ein Sternchen ist (*).

Sie können nun zeilenweise Text in Ihre neue Datei schreiben. Sie müssen die Eingabe jedoch mit dem Befehl I ("Insert") beginnen, der weiter unten in diesem Kapitel beschrieben wird.

Wenn Sie eine bereits vorhandene Datei bearbeiten möchten, müssen Sie als <Dateibezeichnung> den Namen der Datei eingeben, die Sie bearbeiten möchten. Sobald EDLIN diese Datei in dem angegebenen oder im Standardlaufwerk gefunden hat, wird sie in den Arbeitsspeicher geladen. Paßt die ganze Datei in den Arbeitsspeicher, zeigt Ihnen der Zeilen-Editor dies mit folgender Meldung am Bildschirm an:

End of input file

*

Sie können nun mit Hilfe der Bearbeitungsbefehle des Zeilen-Editors die Datei bearbeiten.

Ist die Datei zu umfangreich, um ganz in den Arbeitsspeicher zu passen, dann lädt EDLIN soviel von der Datei in den Arbeitsspeicher, bis dieser zu 3/4 voll ist, und zeigt Ihnen das mit der Eingabeaufforderung am Bildschirm an. Sie können dann den Dateiabschnitt bearbeiten, der im Arbeitsspeicher abgelegt ist.

7

DER ZEILEN-EDITOR EDLIN

Wenn Sie den Teil der Datei, der noch nicht im Arbeitsspeicher steht, bearbeiten wollen, müssen Sie den bereits bearbeiteten Text teilweise auf Diskette abspeichern, um Platz im Arbeitsspeicher freizustellen. Erst dann kann EDLIN einen weiteren Abschnitt bzw. den Rest Ihrer Datei von der Diskette in den Arbeitsspeicher übertragen. Nähere Einzelheiten zu dem Verfahren finden Sie bei der Beschreibung der Befehle "Write" und "Append" in diesem Kapitel.

Wenn Sie Ihre Bearbeitung abgeschlossen haben, können Sie die ursprüngliche und die bearbeitete Fassung Ihrer Datei abspeichern, indem Sie den Befehl "End" verwenden. Der Befehl "End" wird im Abschnitt "Die EDLIN-Befehle" dieses Kapitels besprochen. Die ursprüngliche Datei wird durch das Suffix .BAK gekennzeichnet, während die neue Datei, d.h. die von Ihnen bearbeitete Fassung der ursprünglichen Datei, den Namen und das Suffix erhält, die Sie in dem EDLIN-Befehl angegeben haben. Die Originaldatei mit dem Suffix .BAK wird erst am Ende der Arbeitssitzung gelöscht, oder aber, wenn der Zeilen-Editor mehr Speicherkapazität auf der Diskette benötigt.

Versuchen Sie nicht, Dateien mit dem Suffix .BAK in der Dateibezeichnung zu bearbeiten. EDLIN geht davon aus, daß .BAK-Dateien Sicherungskopien bzw. alte Versionen von Dateien sind. Wenn Sie eine .BAK-Datei bearbeiten wollen, müssen Sie sie zunächst umbenennen, indem Sie das Suffix in der Dateibezeichnung (mit Hilfe des in Kapitel 5 beschriebenen MS-DOS-Befehls RENAME) ändern. Dann können Sie EDLIN starten, indem Sie die geänderte <Dateibezeichnung> angeben.

DER ZEILEN-EDITOR EDLIN

7.3 SONDERBEARBEITUNGSTASTEN IN EDLIN

Die bereits in Kapitel 6 beschriebenen Sonderbearbeitungstasten und der Zeilenspeicher können auch bei der Bearbeitung Ihrer Textdateien mit EDLIN verwendet werden. Diese Sonderbearbeitungstasten werden im folgenden in allen Einzelheiten besprochen.

In Tabelle 7.1 finden Sie eine Übersicht über die den verschiedenen Tastenfunktionen entsprechenden Befehle, die Funktionscodes der Tasten und deren Funktionsbeschreibungen. Eingehendere Beschreibungen der Sonderbearbeitungstasten folgen nach der Tabelle.

HINWEIS

Die Tasten auf Ihrer Tastatur entsprechen möglicherweise nicht den in den folgenden Beispielen angeführten. Deswegen werden wir bei der Beschreibung diese Tasten nicht über ihre Namen, sondern über ihre FUNKTION ansprechen. Wenn also in einem Beispiel gesagt wird, Sie sollen die SKIP1-Taste betätigen, müssen Sie auf Ihrer Tastatur die Taste suchen, die der Bearbeitungsfunktion "1 Zeichen überspringen" entspricht und sie betätigen. Für bestimmte Funktionen müssen Sie unter Umständen zwei Tasten drücken. Schlagen Sie bitte in der Betriebsanleitung Ihres Computers nach, welche Tasten welchen der hier beschriebenen MS-DOS-Bearbeitungsfunktionen entsprechen.

7

DER ZEILEN-EDITOR EDLIN

Tabelle 7.1. Übersicht über die Sonderbearbeitungstasten

Funktion	Funktions- code	Beschreibung
1 Zeichen kopieren	<COPY1>	Kopiert 1 Zeichen aus dem Zeilenspeicher in die neue Befehlszeile.
Kopieren bis zu einem bestimmten Zeichen	<COPYUP>	Kopiert bis zu dem angegebenen Zeichen alle Zeichen aus dem Zeilenspeicher in die neue Befehlszeile.
Inhalt des Zeilenspeichers kopieren	<COPYALL>	Kopiert alle restlichen Zeichen des Zeilenspeichers in die Befehlszeile.
1 Zeichen überspringen	<SKIP1>	Überspringt ein Zeichen, ohne es zu kopieren.
Zeichen bis zu dem angegebenen Zeichen überspringen	<SKIPUP>	Überspringt bis zu dem angegebenen Zeichen alle Zeichen im Zeilenspeicher, ohne sie zu kopieren.
Eingabe beenden	<VOID>	Hebt die momentane Eingabe auf, ohne jedoch den Zeilenspeicher zu tangieren.
Betriebsart Einfügen	<INSERT>	Startet/Beendet die Betriebsart Einfügen.
Betriebsart Ersetzen	<REPLACE>	Beendet die Standardbetriebsart Einfügen.
Zeilenspeicher ersetzen	<NEWLINE>	Ersetzt den Inhalt des Zeilenspeichers durch die neue Befehlszeile.

DER ZEILEN-EDITOR EDLIN

EDLIN

COPY1

BEFEHLSNAME: <COPY1>

VERWENDUNGSZWECK: Kopiert ein Zeichen aus dem Zeilenspeicher in die Befehlszeile.

WIRKUNGSWEISE: Die Betätigung der Sonderbearbeitungstaste <COPY1> bewirkt, daß ein Zeichen aus dem Zeilenspeicher in die Befehlszeile übertragen wird. Bei Betätigung der Taste wird die Betriebsart Einfügen eingeschaltet, ein Zeichen in die Befehlszeile eingefügt, und die Betriebsart Einfügen automatisch wieder beendet.

Beispiel:

Am Bildschirm steht folgendes:

```
1:*Das ist eine Musterdatei.  
1:*_
```

Zu Beginn der Bearbeitung steht die durch den Unterstrich dargestellte Schreibmarke am Anfang der zweiten Zeile. Wenn die Sonderbearbeitungstaste <COPY1> betätigt wird, wird der erste Buchstabe der ersten Zeile D an die entsprechende Stelle in der zweiten Zeile kopiert:

```
<COPY1> 1:*Das ist eine Musterdatei.  
<COPY1> 1:*D_
```

Bei jeder weiteren Betätigung der Bearbeitungstaste <COPY1> wird ein weiterer Buchstabe aus der ersten Zeile in die zweite kopiert:

```
<COPY1> 1:*Da_  
<COPY1> 1:*Das_  
<COPY1> 1:*Das _
```

„Weitergabe sowie Vervielfältigung dieses Unterlags-Verwertungs und Mitteilungsbearbeitungen vorbehalten zu Schadzwecken. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustrentwertung vorbehalten.“

DER ZEILEN-EDITOR EDLIN

COPYUP

BEFEHLSNAME: <COPYUP>

VERWENDUNGSZWECK: Kopiert bis zu dem angegebenen Zeichen alle Zeichen aus dem Zeilenspeicher.

WIRKUNGSWEISE: Wenn die Sonderbearbeitungstaste <COPYUP> betätigt wird, werden bis zu dem angegebenen Zeichen alle Zeichen aus dem Zeilenspeicher in die Befehlszeile kopiert. Das "angegebene" Zeichen ist das unmittelbar nach der Betätigung der Bearbeitungstaste COPYUP eingegebene Zeichen; es wird weder kopiert noch am Bildschirm angezeigt. Sobald Sie die Sonderbearbeitungstaste <COPYUP> betätigen, springt die Schreibmarke auf die Position des im Befehl angegebenen Zeichens. Wenn das angegebene Zeichen im Zeilenspeicher nicht gefunden wird, kann überhaupt nichts kopiert werden. Die Betätigung der Bearbeitungstaste <COPYUP> beendet auch automatisch die Betriebsart Einfügen.

Beispiel:

Am Bildschirm steht folgendes:

```
1:*Das ist eine Musterdatei.  
1:*  
_
```

Zu Beginn der Bearbeitung steht die durch den Unterstrich dargestellte Schreibmarke am Anfang der zweiten Zeile. Sobald Sie die Sonderbearbeitungstaste <COPYUP> betätigen, werden bis zu dem im Befehl angegebenen Zeichen (d.h. dem unmittelbar nach der Betätigung der Bearbeitungstaste eingegebenen Zeichen) alle Zeichen aus der ersten in die zweite Zeile kopiert:

```
1:*Das ist eine Musterdatei.  
<COPYUP> u 1:*Das ist eine M_  
_
```


DER ZEILEN-EDITOR EDLIN

SKIP1

BEFEHLSNAME: <SKIP1>

VERWENDUNGSZWECK: Überspringt ein Zeichen aus dem Zeilenspeicher übersprungen, ohne daß es in die Befehlszeile kopiert wird. Die Wirkungsweise von <SKIP1> ist der von <COPY1> ähnlich mit dem Unterschied, daß das jeweilige Zeichen bei <SKIP1> nicht in die Befehlszeile kopiert wird.

Beispiel:

Am Bildschirm steht folgendes:
1:*Dies ist eine Musterdatei.
1:*

Zu Beginn der Bearbeitung steht die durch den Unterstrich dargestellte Schreibmarke am Anfang der Zeile. Beim Betätigen von <SKIP1> wird das erste Zeichen im Zeilenspeicher (T) übersprungen:

1:*Dies ist eine Musterdatei.
1:*

Die Position der Schreibmarke hat sich nicht geändert, nur der Zeilenspeicher ist betroffen. Um zu sehen, was sich geändert hat, betätigen Sie jetzt die Taste <COPYALL>, wodurch der Rest des Zeilenspeichers in die Befehlszeile kopiert wird und die Schreibmarke hinter dem letzten Zeichen der Zeile steht.

1:*Dies ist eine Musterdatei.
<SKIP1> 1:*\n<COPYALL> 1:ies ist eine Musterdatei.

„Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden, Zuvorherherabgabe, Vervielfältigung, Verbreitung, auch auszugsweise, ist ohne schriftliche Genehmigung der Nixdorf Computer AG.“



DER ZEILEN-EDITOR EDLIN

SKIPUP

BEFEHLSNAME: <SKIPUP>

VERWENDUNGSZWECK: Überspringt die Zeichen im Zeilenspeicher bis zum angegebenen Zeichen.

WIRKUNGSWEISE: Wenn die Sonderbearbeitungstaste <SKIPUP> betätigt wird, so werden alle Zeichen im Zeilenspeicher bis hin zu dem angegebenen Zeichen übersprungen. Das angegebene Zeichen wird nicht kopiert und nicht auf dem Bildschirm angezeigt. Wenn der Zeilenspeicher das angegebene Zeichen nicht enthält, wird nichts übersprungen. Die Wirkungsweise von <SKIPUP> ist ähnlich der Wirkungsweise von <COPYUP> abgesehen davon, daß die Zeichen im Zeilenspeicher übersprungen und nicht kopiert werden.

Beispiel:

Am Bildschirm steht folgendes:

```
1:*Dies ist eine Musterdatei.
```

```
1:*
```

Zu Beginn der Bearbeitung steht die durch den Unterstrich dargestellte Schreibmarke am Anfang der Zeile. Indem man <SKIPUP> und ein Zeichen eingibt, werden alle Zeichen im Zeilenspeicher bis hin zu dem angegebenen Zeichen übersprungen.

```
1:*Dies ist eine Musterdatei.
```

```
<SKIPUP>u 1:*
```

Die Position der Schreibmarke hat sich nicht geändert. Um zu sehen, wieviel Zeichen im Zeilenspeicher übersprungen wurden, betätigen Sie <COPYALL>, um den Rest des Zeilenspeichers zu kopieren. Dies bewegt die Schreibmarke an das Ende der Zeile.

```
1:*Dies ist eine Musterdatei.
```

```
<SKIPUP> n 1:*
```

```
<COPYALL> 1:*ne Musterdatei.
```

DER ZEILEN-EDITOR EDLIN

COPYALL

BEFEHLSNAME: <COPYALL>

VERWENDUNGSZWECK: Kopiert den Inhalt des Zeilenspeichers ab der momentanen Position in die Befehlszeile.

WIRKUNGSWEISE: Wenn man die Bearbeitungstaste <COPYALL> betätigt, werden alle restlichen Zeichen des Zeilenspeichers in die Befehlszeile kopiert. Unabhängig von der Position der Schreibmarke zu dem Zeitpunkt, zu dem die Bearbeitungstaste <COPYALL> betätigt wird, erscheint der Rest der Zeile, und die Schreibmarke wird hinter das letzte Zeichen der Zeile gesetzt.

Beispiel:

Am Bildschirm steht folgendes:

```
1:*Das ist eine Musterdatei.  
1:*_
```

Zu Beginn der Bearbeitung steht die durch den Unterstrich dargestellte Schreibmarke am Anfang der zweiten Zeile. Mittels des Befehls SKIPUP M werden die Zeichen im Zeilenspeicher bis hin zum Zeichen M übersprungen:

```
1:*Das ist eine Musterdatei.  
<SKIPUP> M 1:*_
```

Die Position der Schreibmarke bleibt unverändert, da sich diese Funktion nur auf den Inhalt des Zeilenspeichers auswirkt. Wenn Sie sehen möchten, wieviel Zeichen des Inhalts des Zeilenspeichers Sie bereits übersprungen haben, betätigen Sie die Bearbeitungstaste <COPYALL>. Dadurch wird der Teil des Inhalts des Zeilenspeichers, der noch nicht übersprungen worden ist, in die Befehlszeile kopiert, und die Schreibmarke an das Ende dieser Zeile gesetzt.

```
<SKIPUP> M 1:*Das ist eine Musterdatei.  
<COPYALL> 1:*_Musterdatei._
```

DER ZEILEN-EDITOR EDLIN

VOID

BEFEHLSNAME: <VOID>

VERWENDUNGSZWECK: Beendet die Eingabe und bewirkt die Löschung der Befehlszeile.

WIRKUNGSWEISE: Wenn Sie die Bearbeitungstaste <VOID> betätigen, wird die Befehlszeile rückgängig gemacht (aber noch nicht vom Bildschirm gelöscht); der Zeilenspeicher bleibt dabei unangetastet. Um zu zeigen, daß die Befehlszeile rückgängig gemacht ist, wird nach der Betätigung der Taste <VOID> an deren Ende ein verkehrter Schrägstrich (\) gesetzt, und es erfolgt ein Wagenrücklauf mit Zeilenschaltung; außerdem wird die Betriebsart Einfügen beendet. Die Schreibmarke (durch den Unterstrich dargestellt) steht danach am Anfang einer neuen Zeile. Wenn Sie die Wirkung der Bearbeitungstaste <VOID> wieder rückgängig machen möchten, betätigen Sie die Taste <COPYALL>. Dadurch wird der Inhalt des Zeilenspeichers in die Befehlszeile kopiert und der Inhalt dieser letzteren somit wiederhergestellt.

Beispiel:

Am Bildschirm steht folgendes:

```
1:*Das ist eine Musterdatei.  
1:*_
```

Zu Beginn der Bearbeitung steht die durch den Unterstrich dargestellte Schreibmarke am Anfang der zweiten Zeile. Nehmen wir einmal an, Sie würden in diese Zeile "Musterdatei" schreiben:

```
1:*Das ist eine Musterdatei.  
1:*Musterdatei_
```

DER ZEILEN-EDITOR EDLIN

Wenn Sie die gerade eingegebene Zeile ("Musterdatei") nun wieder löschen und durch den alten Text "Das ist eine Musterdatei" ersetzen möchten, betätigen Sie einfach die Bearbeitungstaste <VOID>. Beachten Sie, daß nun am Ende der Zeile "Musterdatei" ein verkehrter Schrägstrich erscheint und die Schreibmarke an der ersten Schreibstelle der nächsten Zeile steht:

```
<VOID> 1:*Das ist eine Musterdatei.  
1:*Musterdatei$/  
1:_
```

Wenn Sie nun den soeben eingegebenen Zeileninhalt ("Musterdatei") beibehalten möchten oder auch zur Vorbereitung einer anderen Bearbeitungsfunktion, betätigen Sie die RETURN-Taste. Wenn Sie die Taste COPYALL betätigen, wird der ursprüngliche Inhalt des Zeilenspeichers in die Befehlszeile kopiert.

```
<COPYALL> 1: Das ist eine Musterdatei._
```

INSERT

BEFEHLSNAME: <INSERT>

VERWENDUNGSZWECK: Leitet die Betriebsart Einfügen ein oder beendet sie.

WIRKUNGSWEISE: Wenn Sie die Bearbeitungstaste <INSERT> betätigen, bewirken Sie damit, daß im Zeilen-Editor die Betriebsart Einfügen ein- bzw. ausgeschaltet wird. Die Position der Schreibmarke im Zeilenspeicher wird dadurch nicht verändert. Wenn Sie nun Zeichen einfügen, bewegt sich die Schreibmarke bei jeder Einfügung mit. Sobald die Einfügung abgeschlossen ist, kehrt sie allerdings wieder zu der Stelle zurück, auf der sie vor der Einfügung stand. Das bedeutet, daß Einfügungen grundsätzlich immer vor dem Zeichen erfolgen, unter dem die Schreibmarke steht.

DER ZEILEN-EDITOR EDLIN

Beispiel:

Am Bildschirm steht folgendes:

```
1:*Das ist eine Mustervorführung.  
1:*_
```

Zu Beginn der Bearbeitung steht die durch den Unterstrich dargestellte Schreibmarke am Anfang der zweiten Zeile. Betätigen Sie zum Beispiel die Taste <COPYUP> und dann die Taste v:

```
1:*Das ist eine Mustervorführung.  
<COPYUP> v 1:*Das ist eine Muster_
```

Nun drücken Sie die Bearbeitungstaste <INSERT> und fügen die Zeichenfolge "datei" ein:

```
1:*Das ist eine Mustervorführung.  
<COPYUP> v 1:*Das ist eine Muster_  
<INSERT> datei 1:*Das ist eine Musterdatei_
```

Wenn Sie jetzt die Bearbeitungstaste <COPYALL> betätigen, wird der Rest des Inhalts des Zeilenspeichers in die Befehlszeile kopiert:

```
1:*Das ist eine Musterdatei.  
<COPYALL> 1:*Das ist eine Musterdateivorführung._
```

Hätten Sie die RETURN-Taste gedrückt, dann würde der Rest des Inhalts des Zeilenspeichers nicht kopiert, sondern abgetrennt und die Befehlszeile würde mit dem Ende der Einfügung enden:

```
<INSERT> datei <RETURN> 1:*Das ist eine Musterdatei_
```

Um die Betriebsart Einfügen wieder zu verlassen, betätigen Sie einfach erneut die Bearbeitungstaste <INSERT>.

DER ZEILEN-EDITOR EDLIN

REPLACE

BEFEHLSNAME: <REPLACE>

VERWENDUNGSZWECK: Leitet die Betriebsart Ersetzen ein.

WIRKUNGSWEISE: Wenn Sie die Bearbeitungstaste <REPLACE> betätigen, bewirken Sie damit, daß im Zeilen-Editor die Betriebsart Einfügen aus- und die Betriebsart Ersetzen eingeschaltet wird. Alle Zeichen, die Sie nun eingeben, werden die entsprechenden Zeichen im Zeilenspeicher ersetzen bzw. überschreiben. Wenn Sie mit der Bearbeitung einer Zeile beginnen, ist die Betriebsart Ersetzen eingeschaltet. Wenn Sie die RETURN-Taste betätigen, wird der Rest des Inhalts des Zeilenspeichers gelöscht.

Beispiel:

Am Bildschirm steht folgendes:

```
1:*Das ist eine Musterdatei.
1:*_
```

Zu Beginn der Bearbeitung steht die durch den Unterstrich dargestellte Schreibmarke am Anfang der zweiten Zeile. Betätigen Sie nun viermal nacheinander die Bearbeitungstaste <COPY1>, dann die Taste <REPLACE> und schreiben Sie "war", dann betätigen Sie die Taste <COPYALL>:

```
1:*Das ist eine Mustervorführung.
4 x <COPY1> 1:*Das
<REPLACE> war 1:*Das war_
<COPYALL> 1:*Das war_eine Mustervorführung._
```

Beachten Sie bitte, daß Sie die Zeichenfolge "ist" durch die Zeichenfolge "war" ersetzt bzw. überschrieben haben. Wenn Sie so viele Zeichen eingeben, daß sie über das Ende des Texts im Zeilenspeicher hinausführen, werden die im Zeilenspeicher noch verbleibenden Zeichen automatisch an Ihre Zeile angehängt, sobald Sie die Bearbeitungstaste <COPYALL> betätigen.



DER ZEILEN-EDITOR EDLIN

NEWLINE

BEFEHLSNAME: <NEWLINE>

VERWENDUNGSZWECK: Überträgt einen neuen Inhalt in den Zeilenspeicher.

WIRKUNGSWEISE: Wenn Sie die Bearbeitungstaste <NEWLINE> betätigen, wird der Inhalt der momentanen Befehlszeile in den Zeilenspeicher kopiert. Der alte Zeilenspeicherinhalt wird dabei gelöscht. Die Betätigung der Taste <NEWLINE> bewirkt die Ausgabe eines @ ("at"-Zeichen), einen Wagenrücklauf und eine Zeilenschaltung. Dabei wird auch die Befehlszeile rückgängig gemacht (aber nicht gleich am Bildschirm gelöscht), und die Betriebsart Einfügen wird beendet.

HINWEIS

Die Bearbeitungstaste <NEWLINE> hat im wesentlichen dieselbe Funktion wie die Taste <VOID>, mit dem Unterschied, daß der Inhalt des Zeilenspeichers verändert wird und ein @ ("at"-Zeichen) anstatt des \ (verkehrter Schrägstrich) ausgegeben wird.

Beispiel:

Am Bildschirm steht folgendes:

```
1:*Das ist eine Mustervorführung.  
1:*_
```

Zu Beginn der Bearbeitung steht die durch den Unterstrich dargestellte Schreibmarke am Anfang der zweiten Zeile. Wenn die Bearbeitungstaste <COPYUP> betätigt wird, werden bis zu dem danach eingegebenen Zeichen alle Zeichen im Zeilenspeicher kopiert:

```
<COPYUP> i 1:*Das ist eine Mustervorführung.  
<REPLACE> war 1:*Das _  
<COPYALL> 1:*Das war_eine Mustervorführung._
```


DER ZEILEN-EDITOR EDLIN

Nehmen wir einmal an, Sie würden, hier angelangt, nun beschließen, den Inhalt dieser somit fertiggestellten neuen Befehlszeile in den Zeilenspeicher zu übertragen. Dazu brauchen Sie nur die Bearbeitungstaste <NEWLINE> zu betätigen:

```
NEWLINE 1:*Das war eine Mustervorführung.@
```

Das @ weist darauf hin, daß diese Zeile nun den neuen Inhalt des Zeilenspeichers darstellt, den Sie bei Ihrer weiteren Bearbeitung wie gewohnt benutzen können.

EDLIN-BEFEHLE

7.4 ERLÄUTERUNG DER EDLIN-BEFEHLE

Die Befehle des Zeilen-Editors EDLIN ermöglichen die Ausführung von Bearbeitungsfunktionen in Textzeilen. Bevor Sie jedoch mit den EDLIN-Befehlen arbeiten, sollten Sie sich mit den folgenden Besonderheiten vertraut machen.

1. Die EDLIN-Befehle nehmen Suchwegbezeichnungen als Optionen an. So können Sie beispielsweise EDLIN \BIN\BENUTZER\MEIER\TEXT.TXT eingeben, wenn Sie die Datei TEXT.TXT im Unter-Arbeitsbereich MEIER bearbeiten möchten.
2. Wenn Sie eine bestimmte Zeile in Ihrem Text ansprechen möchten, können Sie dies mit Hilfe einer auf die momentane Bearbeitungszeile (die Zeile mit dem Sternchen) bezogenen relativen Zeilennummer. Wenn die betreffende Zeile vor bzw. über der momentanen Bearbeitungszeile steht, geben Sie die entsprechende Zahl von Zeilen mit dem Minuszeichen ein. Steht die gewünschte Zeile hinter bzw. unter der momentanen Bearbeitungszeile, geben Sie die entsprechende Zeilenzahl mit dem Pluszeichen ein.

7

DER ZEILEN-EDITOR EDLIN

Beispiel:

-10,+10L

Dieser Befehl (L = List = Auflisten) bewirkt die Auflistung der 10 Zeilen vor der momentanen Bearbeitungszeile, zuzüglich der momentanen Bearbeitungszeile selbst und der 10 Zeilen nach ihr.

3. Es können mehrere Befehle in eine Befehlszeile geschrieben werden. Wenn Sie einen Befehl zur Bearbeitung einer einzelnen Zeile mit einer Zeilennummer (<Zeile>) schreiben, müssen Sie, wenn Sie mehrere Befehle auf diese Zeile setzen möchten, die verschiedenen Befehle durch Strichpunkte voneinander abgrenzen. In allen anderen Fällen können Sie die verschiedenen Befehle einfach ohne besondere Abgrenzungszeichen hintereinander schreiben. Bei den Befehlen "Search" oder "Replace" kann die Zeichenfolge durch<CONTROL-Z>beendet werden, anstatt durch <RETURN>.

Beispiele:

Die folgende Befehlszeile bearbeitet die Zeile 15 und zeigt anschließend die Zeilen 10 bis 20 am Bildschirm an.

15;-5,+5L

Die Befehlszeile im nächsten Beispiel bewirkt, daß nach der Zeichenfolge "Abrakadabra" gesucht wird und dann 5 Zeilen vor und 5 Zeilen nach der Zeile angezeigt werden, in der diese Zeichenfolge gefunden wird. Wird die Zeichenfolge nicht gefunden, dann werden die 5 Zeilen vor und die 5 Zeilen nach der momentanen Bearbeitungszeile angezeigt:

SAbrakadabra <CONTROL-Z>-5,+L

4. Sie können EDLIN-Befehle mit oder ohne Leerzeichen zwischen der Zeilennummer und dem Befehl schreiben. Wenn Sie zum Beispiel die Zeile 6 löschen möchten, können Sie sowohl 6D als auch 6 D eingeben.
5. In der Betriebsart Einfügen können auch Steuerzeichen (wie beispielsweise CONTROL-C) in den Text eingefügt werden, wenn man den Anführungscode CONTROL-V davorsetzt. Durch CONTROL-V wird MS-DOS mitgeteilt, daß der darauf folgende Großbuchstabe als Steuerzeichen aufzufassen ist. Mit dem besonderen Anführungscode können auch Steuerzeichen in alle Zeichenfolgenargumente der Befehle "Search" und "Replace" eingesetzt werden.

DER ZEILEN-EDITOR EDLIN

Zum Beispiel:

S <CONTROL-V> Z bewirkt, daß CONTROL-Z in der Datei gesucht wird.

R <CONTROL-V> Z <CONTROL-Z> Blume bewirkt, daß alle CONTROL-Z in der Datei durch Blume ersetzt werden.

R <CONTROL-V> C <CONTROL-Z> Baum bewirkt, daß alle CONTROL-C durch Baum ersetzt werden.

Auch CONTROL-V selbst kann in den Text eingefügt werden, indem man eingibt CONTROL-V-V.

