

```

*   ***   *****   ***   ***
* *   *   *   *   *   *   *   *
* *   *   *   *   *   *   *
* *****   *   *   *   *   ***
* *   *   *   *   *   *   *   *
* *   *   *   *   *   *   *   *
***   *   *   *****   ***   ***

```

```

*****
*
*
**
*
*
**
**
**

```

```

***** **
*   *
***** * *
*   *
*   *
*   *
*   *
***** **

```

INTERNE KOMMANDOS

Im vorliegenden Kapitel werden die internen Kommandos von JADOS genau beschrieben. Alle Kommandos sind so aufgebaut, daß sie ohne Parameter aufgerufen werden können. Benötigte Parameter werden vom Anwender interaktiv abgefragt. Der erfahrenere Benutzer, der die Kommandos und deren Möglichkeiten genau kennt, wird die Parameter in der Regel gleich in der Kommandozeile mit angeben.

Zur Nomenklatur sei gesagt, daß optionale Parameter in eckige Klammern gesetzt werden, während Alternativen durch einen senkrechten Strich gekennzeichnet werden. Optionale Parameter werden auch im Interaktivmodus nicht abgefragt.

[...] = Optionaler Parameter

... ; ... = Alternative

1) 0: .. 4: und A: .. Z:

Durch Angabe eines Laufwerksbezeichners mit nachfolgendem Doppelpunkt wird das bezeichnete Laufwerk zum aktuellen Laufwerk. Alle auf diesem Laufwerk vorhandenen Dateien können von da ab ohne Angabe des Laufwerksbezeichners erreicht werden.

Der Bezeichner wird stets im Systemprompt angezeigt.

Es existieren folgende Gruppen von Laufwerken:

0:	-->	Ramdisk
1: bis 4:	-->	Diskettenlaufwerk
A: bis Z:	-->	Festplattenpartition

Falls das gewünschte Laufwerk nicht vorhanden oder ungültig ist, wird eine Fehlermeldung ausgegeben:

Laufwerk n ungültig

wobei n stellvertretend für die oben angegebenen Bezeichner steht.

Nach einem Umladevorgang ist das Laufwerk aktuell, von dem der Bootvorgang stattgefunden hat. Dies ist entweder ein Diskettenlaufwerk oder die Festplattenpartition A.

Nach einem Warmstart ist in der Regel das Laufwerk 1 aktuell. In der Datei CONFIG.SYS kann aber festgelegt werden, daß ein anderes Laufwerk aktuell sein soll.

2) `ass filename [programe]`

ASS ist die Kurzform für ASSEMBLE. Mit diesem Kommando wird der im Grundprogramm vorhandene 68008/68000/68020-Assembler aufgerufen. Hierbei werden eine oder mehrere Dateien mit Assemblerquelltext in den Hauptspeicher (RAM) geladen und vom Assembler in binären Maschinencode übersetzt. Das Ergebnis des Assemblerlaufs ist ein Programm, welches als Datei abgespeichert wird.

Funktionsablauf:

Eine oder mehrere Textdateien werden in den Hauptspeicher geladen. Die Adresse, ab der die Texte abgelegt werden, befindet sich 16 KByte hinter dem Beginn des Userbereichs, welcher davon abhängt, ob im System eine Bankbootkarte aktiv ist. Die genauen Adreßangaben können im Kapitel "Einblick" nachgelesen werden. Vor dem Assemblerlauf werden die Texte im Speicher so weit wie möglich nach hinten geschoben. Dadurch wird Platz für den Maschinencode geschaffen, der ab dem Beginn des Userbereichs abgelegt wird. Da der Assembler aber hinter die Quelltexte noch seine eventuell benötigten Makrotabellen legt, wird zwischen dem Ende der Texte und dem Ende des Userbereichs noch ein Sicherheitsabstand von 8 KByte belassen. Dieser Wert läßt sich in der Datei CONFIG.SYS aber neu festlegen. Nach dem Verschieben wird der Assembler gestartet. Hierbei wird die Option "DEBUG INFO" abgeschaltet. Ausgaben auf den Bildschirm erfolgen nur im Fehlerfall. Sind während der Übersetzung Fehler aufgetreten, so wird der fehlerhafte Maschinencode NICHT abgespeichert. Nach fehlerfreier Übersetzung wird der Maschinencode als Programm abgespeichert. Das Programm hat in der Regel den Dateityp "68K". Zugelassen sind aber auch die Endungen "COM", "SYS" und "OBJ". Die letztere Endung ist bereits im Hinblick auf den Linker erlaubt, welcher demnächst von Klaus Rumrich angeboten wird. Alle anderen Dateieendungen werden automatisch als "68K" gewertet.