6. Das Steuerzeichen CONTROL-Z wird normalerweise dazu verwendet, für den Zeilen-Editor das Dateiende zu kennzeichnen. Wenn Sie also das Steuerzeichen CONTROL-Z in anderen Funktionen in Ihrer Datei verwenden wollen, müssen Sie EDLIN mitteilen, daß diese Steuerzeichen nicht "Dateiende" bedeuten. Dies ist möglich, indem Sie EDLIN mit Hilfe des Befehlszusatzes /B anweisen, alle CONTROL-Z-Zeichen in der Datei nicht zu berücksichtigen und Ihnen die ganze Datei anzuzeigen.

DER ZEILEN-EDITOR EDLIN

EDLIN-BEFEHLE

In der folgenden Übersichtstabelle sind alle EDLIN-Befehle zusammengefaßt. Sie werden nach dem folgenden Abschnitt über die Befehloptionen im einzelnen beschrieben.

Tabelle 7.2 Die EDLIN-Befehle

Befehle	Verwendungszweck
<Zeile>	Bewirkt die Bearbeitung der Zeile mit der Nummer Zeile .
A	Bewirkt das Anhängen von Zeilen.
C	Bewirkt das Kopieren von Zeilen.
D	Bewirkt das Löschen von Zeilen.
E	Bewirkt die ordnungsgemäße Beendigung der Bearbeitung.
I	Bewirkt das Einfügen von Zeilen.
L	Bewirkt das Auflisten von Text.
M	Bewirkt das Umstellen von Zeilen.
P	Bewirkt das Blättern im Text.
Q	Bewirkt den Abbruch der Bearbeitung und die Rückkehr in MS-DOS.
R	Bewirkt das Ersetzen von Text
S	Bewirkt das Suchen im Text.
T	Bewirkt das Übertragen von Text.
W	Bewirkt das Schreiben von Zeilen.

DER ZEILEN-EDITOR EDLIN

7.4.1 Die Befehlsoptionen

Mehrere EDLIN-Befehle verfügen über eine oder mehrere Optionen. Hierbei ist zu beachten, daß die Wirkung einer bestimmten Option, auch wenn die Bezeichnung der Option äußerlich gleich bleibt, sehr verschieden sein kann. Im folgenden beschreiben wir Ihnen die EDLIN-Befehlsoptionen.

<Zeile>

<Zeile> stellt eine Zeilennummer dar, die Sie eingeben. Zeilennummern müssen bei der Eingabe von anderen Zeilennummern, anderen Optionen und von dem Befehl selbst durch Kommas oder Leerzeichen abgegrenzt werden.

<Zeile> kann in drei verschiedenen Schreibweisen angegeben werden:

Zahl

Als Zeilennummer können Sie eine beliebige Zahl kleiner als 65534 eingeben. Wenn eine Zahl eingegeben wird, die größer ist als die höchste verwendete Zeilennummer, bedeutet Zeile einfach nur die Zeile nach der letzten nummerierten Zeile.

Punkt

Wenn Sie für Zeile einen Punkt (.) eingeben, wird diese Eingabe von EDLIN als die Nummer der momentanen Bearbeitungszeile interpretiert. Die momentane Bearbeitungszeile ist die letzte bearbeitete Zeile; sie muß nicht unbedingt auch die letzte der am Bildschirm angezeigten Zeilen sein. Die momentane Bearbeitungszeile wird von EDLIN am Bildschirm durch ein Sternchen (*) zwischen Zeilennummer und erstem Buchstaben der Zeile gekennzeichnet.

Pfundzeichen

Das Pfundzeichen (#) wird ebenfalls von EDLIN auf die Zeile nach der letzten nummerierten Zeile bezogen. Wenn Sie also für Zeile ein # eingeben, dann hat diese Eingabe dieselbe Wirkung wie die Eingabe einer Zeilennummer, die größer ist als die der letzten nummerierten Zeile.

DER ZEILEN-EDITOR EDLIN

<RETURN>

Wenn Sie statt eines der oben beschriebenen Inhalte für <Zeile> einfach nur einen Wagenrücklauf eingeben bzw. die RETURN-Taste betätigen, wird das von EDLIN als Anweisung aufgefaßt, einen für den betreffenden Befehl geeigneten Standardwert zu verwenden.

?

Das Fragezeichen wird nur im Zusammenhang mit den Befehlen "Replace" und "Search" verwendet. Ein Fragezeichen wird von EDLIN als Aufforderung interpretiert, Sie zu fragen, ob die gefundene Zeichenfolge die richtige ist. EDLIN wartet dann mit der Fortsetzung der Verarbeitung, bis Sie entweder mit ja antworten, indem Sie ein Y eingeben oder die RETURN-Taste drücken, oder aber mit nein, indem Sie eine beliebige andere Taste betätigen.

<Zeichenfolge>

Diese <Zeichenfolge> stellt den Text dar, der gesucht oder ersetzt werden soll, wenn zwei Zeichenfolgen hintereinander angegeben werden, den Text, durch den der erste <Zeichenfolge> ersetzt werden soll. Diese Option wird nur im Zusammenhang mit den Befehlen "Search" und "Replace" verwendet. Jede <Zeichenfolge> muß durch das Steuerzeichen <CONTROL-Z> abgeschlossen werden oder durch Betätigung der RETURN-Taste. (Nähere Einzelheiten hierzu finden Sie in der Beschreibung des Befehls "Replace.") Leerzeichen dürfen in den Zeichenfolgen nur dann erscheinen, wenn diese als Teil der Zeichenfolge aufgefaßt werden sollen. Desgleichen darf zwischen der Zeichenfolge und dem entsprechenden Befehlsbuchstaben kein Leerzeichen stehen.

DER ZEILEN-EDITOR EDLIN

7.5 DIE EDLIN-BEFEHLE

Die folgenden Seiten enthalten eine eingehende Beschreibung aller EDLIN-Befehle.

Append

BEFEHLSNAME: (A)ppend

VERWENDUNGSZWECK: Bewirkt, daß die angegebene Zahl von Zeilen von der Diskette gelesen und an die momentan im Arbeitsspeicher stehende zu bearbeitende Datei angehängt wird. Der hinzugefügte Dateiabschnitt wird an das Ende des bereits im Arbeitsspeicher stehenden gestellt.

SYNTAX: [n] A

WIRKUNGSWEISE: Die Verwendung dieses Befehls ist nur dann sinnvoll, wenn die zur Bearbeitung anstehende Datei zu groß ist, um ganz in den Arbeitsspeicher zu passen. Daher werden, wenn Sie EDLIN starten, gerade nur so viele Zeilen in den Arbeitsspeicher übertragen, wie dort hineinpassen, ohne daß EDLIN in der für die Bearbeitung benötigten Speicherkapazität eingeengt wird.

Wenn Sie einen weiteren Abschnitt der Datei, der noch auf Diskette steht, zur Bearbeitung in den Arbeitsspeicher übertragen möchten, müssen Sie einen entsprechend großen Teil der bereits bearbeiteten Datei aus dem Arbeitsspeicher wieder auf Diskette rückübertragen. Erst dann können Sie mit Hilfe des Befehls Append weitere unbearbeitete Zeilen von der Diskette in den Arbeitsspeicher übertragen. Wie man bearbeitete Zeilen auf die Diskette zurückschreibt, entnehmen Sie bitte der Beschreibung des Befehls Write in diesem Kapitel.

DER ZEILEN-EDITOR EDLIN

HINWEISE

1. Wenn Sie die Zahl der in den Arbeitsspeicher zu übertragenden Zeilen nicht angeben, dann überträgt EDLIN gerade so viele Zeilen, daß der Arbeitsspeicher insgesamt zu 3/4 voll wird. Ist der Arbeitsspeicher bereits zu 3/4 voll, dann werden überhaupt keine Zeilen übertragen.

2. Wenn der mit Append in den Arbeitsspeicher übertragene Dateiabschnitt der letzte ist, d.h. die letzte Zeile der Datei enthält, zeigt Ihnen EDLIN dies mit der Meldung "End of input files" an.

Copy

BEFEHLSNAME: (C)opy

VERWENDUNGSZWECK: Kopiert eine Reihe von Zeilen in die Zeile, deren Nummer im Befehl angegeben wird. Mit Hilfe der Option Anzahl kann der Kopiervorgang beliebig oft wiederholt werden.

SYNTAX: [**<Zeile>**] , [**<Zeile>**] ,**<Zeile>** , [**<Anzahl>**] C

WIRKUNGSWEISE: Die erste Option Zeile stellt die erste Zeilenzahl der zu kopierenden Zeilenreihe dar; die zweite Option Zeile stellt die letzte Zeilenzahl der zu kopierenden Zeilenreihe dar. Die dritte Option Zeile stellt die Zeilenzahl der Zeile dar, in die die Zeilenreihe kopiert werden soll. Lassen Sie die erste oder die zweite Option weg, dann wird die momentane Bearbeitungszeile als Standardoption kopiert. Lassen Sie die Option Anzahl weg, dann führt EDLIN den Kopiervorgang nur einmal durch.

Nach dem Kopiervorgang wird die damit erweiterte Datei automatisch neu durchnumeriert.

DER ZEILEN-EDITOR EDLIN

Es ist nicht möglich, eine Zeilenreihe in eine Zeile zu kopieren, die innerhalb der zu kopierenden Reihe steht.
Wenn Sie einen solchen Befehl eingeben würden, bekommen Sie die Fehlermeldung "Entry Error".
Zu einem solchen Eingabefehler würde beispielsweise der Befehl 3,20,15C führen.

Beispiele:

Nehmen wir als Beispiel die folgende bearbeitungsfähige Datei:

- 1: Dies ist eine Musterdatei zur
- 2: Vorführung des Befehls COPY.
- 3: Mit ihr wollen wir zeigen,
- 4: was passiert,
- 5: wenn Sie mit dem Befehl COPY
- 6: Text in Ihre Datei kopieren.

Diesen ganzen Textblock können Sie mit Hilfe des folgenden Befehls auf einmal kopieren:
1,6,7C

Das Ergebnis ist:

- 1: Dies ist eine Musterdatei zur
- 2: Vorführung des Befehls COPY.
- 3: Mit ihr wollen wir zeigen,
- 4: was passiert,
- 5: wenn Sie mit dem Befehl COPY
- 6: Text in Ihre Datei kopieren.
- 7: Dies ist eine Musterdatei zur
- 8: Vorführung des Befehls COPY.
- 9: Mit ihr wollen wir zeigen,.
- 10: was passiert,
- 11: wenn Sie mit dem Befehl COPY
- 12: Text in Ihre Datei kopieren.

Möchten Sie jedoch den kopierten Text innerhalb des anderen Texts angeordnet haben, dann müssen Sie in der dritten Option Zeile die Nummer der Zeile angeben, vor der Sie den kopierten Text eingefügt haben möchten.

DER ZEILEN-EDITOR EDLIN

Benutzen wir noch einmal unsere Beispieldatei, um zu zeigen, wie man Text in einen anderen Text einfügt:

```
1:      Dies ist eine Musterdatei zur
2:      Vorführung des Befehls COPY.
3:      Mit ihr wollen wir zeigen,
4:      was passiert,
5:      wenn Sie mit dem Befehl COPY
6:      Text in Ihre Datei kopieren.
7:      Mit dem Befehl COPY können Sie auch
8:      eine Reihe von Textzeilen in die
9:      Mitte eines Texts kopieren.
10:     Ende der Beispieldatei.
```

Der Befehl 3,6,10C bewirkt folgendes Ergebnis:

```
1:      Dies ist eine Musterdatei zur
2:      Vorführung des Befehls COPY.
3:      Mit ihr wollen wir zeigen,
4:      was passiert,
5:      wenn Sie mit dem Befehl COPY
6:      Text in Ihre Datei kopieren.
7:      Mit dem Befehl COPY können Sie auch
8:      eine Reihe von Textzeilen in die
9:      Mitte eines Texts kopieren.
10:     Mit ihr wollen wir zeigen,
11:     was passiert,
12:     wenn Sie mit dem Befehl COPY
13:     Text in Ihre Datei kopieren.
14:     Ende der Beispieldatei.
```

DER ZEILEN-EDITOR EDLIN

Delete

BEFEHLSNAME: (D)elete

VERWENDUNGSZWECK: Löscht die angegebene Zeilenreihe aus einer Datei.

SYNTAX: [**<Zeile>**] [, **<Zeile>**] D

WIRKUNGSWEISE: Die erste Option **<Zeile>** nimmt die erste Zeilennummer der zu löschenden Zeilenreihe auf. Die zweite Option **<Zeile>** nimmt die letzte Zeilennummer der zu löschenden Zeilenreihe auf. Wenn Sie die erste Option nicht eingeben, wird statt der einzugebenden Zeilenzahl diejenige der momentanen Bearbeitungszeile (d.h. der Zeile, in der zwischen Zeilennummer und Text ein Sternchen steht) angenommen. Wenn die zweite Option nicht eingegeben wird, wird lediglich die in der ersten angegebene oder dafür angenommene Zeile gelöscht. Nach dem Löschvorgang werden die Zeilen ab dem Anfang der gelöschten Zeilenreihe automatisch neu nummeriert, so daß die unmittelbar auf die gelöschte Zeilenreihe folgende Zeile die Nummer der ersten Zeile dieser gelöschten Zeilenreihe erhält. Die erste auf die gelöschten Zeilenreihe folgende Zeile ist nach dem Löschvorgang die momentane Bearbeitungszeile.

Beispiele:

Wir wollen die Wirkungsweise des Befehls Delete an der folgenden, bearbeitungsfähigen Beispieldatei vorführen:

- 1: Mit Hilfe dieser Beispieldatei wollen wir zeigen,
- 2: wie man mit dynamischen Zeilennummern arbeitet.
- 3: Schauen wir uns einmal an, was passiert,
- 4: wenn man die Befehle
- ⋮
- 25: Delete und Insert
- 26: (d.h. die Befehle D und I),
- 27:* zur Bearbeitung verwendet.

DER ZEILEN-EDITOR EDLIN

Nun geben wir den Befehl zum Löschen der durch die drei Auslassungspunkte dargestellten Zeilen ein:
5,24D

Das Ergebnis ist:

```
1:      Mit Hilfe dieser Beispieldatei wollen wir zeigen,  
2:      wie man mit dynamischen Zeilennummern arbeitet.  
3:      Schauen wir uns einmal an, was passiert,  
4:      wenn man die Befehle  
5:      Delete und Insert  
6:      (d.h. die Befehle D und I),  
7:*    zur Bearbeitung verwendet.
```

Wenn Sie nur eine einzelne Zeile löschen wollen, geben Sie beispielsweise ein:
6D

Das Ergebnis ist:

```
1:      Mit Hilfe dieser Beispieldatei wollen wir zeigen,  
2:      wie man mit dynamischen Zeilennummern arbeitet.  
3:*    Schauen wir uns einmal an, was passiert,  
4:      wenn man die Befehle  
5:      Delete und Insert  
6:      zur Bearbeitung verwendet.
```

Als nächstes wollen wir eine Reihe von Zeilen aus der folgenden Datei löschen:

```
1:      Mit Hilfe dieser Beispieldatei wollen wir zeigen,  
2:      wie man mit dynamischen Zeilennummern arbeitet.  
3:*    Schauen wir uns einmal an, was passiert,  
4:      wenn man die Befehle  
5:      Delete und Insert  
6:      (d.h. die Befehle D und I),  
7:      zur Bearbeitung verwendet.  
8:      Es wird ein optimales Ergebnis erzielt.
```

Um eine Reihe von Zeilen aus dieser Datei zu löschen, die mit der momentanen Bearbeitungszeile beginnt, geben wir ein:
,7D

Das Ergebnis ist:

```
1:      Mit Hilfe dieser Beispieldatei wollen wir zeigen,  
2:      wie man mit dynamischen Zeilennummern arbeitet.  
3:*    Es wird ein optimales Ergebnis erzielt.
```

Beachten Sie bitte, daß auch hier die Zeilen automatisch neu durchnummeriert wurden.

DER ZEILEN-EDITOR EDLIN

Zeile

BEFEHLSNAME: Zeile

VERWENDUNGSZWECK: Ermöglicht die Bearbeitung einer Textzeile.

SYNTAX: [**<Zeile>**]

WIRKUNGSWEISE: Wenn einfach nur eine Zeilennummer eingegeben wird, bringt EDLIN diese Zeilennummer und den dazugehörigen Textinhalt auf den Bildschirm. Des weiteren schreibt EDLIN in die darauffolgende Zeile dieselbe Zeilennummer noch einmal. Sie können die Textzeile nun bearbeiten. Dazu können Sie alle hier beschriebenen EDLIN-Bearbeitungsbefehle verwenden. Die angezeigte Textzeile steht im Zeilenspeicher und kann wie weiter oben beschrieben entsprechend verwendet werden, solange die RETURN-Taste nicht betätigt wird.

Wenn Sie diese momentane Bearbeitungszeile ohne Änderungen so übernehmen möchten, wie sie ist, brauchen Sie nur die RETURN-Taste zu betätigen. Voraussetzung dafür ist, daß die Schreibmarke am Ende oder am Anfang der Zeile steht.

HINWEIS

Wenn Sie die RETURN-Taste drücken, während die Schreibmarke in der Mitte der Zeile steht, dann wird der Teil der Zeile hinter der Schreibmarke gelöscht.

Wenn keine Zeilennummer eingegeben wird, d.h. wenn Sie einfach nur die RETURN-Taste betätigen, dann wird die Zeile nach der momentanen Bearbeitungszeile (mit dem Sternchen (*) gekennzeichnete Zeile) zur Bearbeitung auf den Bildschirm gebracht.

DER ZEILEN-EDITOR EDLIN

Beispiel:

Zur Vorführung der Wirkungsweise dieses Befehls verwenden wir die nachstehende bearbeitungsfähige Beispieldatei:

```
1:      Dies ist eine Beispieldatei,  
2:      mit der wir zeigen wollen,  
3:      wie die Zeile  
4:*     vier  
5:      bearbeitet werden kann.
```

Wenn Sie Zeile 4 bearbeiten möchten, geben Sie bitte ein:
4

Nummer und Inhalt dieser Zeile werden nun am Bildschirm angezeigt, die Zeilennummer in der nächsten Zeile noch einmal wiederholt und die Schreibmarke in diese angefangene neue Zeile gestellt:

```
4:*     vier  
4:*     _
```

Als Bearbeitung werden wir nun mit Hilfe der Sonderbearbeitungstaste INSERT und COPYALL ein Wort in diese Zeile einfügen:

```
INSERT Nummer          4:      Nummer_  
<COPYALL> RETURN      4:      Nummer_vier
```

DER ZEILEN-EDITOR EDLIN

End

BEFEHLSNAME: (E)nd

VERWENDUNGSZWECK: Beendet die Arbeitssitzung ordnungsgemäß und schaltet den Zeilen-Editor aus.

SYNTAX: E

WIRKUNGSWEISE: Dieser Befehl bewirkt, daß die bearbeitete Datei auf Diskette zurückgeschrieben, in die Dateibezeichnung der ursprünglichen Datei das Suffix .BAK eingefügt und, schließlich, der Zeilen-Editor EDLIN verlassen wird. Wurde die Datei erst während der Bearbeitungssitzung erstellt, dann wird keine .BAK-Datei (Sicherungsdatei) erstellt.

Der Befehl E kennt keine Optionen. Daher können Sie EDLIN bei Verwendung dieses Befehls auch nicht mitteilen, auf welcher Diskette in welchem Laufwerk die Datei gespeichert werden soll. Das bedeutet, daß Sie bereits zu Beginn der Arbeitssitzung festlegen müssen, auf welche Diskette in welchem Laufwerk Ihre Datei nach der Sitzung gespeichert werden soll. Haben Sie das nicht getan, dann wird die bearbeitete Datei grundsätzlich auf der Diskette im Standardlaufwerk abgespeichert. Sie können jedoch dann trotzdem jederzeit eine andere Datei mit Hilfe des MS-DOS-Befehls COPY auf eine andere Diskette in einem anderen Laufwerk übertragen.

Vergewissern Sie sich bitte, daß die Diskette, auf die gespeichert werden soll, über genügend Speicherkapazität für die komplette Datei verfügt. Ist die Speicherkapazität dort unzureichend, dann wird der Schreibvorgang abgebrochen, sobald der Speicherraum erschöpft ist, und die bearbeitete Datei geht verloren, obwohl bereits ein Teil davon auf der Diskette gespeichert ist.

7

DER ZEILEN-EDITOR EDLIN

Beispiel:

E RETURN

Nach der Ausführung des Befehls E erscheint die MS-DOS-Standardlaufwerk-Eingabeaufforderung (beispielsweise A) am Bildschirm.

Insert

BEFEHLSNAME: (I)nsert

VERWENDUNGSZWECK: Ermöglicht das Einfügen von Text unmittelbar vor der angegebenen Zeile sowie das grundsätzliche Einleiten der Betriebsart Einfügen, bevor Text in EDLIN eingegeben werden kann.

SYNTAX: [**<Zeile>**] I

WIRKUNGSWEISE: Wenn Sie mit dem Zeilen-Editor eine neue Datei erstellen wollen, müssen Sie als erstes den Befehl I eingeben, bevor Sie Text auf den Bildschirm schreiben (einfügen) können. Ihr Text beginnt grundsätzlich mit der Zeilennummer 1.

Alle weiteren Zeilennummern erscheinen automatisch jedesmal, wenn Sie die RETURN-Taste betätigen.

EDLIN bleibt in der Betriebsart Einfügen, bis Sie CONTROL-C eingeben. Wenn die Einfügung abgeschlossen ist und die Betriebsart Einfügen wieder verlassen wurde, wird die auf die eingefügten Zeilen folgende Zeile automatisch zur momentanen Bearbeitungszeile.

Alle auf die eingefügte Zeilenreihe folgenden Zeilennummern werden automatisch um die Zahl der eingefügten Zeilen erhöht.

DER ZEILEN-EDITOR EDLIN

Wenn die Option <Zeile> nicht angegeben wird, interpretiert sie der Zeilen-Editor standardmäßig als Nummer der momentanen Bearbeitungszeile, und die neuen Zeilen werden unmittelbar vor dieser eingefügt. Ist <Zeile> eine beliebige Zahl größer als die letzte Zeilennummer der Datei oder wird anstatt einer Zeilennummer das Pfundzeichen (#) als <Zeile> eingegeben, dann werden die eingefügten Zeilen am Ende der Datei angefügt. In diesem Fall wird die letzte ein- bzw. angefügte Zeile zur momentanen Bearbeitungszeile.

Beispiele:

Zur Vorführung der Wirkungsweise dieses Befehls verwenden wir die nachstehende bearbeitungsfähige Beispieldatei:

```
1:*   Mit Hilfe dieser Beispieldatei wollen wir zeigen,
2:   wie man mit dynamischen Zeilennummern arbeitet.
3:   Schauen wir uns einmal an, was passiert,
4:   wenn man die Befehle
5:   Delete und Insert
6:   (d.h. die Befehle D und I),
7:   zur Bearbeitung verwendet.
```

Wenn Sie Text vor einer bestimmten Zeile einfügen möchten, die nicht die momentane Bearbeitungszeile ist, geben Sie bitte <Zeile>I ein:

```
7I
```

Das Ergebnis ist:

```
7:_
```

Nun geben Sie den neuen Text für Zeile 7 ein:

```
7:   zur Neunumerierung der Zeilen und
```

Um die Betriebsart Einfügen nun wieder zu verlassen, geben Sie in die nächste Zeile das Steuerzeichen CONTROL-Z ein:

```
8:   CONTROL-Z
```

DER ZEILEN-EDITOR EDLIN

Jetzt lassen wir uns die Datei auflisten, indem wir den Befehl L eingeben. Das Ergebnis ist:

```
1:      Mit Hilfe dieser Beispieldatei wollen wir zeigen,
2:      wie man mit dynamischen Zeilennummern arbeitet.
3:      Schauen wir uns einmal an, was passiert,
4:      wenn man die Befehle
5:      Delete und Insert
6:      (d.h. die Befehle D und I),
7:      zur Neumerierung der Zeilen und
8:✘     zur Bearbeitung verwendet.
```

Möchten Sie Zeilen unmittelbar vor der momentanen Bearbeitungszeile einfügen, dann brauchen Sie nur einzugeben:
I

Das Ergebnis ist:

```
8:      _
```

Jetzt fügen wir den folgenden Text ein und beenden die Betriebsart Einfügen durch die Eingabe des Steuerzeichens<CONTROL-Z> in der folgenden Zeile:

```
8:      , bestimmungsgemäß,
9:      <CONTROL-Z>
```

Lassen Sie sich jetzt die Datei wieder auflisten, um das Ergebnis zu sehen, indem Sie den Befehl L eingeben:

Das Ergebnis ist:

```
1:      Mit Hilfe dieser Beispieldatei wollen wir zeigen,
2:      wie man mit dynamischen Zeilennummern arbeitet.
3:      Schauen wir uns einmal an, was passiert,
4:      wenn man die Befehle
5:      Delete und Insert
6:      (d.h. die Befehle D und I),
7:      zur Neumerierung der Zeilen und
8:✘     , bestimmungsgemäß ,
9:      zur Bearbeitung verwendet.
```

Wenn Sie neue Zeilen am Ende Ihrer Datei anfügen möchten, geben Sie bitte ein:
10I

Damit erzielen Sie folgendes Ergebnis:

```
10:    _
```

DER ZEILEN-EDITOR EDLIN

Jetzt können Sie Ihre neuen Zeilen eingeben:

- 10: Der EDLIN-Befehl Insert ermöglicht
- 11: es Ihnen auch, eine Reihe von Zeilen, wie
- 12: beispielsweise die hier geschriebenen, an das
- 13: Ende einer Datei anzufügen.

Beenden Sie nun die Betriebsart Einfügen, indem Sie das Steuerzeichen <CONTROL-Z> in Zeile 14 eingeben. Damit werden die Zeilen, die Sie soeben neu eingegeben haben, an das Ende der in der Datei bereits vorliegenden Zeilen angefügt. Lassen wir uns das Ergebnis anzeigen; geben Sie bitte den Befehl L zum Auflisten bzw. Anzeigen der Datei ein:

L

Das Ergebnis ist:

- 1: Mit Hilfe dieser Beispieldatei wollen wir zeigen,
- 2: wie man mit dynamischen Zeilennummern arbeitet.
- 3: Schauen wir uns einmal an, was passiert,
- 4: wenn man die Befehle
- 5: Delete und Insert
- 6: (d.h. die Befehle D und I);
- 7: zur Neunummerierung der Zeilen und
- 8: , bestimmungsgemäß,
- 9: zur Bearbeitung verwendet.
- 10: Der EDLIN-Befehl Insert ermöglicht
- 11: es Ihnen auch, eine Reihe von Zeilen, wie
- 12: beispielsweise die hier geschriebenen, an das
- 13:✘ Ende einer Datei anzufügen.

DER ZEILEN-EDITOR EDLIN

List

BEFEHLSNAME: (L)ist

VERWENDUNGSZWECK: Dieser Befehl bewirkt, daß eine Reihe von Zeilen angezeigt wird, einschließlich der beiden Zeilen mit den im Befehl selbst angegebenen Nummern.

SYNTAX: [`<Zeile>`] [,`<Zeile>`] L

WIRKUNGSWEISE: Es wird die angegebene Zeilenreihe einschließlich der angegebenen Begrenzungszeichen angezeigt. Wenn die Optionen nicht eingegeben werden, geht der Zeilen-Editor von Standardwerten aus. Wenn die erste Option fehlt, wie beispielsweise:

,`<Zeile>`L

beginnt der Bereich der gezeigten Zeilen 11 Zeilen vor der momentanen Bearbeitungszeile und endet mit der angegebenen `<Zeile>`. Das führende Komma wird benötigt, um anzuzeigen, daß die erste Option nicht eingegeben wurde.

HINWEIS:

Wenn die hier angegebene `<Zeile>` mehr als 11 Zeilen vor der momentanen Bearbeitungszeile liegt, erfolgt die Anzeige so, als ob Sie den Befehl ohne Optionen eingegeben hätten.

Fehlt die zweite Option, wie beispielsweise in:
`<Zeile>` L

dann werden 23 Zeilen angezeigt, beginnend mit der angegebenen `<Zeile>`.

DER ZEILEN-EDITOR EDLIN

Wenn beide Optionen nicht eingegeben werden, wie beispielsweise:
L

dann werden ebenfalls 23 Zeilen angezeigt, und zwar die 11 Zeilen vor der momentanen Bearbeitungszeile, die momentane Bearbeitungszeile und die 11 Zeilen nach der momentanen Bearbeitungszeile. Wenn die Datei keine 11 Zeilen vor der momentanen Bearbeitungszeile enthält, dann werden nach dieser entsprechend mehr als 11 Zeilen angezeigt, so daß die gesamte Länge des angezeigten Texts 23 Zeilen beträgt.

Beispiel:

Zur Vorführung der Wirkungsweise dieses Befehls verwenden wir die nachstehende bearbeitungsfähige Beispieldatei:

```

1:      Mit Hilfe dieser Beispieldatei wollen wir zeigen,
2:      wie man mit dynamischen Zeilennummern arbeitet.
3:      Schauen wir uns einmal an, was passiert,
4:      wenn man die Befehle
5:      Delete und Insert
6:      (d.h. die Befehle D und I),
.
.
.
15:*   Die momentane Bearbeitungszeile ist durch ein
16:   Sternchen gekennzeichnet
.
.
.
26:   zur Bearbeitung verwendet.
27:   Außerdem kann man ...

```

Wollen Sie eine Reihe von Zeilen zur Anzeige bringen, ohne die momentane Bearbeitungszeile anzusprechen, dann geben Sie <Zeile> , <Zeile> L ein
2,6L

Das Ergebnis ist:

```

2:      wie man mit dynamischen Zeilennummern arbeitet.
3:      Schauen wir uns einmal an, was passiert,
4:      wenn man die Befehle
5:      Delete und Insert
6:      (d.h. die Befehle D und I),

```



Weitergabe sowie Vervielfältigung dieses Unterlages, Vervielfältigung und Mitteilung
 ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden. Zuwider-
 handlungen verpflichten zu Schadensersatz. Alle Rechte für den Fall der Patent-
 erteilung oder Gebrauchsmustererteilung vorbehalten.

DER ZEILEN-EDITOR EDLIN

Möchten Sie eine Reihe von Zeilen zur Anzeige bringen, die mit der momentanen Bearbeitungszeile beginnt, dann geben Sie , <Zeile> L ein:
,26L

Das Ergebnis ist:

15:* Die momentane Bearbeitungszeile ist durch ein
16: Sternchen gekennzeichnet.
.
.
26: zur Bearbeitung verwendet.

Wollen Sie eine Reihe von 23 Zeilen zur Anzeige bringen, in deren Mitte die momentane Bearbeitungszeile steht, geben Sie nur L ein:
L

Das Ergebnis ist:

4: wenn man die Befehle
5: Delete und Insert
6: (d.h. die Befehle D und I),
.
.
15:* Die momentane Bearbeitungszeile wird durch ein
16: Sternchen gekennzeichnet.
.
.
26: zur Bearbeitung verwendet.

DER ZEILEN-EDITOR EDLIN

Move

BEFEHLSNAME: (M)ove

VERWENDUNGSZWECK: Bewirkt, daß eine Reihe von Zeilen in die angegebene Zeile (und die folgenden) umgestellt wird.

SYNTAX: [**<Zeile>**] , [**<Zeile>**] , **<Zeile> M**

WIRKUNGSWEISE: Der Befehl Move gibt Ihnen die Möglichkeit, einen Textabschnitt, der mit der ersten angegebenen **<Zeile>** beginnt und mit der zweiten angegebenen **<Zeile>** endet, an eine andere Stelle in der Datei umzustellen. Nach der Umstellung werden die umgestellten Zeilen entsprechend der Richtung der Umstellung neu durchnummeriert.

Beispiel:

1: Dies ist eine Beispieldatei,
2: mit der wir die Wirkungsweise
3: des Befehls Move zeigen wollen.
4: Er erspart es Ihnen, umfangreiche
5: Textpassagen
6: zwecks Umstellung neu
7: schreiben zu müssen.
8:* Move ist ein sehr
9: leistungsfähiger Befehl, der
10: mit drei Optionen arbeitet.

Um den letzten Satz umzustellen, wo er logisch hingehört,
schreiben wir:
8,10,4M

7

DER ZEILEN-EDITOR EDLIN

Dieser Befehl bewirkt, daß die Zeilen 8 bis 10 in die Zeile 4 und die folgenden umgestellt werden; unser Beispiel sieht dann wie folgt aus:

```
1:      Dies ist eine Beispieldatei,  
2:      mit der wir die Wirkungsweise  
3:      des Befehls Move zeigen wollen.  
4:      Move ist ein sehr  
5:      leistungsfähiger Befehl, der  
6:      mit drei Optionen arbeitet.  
7:      Er erspart es Ihnen, umfangreiche  
8:*     Textpassagen  
9:      zwecks Umstellung neu  
10:     schreiben zu müssen.
```

Anstatt die Nummer der Zeile einzugeben, mit der der umzustellende Textabschnitt endet, können Sie sich auch auf die in der ersten Option Zeile angegebene Zeile beziehen und EDLIN anweisen, eine bestimmte Anzahl von Zeilen ab dieser umzustellen, indem Sie die zweite Option Zeile mit einem Pluszeichen vor der Zahl der umzustellenden Zeilen eingeben. Wie bei dem Befehl List können Sie sich auch hier auf die momentane Bearbeitungszeile beziehen, indem Sie die erste Option weglassen.

Um dem Zeilen-Editor mitzuteilen, daß Sie ab der momentanen Bearbeitungszeile 3 Zeilen in Zeile 4 und die folgenden umstellen möchten, schreiben Sie daher:

```
,+3,4M
```

Damit wird der Textabschnitt beginnend mit der momentanen Bearbeitungszeile zuzüglich weiterer 3 Zeilen in die Zeile 4 und die folgenden umgestellt. Das Ergebnis ist dasselbe wie weiter oben beschrieben.

Auch hier ist es nicht möglich, eine Reihe von Zeilen in eine Zeile umzustellen, die in dieser Reihe enthalten ist. Wenn Sie dies versuchen, reagiert der Zeilen-Editor, wie schon einmal gesehen, mit der Meldung: "Entry error".

DER ZEILEN-EDITOR EDLIN

Page

BEFEHLSNAME: (P)age

VERWENDUNGSZWECK: Der Befehl Page ermöglicht es Ihnen, bildschirmseitenweise, d.h. in Abschnitten von je 23 Zeilen, durch eine Datei zu "blättern".

SYNTAX: [**<Zeile>**] [**, <Zeile>**] P

WIRKUNGSWEISE: Wenn die erste **<Zeile>** nicht eingegeben wird, nimmt der Zeilen-Editor als Standardwert die Nummer der momentanen Bearbeitungszeile plus eins.

Die letzte angezeigte Zeile der Datei wird zur momentanen Bearbeitungszeile und entsprechend mit einem Sternchen gekennzeichnet.

DER ZEILEN-EDITOR EDLIN

Quit

BEFEHLSNAME: (Q)uit

VERWENDUNGSZWECK: Mit diesem Befehl brechen Sie die Bearbeitungssitzung ab, im Gegensatz zu dem Befehl End, ohne die bearbeitete Datei zu speichern, und verlassen den Zeilen-Editor, um in das Betriebssystem MS-DOS zurückzukehren.

SYNTAX: Q

WIRKUNGSWEISE: Wenn Sie den Befehl eingeben, fordert EDLIN Sie auf zu prüfen, ob Sie sicher sind, daß Sie die bearbeitete Datei nicht speichern wollen.

Wenn Sie sich sicher sind, daß Sie die Bearbeitungssitzung abbrechen wollen, ohne die bearbeitete Datei zu speichern, geben Sie Y ein. Die Bearbeitungssitzung wird dann beendet; es wird keine bearbeitete Datei gespeichert und keine Sicherungsdatei mit dem Suffix .BAK erstellt. Nähere Einzelheiten bezüglich der .BAK-Datei finden Sie in der Beschreibung des Befehls End in diesem Kapitel.

Wenn Sie es sich überlegt haben und die Bearbeitungssitzung lieber fortsetzen möchten, geben Sie N oder jedes beliebige andere Zeichen mit Ausnahme von Y ein.

HINWEIS

Wenn der Zeilen-Editor gestartet wird, löscht er grundsätzlich eine etwa vorhandene Kopie einer Sicherungsdatei mit dem Suffix Datei freizustellen. Wenn Sie die Frage "Abort edit (Y/N)?" mit Y beantworten, wird Ihre frühere Sicherungskopie gelöscht.

Beispiel:

Q

Abort edit (Y/N)?

Y <RETURN>

A>

DER ZEILEN-EDITOR EDLIN

Replace

BEFEHLSNAME: (R)eplace

VERWENDUNGSZWECK: Dieser Befehl bewirkt, daß in einem angegebenen Textabschnitt, d.h. in einer Reihe von Zeilen, die angegebene Zeichenfolge überall da, wo sie gefunden wird, durch eine andere Zeichenfolge oder eine entsprechende Anzahl von Leerzeichen ersetzt wird.

SYNTAX:

[<Zeile>] [, <Zeile>] [?] R<Zeichenfolge1>
<CONTROL-Z> <Zeichenfolge2>

WIRKUNGSWEISE:

Sobald die <Zeichenfolge1> im Text gefunden wird, wird sie durch die <Zeichenfolge2> ersetzt. Jede Zeile, in der ersetzt wird, wird angezeigt. Wird in einer Zeile mehrmals ersetzt, dann wird die Zeile ebenso oft angezeigt.

Wenn alle <Zeichenfolge1> in der angegebenen Zeilenreihe durch <Zeichenfolge2> ersetzt sind, wird der Befehl R beendet, und das Sternchen als Eingabeaufforderung erscheint wieder am Bildschirm.

Wenn die zweite Zeichenfolgenoption eingegeben werden soll, dann muß die <Zeichenfolge1> von der <Zeichenfolge2> durch das Steuerzeichen CONTROL-Z abgegrenzt werden.

Die <Zeichenfolge2> muß auch mit der Kombination <CONTROL-Z>+ <RETURN> oder mit einem einfachen <RETURN> abgeschlossen werden.

Wird die <Zeichenfolge1> weggelassen, dann wird statt ihrer die zuletzt eingegebene <Zeichenfolge1> ersetzt. Wenn eine <Zeichenfolge1> überhaupt noch nicht eingegeben wurde, d.h. wenn der Befehl R zum ersten Mal angewendet wird, dann wird das Ersetzen in diesem Fall sofort beendet. Wenn <Zeichenfolge2> weggelassen wird, dann kann <Zeichenfolge1> durch Betätigung der RETURN-Taste abgeschlossen werden.

7

DER ZEILEN-EDITOR EDLIN

Wird die erste <Zeile> in der Bereichsangabe weggelassen (wie zum Beispiel wenn Sie schreiben , <Zeile>), dann wird der Zeilen-Editor als erste <Zeile> die Zeile unmittelbar nach der momentanen Bearbeitungszeile nehmen. Wird die zweite <Zeile> weggelassen (beispielsweise wenn Sie schreiben <Zeile> oder <Zeile>,), dann nimmt der Zeilen-Editor als zweite <Zeile> standardmäßig das Zeichen #. Wir erinnern daran, daß # für die Nummer der Zeile nach der letzten Zeile der Datei steht.

Wenn <Zeichenfolge1> mit einem Steuerzeichen <CONTROL-Z> abgeschlossen wurde und keine <Zeichenfolge2> eingegeben wird, dann wird für <Zeichenfolge2> eine "leere Zeichenfolge" angenommen und die <Zeichenfolge1> dadurch ersetzt.

Zum Beispiel:

R <Zeichenfolge1> <CONTROL-Z> <RETURN>
bewirkt, daß die Zeichenfolge1 gelöscht wird, wenn sie irgendwo gefunden wird, aber

R <Zeichenfolge1> <RETURN> bzw.
R <RETURN>

bewirkt, daß Zeichenfolge1 durch die letzte zuvor eingegebene <Zeichenfolge2> ersetzt wird bzw. die zuletzt eingegebene <Zeichenfolge2> . Mit "zuletzt eingegeben" ist hier eine Zeichenfolge gemeint, die bei der letzten früheren Verwendung der Befehle Replace oder Search eingegeben wurde.

DER ZEILEN-EDITOR EDLIN

Wenn Sie die Fragezeichenoption (?) eingegeben haben, wird der Zeilen-Editor in jeder Zeile, in der eine Zeichenfolge gefunden wird, die mit <Zeichenfolge1> übereinstimmt, stehenbleiben, die Zeile am Bildschirm probeweise mit der durch <Zeichenfolge2> ersetzten <Zeichenfolge1> anzeigen und anschließend die Eingabeaufforderung "O.K.?" ausgeben. Wenn Sie die Frage mit Y beantworten oder die RETURN-Taste betätigen, dann wird <Zeichenfolge1> endgültig durch Zeichenfolge2 ersetzt und die nächste <Zeichenfolge1> gesucht. Sobald sie gefunden ist, wird erneut die Eingabeaufforderung "O.K.?" ausgegeben. Der Ablauf wird mit dieser Arbeitsweise fortgesetzt, bis das Ende des angegebenen Textabschnitts oder das Ende der Datei erreicht ist. Nach der Bearbeitung der letzten gefundenen <Zeichenfolge1> zeigt EDLIN wieder das Sternchen als Eingabeaufforderung.

Beantworten Sie die Eingabeaufforderung "O.K.?" durch Betätigung einer anderen Taste als Y oder der RETURN-Taste, dann wird die Zeile so belassen, wie sie war, d.h. mit der <Zeichenfolge1> und Replace wird die nächste Stelle im Text suchen, wo die <Zeichenfolge1> vorkommt. Auch wenn die <Zeichenfolge1> mehr als einmal in ein und derselben Zeile vorkommt, wird sie jedesmal ersetzt, und es wird jedesmal die Eingabeaufforderung "O.K.?" ausgegeben.

Diese Arbeitsweise ermöglicht es Ihnen, die <Zeichenfolge1> nur dann durch den Zeilen-Editor ersetzen zu lassen, wenn es dem Kontext entsprechend sinnvoll erscheint, und so unerwünschte Ersetzungen zu vermeiden.

Beispiele:

Zur Vorführung der Wirkungsweise dieses Befehls verwenden wir die nachstehende bearbeitungsfähige Beispieldatei:

- 1: Mit Hilfe dieser Beispieldatei wollen wir zeigen,
- 2: wie man mit dynamischen Zeilennummern arbeitet.
- 3: Schauen wir uns einmal an, was passiert,
- 4: wenn die Befehle
- 5: Delete und Insert
- 6: (d.h. die Befehle D und I)
- 7: miteinander verbunden werden.
- 8: Der EDLIN-Befehl Insert ermöglicht
- 9: es Ihnen auch, eine Reihe von Zeilen wie
- 10: beispielsweise die hier geschriebenen an das
- 11: Ende einer Datei anzufügen.



Weitergabe sowie Vervielfältigung dieser Unterlagen, Verwertung und Mitteilung der Inhalte ist ohne schriftliche Genehmigung des Nixdorf Computer AG. Alle Rechte vorbehalten. Nachdruck, Vervielfältigung und Verbreitung, auch auszugsweise, ist ohne schriftliche Genehmigung der Nixdorf Computer AG.

DER ZEILEN-EDITOR EDLIN

Probieren wir als erstes den Standardvorgang, bei dem <Zeichenfolge1> durch <Zeichenfolge2> in einem bestimmten Textbereich bzw. in der angegebenen Zeilenreihe ersetzt wird; geben Sie bitte ein:

3,7 Rund <CONTROL-Z> oder <RETURN>

Das Ergebnis ist:

5: Delete oder Insert
6: (d.h. die Befehle D oder I)
7: miteinander verbodern werden.

Beachten Sie bitte, daß in unserem Beispiel auch eine unerwünschte Ersetzung ("verbunden" durch "verbodern") stattgefunden hat. Um das zu vermeiden und zu erreichen, daß Sie bei jeder Ersetzung zuerst gefragt werden, ob ersetzt werden soll, müssen Sie den Befehl etwas anders eingeben.

Um sicher zu sein, daß nur dann ersetzt wird, wenn es sinnvoll ist, geben Sie Ihren Befehl wie folgt ein:

3,7?Rund <CONTROL-Z> oder <RETURN>

Das Ergebnis ist:

5: Delete oder Insert
0.K.? Y
6: (d.h. die Befehle D oder I)
0.K.? Y
7: miteinander verbodern werden.
0.K.? N
*
_

Zum Abschluß geben Sie noch den Befehl List (L) ein, um die durchgeführten Änderungen in unserer Beispieldatei zu prüfen.

5: Delete oder Insert
6: (d.H. die Befehle D oder I)
8: Der EDLIN-Befehl Insert ermöglicht

DER ZEILEN-EDITOR EDLIN

Search

BEFEHLSNAME: (S)earch

VERWENDUNGSZWECK: Bewirkt, daß der angegebene Textabschnitt nach der angegebenen Zeichenfolge abgesucht wird.

SYNTAX:

[<Zeile>] [, <Zeile>] [?] S<Zeichenfolge> <RETURN>

WIRKUNGSWEISE: Die <Zeichenfolge> muß grundsätzlich mit <RETURN> abgeschlossen werden. Die erste Zeile, in der eine Zeichenfolge gefunden wird, die mit Zeichenfolge übereinstimmt, wird angezeigt und wird zur momentanen Bearbeitungszeile. Ist die Fragezeichenoption nicht angegeben, dann wird der Befehl Search beendet, sobald die erste übereinstimmende Zeichenfolge gefunden ist. Wird in dem ganzen angegebenen Textabschnitt keine mit <Zeichenfolge> übereinstimmende Zeichenfolge gefunden, dann wird die Meldung "Not found" angezeigt.

Wurde die Fragezeichenoption (?) bei der Eingabe des Befehls angegeben, dann wird der Zeilen-Editor zunächst einmal die erste Zeichenfolge suchen, die mit <Zeichenfolge> übereinstimmt, und die Zeile, in der diese Zeichenfolge enthalten ist, am Bildschirm anzeigen; danach erfolgt die Eingabeaufforderung "O.K.?". Beantworten Sie diese mit entweder Y oder durch Betätigung der RETURN-Taste, dann wird die angezeigte Zeile zur momentanen Bearbeitungszeile, und die Suche wird beendet. Wenn Sie eine beliebige andere Taste betätigen, sucht der Zeilen-Editor weiter, bis die nächste Zeichenfolge gefunden wird, die mit <Zeichenfolge> übereinstimmt, oder bis der gesamte angegebene und abzusuchende Textabschnitt abgearbeitet ist (und die Meldung "Not found" angezeigt wird).

DER ZEILEN-EDITOR EDLIN

Wird die erste Zeile nicht eingegeben (beispielsweise bei einer Eingabe wie , <Zeile> S<Zeichenfolge>), dann nimmt EDLIN als erste Zeile des abzusuchenden Textabschnitts die Zeile unmittelbar nach der momentanen Bearbeitungszeile. Fehlt die zweite Option Zeile (wie bei der Eingabe <Zeile> S <Zeichenfolge> oder <Zeile> S <Zeichenfolge>), dann nimmt der Zeilen-Editor dafür die Option # (d.h. die Zeile nach der letzten Zeile Ihrer Datei), was dieselbe Wirkung hat wie <Zeile> ,# S Zeichenfolge .

Wird die Option <Zeichenfolge> nicht eingegeben, dann nimmt der Zeilen-Editor die zuletzt eingegebene Zeichenfolge, falls eine eingegeben wurde. ("Zuletzt eingegeben" heißt, bei der letzten Benutzung der Befehle Search oder Replace eingegeben.)

Liegt keine zuletzt eingegebene Zeichenfolge vor (beispielsweise, wenn in der gegenwärtigen Arbeitssitzung bisher weder der Befehl Search noch der Befehl Replace benutzt wurde), dann wird der Befehl sofort beendet.

Beispiele:

Zur Vorführung der Wirkungsweise dieses Befehls verwenden wir die nachstehende bearbeitungsfähige Beispieldatei:

```
1: Mit Hilfe dieser Beispieldatei wollen wir zeigen,
2: wie man mit dynamischen Zeilennummern arbeitet.
3: Schauen wir uns einmal an, was passiert,
4: wenn man die Befehle
5: Delete und Insert
6: (d.h. die Befehle D und I),
7: zur Bearbeitung verwendet.
8: Der EDLIN-Befehl Insert ermöglicht
9: es Ihnen auch, eine Reihe von Zeilen wie
10: beispielsweise die hier geschriebenen an das
11:* Ende einer Datei anzufügen.
```


DER ZEILEN-EDITOR EDLIN

Um die erste Stelle in dem Textabschnitt von Zeile 3 bis 7 zu suchen, an der beispielsweise die Zeichenfolge "und" steht, geben Sie bitte ein:

3,7 Sund <RETURN>

Sofort wird die folgende Zeile angezeigt:

5:* Delete und Insert

Möchten Sie das "und" in Zeile 6, dann müssen Sie den Befehl in der folgenden Form eingeben.

<SKIP1> <COPYALL>,12 Sund <RETURN>

Nun wird die Suche in der ersten Zeile nach der momentanen Bearbeitungszeile (also Zeile 6) aufgenommen, da die erste Option nicht eingegeben wurde.

Das Ergebnis ist:

6:* (d.h. die Befehle D und I),

Möchten Sie den ganzen Text nach allen Zeichenfolgen absuchen lassen, die mit der angegebenen übereinstimmen, dann müssen Sie eingeben:

1, ? Sund

Das Ergebnis ist:

5:* Delete und Insert
O.K.?_

Wenn Sie nun eine beliebige Taste (mit Ausnahme von Y und der RETURN-Taste) betätigen, wird die Suche fortgesetzt. Geben wir also beispielsweise N hier ein:

O.K.? N

Nun wird weiter gesucht:

6:* (d.h. die Befehle D und I)
O.K.?_

DER ZEILEN-EDITOR EDLIN

Nun wollen wir die Suche beenden. Wir nehmen an, daß diese gefundene Zeichenfolge die gesuchte ist und geben Y ein:

O.K.? Y

*

—

Wenn Sie die Zeichenfolge XYZ suchen möchten, ohne daß Sie jedesmal vom Zeilen-Editor zu einer Bestätigung aufgefordert werden ("O.K.?"), geben Sie einfach ein:

SXYZ

EDLIN wird Ihnen in diesem Fall jede Zeichenfolge anzeigen, die mit der gesuchten übereinstimmt, und wird weiterhin nach derselben Zeichenfolge suchen, wenn Sie den Befehl S erneut eingeben:

S

Es erfolgt die Anzeige der nächsten gefundenen Zeichenfolge, die mit der vorgegebenen übereinstimmt.

S

EDLIN sucht weiter, bis die abzusuchende Datei abgearbeitet ist, was Ihnen der Zeilen-Editor mit der Meldung "Not found" mitteilt.

Beachten Sie bitte, daß die Option <Zeichenfolge> als Standardwert durch die zuletzt, d.h. bei der letzten Benutzung des Befehls Search eingegebene, ersetzt wird, wenn Sie keine eingeben.

DER ZEILEN-EDITOR EDLIN

Transfer

BEFEHLSNAME: (T)ransfer

VERWENDUNGSZWECK : Dieser Befehl ermöglicht es Ihnen, den Inhalt der Datei Dateiname mit der in der Zeile Zeile momentan in Bearbeitung befindlichen Datei zusammenzuführen. Wird die Option Zeile nicht angegeben, dann nimmt der Zeilen-Editor statt dessen die momentane Bearbeitungszeile an.

SYNTAX: [**<Zeile>**] T **<Dateiname>**

WIRKUNGSWEISE: Dieser Befehl ist sehr nützlich, wenn Sie den Inhalt einer Datei in eine andere Datei oder in den Text, den Sie gerade erstellen, einfügen möchten. Der solchermaßen übertragene Text wird in der mit Zeile angegebenen Zeile und den folgenden eingefügt, und alle darauffolgenden Zeilen werden neu durchnummeriert.

DER ZEILEN-EDITOR EDLIN

Write

BEFEHLSNAME: (W)rite

VERWENDUNGSZWECK : Dieser Befehl ermöglicht es Ihnen, die angegebene Anzahl von Zeilen aus dem Text, den Sie zur Zeit im Arbeitsspeicher gerade bearbeiten, auf eine Diskette zu speichern. Der auf Diskette geschriebene Textabschnitt beginnt grundsätzlich mit der Zeile 1 Ihres Texts.

SYNTAX: [**<n>**] W

WIRKUNGSWEISE: Dieser Befehl ist im wesentlichen dazu bestimmt, Ihnen zu helfen, wenn die Datei, die Sie bearbeiten, so umfangreich ist, daß sie nicht in den Arbeitsspeicher paßt. Wenn Sie EDLIN starten, liest der Zeilen-Editor, wie Sie bereits wissen, so viele Zeilen in den Arbeitsspeicher ein, bis dieser zu 3/4 voll ist.

Wenn Sie den eingelesenen Text bearbeitet haben und nun auch den Abschnitt Ihrer Datei bearbeiten möchten, der noch auf der Diskette steht, müssen Sie als erstes einen entsprechenden Teil des bearbeiteten Texts aus dem Arbeitsspeicher wieder auf eine Diskette auslagern, um dort Platz zu machen. Erst dann können Sie einen weiteren Textabschnitt von der Diskette in den Arbeitsspeicher übernehmen, um ihn dort zu bearbeiten, was, wie Sie wissen, mit dem Befehl Append gemacht wird. Dieses Auslagern des bearbeiteten Teils Ihres Texts aus dem Arbeitsspeicher auf eine Diskette führen Sie mit Hilfe des Befehls Write durch. Die Option [n] ist für eine Zahl, und zwar die Zahl der Zeilen in Ihrem bearbeiteten Text, gezählt ab der Zeile 1, die Sie aus dem Arbeitsspeicher auslagern und auf Diskette speichern wollen.

DER ZEILEN-EDITOR EDLIN

HINWEIS

Wenn Sie die Zahl der in den Arbeitsspeicher zu übertragenden Zeilen nicht angeben, dann überträgt EDLIN gerade so viele Zeilen, daß der Arbeitsspeicher insgesamt zu 3/4 voll wird. Ist der Arbeitsspeicher bereits zu 3/4 voll, dann werden überhaupt keine Zeilen übertragen. Alle Zeilen werden neu nummeriert, so daß die erste Zeile des im Speicher verbliebenen Textabschnitts zur neuen Zeile 1 wird.

7.6 FEHLERMELDUNGEN IN EDLIN

Sobald EDLIN einen Fehler erkennt, wird die entsprechende der folgenden Fehlermeldungen ausgegeben bzw. am Bildschirm angezeigt:

Cannot edit .BAK file--rename file

Ursache:

Sie haben versucht, eine Datei mit dem Dateinamensuffix .BAK zu bearbeiten. .BAK-Dateien werden durch ihr Suffix als Sicherungsdateien gekennzeichnet, die grundsätzlich nicht bearbeitet werden können.

Abhilfe:

Wenn Sie die .BAK-Datei trotzdem bearbeiten wollen, weil Sie aus bestimmten Gründen auf die Sicherungsdatei zurückgreifen müssen, dann müssen Sie die Sicherungsdatei zuerst mit dem Befehl Rename umbenennen, indem Sie ihr ein anderes Dateinamensuffix zuweisen. Eine andere Möglichkeit wäre, die .BAK-Datei mit dem Befehl Copy zu kopieren und ihr dann einen Dateinamen mit einem anderen Suffix zu geben.

DER ZEILEN-EDITOR EDLIN

No room in directory for file

Ursache:

Sie haben versucht, eine Datei zu erstellen; entweder war der Arbeitsbereich auf der Diskette voll, oder Sie haben eine unzulässige Laufwerksangabe verwendet oder einen unzulässigen Dateinamen eingegeben.

Abhilfe:

Prüfen Sie die Befehlszeile, mit der Sie EDLIN gestartet haben, und stellen Sie insbesondere fest, ob sie einen unzulässigen Dateinamen oder eine unzulässige Laufwerksangabe enthält. Wenn die Befehlszeile nicht mehr auf Ihrem Bildschirm steht und Sie auch noch keine neue Befehlszeile eingegeben haben, können Sie die Befehlszeile, mit der Sie EDLIN gestartet haben, durch Betätigung der Taste COPYALL zurückholen.

Enthält diese Befehlszeile keine unzulässigen Eingaben, dann müssen Sie das CHKDSK-Programm auf dem betreffenden Laufwerk ausführen. Wenn der Prüfbericht Ihnen mitgeteilt, daß das Inhaltsverzeichnis voll ist, nehmen Sie die Diskette heraus. Legen Sie eine neue Diskette ein und formatieren sie.

Entry error

Ursache:

In der zuletzt eingegebenen Befehlszeile liegt ein Syntaxfehler vor.

Abhilfe:

Geben Sie den Befehl erneut ein und achten Sie dabei auf die korrekte Syntax. Schließen Sie mit <RETURN> ab.

Line too long

Ursache:

Bei der Benutzung des Befehls Replace haben Sie als Ersatz eine Zeichenfolge eingegeben, die so lang ist, daß die Zeilenlänge von 253 Zeichen überschritten wurde. Der Zeilen-Editor hat daher die Ausführung des Befehls Replace abgebrochen.

Abhilfe:

Unterteilen Sie die zu lange Zeile in zwei Zeilen und versuchen Sie das Ersetzen mit dem Befehl Replace in zwei Arbeitsgängen.

DER ZEILEN-EDITOR EDLIN

Disk full--file write not completed

Ursache:

Sie haben den Befehl End eingegeben, aber auf der Diskette/Platte war nicht mehr genug Platz für die ganze Datei. EDLIN hat daher die Ausführung des Befehls E abgebrochen und Sie zum Betriebssystem zurückverwiesen. Möglicherweise wurde jedoch ein Teil Ihrer Datei auf der Diskette/Platte gespeichert.

Abhilfe:

Es wurde nur ein Teil Ihrer Datei (wenn überhaupt) gespeichert. Es empfiehlt sich, diesen gespeicherten Teil der Datei zu löschen und die Bearbeitungssitzung neu zu starten. Die Datei wird nach diesem Fehler nicht verfügbar sein. Vergewissern Sie sich immer, daß auf Ihrer Diskette/Platte genügend Platz für das Abspeichern Ihrer Datei ist, bevor Sie Ihre Bearbeitungssitzung beginnen.

Incorrect DOS version

Ursache:

Sie haben versucht, EDLIN in einer anderen bzw. älteren Fassung von MS-DOS zu starten als 2.0.

Abhilfe:

Vergewissern Sie sich, daß die MS-DOS-Fassung, die Sie verwenden, die Fassung 2.0 oder eine neuere ist.

Invalid drive name or file

Ursache:

Sie haben beim Starten des Zeilen-Editors eine ungültige Laufwerksangabe oder einen ungültigen Dateinamen eingegeben.

Abhilfe:

Geben Sie die richtige Laufwerksangabe bzw. den richtigen Dateinamen ein.

Filename must be specified

Ursache:

Sie haben, als Sie EDLIN starteten, keinen Dateinamen eingegeben.

Abhilfe:

Geben Sie den Dateinamen ein.

DER ZEILEN-EDITOR EDLIN

Invalid parameter

Ursache:

Sie haben, als Sie EDLIN starteten, einen Befehlszusatz eingegeben, der nicht /B war.

Abhilfe:

Geben Sie beim Start des Zeilen-Editors den Befehlszusatz /B ein.

Insufficient memory

Ursache:

Ihre Speicherkapazität reicht für die Ausführung des Zeilen-Editors nicht aus.

Abhilfe:

Stellen Sie Speicherkapazität frei, indem Sie Teile des Arbeitsspeichers auf die Diskette schreiben.

File not found

Ursache:

Der bei der Eingabe des Befehls Transfer angegebene Dateiname ist nicht vorhanden.

Abhilfe:

Geben Sie bei der Eingabe des Befehls Transfer einen gültigen Dateinamen ein.

Must specify destination number

Ursache:

Sie haben bei der Eingabe des Befehls Copy oder Move keine Zielzeilennummer angegeben.

Abhilfe:

Geben Sie den Befehl erneut mit der entsprechenden Zielzeilennummer ein.

DER ZEILEN-EDITOR EDLIN

Not enough room to merge the entire file

Ursache:

Es fehlt ausreichende Speicherkapazität bei der Ausführung des Befehls Transfer.

Abhilfe:

Stellen Sie Speicherkapazität frei, indem sie einen Teil der bearbeiteten Datei mittels des W-Befehls auf die Diskette/Platte schreiben.

File creation error

Ursache:

Die vorläufige EDLIN-Datei kann nicht erstellt werden.

Abhilfe:

Vergewissern Sie sich, daß Sie in Ihrem Arbeitsbereich genügend Platz für die vorläufige Datei haben, und prüfen Sie, ob Sie Ihrer Datei nicht zufällig denselben Namen gegeben haben, wie einem Unter-Arbeitsbereich des Arbeitsbereichs, in dem sich die zu bearbeitende Datei befindet.



DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8 DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8.1	Einführung.....	8 - 2
8.1.1	Einschränkungen beim Vergleich von Binärdateien.....	8 - 3
8.2	Syntax und Datenbezeichnungen.....	8 - 3
8.3	Wie man das Dienstprogramm FC aufruft.....	8 - 4
8.4	Die FC-Befehlszusätze.....	8 - 4
8.5	Wie erkannte Unterschiede angezeigt werden.....	8 - 7
8.6	Neubestimmung des Ausgabeziels für FC.....	8 - 8
8.7	Beispiele.....	8 - 9
8.8	Fehlermeldungen in FC.....	8 - 13

Wesentliche Teile dieses Dokuments sind urheberrechtlich geschützt durch Nixdorf Computer AG. Alle Rechte für den Fall der Patentverletzung oder Gebrauchsmusterinfraktion vorbehalten.



DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8.1 EINFÜHRUNG

Manchmal kann es nützlich sein, zwei Dateien mit Hilfe des Computers vergleichen zu können. Wenn Sie beispielsweise von einer Datei mehrere Kopien angefertigt haben und diese Kopien verschiedentlich bearbeitet wurden, kann es sich als sinnvoll erweisen, sie nach der Bearbeitung zu vergleichen, um festzustellen, welche der Kopien sich auf dem neuesten Stand befindet. Genau dies ermöglicht Ihnen das MS-DOS-Dienstprogramm FC für den Dateienvergleich (FC = File Comparison = Dateienvergleich).

Das Dienstprogramm FC vergleicht den Inhalt zweier Dateien und zeigt alle Unterschiede an, die es vorfindet. Diese Unterschiede können entweder am Bildschirm angezeigt oder in eine weitere Datei geschrieben werden. Die zu vergleichenden Dateien können entweder hochsprachliche Dateien sein, (d.h. Dateien, die Anweisungen in einer höheren Programmiersprache enthalten) oder auch Binärdateien (d.h. Dateien, wie sie vom MACRO-86-Assembler oder vom Bindeprogramm MS-LINK oder aber auch von einem Microsoft-Hochsprachenkompilierer erzeugt werden).

Für den Vergleich gibt es zwei Möglichkeiten, die wahlweise zur Verfügung stehen: zeilenweiser Vergleich oder byteweiser Vergleich. Wird der zeilenweise Vergleich gewählt, dann vergleicht FC die erste Zeile des zu vergleichenden Abschnitts in der Ausgangsdatei mit der ersten Zeile des zu vergleichenden Abschnitts in der Zieldatei. Ist die erste Zeile in der Zieldatei identisch mit der ersten Zeile in der Ausgangsdatei, geht FC zum Vergleich der zweiten Zeile der Ausgangsdatei mit der zweiten Zeile der Zieldatei über usw. Stellt FC beispielsweise beim Vergleich der dritten Zeile der Ausgangsdatei mit der dritten Zeile der Zieldatei fest, daß diese letztere nicht identisch ist mit der ersteren, dann gibt es den Inhalt der nicht identischen Zeile der Zieldatei am Bildschirm aus oder schreibt sie in die vorgegebene Datei. Dann geht FC zum Vergleich der dritten Zeile der Ausgangsdatei mit der vierten Zeile der Zieldatei über. Ist auch diese nicht identisch mit der ersteren, dann wird auch sie am Bildschirm ausgegeben oder in die Datei geschrieben. Wird bei dem darauf folgenden Vergleich der dritten Zeile der Ausgangsdatei mit der fünften Zeile der Zieldatei Identität festgestellt, dann geht FC zum Vergleich der vierten Zeile der Ausgangsdatei mit der sechsten Zeile der Zieldatei über usw. Beim byteweisen Vergleich wird analog verfahren mit dem Unterschied, daß das erste Byte der Ausgangsdatei mit dem ersten Byte der Zieldatei verglichen wird usw.

DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8.1.1 Einschränkungen beim Vergleich von Binärdateien

Beim Vergleich von hochsprachlichen Dateien benötigt FC viel Speicherkapazität für die Aufnahme der kompletten Dateien. Sind die zu vergleichenden Dateien größer als die verfügbare freie Speicherkapazität, dann wird FC zunächst einmal den Teil der Dateien vergleichen, der in den Arbeitspuffer paßt. Findet FC in den im Puffer abgestellten Abschnitten der beiden zu vergleichenden Dateien keine übereinstimmenden Zeilen, dann wird lediglich die Meldung angezeigt:

***** Files are different *****

Wenn hochsprachliche Dateien so groß sind, daß sie nicht in den verfügbaren Speicherraum passen, vergleicht FC beide Dateien vollständig, indem die Dateien segmentweise von der Diskette in den Arbeitsspeicher geholt werden. Alle Unterschiede werden genauso angezeigt bzw. in eine Datei ausgegeben wie bei Dateien, die vollständig in den Speicher passen.

8.2 SYNTAX UND DATEIBEZEICHNUNGEN

In allen Dateibezeichnungen ist die folgende Syntax bzw. Schreibweise anzuwenden:

[d:] <Dateiname> [<.suf>]

wobei:

d die Buchstabenbezeichnung der Laufwerksangabe darstellt. Wenn eine Laufwerksangabe nicht erfolgt, geht FC davon aus, daß das momentane Standardlaufwerk des Betriebssystems gemeint ist.

Dateiname ist der bis zu 8 Zeichen lange Name der Datei.



DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8.3 WIE MAN DAS DIENSTPROGRAMM FC AUFRUFT

Die Schreibweise für den Aufruf von FC ist wie folgt:

```
FC [# /B /W /C] <Dateiname1> <Dateiname2>
```

FC vergleicht die erste Datei (Dateiname1) mit der zweiten (Dateiname2) und weist alle Unterschiede zwischen den beiden Dateien aus. Anstatt der beiden Dateinamen können Sie auch zwei Suchwegbezeichnungen angeben, wie zum Beispiel:

```
FC B: \NEU \TOPDATEI1.TXT \ TOP \ DATEI2.TXT
```

Mit der Eingabe dieses Befehls bewirken Sie, daß die DATEI1.TXT im Arbeitsbereich \NEU \TOP auf der Diskette in Laufwerk B mit der DATEI2.TXT im Arbeitsbereich \TOP verglichen wird. Da für Dateiname2 keine Laufwerksangabe erfolgte, geht FC automatisch davon aus, daß sich der Arbeitsbereich \TOP auf der Diskette im Standardlaufwerk befindet.

8.4 DIE FC-BEFEHLSZUSÄTZE

Vier Befehlszusätze

stehen Ihnen zur Verwendung mit dem Dienstprogramm FC zur Verfügung:

/B Im Normalfall, d.h., wenn nicht durch die Eingabe eines entsprechenden Befehlszusatzes etwas Anderes veranlaßt wird, vergleicht FC die beiden Dateien zeilenweise. Der Befehlszusatz /B bewirkt davon abweichend, den byteweisen Vergleich der beiden Dateien, wobei das Vergleichsprogramm nicht versucht, wieder Tritt zu fassen, wenn einmal ein Übereinstimmungsfehler erkannt wurde.

DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

Abweichende Dateiinhalte, d.h. Bytes, die in der Zieldatei vorkommen, in der Ausgangsdatei aber nicht, werden wie folgt ausgegeben:

```
--ADDRS ----F1- ---F2-  
xxxxxxx   y      z
```

(wobei xxxxxxx die relative Adresse des Bytepaars, bezogen auf den Anfang der Datei, darstellt.) Anfangsadresse ist 00000000; y und z sind die nicht übereinstimmenden Bytes, jeweils in Datei1 und Datei2. Ist eine der Dateien kleiner als die andere und enthält somit weniger Daten, wird eine entsprechende Meldung ausgegeben. Wenn zum Beispiel das Ende von Datei1 erreicht wird, bevor auch Datei2 zu Ende ist, gibt FC folgende Meldung aus:

*** Data left in F2 ***

stellt eine Zahl von 1 bis 9 dar. Dieser Befehlszusatz ermöglicht es Ihnen, dem Programm mitzuteilen, wieviel Zeilen weiter verglichen werden sollen, wenn ein Übereinstimmungsfehler erkannt wurde. Wird dieser Befehlszusatz nicht verwendet, dann geht FC von dem Standardwert 3 aus. Dieser Befehlszusatz ist nur beim Vergleich von hochsprachlichen Dateien zu verwenden.

/W Bewirkt, daß FC beim Vergleich Leerräume (Tabsprünge und Leerzeichen) im Text verdichtet. Damit wird erreicht, daß mehrere aufeinanderfolgende Leerzeichen oder sonstige Leerräume in einer Zeile als ein einziges Leerzeichen betrachtet werden. Beachten Sie bitte, daß FC, obwohl es Leerräume verdichtet, die vorhandenen Leerzeichen nicht unberücksichtigt läßt. Zwei Ausnahmen sind führende und schließende Leerzeichen am Anfang und am Ende einer Zeile, die von FC nicht berücksichtigt werden. Beispiele (in unseren Beispielen sind die Leerzeichen durch Unterstriche dargestellt):

 Nach weiteren Daten suchen

wird mit

Nach weiteren Daten suchen

und mit Nach weiteren Daten suchen
als identisch angesehen, jedoch nicht mit

 Nach weiteren Daten suchen

Auch dieser Befehlszusatz ist nur beim Vergleich von hochsprachlichen Dateien zu verwenden.



„Weitergabe sowie Verwertung dieser Unterlagen, Vervielfältigung und Mitteilung ohne schriftliche Genehmigung der Nixdorf Computer AG ist ausdrücklich untersagt. Die Nixdorf Computer AG übernimmt keine Haftung für die Richtigkeit der Angaben. Alle Rechte vorbehalten. Nachdruck, Vervielfältigung und Verbreitung, auch auszugsweise, ist ohne schriftliche Genehmigung der Nixdorf Computer AG.“

DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

/C Bewirkt, daß beim Vergleich die Buchstabenart (Groß- oder Kleinbuchstabe) unberücksichtigt bleibt. Alle Buchstaben in den Dateien werden grundsätzlich als Großbuchstaben angesehen. Zum Beispiel:

NACH_WEITEREN_DATEN_suchen

wird mit

Nach_weiteren_Daten_suchen

als identisch angesehen.

Wenn beide Befehlszusätze, d.h. sowohl /W als auch /C vorgegeben werden, dann wird FC entsprechend gleichzeitig Leerräume verdichten und die Buchstabenart unberücksichtigt lassen.

Beispiel:

___Nach_WEITEREN_Daten_suchen___

wird mit

Nach_weiteren_Daten_suchen

als identisch angesehen.

Auch dieser Befehlszusatz ist nur beim Vergleich von hochsprachlichen Dateien zu verwenden.

DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8.5 WIE ERKANNTE UNTERSCHIEDE ANGEZEIGT WERDEN

Die Unterschiede zwischen den beiden von Ihnen angegebenen Dateien zeigt das Dienstprogramm FC für den Dateienvergleich an, indem es zunächst den Namen der Datei 1 ausgibt, deren Inhalt mit dem der anderen verglichen wird, gefolgt von den Zeilen der verglichenen Datei 2, die nicht mit der betreffenden Zeile der Datei 1 übereinstimmen und, schließlich, der ersten Zeile der verglichenen Datei 2, die mit der betreffenden der anderen übereinstimmt. Dann wird die Datei 2 mit der Datei 1 auf dieselbe Weise verglichen. Es wird ihr Name ausgegeben, gefolgt von den Zeilen der Datei 2, die nicht mit der betreffenden Zeile der Datei 1 übereinstimmen und, sobald eine Übereinstimmung gefunden wurde, von der ersten Zeile der Datei 2, die mit der betreffenden der Datei 1 übereinstimmt. Standardvorgabe für die Anzahl der bis zur Erzielung einer Übereinstimmung zu vergleichenden Zeilen ist 3. (Wenn Sie diese Standardvorgabe ändern möchten, geben Sie die gewünschte Anzahl von Zeilen mit Hilfe des Befehlszusatzes /# ein.)

Beispiel:

```
-----<Dateiname1>
<nicht übereinstimmende Zeile>
<1. Zeile in der Datei1, die mit Datei2übereinstimmt>
-----<Dateiname2>
<nicht übereinstimmende Zeile>
<1. Zeile in der Datei2, die mit Datei1 übereinstimmt>
-----
```

FC wird, weiterhin nach diesem Schema verfahrend, alle Unterschiede zwischen den beiden Dateien ausweisen.

Ist die Anzahl der nicht übereinstimmenden Zeilen zu groß (größer als die mit dem Befehlszusatz /# vorgegebene Zahl bzw. als die Standardvorgabe 3), dann wird das Programm einfach nur mitteilen, daß die Dateien verschieden sind und seine Tätigkeit beenden. Werden nach Erkennung des ersten Übereinstimmungsfehlers keine übereinstimmenden Zeilen mehr in den beiden Dateien gefunden, dann wird FC die Meldung ausgeben:

```
*** Files are different ***
```

und Sie in MS-DOS zurückverweisen bzw. die Standardlaufwerk-Eingabeaufforderung von MS-DOS (beispielsweise A>) anzeigen.



DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8.6 NEUBESTIMMUNG DES AUSGABEZIELS FÜR FC

Wenn Sie nichts anderes angeben, werden alle weiter oben erwähnten Ausgaben von FC, d.h. die Angabe der übereinstimmenden und der nicht übereinstimmenden Zeilen der beiden Dateien, an Ihrem Bildschirm erfolgen. Sie können dies jedoch ändern, indem Sie das Ausgabeziel neu bestimmen, was in derselben Weise erfolgt wie die Neubestimmung eines Ausgabeziels für einen MS-DOS-Befehl (siehe Kapitel 4 "Wissenswertes über MS-DOS-Befehle").

Wenn Sie beispielsweise Datei1 und Datei2 vergleichen und das Ergebnis des Vergleichs von FC in die Datei UNTERSCH.TXT ausgeben möchten, geben Sie bitte ein:

```
FC Datei1 Datei2 >UNTERSCH.TXT
```

Die Angabe der übereinstimmenden und der nicht übereinstimmenden Zeilen in Datei1 und Datei2 wird dann in die Datei UNTERSCH.TXT auf der Diskette im Standardlaufwerk geschrieben.

DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8.7 BEISPIELE

Beispiel 1:

Nehmen wir an, auf Ihrer Diskette würden folgende beiden ASCII-Dateien stehen:

ALPHA.ASM	BETA.ASM
DATEI A	DATEI B

A	A
B	B
C	C
D	G
E	H
F	I
G	J
H	1
I	2
M	P
N	Q
O	R
P	S
Q	T
R	U
S	V
T	4
U	5
V	W
W	X
X	Y
Y	Z
Z	

Wenn Sie veranlassen möchten, daß FC für diese beiden Dateien vergleicht und die Unterschiede am Bildschirm anzeigt, geben Sie ein:

```
FC ALPHA.ASM BETA.ASM
```

FC vergleicht nun ALPHA.ASM mit BETA.ASM und zeigt Ihnen die Unterschiede am Bildschirm Ihres Computers an. Sämtliche anderen Standardvorgaben bleiben unverändert (also zum Beispiel: "Tabsprünge, Leerzeichen und Kommentarzeilen sind nicht in den Vergleich einzubeziehen", und es ist ein byteweiser Vergleich zwischen den beiden Dateien durchzuführen".)



DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

Die entsprechende Ausgabe wird Ihnen FC in der folgenden Form auf Ihrem Bildschirm bringen (die Erläuterungen rechts sind natürlich nicht Teil der Anzeige):

```
-----ALPHA.ASM
D                                ANMERKUNG: Die Datei
E                                ALPHA enthält defg;
F                                BETA enthält nur g.
G
-----BETA.ASM
G
-----
-----ALPHA.ASM
M                                ANMERKUNG: Die Datei
N                                ALPHA enthält mno,
O                                während BETA j12
P                                enthält.
-----BETA.ASM
J
1
2
P
-----
-----ALPHA.ASM
W                                ANMERKUNG: Die Datei
-----BETA.ASM                ALPHA enthält w
4                                während BETA 45W
5                                enthält.
W
```

Beispiel 2:

Ausgehend von einem Vergleich derselben beiden Dateien können Sie die Unterschiede auch auf Ihrem Drucker ausgeben lassen. In diesem Beispiel müssen vier aufeinanderfolgende Zeilen identisch sein, um als Übereinstimmung erkannt zu werden.

Eingabe:

```
FC /4 ALPHA.ASM BETA.ASM PRN
```

DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

Nun wird folgende Ausgabe auf Ihrem Drucker erscheinen:

-----ALPHA.ASM

D
E
F
G
H
I
M
N
O
P

ANMERKUNG: p ist die
1. einer Reihe von
4 Übereinstimmungen.

-----BETA.ASM

G
H
I
J
1
2
P

-----ALPHA.ASM

W

ANMERKUNG: w ist die
1. einer Reihe von
4 Übereinstimmungen.

-----BETA

4
5
W

Beispiel 3:

In diesem Beispiel wird ein bitweiser Vergleich derselben in dem vorhergehenden Beispiel bereits verwendeten Dateien veranlaßt, und die Unterschiede werden am Bildschirm angezeigt.

Eingabe:

FC /B ALPHA.ASM BETA.ASM



Weitergabe sowie Vervielfältigung dieses Unterlags, Verwertung und Mitteilung
 ihres Inhaltes nicht gestattet. Soweit nicht ausdrücklich zugelassen, Zuwider-
 handeln ist strafbar. Die Verantwortung für den Fall der Patent-
 erteilung oder Gebrauchsmusterverleihung verbleibt bei dem Erfinder.

DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

Der Befehlszusatz /B bewirkt in diesem Beispiel, daß ein byteweiser Vergleich durchgeführt wird. Dieser Befehlszusatz sowie auch alle anderen müssen in der FC-Befehlszeile vor den Dateinamen eingegeben werden. Nun müßte die folgende Anzeige auf Ihrem Bildschirm erscheinen:

--ADDRS-----	F1----	F2--
00000009	44	47
0000000C	45	48
0000000F	46	49
00000012	47	4A
00000015	48	31
00000018	49	32
0000001B	4D	50
0000001E	4E	51
00000021	4F	52
00000024	50	53
00000027	51	54
0000002A	52	55
0000002D	53	56
00000030	54	34
00000033	55	35
00000036	56	57
00000039	57	58
0000003C	58	59
0000003F	59	5A
00000042	5A	1A

DAS DIENSTPROGRAMM FC FÜR DEN DATEIENVERGLEICH

8.8 FEHLERMELDUNGEN IN FC

Wenn das Dienstprogramm FC für den Dateienvergleich einen Fehler erkennt, wird eine oder mehrere der folgenden Fehlermeldungen angezeigt oder ausgedruckt:

Incorrect DOS version

Sie haben versucht, FC unter einer Fassung von MS-DOS auszuführen, die älter als 2.0 ist.

Invalid parameter: <Option> !

Einer der Befehlszusätze, die Sie eingegeben haben, ist ungültig.

Files not found: <Dateiname>

FC konnte eine der von Ihnen angegebenen Dateien nicht finden.

Read error in: <Dateiname>

FC war nicht in der Lage, die ganze Datei zu lesen.

Invalid number of parameters

Sie haben in der FC-Befehlszeile eine unzulässige Zahl von Optionen eingegeben.



DAS BINDEPROGRAMM MS-LINK

9 DAS BINDEPROGRAMM MS-LINK

9.1	Einführung.....	9	- 2
9.2	Kurzbeschreibung von MS-LINK.....	9	- 2
9.3	Einige Begriffe, die Sie kennen sollten.....	9	- 4
9.4	Die Dateien, mit denen MS-LINK arbeitet.....	9	- 6
9.4.1	Die Suffixe der Eingabedateien.....	9	- 6
9.4.2	Die Suffixe der Ausgabedateien.....	9	- 6
9.4.3	Die vorläufige Arbeitsdatei VM.TMP.....	9	- 7
9.5	Wie man MS-LINK startet.....	9	- 8
9.5.1	Erstes Verfahren: Beantwortung der Eingabeaufforderungen	9	- 8
9.5.2	Zweites Verfahren: die lange Befehlszeile.....	9	- 10
9.5.3	Drittes Verfahren: die Antwortdateien.....	9	- 11
9.6	Hilfs- und Steuerzeichen in MS-LINK.....	9	- 13
9.7	Die MS-LINK-Eingabeaufforderungen.....	9	- 15
9.8	MS-LINK-Befehlszusätze.....	9	- 17
9.9	Beispiele eines MS-LINK-Bindelaufs.....	9	- 21
9.10	Fehlermeldungen in MS-LINK.....	9	- 24

Weitergabe, sowie Verzichtigung, Übersetzung, Vervielfältigung und Mitteilung
 in jeglicher Form ist ohne schriftliche Genehmigung ausdrücklich untersagt. Zuwider-
 handlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-
 erteilung oder Gebrauchsmustererteilung vorbehalten.



DAS BINDEPROGRAMM MS-LINK

9.1 EINFÜHRUNG

In diesem Kapitel möchten wir Sie mit dem Bindeprogramm MS-LINK vertraut machen. Bevor Sie dieses Programm benutzen, sollten Sie dieses Kapitel aufmerksam und lückenlos lesen.

HINWEIS

Wenn Sie nicht die Absicht haben, Programme zu kompilieren und zu binden, brauchen Sie dieses Kapitel natürlich nicht zu lesen.

Das MS-DOS-Bindeprogramm MS-LINK leistet folgende Funktionen:

- Es verbindet getrennt erstellte maschinensprachliche Programmmodule zu einem verschiebbaren Lademodul, d.h. zu einem lauffähigen Programm.
- Es sucht die Schlüssel noch unentschlüsselter Externverweise in Modulbibliotheken.
- Es entschlüsselt externe Querverweise.
- Es erstellt eine Liste, aus der sowohl die Schlüssel der Externverweise als auch entsprechende Fehlermeldungen hervorgehen.

9.2 KURZBESCHREIBUNG VON MS-LINK

Wenn Sie ein Programm schreiben, werden Sie dies wahrscheinlich in einer höheren Programmiersprache tun, d.h., Sie erstellen einen hochsprachlichen Code. Dieser hochsprachliche Code muß dann als erstes mit Hilfe eines Kompilierers in maschinensprachlichen Code, und zwar zunächst in maschinensprachliche Teilprogramme oder Module, umgesetzt werden. Diese maschinensprachlichen Module müssen dann ihrerseits einen Bindeprozeß durchlaufen, um ein maschinensprachliches Programm zu erstellen, das der Computer unmittelbar lesen kann, d.h., ein sogenanntes lauffähiges Programm.

Auch wenn Sie mehrere Programme erstellen wollen, die Sie in verschiedenen Kombinationen in einem Durchlauf ausführen möchten, wird Ihnen das Bindeprogramm MS-LINK Vorteile bringen. Jedes Ihrer Teilprogramme kann dann auf ein Symbol verweisen, das in einem anderen Programmmodul verschlüsselt ist. Diesen Verweis von einem Teilprogramm auf ein anderes nennen wir einen Externverweis.

DAS BINDEPROGRAMM MS-LINK

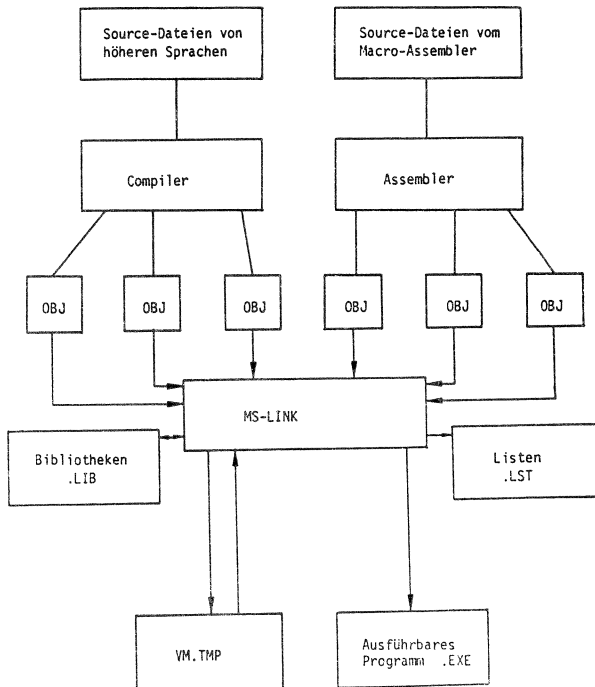
MS-LINK ermöglicht es Ihnen, mehrere solche maschinensprachliche Teilprogramme zu einem verschiebbaren Lademodul bzw. zu einem lauffähigen Programm zusammenzufassen. Bei dieser Zusammenfassung von Teilprogrammen stellt MS-LINK sicher, daß alle Externverweise zwischen den verschiedenen Programmmodulen entschlüsselt sind. Schlüssel, die in den Programmmodulen selbst nicht enthalten sind, kann MS-LINK in mehreren Modulbibliotheken suchen.

Des weiteren erzeugt MS-LINK eine Auflistung, die in eine Datei geschrieben wird, in der die Schlüssel sämtlicher Externverweise und auch die entsprechenden Fehlermeldungen aufgezeigt werden.

MS-LINK nutzt die verfügbare Speicherkapazität optimal. Reicht sie jedoch nicht aus, dann erstellt MS-LINK eine vorläufige Diskettendatei mit dem Namen VM.TMP.

Abbildung 9 veranschaulicht die verschiedenen Abschnitte des Bindepzesses.

Abbildung 9. Der Bindepzess mit MS-LINK



„Weitergabe sowie Verwältigung dieser Unterlagen, Verwertung und Mitteilung
 handlungen verpflichten zu Schriftensatz. Alle Rechte für den Fall der Por-
 erteilung oder Gebrauchsmusterreimung vorbehalten.“



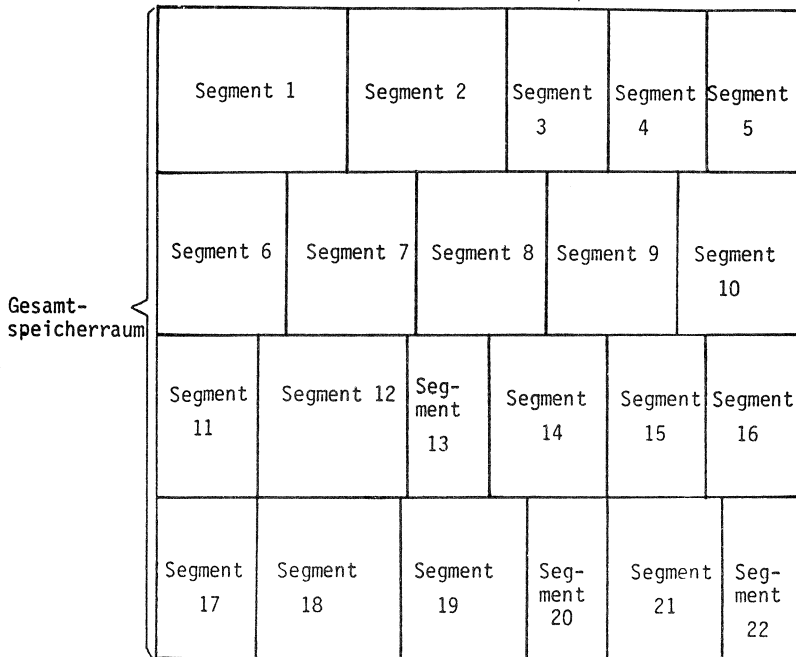
DAS BINDEPROGRAMM MS-LINK

9.3 EINIGE BEGRIFFE, DIE SIE KENNEN SOLLTEN

Wenn die maschinensprachlichen Module, die Sie mit Hilfe von MS-LINK binden möchten, aus hochsprachlichem Code wie BASIC, Pascal oder einer anderen höheren Programmiersprache kompiliert werden, brauchen Sie die Begriffe, die die interne Arbeitsweise von MS-LINK beschreiben, nicht zu kennen. Wenn Sie jedoch Programme in Assembler Sprache schreiben und kompilieren möchten, ist es erforderlich, daß Sie die Arbeitsweise von MS-LINK auch im einzelnen verstehen. Daher möchten wir Ihnen im folgenden die wichtigsten Begriffe, die in diesem Kapitel zur Beschreibung der Arbeitsweise von MS-LINK verwendet werden, kurz erläutern. Nützliche Informationen hierzu finden Sie auch in dem MS-LINK gewidmeten Abschnitt des Macro-Assembler-Handbuchs.

Unter MS-DOS kann der Speicherraum in Segmente, Klassen und Gruppen unterteilt werden. Diese drei Begriffe werden durch Abbildung 10 veranschaulicht.

Abbildung 10. Die Unterteilung des Speicherraums unter MS-DOS.



DAS BINDEPROGRAMM MS-LINK

Beispiel:

	Segmentname	Klassenname
Segment 1	PROG.1	CODE
Segment 2	PROG.2	CODE
Segment 12	PROG.3	DATEN

Die Segmente 1, 2 und 12 bilden in unserem Beispiel eine Gruppe mit einer Gruppenadresse, die der niedrigsten Adresse des Segments 1 (d.h. der niedrigsten Adresses überhaupt im gesamten Speicherbereich) entspricht. Jedes Segment ist durch einen Segmentnamen und einen Klassennamen gekennzeichnet. Beachten Sie bitte, daß die Segmente 1, 2 und 12 verschiedene Segmentnamen haben, daß sie aber nicht unbedingt auch denselben Klassennamen haben müssen.

MS-LINK lädt alle Segmente nach Klassennamen in den Speicher, und zwar angefangen bei dem ersten angetroffenen Segment bis zu dem letzten. Alle derselben Klasse angehörigen Segmente werden in zusammenhängende Speicherbereiche geladen.

Während der Verarbeitung spricht MS-LINK die Segmente über ihre Speicheradressen an (wo sie geladen wurden). MS-LINK erreicht dies, indem es die Segmente über ihre Gruppen adressiert.

Eine Gruppe ist eine Anzahl von Segmenten, die in einen 64 kbyte großen Speicherbereich passen. Die Segmente brauchen nicht nebeneinander zu stehen, um eine Gruppe zu bilden (siehe Abbildung). Die Adresse einer Gruppe ist die niedrigste Adresse des niedrigsten Segments in dieser Gruppe. Bei dem Bindelauf untersucht MS-LINK zuerst die Gruppe und spricht dann die Segmente über die Speicheradressen der Gruppe an. Ein Programm kann aus einer oder mehreren Gruppen bestehen.

Wenn Sie ein Programm in Assemblerersprache schreiben, können Sie die Gruppen- und Klassennamen in Ihrem Programm selbst bestimmen. Wenn Sie Ihr Programm in einer höheren Programmiersprache schreiben (wie BASIC, COBOL, FORTRAN, PASCAL), erfolgt die Vergabe der Namen durch den Compiler.

Nähere Informationen bezüglich der Vergabe von Gruppen- und Klassennamen sowie über die Art und Weise, wie MS-LINK die Segmente im Speicher anordnet und zusammenfaßt, finden Sie im Macro-Assembler-Handbuch.



DAS BINDEPROGRAMM MS-LINK

9.4 DIE DATEIEN, MIT DENEN MS-LINK ARBEITET

MS-LINK nimmt eine oder mehrere Eingabedateien an, erstellt zwei Ausgabedateien, erstellt bei Bedarf eine vorläufige Diskettendatei und kann dazu angewiesen werden, bis zu 8 Modulbibliotheksdateien abzusuchen.

Für jede Dateiart kann der Anwender eine dreiteilige Dateibezeichnung angeben. Das Format der MS-LINK-Dateibezeichnung ist dasselbe wie dasjenige einer Diskettendatei:

[d:] <Dateiname> [<.suf>]

wobei:

d die Laufwerksangabe darstellt. Zulässige Laufwerksangaben sind in MS-LINK A bis O. Jede Laufwerksangabe muß grundsätzlich mit einem Doppelpunkt (A:) als Bestandteil der Laufwerksangabe abgeschlossen werden.

Dateiname ein beliebiger zulässiger Dateiname mit bis zu 8 Zeichen Länge ist.

ist. Der Punkt (.) muß grundsätzlich als Teil des Suffixes angegeben werden.

9.4.1 Die Suffixe der Eingabedateien

Wenn in den Eingabedateibezeichnungen (Maschinencoddateien) keine Suffixe angegeben werden, erkennt MS-LINK standardmäßig die folgenden Suffixe:

9.4.2 Die Suffixe der Ausgabedateien

MS-LINK vergibt automatisch die folgenden Standardsuffixe an seine Ausgabedateien (lauffähige Dateien und Auflistungen):

DAS BINDEPROGRAMM MS-LINK

9.4.3 Die vorläufige Arbeitsdatei VM.TMP

Beim Bindeprozeß nutzt MS-LINK die zur Verfügung stehende Speicherkapazität optimal aus. Wenn die durch MS-LINK gebundenen Dateien jedoch zu einer Ausgabedatei führen, die so groß ist, daß sie nicht mehr in den Arbeitsspeicher paßt, dann erstellt MS-LINK eine vorläufige Arbeitsdatei mit dem Namen VM.TMP auf der Diskette im Standardlaufwerk. In diesem Fall warnt Sie MS-LINK mit der folgenden Meldung:

**VM.TMP has been created.
Do not change diskette in drive, d:**

Wenn diese Meldung angezeigt worden ist, dürfen Sie die Diskette in Ihrem Standardlauf bis zum Ende des Bindelaufs nicht wechseln. Wenn Sie die Diskette trotzdem herausnehmen, wird das weitere Verhalten von MS-LINK unberechenbar. Mit größter Wahrscheinlichkeit wird folgende Meldung angezeigt:

Unexpected end of file on VM.TMP

Der Inhalt von VM.TMP wird grundsätzlich in die Datei übertragen, die Sie als Antwort auf die Eingabeaufforderung "Ausgabedatei:" angeben. Denn VM.TMP ist lediglich eine Arbeitsdatei, die am Ende des Bindelaufs automatisch wieder gelöscht wird.

HINWEIS

Benutzen Sie die Bezeichnung VM.TMP bitte nie als Dateiname für eine Ihrer Dateien. Wenn Sie eine Datei mit diesem Namen auf der Diskette im Standardlaufwerk haben und MS-LINK die Datei VM.TMP aufruft, wird die alte, noch auf der Diskette gespeicherte Datei VM.TMP (das wäre Ihre Datei!) automatisch gelöscht und eine neue erstellt. Das heißt, der Inhalt Ihrer Datei würde unwiderruflich verlorengehen.



DAS BINDEPROGRAMM MS-LINK

9.5 WIE MAN MS-LINK STARTET

Um aktiv zu werden, benötigt MS-LINK zwei Arten von Eingaben: einen Befehl, mit dem es gestartet wird, und eine Reihe von Antworten auf entsprechende Eingabeaufforderungen. Außerdem können Sie die Feinsteuerung von MS-LINK mit Hilfe von 7 Befehlszusätzen beeinflussen. Die normale Arbeitsweise ist, daß Sie alle Befehle und Antworten auf die MS-LINK-Eingabeaufforderung über die Bedienungsplatztastatur eingeben. Alternativ dazu können Sie die Antworten auf die MS-LINK-Eingabeaufforderungen sowie alle gewünschten Befehlszusätze in einer Antwortendatei zur Verfügung stellen. Während des Dialogs mit MS-LINK können Sie sich einer Reihe von Hilfs- und Steuerzeichen bedienen.

Es gibt drei Möglichkeiten, MS-LINK zu starten. Die erste besteht darin, daß Sie die Befehle und Antworten auf die verschiedenen Eingabeaufforderungen unmittelbar selbst über die Tastatur eingeben. Die zweite Möglichkeit besteht darin, daß Sie alle Befehle und Antworten in eine Befehlszeile schreiben. Die dritte Möglichkeit setzt voraus, daß Sie eine Antwortendatei erstellen, die alle erforderlichen Befehle und Antworten enthält, und beim Start MS-LINK mitteilen, wo diese Datei sich befindet.

Zusammenfassung der Startmöglichkeiten für MS-LINK:

```
-----  
Verfahren 1      LINK  
Verfahren 2      LINK <Dateiname> [/Befehlszusätze]  
Verfahren 3      LINK @ <Dateibezeichnung>  
-----
```

9.5.1 Erstes Verfahren: Beantwortung der Eingabeaufforderungen

Wenn Sie MS-LINK nach dem ersten Verfahren starten möchten, geben Sie bitte ein:

LINK

Danach wird MS-LINK in den Arbeitsspeicher geladen. MS-LINK wird Ihnen nun nacheinander vier Eingabeaufforderungen zeigen, die Sie einzeln beantworten müssen.

Am Ende jeder Eingabezeile können Sie zusätzlich noch einen oder mehrere Befehlszusätze eingeben, vor denen jeweils der Zusatzcode, d.h. der normale Schrägstrich, stehen muß.

DAS BINDEPROGRAMM MS-LINK

Für den schnellen Überblick haben wir die vier möglichen Eingabeaufforderungen nachstehend kurz zusammengefaßt. Eine eingehendere Beschreibung finden Sie im Abschnitt 9.7 "Die MS-LINK-Eingabeaufforderungen".

EINGABEAUFFORDERUNG

ANTWORTEN

Eingabemodule [OBJ] :

Hier müssen Sie die .OBJ-Teilprogramme anführen, die Sie binden möchten. Sie müssen bei der Eingabe durch Leerzeichen oder Pluszeichen (+) gegeneinander abgegrenzt sein. Wenn Sie als letztes Zeichen ein Pluszeichen eingeben, wird die Eingabeaufforderung erneut erscheinen. Eine Standardannahme ist nicht vorgesehen, eine Antwort auf jedem Fall erforderlich.

Ausgabedatei [Erster
Modulname.EXE] :

Hier müssen Sie einen Namen für die lauffähige Ausgabedatei angeben. Standardvorgabe ist der Name des ersten, in Beantwortung der ersten Eingabeaufforderung angegebenen Eingabemoduls mit dem Suffix .EXE. (Das Suffix des Ausgabedateinamens ist Ihrem Zugriff entzogen.)

Listendatei [Name der
Ausgabedatei.MAPS] :

Hier müssen Sie den Namen für die Datei eingeben, in der die Auflistung niedergeschrieben werden soll. Standardvorgabe ist der Name der Ausgabedatei.

Modulbibliotheken [] :

Hier müssen die Dateinamen der verschiedenen Modulbibliotheken angegeben, die nach den Verweisungsschlüsseln abgesucht werden sollen. Bitte die Dateinamen durch Leerzeichen oder Pluszeichen (+) gegeneinander abgrenzen. Wenn Sie als letztes Zeichen ein Pluszeichen eingeben, erscheint die Eingabeaufforderung erneut. Standardvorgabe sind die Standardmodulbibliotheken in den Maschinencodemodulen. (Die Suffixe werden automatisch in .LIB umgewandelt.)



DAS BINDEPROGRAMM MS-LINK

9.5.2 Zweites Verfahren: die lange Befehlszeile

Um MS-LINK nach dem zweiten Verfahren zu starten, geben Sie bitte alle Befehle und Antworten gleich in der ersten Befehlszeile ein. Die auf LINK folgenden Eingaben sind die Antworten auf die MS-LINK-Eingabeaufforderungen. Die Eingabefelder der verschiedenen Eingabeaufforderungen müssen durch Kommas gegeneinander abgegrenzt sein. Wenden Sie bitte die folgende Schreibweise an:

```
LINK  
<Liste der Eingabemodule> , <Ausgabedatei> , <Listendatei> ,  
<Liste der Modulbibliotheken> [/Befehlszusatz...]
```

Die Antworten auf die verschiedenen MS-LINK-Eingabeaufforderungen werden also unmittelbar nach dem Befehl LINK in derselben Zeile eingegeben. Die verschiedenen Antworten müssen voneinander durch Kommas abgegrenzt sein.

Erläuterung der verschiedenen Optionen:

Liste der Eingabemodule ist die Liste der verschiedenen Teilprogramme im Maschinencode, durch Pluszeichen gegeneinander abgegrenzt.

Ausgabedatei ist der Name der Ausgabedatei für das lauffähige Maschinencodeprogramm.

Listendatei ist der Name der Datei für die von MS-LINK auszugebende Auflistung.

Liste der Modulbibliotheken ist die Liste der abzusuchenden Modulbibliotheksdateien.

/Befehlszusatz verweist auf die wahlfrei einzusetzenden Befehlszusätze, die unmittelbar im Anschluß an die verschiedenen Antworten eingegeben werden können (entweder ohne Leerzeichen vor jedem der Kommas oder nach der letzten Antwort Liste der Modulbibliotheken, wie dargestellt).

Wenn Sie anstatt einer bestimmten Antwort die entsprechende Standardantwort wählen möchten, geben Sie einfach ein zweites Komma ohne Zwischenraum ein.

DAS BINDEPROGRAMM MS-LINK

Beispiel:

```
LINK  
BLAU+GRUN+ROT+GELB/P/M,,NEULISTE,TOPLIB.LIB
```

Mit diesem Befehl bewirken Sie, daß als erstes MS-LINK und gleich anschließend die Maschinencodemodule BLAU.OBJ, GRUN.OBJ, ROT.OBJ und GELB.OBJ geladen werden. Danach macht MS-LINK (aufgrund des Zusatzes /P) eine Pause. Der Bindelauf kann nun durch Betätigung einer beliebigen Taste gestartet werden, worauf MS-LINK (aufgrund des Zusatzes /M) sämtliche allgemeinen Symbole auflistet, dann für die Ausgabedatei die Standardvorgabe BLAU.EXE wählt, schließlich eine Listendatei mit dem Namen NEULISTE.MAP anlegt und endlich die Modulbibliotheksdatei TOPLIB.LIB absucht.

9.5.3 Drittes Verfahren: die Antwortendatei

Um MS-LINK nach dem dritten Verfahren zu starten, geben Sie bitte folgendes ein:

```
LINK @ <Dateibezeichnung>
```

wobei:

Dateibezeichnung der Name einer Antwortendatei ist. Die Antwortendatei muß alle Antworten auf die MS-LINK-Eingabeaufforderungen (wie unter Verfahren 1 beschrieben) enthalten. Sie kann außerdem noch beliebige Befehlszusätze enthalten. Bei der Benennung einer Antwortendatei können Sie ein Suffix festlegen oder auch nicht. Wenn Sie mit dem Verfahren 3 arbeiten, können Sie den Befehl, mit dem Sie MS-LINK starten, entweder über die Tastatur oder über einen Stapelprogrammbaustein eingeben, ohne daß Sie in diesem letzteren Fall sonst noch etwas dazu tun müßten.

Wenn Sie MS-LINK nach diesem dritten Verfahren starten möchten, müssen Sie eine Antwortendatei erstellen, die mehrere Textzeilen enthält, von denen jede die Antwort auf eine der MS-LINK-Eingabeaufforderungen enthält. Die Antworten müssen in derselben Reihenfolge angeordnet sein, wie die in der Beschreibung des ersten Verfahrens aufgeführten MS-LINK-Eingabeaufforderungen. Falls erforderlich und gewünscht, können Sie Ihre Antworten auf die Eingabeaufforderungen zu den Eingabemodulen und zu den Modulbibliotheken über mehrere Zeilen fortsetzen, indem Sie als Anschlußzeichen für die jeweils weitere Zeile ein Pluszeichen (+) setzen.



DAS BINDEPROGRAMM MS-LINK

Befehlszusätze sowie Hilfs- und Steuerzeichen können Sie in der Antwortendatei genauso verwenden wie bei der Eingabe Ihrer Antworten über die Bedienungsplatztastatur.

Sobald der Bindelauf mit MS-LINK beginnt, werden die Eingabeaufforderungen nacheinander mit den der Antwortendatei entnommenen Antworten angezeigt. Enthält die Antwortendatei keine Antworten für alle Eingabeaufforderungen (entweder in der Form von Dateinamen, von Strichpunkten als Hilfszeichen oder von Wagenrücklaufzeichen), dann zeigt MS-LINK die Eingabeaufforderung, zu der die Antwort fehlt, am Bildschirm an und wartet, bis Sie eine gültige Antwort eingegeben haben. Sobald die gültige Antwort eingegeben wird, fährt MS-LINK mit dem Bindelauf fort.

Beispiel:

```
BLAU GRUN ROT GELB  
/PAUSE/MAP  
NEULISTE  
TOPLIB.LIB
```

Mit dieser Antwortendatei weisen Sie MS-LINK an, die vier Maschinencodemodule mit den Namen BLAU, GRUN, ROT und GELB zu laden. Dann beginnt MS-LINK mit der Verarbeitung und stoppt vor der Ausgabe des Verzeichnisses der allgemeinen Symbole, um Ihnen das Auswechseln der Disketten zu ermöglichen (siehe Beschreibung unter /PAUSE im Abschnitt 9.8 "MS-LINK-Befehlszusätze" in diesem Kapitel, bevor Sie diese Funktion benutzen). Wenn Sie dann eine beliebige Taste betätigen, wird die Verarbeitung fortgesetzt, und es erfolgt die Ausgabe der Ausgabedateien mit den Namen BLAU.EXE und NEULISTE.MAP. Die Verweisungsschlüssel wird sich MS-LINK aus der Modulbibliotheksdatei TOPLIB.LIB suchen und als Befehlszusätze die Standardvorgaben verwenden.

DAS BINDEPROGRAMM MS-LINK

9.6 HILFS- UND STEUERZEICHEN IN MS-LINK

MS-LINK versteht drei Hilfs- bzw. Steuerzeichen:

Pluszeichen

Mit dem Pluszeichen (+) können Sie einerseits mehrere Antworten auf dieselbe Eingabeaufforderung voneinander abgrenzen und andererseits bei der Eingabe der Antworten auf die Eingabeaufforderungen bezüglich der Eingabemodule und der Modulbibliotheken die momentane Schreibzeile verlängern. (Um die verschiedenen angegebenen Eingabemodule voneinander abzugrenzen, können Sie auch ein Leerzeichen benutzen.) Wenn Sie eine größere Anzahl von längeren Antworten (jede einzelne kann sehr lang sein) eingeben möchten, setzen Sie am Ende jeder Zeile ein Pluszeichen, das Sie mit <RETURN> abschließen, um die Zeile zu verlängern. Geben Sie diese Zeichenkombination als letzte Eingabe zu einer dieser beiden Eingabeaufforderungen ein, dann wird MS-LINK Sie auffordern, weitere Modulnamen zu nennen. Wenn die jeweilige Eingabeaufforderung bezüglich der Eingabemodule oder der Modulbibliotheken erneut erscheint, geben Sie weitere Antworten ein. Wenn alle zu bindenden Eingabemodule und alle abzusuchenden Modulbibliothekdateien angegeben sind, vergewissern Sie sich, daß die letzte Antwortzeile mit einem Modulnamen und RETURN , und nicht mit einem Pluszeichen und RETURN endet.

Beispiele:

```

Eingabemodul  [ .OBJ ] : BLAU GRUN ROT GELB+ <RETURN>
Eingabemodul  [ .OBJ ] : BAUM+BLUME+BACH+ <RETURN>
Eingabemodul  [ .OBJ ] : SEERAUBER <RETURN>
  
```

Strichpunkt

Der Strichpunkt, unmittelbar durch <RETURN> abgeschlossen, ermöglicht es Ihnen, an einer beliebigen Stelle nach der ersten Eingabeaufforderung (Eingabemodule:) für alle noch zu beantwortenden Eingabeaufforderungen als Antwort die Standardvorgaben zu wählen. Diese Funktion spart Zeit, da die Notwendigkeit entfällt, eine Reihe von <RETURN> einzugeben.



Weitergabe sowie Vervielfältigung dieses Unterlages, Vervielfältigung und Mitteilung
 ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden, Zuwider-
 handeln kann strafbar sein. Alle Rechte vorbehalten. © 1984 Nixdorf Computer AG.
 Erlaubung oder Gebrauchszustimmung vorbehalten.

DAS BINDEPROGRAMM MS-LINK

HINWEIS

Sobald Sie den Strichpunkt geschrieben und (durch Drücken der RETURN-Taste) eingegeben haben, können Sie auf keine der verbleibenden Eingabeaufforderungen bei diesem Bindelauf mehr einzeln antworten. Benutzen Sie daher den Strichpunkt auf keinen Fall, um Eingabeaufforderungen zu überspringen. Verwenden Sie dazu vielmehr die RETURN-Taste.

Beispiel:

```
Eingabemodule  
  [.OBJ] : BLAU GRUN ROT GELB+ <RETURN>  
Ausgabedatei [BLAU.EXE] : ; <RETURN>
```

Ab hier wird keine weitere Eingabeaufforderung mehr erscheinen, und MS-LINK wird für alle noch ausstehenden Eingabeaufforderungen die Standardantworten verwenden (einschließlich BLAU.MAP für die Listendatei).

<CONTROL-C>

Die Tastenkombination <CONTROL-C> ermöglicht es Ihnen, den Bindelauf zu einem beliebigen Zeitpunkt abubrechen. Wenn Sie eine falsche Antwort, wie beispielsweise einen verkehrten Dateinamen oder auch einen falsch geschriebenen Dateinamen eingegeben haben, bleibt Ihnen nämlich nichts anderes übrig, als MS-LINK zu verlassen, indem Sie die Tastenkombination CONTROL-C benutzen und anschließend MS-LINK erneut starten. Haben Sie den Fehler lediglich geschrieben, aber noch nicht durch Betätigung der RETURN-Taste eingegeben, dann haben Sie noch die Möglichkeit, die fehlerhaften Zeichen mit der Rücktaste zu löschen. Das geht jedoch jeweils nur in der Zeile, in der Sie sich gerade befinden.

DAS BINDEPROGRAMM MS-LINK

9.7 DIE MS-LINK-EINGABEAUFFORDERUNGEN

MS-LINK erwartet von Ihnen Antworten auf vier Texteingabeaufforderungen. Sobald Sie die Antwort auf eine dieser Eingabeaufforderungen geschrieben und die RETURN-Taste gedrückt haben, um Ihre Antwort einzugeben, erscheint die nächste Eingabeaufforderung. Sobald die Antwort auf die letzte Eingabeaufforderung eingegeben worden ist, beginnt automatisch der Bindelauf mit MS-LINK, ohne daß Sie dazu irgendwelche weiteren Befehle einzugeben brauchten. Nach Abschluß des Bindelaufs übergibt MS-LINK ebenso automatisch die Steuerung wieder an das Betriebssystem. Das heißt, daß der Bindelauf erfolgreich abgelaufen ist, sobald Sie die MS-DOS-Eingabeaufforderung am Bildschirm sehen. Konnte der Bildlauf nicht erfolgreich abgeschlossen werden, dann zeigt Ihnen MS-LINK am Bildschirm die entsprechende Fehlermeldung an.

Die MS-LINK-Eingabeaufforderungen dienen dazu, Ihnen die Eingabe der Eingabemodule, der Ausgabedatei, der Listendatei und der Modulbibliotheksdateien zu ermöglichen. Die Eingabeaufforderungen sind nachstehend in der Reihenfolge Ihres Erscheinens aufgeführt und beschrieben. Die jeweilige Standardantwort ist bei Eingabeaufforderungen, die mit Standardantworten arbeiten können, zwischen eckigen Klammern ([]) unmittelbar hinter dem Text der Eingabeaufforderung eingegeben. Die Eingabeaufforderung bezüglich der Eingabemodule verfügt über keinerlei Standardantwort; Sie müssen hier grundsätzlich die entsprechenden Dateinamen eingeben.

Eingabemodule [.OBJ] :

Geben Sie hier die Namen der Eingabemodule ein, die Sie binden möchten. MS-LINK geht standardmäßig davon aus, daß das Suffix jedes eingegebenen Dateinamens .OBJ lautet. Hat eines der von Ihnen ein anderes Dateinamensuffix, dann muß dieses Suffix eigens eingegeben werden. In allen anderen Fällen braucht das Suffix nicht eingegeben zu werden. Die verschiedenen angegebenen Eingabemodulnamen müssen durch Pluszeichen (+) gegeneinander abgegrenzt sein.

Wir erinnern daran, daß MS-LINK Segmente in der vorgefundenen Reihenfolge klassenweise lädt. Sie können sich daher dieser Informationen bedienen, um die Reihenfolge zu bestimmen, in der die Eingabemodule von MS-LINK gelesen werden. Näheres über diese Möglichkeit entnehmen Sie bitte dem Macro-Assembler-Handbuch.

DAS BINDEPROGRAMM MS-LINK

Ausgabedatei [Erster Modulname.EXE] :

Hier müssen Sie einen Namen für die Datei eingeben, die zur Aufnahme des lauffähigen Programms erstellt werden muß, das der Bindelauf erzeugt. Die Ausgabedatei erhält auf jeden Fall das Dateinamensuffix .EXE, auch wenn Sie ein anderes eingeben.

Wenn Sie die Eingabeaufforderung nach dem Namen der Ausgabedatei nicht beantworten, verwendet MS-LINK automatisch den Namen der ersten Datei, die Sie als Antwort auf die Eingabeaufforderung nach den Eingabemodulen benannt haben, als Namen für diese Datei.

Beispiel:

Ausgabedatei [BLAU.EXE] : B:GEHALT/P

Die Eingabe dieser Antwort bewirkt, daß MS-LINK auf der Diskette in Laufwerk B die Ausgabedatei GEHALT.EXE anlegt. Des weiteren wird MS-LINK dann eine Pause einlegen, die es Ihnen ermöglicht, zur Aufnahme der Ausgabedatei eine neue Diskette in Laufwerk B einzulegen.

Listendatei [Name der Ausgabedatei.MAP]

Die Listendatei enthält einen Eintrag für jedes Segment in den Eingabemodulen. Jeder Eintrag enthält außerdem die entsprechende Adresse in der Ausgabedatei.

Standardantwort ist der Dateiname der Ausgabedatei mit dem Standardsuffix .MAP.

Modulbibliotheken [] :

Als gültige Antworten sind bis zu 8 Namen von Modulbibliotheksdateien möglich oder einfach nur ein <RETURN>. (Wenn Sie statt der Eingabe einer Antwort lediglich die RETURN-Taste betätigen, wird dies von MS-LINK so verstanden, daß in der Standardmodulbibliothek gesucht werden soll.) Die Modulbibliotheksdateien müssen grundsätzlich mit dem Bibliotheksprogramm erstellt worden sein. (Nähere Einzelheiten über Bibliotheksdateien finden Sie in dem Abschnitt MS-LIB des Macro-Assembler-Handbuchs.) MS-LINK geht standardmäßig davon aus, daß das Dateinamensuffix der Modulbibliotheksdateien .LIB ist.

Wenn Sie mehrere Bibliotheksdateien eingeben, müssen Sie diese durch Leerzeichen oder Pluszeichen (+) voneinander abgrenzen.

DAS BINDEPROGRAMM MS-LINK

MS-LINK sucht die Modulbibliotheksdateien in der Reihenfolge ab, in der Sie sie benannt haben. Sobald es ein Modul findet, das den Schlüssel zu dem betreffenden Externverweis enthält, verarbeitet MS-LINK dieses Modul als nächstes Eingabemodul.

Kann MS-LINK auf den Disketten in den beiden Laufwerken keine Modulbibliothek finden, dann weist es Sie durch die folgenden Meldungen darauf hin:

**Cannot find library library-name
Type new drive + letter:**

Geben Sie nun die entsprechende Laufwerksangabe ein (zum Beispiel B).

9.8 MS-LINK-BEFEHLSZUSÄTZE

Es stehen Ihnen sieben Befehlszusätze zur Feinsteuerung verschiedener Funktionen von MS-LINK zur Verfügung. Die Befehlszusätze müssen jeweils am Ende einer Antwort auf eine Eingabeaufforderung eingegeben werden, völlig unabhängig von dem für das Starten von MS-LINK verwendeten Verfahren. Sie können alle Befehlszusätze am Ende einer beliebigen Antwort zusammenfassen oder sie auf die Enden mehrerer Antworten verteilen. Wenn Sie mehrere Befehlszusätze am Ende einer Antwort eingeben wollen, müssen die verschiedenen Zusätze jeweils durch einen normalen Schrägstrich (/) voneinander abgegrenzt sein.

DAS BINDEPROGRAMM MS-LINK

Sie können die Namen aller Befehlszusätze bei der Eingabe abkürzen. Sie müssen dabei lediglich darauf achten, daß die in der Abkürzung verwendeten Buchstaben exakt in der Reihenfolge eingegeben werden müssen, wie sie im Namen des Befehlszusatzes vorkommen. Es sind also weder Lücken noch verstellte Buchstaben zulässig.

Beispiel:

Zulässig	Unzulässig
/D	/DSL
/DS	/DAL
/DSA	/DLC
/DSALLOCA	/DSALLOCT
/DSALLOCATE	

Mit dem Befehlszusatz /DSALLOCATE können Sie MS-LINK anweisen, alle Daten am oberen Ende des Datensegments zu laden. Im Normalfall lädt MS-LINK alle Daten am unteren Ende des Datensegments. Beim Bindelauf wird der DS-Zeiger auf die niedrigstmögliche Adresse gesetzt, um sicherzustellen, daß das ganze DS-Segment genutzt werden kann. Die Verwendung des Zusatzes /DSALLOCATE in Verbindung mit der Standardvorgabe tiefladen (wenn der Befehlszusatz /HIGH nicht benutzt wird) ermöglicht es dem Benutzeranwendungsprogramm, die freibleibende Speicherkapazität unterhalb des spezifisch der D-Gruppe zugewiesenen Speicherbereichs dynamisch zuzuweisen und trotzdem über denselben DS-Zeiger adressierbar zu bleiben. Diese dynamische Speicherraumzuweisung wird für Pascal- und Fortran-Programme benötigt.

Hinweis

Ihr Anwendungsprogramm kann bis zu 64 kByte Speicher (bzw. den momentan tatsächlich verfügbaren Speicherraum) dynamisch zuweisen, abzüglich jedoch des innerhalb der D-Gruppe zugewiesenen Speicherraums.

/HIGH

Mit dem Befehl /HIGH kann MS-LINK die Ausgabedatei so hoch wie möglich im Speicher laden. Normalerweise lädt MS-LINK die Ausführungsdatei so tief wie möglich im Speicher.

DAS BINDEPROGRAMM MS-LINK

ACHTUNG

Verwenden Sie niemals den Befehlszusatz /HIGH in Pascal- oder Fortran-Programmen.

/LINENUMBERS

Der Befehlszusatz /LINENUMBERS bewirkt, daß MS-LINK in der Listendatei die Zeilennummern und -adressen der hochsprachlichen Anweisungen in den Eingabemodulen einfügt. Im Normalfall werden die Zeilennummern in der Listendatei nicht ausgegeben.

HINWEIS

Es gibt Kompilierer, die Maschinencodemodule ohne Zeilennummern erzeugen. Werden Ausgabedateien eines solchen Compilers verwendet, dann kann Ms-LINK natürlich auch keine Zeilennummern in die Ausgabedatei schreiben.

/MAP

Wenn Sie den Befehlszusatz /MAP eingeben, weisen Sie damit MS-LINK an, alle allgemeinen, in den Eingabemodulen verschlüsselten Symbole in der Listendatei aufzulisten. Ohne diesen Zusatz /MAP führt MS-LINK in der Listendatei lediglich Fehlermeldungen auf (einschließlich der nicht entschlüsselten allgemeinen Symbole).

Die Symbole werden in alphabetischer Reihenfolge aufgeführt. Für jedes Symbol schreibt MS-LINK den Wert und die relative Segmentadresse in der Ausgabedatei in die Liste. Die Symbole werden am Ende der Listendatei aufgeführt.

/PAUSE

Mit Hilfe des Befehlszusatzes /PAUSE können Sie MS-LINK veranlassen, im Bindelauf eine Pause einzulegen, sobald der Befehlszusatz gelesen wird. Normalerweise führt MS-LINK den Bindelauf von Anfang bis zum Ende ohne Unterbrechung durch. Dieser Befehlszusatz ermöglicht es dem Anwender, den Bindelauf zu unterbrechen, um die Disketten auszuwechseln, bevor MS-LINK die Ausgabedatei, d.h. das lauffähige .EXE-Programm ausgibt.

DAS BINDEPROGRAMM MS-LINK

Sobald MS-LINK den Befehlszusatz /PAUSE erkennt, gibt es folgende Meldung aus:

```
About to generate .EXE file
Change disks hit any key
```

Sobald Sie nach dem Wechseln der Disketten eine Taste betätigen, wird der Bindelauf mit der Ausgabe der Ausgabedatei durch MS-LINK fortgesetzt.

ACHTUNG!

Sie dürfen auf keinen Fall die Diskette entfernen, auf die die Listendatei geschrieben werden soll, oder diejenige, die MS-LINK für die VM.TMP-Datei verwendet hat, sofern eine solche erstellt wurde.

/STACK: Zahl

Dieser Befehlszusatz ermöglicht es Ihnen, die Größe des Stapelspeicherbereichs für den Bindelauf zu bestimmen. Die Option Zahl steht für einen beliebigen positiven numerischen Wert (in hexadezimaler Schreibweise) bis zu 65536 Bytes. Wird ein Wert zwischen 1 und 511 eingegeben, dann wählt MS-LINK 512. Wird der Befehlszusatz /STACK nicht verwendet, dann errechnet sich MS-LINK die erforderliche Stapelbereichsgröße für den Bindelauf automatisch.

Zu diesem Zweck sollten alle Compiler und Assembler in den Maschinencodemodulen die Informationen bereitstellen, die es dem Bindeprogramm ermöglichen, die erforderliche Stapelbereichsgröße zu berechnen.

Mindestens eines der Eingabemodule muß eine Stapelzuweisungsanweisung enthalten. Ist dies nicht der Fall, dann zeigt MS-LINK die folgende Fehlermeldung an:

WARNING: NO STACK STATEMENT

/NO

/NO steht für NODEFAULTLYBRARYSEARCH (= Standard-Modulbibliotheksdateien nicht absuchen). Mit Hilfe dieses Befehlszusatzes können Sie MS-LINK anweisen, die Standard-Modulbibliotheken in den Maschinencodemodulen nicht abzusuchen. Wenn Sie zum Beispiel einen Bindelauf mit Maschinencodemodulen in Pascal durchführen möchten, können Sie durch Eingabe des Befehlszusatzes /NO MS-LINK anweisen, die Bibliotheksdatei mit dem Namen PASACL.LIB nicht automatisch nach dem Schlüssel von Externverweisen abzusuchen.

DAS BINDEPROGRAMM MS-LINK

9.9 BEISPIEL EINES MS-LINK-BINDELAUFS

In diesem Beispiel zeigen wir Ihnen die Informationen, die während eines MS-LINK-Bindelaufs am Bildschirm angezeigt werden können.

In Beantwortung der MS-DOS-Eingabeaufforderung schreiben Sie:
LINK

Nun bringt das System die folgenden Meldungen und Eingabeaufforderungen auf den Bildschirm (Ihre Antworten sind jeweils unterstrichen):

Microsoft Object Linker V.2.00
(C) Copyright 1982 by Microsoft Inc.

Object Modules [.OBJ] : EA SYSINIT
Run File [EA.EXE] :
List File [NUL.MAP] : EA/MAP
Libraries [.LIB] : ;

Hinweise:

1. Durch die Angabe von /MAP erhalten Sie sowohl eine alphabetische als auch chronologische Auflistung der allgemeinen Symbole.
2. Wenn Sie auf die Listendatei-Eingabeaufforderung mit PRN antworten würden, würde die Ausgabe auf Ihrem Drucker erfolgen.
3. Wenn Sie den Befehlszusatz /LINE angeben, dann würde MS-LINK Ihnen sämtliche Zeilennummern aller Module auflisten. (Beachten Sie bitte, daß dadurch ein beträchtliches Ausgabevolumen zustandekommen kann!)

DAS BINDEPROGRAMM MS-LINK

4. Durch die Eingabe eines Strichpunkts (;), der mit RETURN abgeschlossen wurde, als Antwort auf die Modulbibliotheken-Eingabeaufforderung bewirken Sie, daß MS-LINK die Modulbibliotheken automatisch absucht. Sobald MS-LINK alle Modulbibliotheken gefunden hat, wird der Speicherbelegungsplan des Bindeprogramms in Form einer Liste von Segmenten in der Reihenfolge ihrer Anordnung innerhalb des Lademoduls ausgegeben. Die Auflistung dürfte in etwa wie folgt aussehen:

Anfang	Ende	Länge	Name
00000H	009ECH	09EDH	CODE
009FOH	01166H	077H	SYSINISEG

Die Informationen in den Spalten Anfang und Ende sind die 20-Bit-Hexadezimaladressen der einzelnen Segmente, bezogen auf Speicherstelle 0. Speicherstelle 0 ist der Anfang des Lademoduls.

Die angezeigten Adressen sind nicht die absoluten Adressen, an denen diese Segmente geladen sind. Nähere Hinweise dazu, wie man feststellt, wo die relative Nulladresse sich tatsächlich befindet und auch wie man die absolute Adresse eines Segments bestimmt, entnehmen Sie bitte dem Macro-Assembler-Handbuch.

DAS BINDEPROGRAMM MS-LINK

Da der Befehlszusatz /MAP eingegeben wurde, bringt MS-LINK die allgemeinen Symbole sowohl nach Namen als auch nach Wert geordnet zur Anzeige.

Zum Beispiel:

ADRESSE	ALLGEMEINE SYMSBOLE NACH NAMEN
009F:0012	PUFFER
009F:0005	MOMENTANE LAGE VON DOS
009F:0011	STANDARDLAUFWERK
009F:000B	GERÄTEVERZEICHNIS
009F:0013	DATEIEN
009F:0009	ENDLAGE VON DOS
009F:000F	SPEICHERKAPAZITÄT
009F:0000	SYSINIT

ADRESSE	ALLGEMEINE SYMSBOLE NACH NAMEN
009F:0000	SYSINIT
009F:0005	MOMENTANE LAGE VON DOS
009F:0009	ENDLAGE VON DOS
009F:000B	GERÄTEVERZEICHNIS
009F:000F	SPEICHERKAPAZITÄT
009F:0011	STANDARDLAUFWERK
009F:0012	PUFFER
009F:0013	DATEIEN

Nähere Informationen zu MS-LINK entnehmen Sie bitte dem Macro-Assembler-Handbuch.



DAS BINDEPROGRAMM MS-LINK

9.10 FEHLERMELDUNGEN IN MS-LINK

Jeder erkannte Fehler bewirkt den Abbruch des Bindelaufs. Wenn Sie die Ursache des Fehlers gefunden und beseitigt haben, müssen Sie MS-LINK neu starten. Die Fehler werden von MS-LINK mit Hilfe folgender Fehlermeldungen angezeigt:

ATTEMPT TO ACCESS DATA OUTSIDE OF SEGMENT BOUNDS, POSSIBLY BAD OBJECT MODULE

Es liegt sehr wahrscheinlich ein fehlerhaftes Eingabemodul vor.

BAD NUMERIC PARAMETER

Der numerische Wert wurde nicht in Ziffern eingegeben.

CANNOT OPEN TEMPORARY FILE

Es ist MS-LINK unmöglich, die vorläufige Datei VM.TMP zu erstellen, weil der Diskettenarbeitsbereich voll ist. Legen Sie bitte eine neue Diskette ein. Entfernen Sie jedoch auf keinen Fall die Diskette, auf die die Listendatei (.MAP-Datei) geschrieben werden soll!

ERROR: DUP RECORD TOO COMPLEX

Der DUP-Satz im Assemblierer-Modul ist zu umfangreich. Vereinfachen Sie den DUP-Satz im Assemblierer-Programm.

ERROR: FIXUP OFFSET EXCEEDS FILED WIDTH

Ein Assemblierer-Befehl bezieht sich auf eine Adresse mit einem kurzen Befehl anstatt eines langen. Berichtigen Sie die Assembliererdatei und assemblieren sie erneut.

INPUT FILE READ ERROR

Es liegt sehr wahrscheinlich eine fehlerhafte Eingabedatei vor.

INVALID OBJECT MODULE

Eines oder mehrere der Eingabemodule ist bzw. sind unzulässig aufgebaut oder unvollständig (was vorkommt, wenn die Assemblierung nicht zu Ende geführt wurde).

SYMBOL DEFINED MORE THAN ONCE

MS-LINK hat zwei oder mehr Module gefunden, die verschiedene Schlüssel für dasselbe Symbol enthalten.

PROGRAM SIZE OR NUMBER OF SEGMENTS EXCEEDS CAPACITY OF LINKER

Der Gesamtumfang darf 384 kByte und die Zahl der Segmente darf 255 nicht überschreiten.

DAS BINDEPROGRAMM MS-LINK

REQUESTED STACK SIZE EXCEEDS 64K

Legen Sie einen Stapelbereich größer oder gleich 64 kByte mit Hilfe des Befehlszusatzes /STACK fest.

SEGMENT SIZE EXCEEDS 64K

64 kByte stellt die Grenze für das Adressensystem dar.

SYMBOL TABLE CAPACITY EXCEEDED

Es wurden zu viele und/oder zu lange Namen eingegeben, so daß die Grenze, die bei ungefähr 25 kByte liegt, überschritten wurde.

TOO MANY EXTERNAL SYMBOLS IN ONE MODULE

Die Grenze liegt bei 256 Externsymbole pro Modul.

TOO MANY GROUPS

Die Grenze liegt bei 10 Gruppen.

TOO MANY LIBRARIES SPECIFIED

Die Grenze liegt bei 8 Modulbibliotheken.

TOO MANY PUBLIC SYMBOLS

Die Grenze liegt bei 1024 allgemeinen Symbolen.

TOO MANY SEGMENTS OR CLASSES

Die Grenze liegt bei 256 Segmenten und/oder Klassen.

UNRESOLVED EXTERNALS: list

Für die aufgeführten Externsymbole konnte in den angegebenen Modulen bzw. Modulbibliotheken kein Schlüssel gefunden werden.

VM READ ERROR

Es handelt sich hier um einen Diskettenfehler, der nicht durch MS-LINK verursacht wurde.

WARNING: NO STACK SEGMENT

Sie haben den Befehlszusatz /STACK eingegeben, aber keines der angegebenen Eingabemodule enthält eine Anweisung zur Zuweisung eines Stapelbereichs.

WARNING: SEGMENT OF ABSOLUTE OR UNKNOWN TYPE

Es liegt entweder ein fehlerhaftes Eingabemodul vor, oder es wurde versucht, Eingabemodule zu binden, die MS-LINK nicht bearbeiten kann (zum Beispiel ein Eingabemodul mit absoluter Adresse).

WRITE ERROR IN TMP FILE

Auf der Diskette ist kein Platz mehr für eine notwendige Erweiterung der VM.TMP-Datei.

WRITE ERROR ON RUN FILE

Wenn diese Meldung erscheint, ist im allgemeinen auf der Diskette kein Platz mehr für die Ausgabedatei.

DEBUG

10. DEBUG

10.1	EINLEITUNG.....	10 - 2
10.2	STARTEN DES DEBUG-PROGRAMMES.....	10 - 3
10.3	BEFEHLSPARAMETER.....	10 - 4
10.4	DEBUG-BEFEHLE.....	10 - 7
	Assemble.....	10 - 8
	Compare.....	10 - 11
	Dump.....	10 - 12
	Enter.....	10 - 14
	Fill.....	10 - 16
	Go.....	10 - 17
	Hex.....	10 - 19
	Input.....	10 - 20
	Load.....	10 - 21
	Move.....	10 - 23
	Name.....	10 - 24
	Output.....	10 - 26
	Quit.....	10 - 27
	Register.....	10 - 28
	Search.....	10 - 30
	Trace.....	10 - 31
	Unassemble.....	10 - 32
	Write.....	10 - 34

Wiedergabe sowie Vervielfältigung dieses Unterlags, Vorwertung und Mitteilung
 ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden. Zuwider-
 handlungen verpflichten zu Schadensersatz. Alle Rechte für den Fall der Patent-
 erteilung oder Gebrauchsmustererteilung vorbehalten.

DEBUG

10.1 E I N L E I T U N G

DEBUG ist ein interaktives Werkzeug, das zum Austesten von Anwender-Software entwickelt wurde. DEBUG erlaubt dem Anwender, binäre Programme zu prüfen, zu ändern und zu testen, ohne das Programm nach einer Änderung neu zu übersetzen.

DEBUG benötigt lediglich 13K Bytes des Speichers und kann mit einem Diskettenlaufwerk arbeiten.

Jeder DEBUG-Befehl kann durch ein "Ctrl""C" beendet (abgebrochen) werden. Das Aufrollen der Bildschirmanzeige kann durch "CTRL" "S" angehalten werden, um das Lesen zu erleichtern. Wenn irgendeine andere Taste gedrückt wird, setzt der Bildschirm das Aufrollen fort.

DEBUG

10.2 STARTEN DES DEBUG - PROGRAMMES

DEBUG kann durch Eingabe von:

DEBUG

gestartet werden.

Das DEBUG-Dienstprogramm wird den Anwender durch Anzeigen eines Bindestriches (-) auffordern, Befehle einzugeben. Da bis jetzt kein Dateiname angegeben wurde, können die DEBUG-Befehle NAME und LOAD verwendet werden, um die zu untersuchende Datei zu bestimmen. (Siehe NAME- und LOAD-Befehl.)

Eine weitere Möglichkeit ist die, den Dateinamen beim Start von DEBUG mit anzugeben:

DEBUG [d:] [Pfad] [Dateiname [parm1] [parm2]

parm 1 und parm 2 sind optionale Parameter.

DEBUG setzt Register und Kennzeichen auf die folgenden Anfangswerte:

Die Segmentregister CS, DS, ES und SS werden auf den ersten angrenzenden Speicherbereich nach dem DEBUG-Programm gesetzt.

Der Befehlszeiger (IP) wird auf den Wert 0100H gesetzt.

Der Stapelzeiger (SP) wird an das Segmentende oder an den unteren Rand des übertragenen Bereiches des Programmgladers gesetzt, je nachdem welcher von beiden niedriger ist.

Die restlichen Register (AX, BX, CX, DX, BP, SI und DI) werden auf Null gesetzt. Wenn Sie eine Dateispezifizierung beim Start von DEBUG vorgeben, enthält das CS-Register die Dateigröße in Bytes. Wenn die Datei größer als 64K ist, ist die Größe in den Registern BX und CX enthalten (der größere Teil in BX).

Die Kennzeichenbits werden gelöscht (Siehe den Registerbefehl.)

Die vorgegebene Disketten-Übergabeadresse wird im Code-Segment auf 80H gesetzt.

DEBUG

10.3 B E F E H L S P A R A M E T E R

Parameter	Definition
Adresse	<p>Alle numerischen Werte sind hexadezimal. Geben Sie eine einteilige oder zweiteilige Bezeichnung in einem der folgenden Formate ein:</p> <p>Eine alphabetische Bezeichnung des Segmentregisters, einen Doppelpunkt und einen Offsetwert:</p> <p>CS:0100</p> <p>Eine Segmentadresse, einen Doppelpunkt und dann einen Offsetwert:</p> <p>4BA:0100</p> <p>Nur einen Offsetwert:</p> <p>100</p> <p>In diesem Fall wird das vorgegebene Segment verwendet. CS ist für die Befehle G, L, T, U und W das vorgegebene Segment. Für alle anderen Befehle ist DS das vorgegebene Segment.</p> <p>BEMERKUNG: Speicherplatzadressen müssen gültig sein, da sonst unvorhersehbare Ergebnisse auftreten.</p>
Byte	Ein Hexadezimalwert, der aus einem oder zwei Zeichen besteht.
Laufwerk	Ein einstelliger Hexadezimalwert, um anzuzeigen, von welchem Laufwerk eine Datei geladen oder auf welches eine Datei geschrieben wird. Die gültigen Werte sind 0-3. Diese Werte bezeichnen die Laufwerke wie folgt: 0=A:, 1=B:, 2=C:, 3=D:.

DEBUG

Dateiname Ein Dateiname, der eine Laufwerksbezeichnung und eine Dateinamenerweiterung enthalten kann. (Siehe den Befehl Name.)

Liste Ein oder mehrere Werte für Byte- und/oder Zeichenfolgen. Zum Beispiel:

ECS:100 42 45 52 54 41

Bereich Adresse Adresse

Ein Bereich kann durch eine untere und obere Grenzadresse beschrieben werden.
Zum Beispiel:

DS:510 590

Adresse L Wert

Ein Bereich kann auch durch eine untere Grenzadresse und eine Hexadezimalzahl zur Vorgabe der enthaltenen Bytes spezifiziert werden.
Zum Beispiel:

CS:400 L 11

BEMERKUNG: Die Bereichsgrenze ist hex 10000. Um den Wert von 10000 hex durch vier Hexadezimalzeichen vorzugeben, geben Sie 0000 (oder 0) ein.

Registername Siehe Registerbefehl.

„Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet. Soweit nicht aus dem Zusammenhang ersichtlich, sind alle Rechte vorbehalten.“
„Reproduction, distribution, diffusion, communication, vente ou utilisation sous quelque forme que ce soit, sans l'autorisation écrite préalable de Nixdorf Computer AG, est formellement interdite.“

DEBUG

- Sector** Ein 1- bis 3-stelliger Hexadezimalwert, den Sektor auf der Diskette relativ zum Diskettenanfang oder die Anzahl der Diskettensektoren anzugibt, die gelesen oder geschrieben werden sollen.
- String** Eine beliebige Anzahl von Zeichen, die in Anführungsstriche eingeschlossen ist. Es können einzelne (') oder doppelte (") Anführungsstriche sein. Wenn beim Begrenzer Anführungsstriche innerhalb einer Zeichenfolge erscheinen, müssen die Anführungsstriche doppelt sein. Die folgenden Zeichenfolgen sind zum Beispiel gültig:
- ```
'This is a "string" is okay'.
'This is a 'string' is okay'.
```
- Diese Zeichenfolge ist jedoch ungültig:
- ```
'This is a 'string' is not'.
```
- Ähnliche Zeichenfolgen sind gültig:
- ```
"This is a 'string' is okay".
"This is a ""string"" is okay".
```
- Wert** Entweder ein, bis zu 4-stelliger, Hexadezimalwert, der verwendet wird, um die Anschlußnummer oder die Anzahl der Wiederholungen eines Befehls vorzugeben.



## DEBUG

### 10.4 DEBUG - BEFEHLE

Die folgenden Anmerkungen sind für alle DEBUG-Befehle gültig:

Ein Befehl wird mit einem einzelnen großen oder kleinen Buchstaben eingegeben. Es können ein oder mehrere Parameter folgen.

Begrenzer werden nur zwischen zwei aufeinanderfolgenden Hexadezimalwerten benötigt. Begrenzer können verwendet werden, um Befehle und Parameter zu trennen. Die folgenden Befehle sind gleichbedeutend:

```
dcS:200 210
d cs:200 210
d,cs:200,210
```

Um einen Befehl abzubrechen, drücken Sie "Ctrl" "Break".

Verwenden Sie die Taste "ENTER", um einen Befehl zu aktivieren.

Um das Aufrollen der Anzeige zu stoppen, verwenden Sie "Ctrl" "NumLock". Drücken Sie irgendeine andere Taste, um das Aufrollen wieder aufzunehmen.

Das Aufforderungszeichen des DEBUG-Programmes ist der Bindestrich (-).

Das DEBUG-Programm befindet sich auf Ihrer Systemdiskette.

## DEBUG

### Assemble

---

**Syntax:** A Adresse

**Zweck:** Übersetzt Anweisungen in Assemblersprache in den Speicher. DEBUG unterstützt die Standard-Assemblersprache für 8086/8087/8088.

**Bemerkungen:** DEBUG akzeptiert numerische Hexadezimal-Eingaben. Alle Assembler-Anweisungen befinden sich an benachbarten Plätzen (Adressen) im Speicher, beginnend bei der vorgegebenen Adresse. Die vorgegebene Adresse ist der Platz, der der letzten Übersetzung folgt oder der Bereich bei CS:0100, falls kein vorhergehender Assembler-Befehl verwendet wurde. Um zum DEBUG-Aufforderungszeichen zurückzukehren, drücken Sie, wenn alle Anweisungen eingegeben wurden, "ENTER".

DEBUG reagiert auf ungültige Anweisungen durch Anzeigen von:

^Error

Alle numerischen Werte werden hexadezimal eingegeben und können mit 1-4 Zeichen eingegeben werden.

Die sich über das Segment hinwegsetzenden Mnemonics sind CS:, DS:, ES: und SS:.

Vorangestellte Mnemonics müssen vor dem Operationcode, auf den sie sich beziehen, eingegeben werden und können auf einer separaten Zeile erscheinen.

Mnemonics, die Zeichenfolgen beeinflussen, müssen die Größe der Zeichenfolge angeben. Zum Beispiel:

MOVSW

um Wort-Zeichenfolgen zu bewegen.

MOVSB

um Byte-Zeichenfolgen zu bewegen.

## DEBUG

Das Mnemonic für "far return" ist retf.

Der Assembler übersetzt automatisch kurze, nahe oder ferne Verzweigungen und ruft, abhängig von der Byte-Distanz, die Zieladresse auf. Mit vorangestelltem NEAR oder FAR kann man sich über diese Festsetzung hinwegsetzen. Zum Beispiel:

```
0100:0500 JMP 502 ;a 2 byte short jump
0100:0502 JMP NEAR 505 ;a 3 byte near jump
0100:0505 JMP FAR 50A ;a 5 byte far jump
```

Das vorangestellte NEAR kann mit NE abgekürzt werden, das vorangestellte FAR nicht.

Die Operanden müssen entweder Wort- oder Byte-Speicherplätze spezifizieren. Der Typ muß mit dem vorangestellten "WORD PRT" oder "BYTE PTR" angewiesen sein. DEBUG akzeptiert die Voranstellungen "WO" und "BY". Zum Beispiel:

```
NEG BYTE PRT 128]
DEC WO S1]
```

Um anzuzeigen, ob ein Operand sich auf einen Speicherplatz oder eine unmittelbaren Operanden bezieht, verwendet DEBUG die Konvention, daß in eckige Klammern eingeschlossene Operanden sich auf Speicherstellen beziehen.

Zum Beispiel:

```
MOV AX,21; Lade AX mit 21H
MOV AX, [21]; Lade AX mit dem Inhalt von
Speicherstelle 21H
```

## DEBUG

Es sind die Pseudo-Anweisungen DB und DW verfügbar. Der DB-Operationscode übersetzt Byte-Werte. Der DW-Operationscode übersetzt Wortwerte. Zum Beispiel:

```
DB 1,2,3,4, "THIS IS AN EXAMPLE"
DB 'THIS IS A QUOTE:" '
DB "THIS IS A QUOTE:' "

DW 1000,2000,3000,"BACH"
```

Für die Operationscodes des 8087 müssen die Voranstellungen WAIT oder FWAIT eindeutig vorgegeben werden. Zum Beispiel:

```
FWAIT FADD ST,ST(3) ; This line will
 ; assemble
 ; an FWAIT
 ; prefix.

LD TBYTE PRT [BX] ; This line will not.
```

Assemble unterstützt alle Befehlsformen für registerindirekte Adressierung. Zum Beispiel:

```
ADD BX,34 [BP+2] . [S1-1]
POP [BP+DI]
PUSH [S1]
```

Es werden alle Operationscode-Synonyme unterstützt. Zum Beispiel:

```
LOOPZ 100
LOOPE 100

JA 200
JNBE 200
```

## DEBUG

### Compare

---

- Syntax:** C Bereich Adresse
- Zweck:** Wird zum Vergleichen zweier Blöcke im Speicher verwendet.
- Bemerkungen:** Bereiche werden verwendet, um zwei Blöcke für den Vergleich auszuwählen. Wenn die Bereiche des Speichers identisch sind, erscheint das Aufforderungszeichen. Wenn sie unterschiedlich sind, werden seine Adressen und sein Inhalt in folgender Form angezeigt:

Adr1 Byte1 Byte2 Adr2

|            |                                                                                  |
|------------|----------------------------------------------------------------------------------|
| Adr1 Byte1 | bezieht sich auf den Platz und den Inhalt des nicht übereinstimmenden Bereiches. |
| Byte2 Adr2 | bezieht sich auf das Byte, das in der Adresse gefunden wurde.                    |

**Beispiel:** C100, 1FF 300  
oder  
C100L100 300

Die 100H-Bytes des Speichers, die bei DS:100 beginnen, werden mit den 100H-Bytes, die bei DS:300 beginnen, verglichen.

## DEBUG

### Dump

---

**Syntax:** D [Adresse]  
oder  
D [Bereich]

**Zweck:** Wird zum Anzeigen des Blockinhaltes des Speichers verwendet.

**Bemerkungen:** Die Hexadezimal- und ASCII-Darstellung des vorgegebenen Blockes wird angezeigt. Die nicht druckbaren Zeichen des ASCII-Teils werden durch einen Punkt (.) dargestellt.

Jede Zeile zeigt 16 Bytes an, wobei sich zwischen dem 8. und 9. Byte jeweils ein Bindestrich befindet.

**BEMERKUNG:** Wenn sich die Startadresse der Ausgabe nicht auf einer Bereichsgrenze befindet, kann die erste Zeile weniger als 8 oder 16 Bytes haben. In diesem Fall beginnt die zweite Zeile der Ausgabe auf einer Bereichsgrenze.

Der Ausgabebefehl besitzt zwei Formatzsätze:

**Zusatz 1**            D Adresse

oder

D Bereich

Der Inhalt wird, beginnend bei der vorgegebenen Adresse, angezeigt.

Wenn keine Adresse angegeben wird, so ist die Startadresse diejenige Adresse, die der letzten (mittels des D-Befehls) angezeigten Adresse unmittelbar folgt. Jedes nachfolgende D zeigt die Bytes an, die den zuletzt angezeigten unmittelbar folgen. Wenn vorher kein D-Befehl ausgeführt wurde, wird als Offsetwert 0100H verwendet.

**DEBUG**

**BEMERKUNG:** Wenn Sie für die Startadresse nur einen Offsetwert angeben, so wird das Segment im DS-Register verwendet.

**Zusatz 2**            **D** Bereich

Der Inhalt des spezifizierten Adressbereiches wird angezeigt.

**Beispiel:**

Wenn Sie den Befehl

**DCS: 100 110**

eingeben, führt DEBUG die Ausgabe in folgendem Format durch:

**04BA:0100 42 45 52 54 41 ... 4E 44 TOM SAWYER**

Wenn Sie den Befehl

**DCS:100 L 20**

eingeben, wird die Anzeige, wie oben beschrieben, formatiert, aber es werden 20H Bytes angezeigt.

## DEBUG

### Enter

---

- Syntax:** E Adresse [Liste]
- Zweck:** Der Befehl Enter kann auf zwei Arten verwendet werden:
- Um den Inhalt von einem oder mehreren Bytes mit den in der Liste enthaltenen Werten zu ersetzen. Begonnen wird bei der vorgegebenen Adresse.
- Zeigt die Bytes an und erlaubt die Durchführung einer Änderung der Reihenfolge nach.
- Bemerkungen:** Wenn die optionale Liste von Werten eingegeben wird, wird der Austausch der Byte-Werte automatisch vorgenommen.

Zum Beispiel,

E ds:100 E3 "abc" 7E

ersetzt den Inhalt des Speicherbereichs von ds:100 bis ds:104 durch E3 61 62 63 7E.

Wenn die Adresse ohne die zusätzliche Liste eingegeben wird, zeigt DEBUG die Adresse und deren Inhalt an und wartet auf Ihre Eingabe. An diesem Punkt können Sie durch den Befehl Enter eine der folgenden Aktionen ausführen:

1. Ein Byte-Wert wird durch den Wert, den Sie eingegeben haben, ersetzt. Geben Sie einfach den Wert hinter dem aktuellen Wert ein. Wenn der eingegebene Wert kein gültiger Hexadezimalwert ist oder wenn mehr als zwei Stellen eingegeben wurden, werden die ungültigen Zeichen oder Extra-Zeichen nicht angenommen.
2. Drücken Sie die lange "SPACEBAR"-Taste, um zum nächsten Byte überzugehen. Um den Wert zu ändern, geben Sie einfach den neuen Wert, wie oben unter (1.) beschrieben, ein. Wenn Sie mit den Abständen jenseits einer 8-Byte-Bereichsgrenze kommen, beginnt DEBUG eine neue Anzeigezeile; am Anfang dieser Zeile steht dabei die Adresse.



## DEBUG

3. Geben Sie einen Bindestrich (-) ein, um zu dem vorangehenden Byte zurückzukehren. Wenn Sie sich entscheiden, ein Byte hinter der aktuellen Position zu ändern, kehrt die aktuelle Position zu dem vorherigen Byte zurück, wenn sie den Bindestrich eingeben. Wenn der Bindestrich geschrieben ist, wird eine neue Zeile begonnen und die Adresse mit ihrem Byte-Wert angezeigt.
  
4. Drücken Sie die Taste "ENTER", um den Eingabebefehl zu beenden. Die Taste "ENTER" kann in jeder Byte-Position gedrückt werden.

### Beispiel:

E cs:100

DEBUG zeigt an:

04BA:0100 EB.\_

Um EBH in 41H zu ändern, geben Sie 41 ein.

04BA:0100 EB.41\_

Um zu den nachfolgenden Bytes zu gelangen, drücken Sie "SPACEBAR". Folgende Anzeige können Sie z.B. sehen:

04BA:0100 EB.41 10. 00. BC.\_

Um hex BC in hex 42 zu ändern, geben Sie 42 ein.

04BA:0100 EB.41 10. 00. BC.42\_

Um hex 10 in hex 6F zu ändern, geben Sie zwei Bindestriche und hex 6F ein.

04BA:0100 EB.41 10. 00. BC.42\_  
 04BA:0102 00.  
 04BA:0101 10.6F\_

Drücken Sie die Taste "ENTER", um den Eingabebefehl zu beenden. Das Aufforderungszeichen Bindestrich (-) wird angezeigt.

## DEBUG

### Fill

---

**Syntax:** F Bereich Liste

**Zweck:** Dieser Befehl wird verwendet, um einen Bereich von Bytes im Speicher durch eine Liste auszutauschen.

**Bemerkungen:** Wenn die Anzahl der Bytes, die im Speicherbereich angegeben wurden, größer als die Anzahl der Bytes in der Liste ist, so wird DEBUG versuchen, die Liste solange zu wiederholen, bis die Speicher-Bytes gefüllt sind. Wenn die Liste größer ist als die Anzahl der Bytes, die im Bereich angegeben wurden, dann wird die Liste gekürzt.

**BEMERKUNG:** Wenn Sie nur einen Offsetwert für die Startadresse des Bereiches spezifizieren, wird das Segment im DS-Register verwendet.

**Beispiel:** F04BA:100 L 5 F8 "ABC" 2D

Die Speicherplätze von 04BA:100 bis 04BA:104 werden mit den 5 Bytes in der Liste aufgefüllt. Die ASCII-Werte der Listenzeichen sind gespeichert. Die Plätze 100-104 enthalten F8 41 42 43 2D.

**DEBUG**

3. Geben Sie einen Bindestrich (-) ein, um zu dem vorangehenden Byte zurückzukehren. Wenn Sie sich entscheiden, ein Byte hinter der aktuellen Position zu ändern, kehrt die aktuelle Position zu dem vorherigen Byte zurück, wenn sie den Bindestrich eingeben. Wenn der Bindestrich geschrieben ist, wird eine neue Zeile begonnen und die Adresse mit ihrem Byte-Wert angezeigt.
4. Drücken Sie die Taste "ENTER", um den Eingabebefehl zu beenden. Die Taste "ENTER" kann in jeder Byte-Position gedrückt werden.

**Beispiel:**

E cs:100

DEBUG zeigt an:

04BA:0100 EB.\_

Um EBH in 41H zu ändern, geben Sie 41 ein.

04BA:0100 EB.41\_

Um zu den nachfolgenden Bytes zu gelangen, drücken Sie "SPACEBAR". Folgende Anzeige können Sie z.B. sehen:

04BA:0100 EB.41 10. 00. BC.\_

Um hex BC in hex 42 zu ändern, geben Sie 42 ein.

04BA:0100 EB.41 10. 00. BC.42\_

Um hex 10 in hex 6F zu ändern, geben Sie zwei Bindestriche und hex 6F ein.

04BA:0100 EB.41 10. 00. BC.42\_  
04BA:0102 00.\_  
04BA:0101 10.6F\_

Drücken Sie die Taste "ENTER", um den Eingabebefehl zu beenden. Das Aufforderungszeichen Bindestrich (-) wird angezeigt.

## DEBUG

### Fill

---

**Syntax:** F Bereich Liste

**Zweck:** Dieser Befehl wird verwendet, um einen Bereich von Bytes im Speicher durch eine Liste auszutauschen.

**Bemerkungen:** Wenn die Anzahl der Bytes, die im Speicherbereich angegeben wurden, größer als die Anzahl der Bytes in der Liste ist, so wird DEBUG versuchen, die Liste solange zu wiederholen, bis die Speicher-Bytes gefüllt sind. Wenn die Liste größer ist als die Anzahl der Bytes, die im Bereich angegeben wurden, dann wird die Liste gekürzt.

**BEMERKUNG:** Wenn Sie nur einen Offsetwert für die Startadresse des Bereiches spezifizieren, wird das Segment im DS-Register verwendet.

**Beispiel:** F04BA:100 L 5 F8 "ABC" 2D

Die Speicherplätze von 04BA:100 bis 04BA:104 werden mit den 5 Bytes in der Liste aufgefüllt. Die ASCII-Werte der Listenzeichen sind gespeichert. Die Plätze 100-104 enthalten F8 41 42 43 2D.

**DEBUG**

Go

---

**Syntax:** G [=Adresse] [Adresse] .....

**Zweck:** Dieser Befehl wird verwendet, um das aktuelle Programm auszuführen. Das Programm stoppt und zeigt Register und Kennzeichen an, wenn es auf einen BREAKPOINT (Anhaltepunkt) auftritt.

**Bemerkungen:** Der Befehl Go hat zwei Formatoptionen:

**Zusatz 1** G [=Adresse]

Wenn Sie keine Anhaltepunkte setzen, führt dieser Zusatz das Programm aus, das Sie austesten.

Wenn Sie das Programm mit jeweils v verschiedenen Parametern austesten, verwenden Sie diesen Zusatz. (Wir verweisen auf den Befehl Name). Wenn Sie [=Adresse] nicht verwenden, vergewissern Sie sich, daß die CS:IP Werte richtig gesetzt sind, bevor Sie den G-Befehl ausführen.

**Zusatz 2** G [=Adresse] [Adresse] .....

Dieser Zusatz hält die Programmausführung an einem Anhaltepunkt an. Der Zustand der aktuellen Register und Kennzeichen wird angezeigt. Bis zu zehn Anhaltepunkte sind erlaubt.

## DEBUG

**BEMERKUNGEN:** Wenn ein Programm normal beendet wurde (die Meldung "Program terminated normally" wurde angezeigt), müssen Sie das Programm neu laden, um es nochmals auszuführen.

Anhaltepunkte dürfen nur bei Adressen gesetzt werden, die das erste Byte eines 8088-Operationscodes enthalten.

Der Stapelzeiger muß gültig sein und zum Befehl Go 6 Bytes zur Verfügung haben.

Wenn Sie nur für einen Anhaltepunkt einen Relativzeiger spezifizieren, wird das Segment in dem CS-Register verwendet.

**Beispiel:**

**GCS:7550**

Das Programm im Speicher wird bis zur Adresse 7750 im CS-Segment ausgeführt. DEBUG zeigt die Register und Kennzeichen an. Um das Programm bei der Anweisung hinter dem Anhaltepunkt wieder aufzunehmen, geben Sie den GO-Befehl ein.

**DEBUG**

**Hex**

---

**Syntax:** H Wert Wert

**Zweck:** Dieser Befehl zeigt die Summe und Differenz der zwei Hexadezimalwerte an.

**Beispiel:** H 0A 8  
0012 0002

Die Hexadezimalsumme von 000A und 0008 ist 0012 und deren Differenz ist 0002.

Wiederholt sich, Verweigerung durch Unklarheit, Verweigerung und Mitteilung  
ihres Inhalts nicht gestattet. Die Rechte für den Fall der Patent-  
erteilung oder Gebrauchsmustererteilung vorbehalten.

## DEBUG

### Input

---

**Syntax:** I Wert

**Zweck:** Von dem durch den Wert bestimmten Anschluss wird ein Byte eingelesen und in hexadezimal angezeigt.

**Beispiel:** 12F8

Wenn das Byte am Anschluß 42H ist, dann zeigt DEBUG an:

42



**DEBUG**

**Load**

---

**Syntax:** L [Adresse [Laufwerk Sektor Sektor]]

**Zweck:** Dieser Befehl wird verwendet, um eine Datei in den Speicher zu laden. Er kann auch verwendet werden, um spezielle Sektoren von einer Diskette zu laden.

**Bemerkungen:** Setzen Sie BX:CX für die Anzahl der Bytes, die gelesen werden sollen. Die Datei muß entweder beim Aufruf von DEBUG mit angegeben werden oder mit dem N-Befehl benannt sein. Beide Möglichkeiten formatieren einen Dateinamen in das richtige, normale Format des Dateisteuerungsblocks bei CS:5C.

Wenn der L-Befehl ohne irgendeinen Parameter geschrieben wurde, lädt DEBUG die Datei, beginnend mit Adresse CS:100, in den Speicher und setzt BX:CX auf die Anzahl der Bytes, die geladen wurden. Wenn der L-Befehl mit einem Adressenparameter geschrieben wurde, beginnt das Laden bei der angegebenen Speicheradresse. Wenn L mit allen Parametern geschrieben wurde, werden absolute Diskettensektoren geladen, keine Datei. Die Sektoren werden von dem angegebenen Laufwerk geladen (die numerische Laufwerksbezeichnung ist 0=A; 1=B; 2=C; usw.); DEBUG beginnt das Laden mit dem ersten angegebenen Sektor und fährt fort, bis die angegebene Anzahl von Sektoren geladen sind.

**Beispiel:** Nehmen wir an, daß die folgenden Befehle eingegeben wurden:

```
A>DEBUG
-NFILE.COM
```

Um FILE.COM zu laden, geben Sie nun ein:

L

## DEBUG

DEBUG lädt die Datei und zeigt dann das Aufforderungszeichen von DEBUG an. Nehmen wir an, daß Sie lediglich Teile einer Datei oder gewisse Sektoren von der Diskette laden wollen. Um dieses durchzuführen, geben Sie folgendes ein:

```
L04BA:100 2 OF 6D
```

DEBUG lädt dann 109 (6D hex) Sektoren, beginnend mit dem Sektor 15 (relativ zum Diskettenanfang) in den Speicher, beginnend bei der Adresse 04BA:0100. Wenn die Aufzeichnung geladen wurde, geht DEBUG einfach wieder zum Aufforderungszeichen (-) über.

Wenn die Datei die Erweiterung .EXE hat, wird sie zu der in der Ladeadresse festgelegten Rubrik für .EXE-Dateien verschoben: die Adressenparameter für .EXE-Dateien werden immer ignoriert. Die Rubrik für .EXE-Dateien selbst wird, bevor sie in den Speicher geladen wird, entfernt. Dadurch ist der Umfang einer .EXE-Datei auf der Diskette anders als der im Speicher.

Die Datei, die Sie mit dem Befehl Name benannt oder beim Start von DEBUG mit angegeben haben, sei eine .HEX Datei. Wenn Sie dann einen L-Befehl ohne Parameter eingeben, wird DEBUG veranlaßt, die Datei von der Adresse an zu laden, die in der .HEX-Datei vorgegeben wurde. Wenn der L-Befehl eine Zusatzadresse enthält, fügt DEBUG, die im L-Befehl festgelegte Adresse zu der, in der .HEX-Datei gefundenen Adresse hinzu, um die Startadresse zum Laden der Datei zu bestimmen.

DEBUG

Move

---

**Syntax:** M Bereich Adresse

**Zweck:** Dieser Befehl wird verwendet, um einen Block vom Speicher, festgelegt durch den Bereich (range) und der Startadresse (address), zu verschieben.

**Bemerkung:** Überlappende Verschiebungen (Verschiebungen, bei denen Teile des Blockes bestehende Adressen überlappen) werden immer durchgeführt, ohne Daten zu verlieren. Adressen, die überschrieben werden könnten, werden zuerst verschoben.

Wenn Sie von höheren Adressen zu niedrigeren Adressen verschieben wollen, beginnen Sie mit dem Verschieben der Blöcke aus den unteren Adressen und fahren bis zu den höchsten Adressen fort. Wenn Sie von unteren Adressen zu höheren Adressen verschieben, beginnen Sie mit der höchsten Adresse und arbeiten sich bis zur niedrigsten Adresse vor. Die Reihenfolge der Verschiebungen ist entscheidend, da der MOVE-Befehl die Daten von einem Bereich in einen anderen in der beschriebenen Reihenfolge kopiert und die neue Adresse überschreibt.

**Beispiel:** MCS:100 110 CS:500

Zuerst verschiebt DEBUG die Adresse CS:110 zur Adresse CS:510, dann DS:10F zu CS:50F und so weiter, bis CS:100 nach CS:500 verschoben wurde. Der DUMP-Befehl kann verwendet werden, um das Ergebnis des MOVE-Befehls zu betrachten.

## DEBUG

### Name

---

- Syntax:** N Dateiname [Dateiname..]
- Zweck:** Dieser Befehl legt den Dateinamen der Datei fest, die später im Zusammenhang mit dem LOAD- und GO-Befehl verwendet wird. Dieser Befehl muß verwendet werden, wenn DEBUG ohne einen festgelegten Dateinamen gestartet wird.
- Bemerkungen:** Alle Dateibezeichnungen und Parameter werden genau so in einem Parameter-Sicherungsbereich eingefügt wie eingegeben, einschließlich Begrenzer. CS:80 enthält die Anzahl der eingegebenen Zeichen.
- Beispiel:**
- NPROGA.EXE
  - L
  - NFILE1.DAT FILE2.DAT
  - G

Das Ergebnis dieser Befehle ist:

NAME legt PROGA.EXE als den Dateinamen fest.  
LOAD lädt PROGA.EXE in den Speicher.  
NAME wird nochmals verwendet, um die von PROGA.EXE verwendeten Parameter festzulegen.  
GO führt PROGA.EXE aus, als wenn FILE1.DAT und FILE2.DAT in der DOS-Befehlsebene eingegeben wurden.

**BEMERKUNG:** Wenn zu diesem Zeitpunkt ein WRITE-Befehl ausgeführt wurde, würde die Datei unter dem Dateinamen FILE1.DAT gespeichert. Um dieses Problem zu umgehen, verwenden Sie den NAME-Befehl vor LOAD oder WRITE.

**DEBUG**

Die Bereiche des Speichers, die durch den NAME-Befehl beeinflusst werden können, sind:

- CS:5C Dateisteuerblock für Datei 1.
- CS:6C Dateisteuerblock, wenn Datei 2 festgelegt wurde.
- CS:80 Zählen der Zeichen im NAME-Befehl.
- CS:81 Eingegebene Zeichen im NAME-Befehl.

Um PROG so auszuführen, wie die folgende Zeile dieses darstellt:

PROG PARM1 PARM2/C

geben Sie ein:

DEBUG PROG.COM  
-NPARM1 PARM2/C  
-G  
-

## DEBUG

### Output

---

**Syntax:**            0 Wert Byte

**Zweck:**            Dieser Befehl überträgt das Byte zu dem Ausgangskanal, der durch den Wert (value) festgelegt wurde.

**Beispiel:**            02F8 4F

Ausgangskanal, der durch den Wert festgelegt wurde.

**DEBUG**

**Quit**

---

- Syntax:** Q
- Zweck:** Dieser Befehl beendet das DEBUG-Programm.
- Bemerkungen:** Bedenken Sie, daß der QUIT-Befehl den DEBUG-Vorgang beendet, ohne das aktuelle Programm zu speicherrn. Der Write-Befehl muß verwendet werden, um das Programm zu speichern, bevor DEBUG beendet wird.
- DEBUG kehrt zur DOS-Befehlsebene zurück.

**Beispiel:** -Q  
A>

## DEBUG

### Register

---

**Syntax:** R [Registername]

**Zweck:** Dieser Befehl kann verwendet werden, um:  
Den Hex-Inhalt irgendeines einzelnen Registers anzuzeigen und zu modifizieren.  
Den Hex-Inhalt aller Register, Kennzeichen und die nächste Anweisung, die ausgeführt werden soll, anzuzeigen.  
Die Kennzeichenvorgaben anzuzeigen und zu modifizieren.

**Bemerkungen:** Wenn kein Registername angegeben wurde, zeigt der R-Befehl den Inhalt aller Register und Kennzeichen an.

Die gültigen Registernamen sind (IP und PC beziehen sich beide auf den Befehlszeiger):

|    |    |    |
|----|----|----|
| AX | BP | SS |
| BX | SI | CS |
| CX | DI | IP |
| DX | DS | PC |
| SP | ES | F  |

Die Kennzeichen sind:

| KENNZEICHENNAME | SETZEN         | LÖSCHEN         |
|-----------------|----------------|-----------------|
| Overflow        | OV             | NV              |
| Direction       | DN (dekrement) | IN (inkrement)  |
| Interrupt       | EI (verfügbar) | DI (gesperrt)   |
| Sign            | NG (negativ)   | PL (plus)       |
| Zero            | ZR (Null)      | NZ (nicht Null) |
| Auxillary Carry | AC             | NA              |
| Parity          | PE (gerade)    | PO (ungerade)   |
| Carry           | CY             | NC              |



**DEBUG**

**Beispiel:** R

DEBUG zeigt alle Register, Kennzeichen und die Anweisung für die aktuelle Speicherstelle an.

RF

DEBUG zeigt sämtliche Kennzeichen an. Um den Wert der Kennzeichen zu ändern, geben Sie lediglich einen gültigen Kennzeichenwert ein. Um den R-Befehl zu verlassen, drücken Sie "ENTER". Die Kennzeichen, für die Sie keine Werte aufgeführt haben, bleiben unverändert.

R BX

Zeigt den Inhalt des BX-Registers an. Um den Inhalt des Registers zu ändern, geben Sie lediglich einen Hexadezimalwert aus 1-4 Zeichen ein. Um den R-Befehl zu verlassen, drücken Sie "ENTER". Wenn Sie keinen neuen Wert eingeben, bleibt der Inhalt des Registers unverändert.

## DEBUG

### Search

---

**Syntax:** S Bereich Liste

**Zweck:** Dieser Befehl wird verwendet, um nach einer bestimmten Zeichenfolge innerhalb eines Speicherbereiches zu suchen.

**Bemerkungen:** Jedesmal wenn eine Übereinstimmung auftritt, wird die Adresse angezeigt. Wenn keine Übereinstimmung gefunden wird, wird ein Bindestrich (-) angezeigt.

**Beispiel:** S CS:100 110 41

Sucht nach hex 41 von der Adresse CS:100 bis CS:110.

Die Adressen der Übereinstimmungen werden angezeigt:

04BA:0210  
04BA:021D

**DEBUG**

**Trace**

**Syntax:** T [=Adresse] [Wert]

**Zweck:** Dieser Befehl führt Anweisungen aus und zeigt den Inhalt aller Register und Kennzeichen, nach jeder Anweisung, an.

**Bemerkungen:** Die Anzahl, die durch value (Wert) vorgegeben ist, bestimmt die Anzahl der auszuführenden Anweisungen. Durch Drücken von "Ctrl" "NumLock" können Sie das Aufführen stoppen, sodaß Sie die Register und Kennzeichen für jede einzelne Anweisung untersuchen können.

**Beispiel:** T

Nehmen wir an, daß die momentane Position 04BA:011A ist. DEBUG wird folgendes anzeigen:

```
AX=0E00 BX=00FF CX=0007 DX=01FF
SP=039D BP=0000 SI=005C DI=0000
DS=04BA ES=04BA SS=04BA CS=04BA
IP=011A NV UP DI NG NZ AC PE NC
04BA:011A CD21 INT 21
```

T=110A 10

DEBUG führt sechszehn Anweisungen (10 in hex) aus. Die Register und Kennzeichen werden für jede Anweisung angezeigt.

Keine Haftung für Schäden, die durch die Benutzung dieses Produkts entstehen. Die Haftung für Schäden, die durch die Benutzung dieses Produkts entstehen, ist ausschließlich Sache des Herstellers. Alle Rechte für den Fall der Patent- erteilung oder Gebrauchsmustererteilung vorbehalten.

## DEBUG

### Unassemble

---

|                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Syntax:</b>      | U [Adresse]<br>oder<br>U [Bereich]                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>Zweck:</b>       | Dieser Befehl wird verwendet, um den betreffenden Speicherinhalt in entsprechende Assembler-Befehle zu übersetzen. Die Adressen- und Hexadezimalwerte werden mit assembler-ähnlichen Anweisungen angezeigt.                                                                                                                                                                                                                                                                                                               |
| <b>Bemerkungen:</b> | <p>Wenn der Inhalt der festgelegten Adressen keine gültigen Befehlscodes enthält, treten Fehler auf.</p> <p>Wenn Sie keinen Bereich (range) von Bytes vorgeben, übersetzt DEBUG 20 hex Bytes. Wenn keine Adresse festgelegt ist, dann ist die vorgegebene Adresse die Adresse hinter der letzten, durch einen U-Befehl, festgelegten. Auf diese Art und Weise ist es möglich, eine fortlaufende Übersetzung durchzuführen.</p> <p>Wenn Sie den Bereich vorgeben, bleiben alle Anweisungen in dem Bereich unübersetzt.</p> |

**NIXDORF**  
**COMPUTER**

**DEBUG**

**Beispiel:**

**U 04BA:0100 L10**

DEBUG zeigt an:

|           |    |    |    |
|-----------|----|----|----|
| 04BA:0109 | 65 | DB | 65 |
| 04BA:010A | 63 | DB | 63 |
| 04BA:010B | 69 | DB | 69 |
| 04BA:010C | 66 | DB | 66 |
| 04BA:010D | 69 | DB | 69 |
| 04BA:010E | 63 | DB | 63 |
| 04BA:010F | 61 | DB | 61 |

Wenn Sie 04BA:0100 0108 eingeben, wird  
DEBUG folgendes anzeigen:

|           |        |                   |
|-----------|--------|-------------------|
| 04BA:0100 | 206472 | AND [SI+72],AH    |
| 04BA:0103 | 69     | DB 69             |
| 04BA:0104 | 7665   | JBE 016B          |
| 04BA:0106 | 207370 | [AND BP+DI+70],DH |

## DEBUG

### Write

---

- Syntax:** W [Adresse [Laufwerk Sektor Sektor]]
- Zweck:** Dieser Befehl wird verwendet, um den Inhalt des festgelegten Speicherbereiches auf das angegebene Laufwerk zu schreiben.
- Bemerkungen:** Das Schreiben auf absolute Sektoren ist sehr gefährlich, weil Sie den Dateiverwalter übergehen. Stellen Sie sicher, daß die festgelegten Sektoren auf der Diskette KEINE Daten enthalten, die Sie behalten möchten.
- Die momentane Datei muß entweder beim Start von DEBUG oder durch den NAME-Befehl bezeichnet werden.
- Wenn W ohne Parameter verwendet wird,
- muß BX:CX die Anzahl der Bytes, die geschrieben werden sollen, enthalten.  
wird die Datei ab CS:100 geschrieben.
- Wenn W nur mit einer Adresse verwendet wird,
- wird die Datei geschrieben, beginnend mit der angegebenen Adresse.
- Wenn W mit Parametern verwendet wird,
- beginnt das Schreiben bei der vorgegebenen Adresse,  
wird die Datei auf das festgelegte Laufwerk geschrieben.  
(Die Laufwerkbestimmung ist numerisch  
0=A:  
1=B:  
2=C: usw.)
- Ein einzelner W-Befehl kann eine maximale Anzahl von 80 (hex) Sektoren schreiben.

**DEBUG**

**Beispiel:**

**W**

DEBUG schreibt die Datei auf die Diskette und zeigt dann das DEBUG-Aufforderungszeichen an.

**WCS:100 1 37 2B**

DEBUG schreibt den Inhalt des Speichers, beginnend bei CS:100. Die Daten werden auf die Diskette in Laufwerk B geschrieben. Insgesamt werden 43 ( 2B hex ) Sektoren geschrieben.





## A N H A N G

### ANHANG TEIL A

#### HINWEISE FÜR BENUTZER VON SYSTEMEN MIT DISKETTEN-EINZELLAUFWERK

##### BEMERKUNG

Die in diesem Teil des Anhangs gegebenen Hinweise sind anlagenspezifisch und können nicht auf allen Gerätefabrikaten verwirklicht werden.

Wenn Sie mit einem System mit Disketten-Einzellaufwerk arbeiten, geben Sie die Befehle genauso ein, wie Sie es mit einem Disketten-Doppellaufwerk tun würden.

Sie arbeiten also mit Ihrem Disketten-Einzellaufwerk so, als ob Sie zwei Laufwerke, und zwar Laufwerk A und Laufwerk B, hätten. Anstatt Laufwerk A und Laufwerk B, d.h., anstatt zweier physisch verschiedener Laufwerke bei einem Computer mit Disketten-Doppellaufwerk verwenden Sie ganz einfach verschiedene Disketten, und zwar eine Diskette A und eine Diskette B, in demselben Laufwerk.

Wenn Sie zum Beispiel Laufwerk B angeben, während die "Laufwerk-A-Diskette" zuletzt benutzt wurde, werden Sie aufgefordert, die Diskette für Laufwerk B einzulegen.

Zum Beispiel:

```
A> COPY COMMAND.COM B:
Insert diskette for drive B:
and strike any key when ready
1 File(s) copied
A>
```

Wenn Sie Laufwerk A angeben, während als letztes die "Laufwerk-B-Diskette" benutzt wurde, werden Sie abermals aufgefordert, die Diskette zu wechseln. Dieses Mal fordert Sie MS-DOS auf, die "Laufwerk-A-Diskette" einzulegen.

Dasselbe Verfahren gilt, wenn ein Befehl aus einem Stapelprogrammabstein ausgeführt wird. MS-DOS wartet, bis Sie die geeignete Diskette angelegt haben und eine beliebige Taste drücken, bevor es mit der Ausführung fortfährt. Zu dem Diskettenwechsel werden Sie zum geeigneten Zeitpunkt aufgefordert.

Wiederholte, sowie Verletzung, dieser Urheberrechte, Verwertung und Mitschulung  
ihres Inhalts nicht gestattet. Soweit nicht ausdrücklich zugestanden, zuwider-  
handlungen verpflichten zu Schadensersatz. Alle Rechte für den Fall der Patent-  
erteilung oder Gebrauchsmusteranmeldung vorbehalten.

## A N H A N G

### HINWEISE FÜR BENUTZER VON SYSTEMEN MIT DISKETTEN EINZELLAUFWERK

Die Laufwerksangabe in der MS-DOS-System-Eingabeaufforderung steht für das Standardlaufwerk, d.h. das Laufwerk, in dem MS-DOS eine Datei sucht, deren Name ohne Laufwerksangabe eingegeben wurde. Die Laufwerksangabe in der System-Eingabeaufforderung steht nicht für die Angabe der zuletzt verwendeten Diskette!

Nehmen wir beispielsweise an, daß Laufwerk A das Standardlaufwerk ist. Wenn der zuletzt durchgeführte Befehl DIR B: war, geht MS-DOS davon aus, daß die "Laufwerk-B-Diskette" sich immer noch in diesem Laufwerk befindet. Dessenungeachtet lautet die System-Eingabeaufforderung nach wie vor A:, weil Laufwerk A immer noch das Standardlaufwerk ist. Wenn Sie also den Befehl DIR eingeben, fordert MS-DOS Sie auf, die Laufwerk-A-Diskette einzulegen, weil Laufwerk A das Standardlaufwerk ist, und Sie in Ihrem DIR-Befehl kein anderes Laufwerk angegeben haben.

## A N H A N G

### ANHANG TEIL B

#### DISKETTENFEHLER

Wenn zu einem beliebigen Zeitpunkt während der Bearbeitung eines Befehls oder Programms ein Diskette- oder ein sonstiger Gerätefehler auftritt, teilt MS-DOS Ihnen dies durch die Ausgabe einer entsprechenden Fehlermeldung am Bildschirm mit. Diese Fehlermeldungen haben grundsätzlich die folgende Form:

<xyz> ERROR WHILE <E/A-Vorgang> ON DRIVE x  
Abort, Ignore, Retry:

In dieser Meldung kann <xyz> durch einen der folgenden Texte ersetzt werden:

WRITE PROTECT  
BAD UNIT  
NOT READY  
BAD COMMAND  
DATA  
BAD CALL FORMAT  
SEEK  
NON-DOS DISK  
SECTOR NOT FOUNT  
NO PAPER  
WRITE FAULT  
READ FAULT  
DISK

Die Zeichenfolge <E/A-Vorgang> kann durch eines der beiden folgenden Wörter ersetzt werden:

READING  
WRITING

Die Laufwerksangabe <x> bezeichnet das Laufwerk, in dem der Fehler festgestellt wurde.

## A N H A N G

Nach der Ausgabe einer solchen Fehlermeldung erwartet MS-DOS, daß Sie eine der drei folgenden Antworten eingeben:

- A      Abbrechen: Damit wird die Ausführung des Programms, für die auf der Diskette gelesen oder geschrieben werden mußte, beendet.
- I      Ignorieren: Dies ist die Anweisung an MS-DOS, den fehlerhaften Sektor zu überspringen und weiterzumachen, so als ob der Fehler nicht aufgetreten wäre.
- R      Wiederholen: Hiermit wird MS-DOS angewiesen, den ganzen Vorgang zu wiederholen. Diese Antwort können Sie eingeben, wenn Sie in der Zwischenzeit die Ursache für den festgestellten Fehler beseitigt haben (wie beispielsweise bei NOT READY oder WRITE PROTECT).

Im allgemeinen empfiehlt es sich zu versuchen, die Situation zu retten, indem man folgende Antworten in dieser Reihenfolge eingibt:

R (Ein neuer Versuch kostet nicht viel!)

A (Wenn W nichts bringt, ist es ratsam, das Programm zu beenden und eine neue Diskette zu verwenden!)

Eine weitere Fehlermeldung könnte unter Umständen auch mit fehlerhaften Lese- bzw. Schreibvorgängen auf der Diskette zusammenhängen, und zwar die folgende:

### FILE ALLOCATION TABLE BAD FOR DRIVE x

Mit dieser Meldung teilt Ihnen MS-DOS mit, daß die im Speicher stehende Kopie einer der Dateispeicherplatztabellen-Zeiger nicht vorhandene Daten enthält. Möglicherweise wurde die Diskette nicht ordnungsgemäß oder überhaupt nicht formatiert, bevor sie benutzt wurde. Wenn der Fehler weiterbesteht, ist die Diskette in diesem Zustand unbrauchbar und muß grundsätzlich neu formatiert werden.

## A N H A N G

### ANHANG TEIL C

#### ANSI-ESCAPE-ZEICHENFOLGEN

##### BEMERKUNG

Die in diesem Teil des Anhangs gegebenen Hinweise sind anlagenspezifisch und können nicht auf allen Gerätefabrikaten verwirklicht werden.

Eine ANSI-ESCAPE-Zeichenfolge ist eine Zeichenfolge, die durch ein ESCAPE-Zeichen bzw. durch eine Betätigung der ESCAPE-Taste eingeleitet wird, und mit deren Hilfe Sie bestimmte Funktionen von MS-DOS steuern können. Insbesondere können Sie mit ihrer Hilfe die Tastenbelegung Ihrer Tastatur ändern, die grafischen Funktionen beeinflussen und die Bewegungen der Schreibmarke steuern.

In diesem Teil des Anhangs möchten wir kurz erläutern, wie die ANSI-ESCAPE-Zeichenfolgen in der Fassung 2.0 von MS-DOS festgelegt sind. Beispiele, die zeigen, wie man mit ANSI-ESCAPE-Zeichenfolgen umgeht, finden Sie am Ende dieses Teils des Anhangs.

##### Hinweise:

1. Wenn Sie keinen besonderen Wert oder den Wert 0 eingeben, dann wird der Standardwert verwendet.
2. Pn steht für einen numerischen Parameter. In diesem Fall ist der Parameter eine mit Hilfe von ASCII-Ziffern anzugebende Dezimalzahl.
3. Ps steht für einen Wahlparameter. Das ist eine beliebige Dezimalzahl, die für den Aufruf einer Unterfunktion eingegeben wird. Es können auch mehrere Unterfunktionen gleichzeitig aufgerufen werden, indem Sie die Parameter durch Strichpunkte gegeneinander abgrenzen.

# A N H A N G

## C.1 Schreibmarkenfunktionen

Mit den folgenden ESCAPE-Zeichenfolgen können Sie die Schreibmarkenposition am Bildschirm beeinflussen.

CUP - Cursor Position = Schreibmarkenposition  
ESC [P1 ; Pc H

HVP - Horizontal & Vertical Position = Waagerechte und senkrechte Position

ESC [P1 ; Pc f

CUP und HVP führen die Schreibmarke auf die durch die Parameter festgelegte Position am Bildschirm. Der erste Parameter bestimmt die Zeilennummer, während der zweite Parameter die Spaltennummer darstellt. Standardwert ist grundsätzlich 1. Wird kein Parameter angegeben, dann wird die Schreibmarke an den Bildschirmfang geführt.

CUU - Cursor Up = Schreibmarke nach oben  
ESC [Pn A

Diese Steuerzeichenfolge bewirkt, daß die Schreibmarke ohne Spaltenwechsel um eine bestimmte Anzahl von Zeilen nach oben bewegt wird. Die Zahl der Zeilen, um die die Schreibmarke nach oben bewegt wird, wird durch den Parameter Pn bestimmt.

Standardwert für Pn ist 1; d.h., wenn Sie Pn nicht eingeben, wird die Schreibmarke grundsätzlich um 1 Zeile nach oben versetzt. Die Steuerzeichenfolge CUU wird nicht berücksichtigt, wenn die Schreibmarke sich bereits in der obersten Zeile des Bildschirms befindet.

CUD - Cursor Down = Schreibmarke nach unten  
ESC [Pn B

Diese Steuerzeichenfolge bewirkt, daß die Schreibmarke ohne Spaltenwechsel um eine bestimmte Anzahl von Zeilen nach unten bewegt wird. Die Zahl der Zeilen, um die die Schreibmarke nach unten bewegt wird, wird durch den Parameter Pn bestimmt. Standardwert für Pn ist 1; d.h., wenn Sie Pn nicht eingeben, wird die Schreibmarke grundsätzlich um 1 Zeile nach unten versetzt. Die Steuerzeichenfolge CUU wird nicht berücksichtigt, wenn die Schreibmarke sich bereits in der obersten Zeile des Bildschirms befindet.

CUF - Cursor Forward = Schreibmarke nach rechts  
ESC [Pn C

**A N H A N G**

Die Steuerzeichenfolge CUF bewirkt, daß die Schreibmarke ohne Zeilenwechsel um eine bestimmte Anzahl von Spalten nach rechts bewegt wird. Die Zahl der Spalten, um die die Schreibmarke nach rechts versetzt wird, ist in Pn festzulegen. Standardwert für Pn ist 1; d.h., wenn Sie Pn nicht eingeben, wird die Schreibmarke grundsätzlich um 1 Spalte nach rechts versetzt. Die Steuerzeichenfolge CUF wird nicht berücksichtigt, wenn die Schreibmarke sich bereits in der äußersten rechten Spalte des Bildschirms befindet.

CUB - Cursor Backward = Schreibmarke nach links  
ESC [Pn D

Die Steuerzeichenfolge CUF bewirkt, daß die Schreibmarke ohne Zeilenwechsel um eine bestimmte Anzahl von Spalten nach links bewegt wird. Die Zahl der Spalten, um die die Schreibmarke nach links versetzt wird, ist in Pn festzulegen. Standardwert für Pn ist 1; d.h., wenn Sie Pn nicht eingeben, wird die Schreibmarke grundsätzlich um 1 Spalte nach links versetzt. Die Steuerzeichenfolge CUB wird nicht berücksichtigt, wenn die Schreibmarke sich bereits in der äußersten linken Spalte des Bildschirms befindet.

DSR - Device Status Report = Gerätezustandsbericht  
ESC [6 n

Das Bedienungsplatzsteuerprogramm überträgt eine CPR-Zeichenfolge (siehe weiter unten), sobald es die Steuerzeichenfolge DSR erhält.

CPR - Cursor Position Report = Schreibmarkenpositionsmeldung  
(vom Bedienungsplatzsteuerprogramm an das Betriebssystem)  
ESC [Pn ; Pn R

Die Steuerzeichenfolge CPR bewirkt die Meldung der momentanen Schreibmarkenposition über die Standardeingabefunktionen. Der erste Parameter stellt die momentane Zeile und der zweite die momentane Spalte dar.

SCP - Save Cursor Position = Speicherung der Schreibmarkenposition  
ESC [s

Die momentane Schreibmarkenposition wird gespeichert. Diese Schreibmarkenposition kann mit der ESCAPE-Steuerzeichenfolge RCP (siehe weiter unten) wiederhergestellt werden.

RCP - Restore Cursor Position = Wiederherstellung der Cursorposition  
ESC [u

Mit dieser Steuerzeichenfolge wird die Schreibmarkenposition auf den Wert zurückgesetzt, den Sie hatte, als das Bedienungsplatzsteuerprogramm die Steuerzeichenfolge SCP empfing.

11

Wechselt diese Beschreibung in einer Nachfolge von Bildschirmen auf, bis Sie ein Bild nicht vollständig zugestimmt. Zurückhandlungen verpflichten zu Schreibersatz. Alle Rechte für den Fall der Patentierung oder Gebrauchsmusteranmeldung vorbehalten.

## A N H A N G

### C.2 LÖSCHFUNKTIONEN

Die folgende ESCAPE-Steuerzeichenfolgen ermöglichen es, Löschfunktionen zu steuern.

ED - Erase Display = Bildschirm löschen  
ESC [2 J

Die Steuerzeichenfolge ED bewirkt, daß der Bildschirm gelöscht wird und die Schreibmarke an den Bildschirmanfang zurückkehrt.

EL - Erase Line = Zeile löschen  
ESC [K

Diese Steuerzeichenfolge bewirkt, daß der Zeileninhalt von der Schreibmarke bis zum Ende der Zeile, einschließlich des Inhalts der Schreibstelle, auf der die Schreibmarke steht, gelöscht wird.

### C.3 BILDSCHIRMBETRIEBSARTEN

Mit den folgenden Steuerzeichenfolgen können Sie die Betriebsart des Bildschirms, d.h. die Darstellungsform von Zeichen am Bildschirm, beeinflussen.

SGR - Set Graphics Rendition = Einstellung der Bildschirmdarstellung  
ESC [Ps ; ... ; Ps m



**A N H A N G**

Die Steuerzeichenfolge SGR ermöglicht über den Parameter (s) die Einstellung der nachstehend beschriebenen Darstellungsformen. Die jeweils eingestellte Bildschirmdarstellungsform bleibt eingestellt, bis die nächste SGR-Steuerzeichenfolge eingegeben wird.

| Parameter | Einstellungsfunktion              |                             |
|-----------|-----------------------------------|-----------------------------|
| 0         | Alle Attribute aus                |                             |
| 1         | Fettdruck ein                     |                             |
| 4         | Unterstreichen ein                | (nur Monochrom-Bildschirme) |
| 5         | Blinken ein                       |                             |
| 7         | Negativdarstellung ein            |                             |
| 8         | Abgedunkelt ein                   | (ISO 6429)                  |
| 30        | Schwarzer Vordergrund             | (ISO 6429)                  |
| 31        | Roter Vordergrund                 | (ISO 6429)                  |
| 32        | Grüner Vordergrund                | (ISO 6429)                  |
| 33        | Gelber Vordergrund                | (ISO 6429)                  |
| 34        | Blauer Vordergrund                | (ISO 6429)                  |
| 35        | Purpurroter (Magenta) Vordergrund | (ISO 6429)                  |
| 36        | Cyanblauer Vordergrund            | (ISO 6429)                  |
| 37        | Weißer Vordergrund                | (ISO 6429)                  |
| 40        | Schwarzer Hintergrund             | (ISO 6429)                  |
| 41        | Roter Hintergrund                 | (ISO 6429)                  |
| 42        | Grüner Hintergrund                | (ISO 6429)                  |
| 43        | Gelber Hintergrund                | (ISO 6429)                  |
| 44        | Blauer Hintergrund                | (ISO 6429)                  |
| 45        | Purpurroter (Magenta) Hintergrund | (ISO 6429)                  |
| 46        | Cyanblauer Hintergrund            | (ISO 6429)                  |
| 77        | Weißer Hintergrund                | (ISO 6429)                  |

SM - Set Mode = Betriebsart einstellen  
 oder ESC [ = Ps h  
 oder ESC [ = h  
 oder ESC [ = 0 h  
 oder ESC [ ? 7 h

Wenn Sie diese Veröffentlichung ohne schriftliche Genehmigung und Mitteilung  
 an den Hersteller zu verwenden, sind Sie nicht ausdrücklich zugestanden. Zusätz-  
 liche Informationen zu Schadensersatz. Alle Rechte für den Fall der Patent-  
 erteilung oder Gebrauchsmustererteilung vorbehalten.



## A N N A N G

Die Steuerzeichenfolge SM ermöglicht es Ihnen, die Bildschirmbreite oder die Bildschirmfarbenkombination wie folgt einzustellen:

| Parameter | Funktion                                   |
|-----------|--------------------------------------------|
| 0         | 40 x 25 schwarz-weiß                       |
| 1         | 40 x 25 Farbe                              |
| 2         | 80 x 25 schwarz-weiß                       |
| 3         | 80 x 25 Farbe                              |
| 4         | 320 x 200 Farbe                            |
| 5         | 320 x 200 schwarz-weiß                     |
| 6         | 640 x 200 schwarz-weiß                     |
| 7         | Automatische Zeilenschaltung am Zeilenende |

RM - Reset Mode = Betriebsart zurücksetzen

- ESC [ = Ps 1
- oder ESC [ = 1
- oder ESC [ = 0 1
- oder ESC [ ? 7 1

Die Parameter für die Steuerzeichenfolge RM sind dieselben wie für die Steuerzeichenfolge SM (Set Mode = Betriebsart einstellen) mit der Ausnahme, daß der Parameter 7 die automatische Zeilenschaltung am Zeilenende zurücksetzen, d.h. ausschalten wird.

Die folgenden Steuerzeichenfolgen zur Tastaturumbelegung sind zwar nicht in den Normen ANSI 3.64-1979 oder ISO 6429 aufgeführt, sind aber mit den Vorschriften dieser Normen kompatibel.

Die Steuerzeichenfolge lautet:

- ESC [ Pn ; Pn ; ... Pn p
  - oder ESC [ "Zeichenfolge" ; p
  - oder ESC [ Pn ; "Zeichenfolge" ; Pn ; Pn ; "Zeichenfolge" ; Pn
- p oder eine beliebige andere Kombination von Zeichenfolgen und Dezimalzahlen.

Der Endcode in der Steuerzeichenfolge (p) ist laut ANSI-Norm 3.64-1979 dem persönlichen Gebrauch des Benutzers vorbehalten. Der erste ASCII-Code in der Steuerzeichenfolge bestimmt, welcher Code belegt werden soll. Die darauffolgenden Zahlen bestimmen die Reihe von ASCII-Codes, die erzeugt werden sollen, wenn diese Taste abgefragt wird. Mit einer Ausnahme: Wenn der erste Code in der Zeichenfolge eine Null ist, dann bedeutet dies, daß der erste und der zweite Code zusammen einen erweiterten ASCII-Code darstellen.

## A N H A N G

Beispiele:

1. Die Taste A/a umbelegen auf Q/q (und umgekehrt):

|     |                 |             |
|-----|-----------------|-------------|
| ESC | [ 6 5 ; 8 1 p   | A wird zu Q |
| ESC | [ 9 7 ; 1 1 3 p | a wird zu q |
| ESC | [ 8 1 ; 6 5 p   | Q wird zu A |
| ESC | [ 1 1 3 ; 9 7 p | q wird zu a |

2. Umbelegung der Taste F10, so daß sie einen mit RETURN abgeschlossenen DIR-Befehl eingibt:

ESC [ 0 ; 6 8 ; " d i r " ; 1 3 p

Die Zeichenfolge 0;68 stellt den erweiterten ASCII-Code für die Taste F10 dar; der Dezimalcode 13 bedeutet RETURN.



# A N H A N G

## ANHANG TEIL D

### WIE SIE IHR SYSTEM KONFIGURIEREN

In vielen Fällen gibt es eine Reihe von anlagenspezifischen Einstellungen für den Einsatz von MS-DOS, die beim Systemstart vorgenommen werden müssen. Zum Beispiel die Installation eines Standard-Gerätesteuerprogramms für den Systemdrucker.

Die MS-DOS-Konfigurierungsdatei (CONFIG.SYS) ermöglicht es Ihnen, Ihr System mit geringstmöglichen Aufwand zu konfigurieren. Mit Hilfe dieser Datei können Sie beim Systemstart beliebige Gerätesteuerprogramme in Ihrem System installieren. Die Konfigurierungsdatei ist ganz einfach eine ASCII-Datei mit einer Reihe von Befehlen für das Umladen von MS-DOS. Der Umladevorgang läuft wie folgt ab:

1. Der Diskettenurladesektor wird gelesen. Dieser Sektor enthält Codes, die das Lesen des MS-DOS-Codes und des BIOS (maschinenabhängiger Code) der Anlage ermöglichen.
2. Nun werden der MS-DOS-Code und der BIOS-Code gelesen.
3. Eine Reihe von BIOS-Initialisierungen findet statt.
4. Ein Systeminitialisierungsunterprogramm liest die Konfigurierungsdatei (CONFIG.SYS), falls eine solche vorhanden ist, und führt die darin veranlaßte Installation von Gerätesteuerprogrammen und anderen Benutzeroptionen durch. Abschließend führt das Unterprogramm den Befehlsinterpreter aus, der dann seinerseits den MS-DOS-Umladevorgang beendet.

#### D.1 WIE MAN DIE KONFIGURATIONSDATEI CONFIG.SYS ÄNDERT

Wenn sich auf Ihrer MS-DOS-System-Diskette keine Konfigurierungsdatei mit dem Namen CONFIG.SYS befindet, können Sie eine solche Datei mit Hilfe des MS-DOS-Zeilen-Editors EDLIN erstellen. Speichern Sie diese Datei, sobald Sie sie erstellt haben, in Ihrem Grundarbeitsverzeichnis auf der MS-DOS-System-Diskette.

## AN H A N G

### WIE SIE IHR SYSTEM KONFIGURIEREN

Im folgenden führen wir die Befehle auf, die Sie in der Konfigurationsdatei CONFIG.SYS verwenden können:

**BUFFERS: = <Zahl>**

Dies ist die Zahl der Sektorpuffer, die die Systemtabelle enthalten. Sie ist anlagenspezifisch. Wenn sie noch nicht vorgegeben ist, wäre 10 ein vernünftiger Wert.

**FILES = <Zahl>**

Dies ist die Zahl der offenen Dateien, auf die die XENIX-Systemaufrufe zugreifen können. Auch sie ist anlagenspezifisch. Falls sie nicht vorgegeben ist, wäre auch hier 10 ein vernünftiger Wert.

**DEVICE = <Dateiname>**

Hiermit wird das Gerätsteuerprogramm aus der Datei <Dateiname> in der Systemtabelle installiert (siehe weiter unten).

**BREAK = <ON oder OFF>**

Wenn hier ON angegeben ist (Standardvorgabe ist OFF), wird bei jedem Systemaufruf geprüft, ob CONTROL-C eingegeben wurde. ON verbessert die Möglichkeit, laufende Programme abzubrechen gegenüber älteren Versionen von MS-DOS.

**SHELL = <Dateiname>**

Mit diesem Befehl wird die Ausführung des Hauptbefehlsprozessors aus µDateiname<sup>o</sup> gestartet.

Eine typische Konfigurationsdatei sieht in etwa wie folgt aus:

```
Buffers = 10 Files = 10 Device = \BIN \NETWORK.SYS
BREAK = ON Shell = A: BIN COMMAND.COM /P
```

Beachten Sie, daß hier die Parameter Buffers und Files auf 10 gesetzt sind. Das System-Initialisierungsprogramm sucht nach dem Dateinamen \BIN\NETWORK.SYS, um das Gerät herauszufinden, dessen Steuerprogramm im System installiert werden soll. Diese Datei wird normalerweise mit dem betreffenden Gerät auf Diskette geliefert. Vergewissern Sie sich, daß Sie die Datei mit dem Gerätsteuerprogramm in dem im DEVICE-Parameter angegebenen Arbeitsbereich gespeichert haben.

Diese Konfigurationsdatei setzt außerdem den MS-DOS-Befehl EXEC in die COMMAND.COM-Datei auf der Diskette in Laufwerk A im Arbeitsbereich \BIN. Mit A:\BIN wird dem Programm COMMAND.COM mitgeteilt, wo es sich selbst finden kann, wenn es sich von der Diskette erneut einlesen muß. Mit /P wird COMMAND.COM mitgeteilt, daß es das erste im System auszuführende Programm ist, womit sichergestellt ist, daß es den MS-DOS-Befehl EXIT zum Verlassen des Systems ausführen kann.