Um die Aufteilung der Speicheradressen nicht durcheinander zu bringen, dürfen die Assembleranweisungen "ORG" und "OFFSET" NICHT benutzt werden.

Die während der Assemblierung benötigte Symboltabelle liegt hinter dem Grundprogramm. Es ist dafür zu sorgen, daß hinter dem Grundprogramm genug RAM vorhanden ist. Mehr als 64 KByte Symboltabelle kann das Grundprogramm allerdings zur Zeit nicht verwalten.

Parameter:

Der erste Parameter "filename" bezeichnet die zu übersetzende Quelltextdatei. Falls dem Dateinamen das Zeichen "@" vorangestellt wird, so wird die Datei als Antwortdatei gewertet. Die Antwortdatei enthält die Namen aller Quelltextdateien, die assembliert werden sollen. Pro Zeile darf nur ein Dateiname angegeben werden. Während des Ladens werden die Namen der Textdateien und dahinter die aktuelle Speicheradresse angezeigt. Die Antwortdatei wird mit dem normalen Editor erzeugt. Der Name der Antwortdatei ist beliebig, ebenso der Dateityp.

Der zweite Parameter kennzeichnet den Namen der Programmdatei unter dem der vom Assembler erzeugte Maschinencode gespeichert werden soll. Wird der Programmname nicht angegeben, so wird er aus dem ersten Parameter gebildet, wobei das eventuell vorhandene Zeichen "@" abgezogen wird. Als Dateitypen sind "68K", "COM", "SYS" und "OBJ" zugelassen. Alle anderen Endungen oder wenn keine Endung angegeben wird, wird als "68K" angenommen.

Werden keine Parameter angegeben, so wird das Kommando im Interaktivmodus betrieben. Nach der Meldung

A S S E M B L E R

!! KEINE ORG- ODER OFFSETANWEISUNG BENUTZEN !!
!! STARTADRESSE IST: xxxxxx

wird nach der Textablageadresse gefragt:

Quellprogramme laden
Welche Startadresse ? (Default = \$xxxxxx)

Nach Eingabe der Adresse oder Bestätigung der Defaultadresse wird der Anwender aufgefordert, die Quelltexte zu laden. Dies geschieht über ein kleines Menü:

1 - TEXT LADEN
2 - automatisch
9 - beenden

Durch Eingabe einer *1* wird genau ein Text geladen. Dazu wird der Name der Datei abgefragt:

WELCHE TEXTDATEI LADEN ?

Nach Eingabe des Namens erscheint erneut das obige Menü.

Durch Eingabe einer *2* wird eine Antwortdatei geladen, die die Namen der zu ladenden Quelltextdateien enthält. Der Name der Antwortdatei wird abgefragt:

Name der Antwortdatei ?

Nach Eingabe des Namens werden alle in der Antwortdatei verzeichneten Dateien geladen. Anschließend erscheint wieder obiges Menü.

Durch Eingabe einer *9* wird das Textlademenü beendet. Anschließend erscheint ein Menü, in dem der Anwender die Assemblerausgabe auswählen kann:

Der in Großbuchstaben geschriebene Menüpunkt ist voreingestellt. Er kann direkt durch <RETURN> ausgewählt werden.

A U S G A B E
1 - NUR FEHLER
2 - Bildschirm
3 - Drucker
9 - beenden

Durch Eingabe einer *1* wird festgelegt, daß nur Fehlermeldungen ausgegeben werden.

Durch Eingabe einer *2* wird das Listing auf den Bildschirm ausgegeben. Es kann mit ^S angehalten und mit ^Q fortgesetzt werden. Durch ^C wird der

Assemblerlauf abgebrochen.

Durch Eingabe einer 3 wird das Listing ausgedruckt.

Durch Eingabe einer 9 wird das Assemblerkommando abgebrochen ohne daß eine Übersetzung erfolgt.

Nach einer fehlerfreien Übersetzung wird der Anwender nach dem Programmnamen gefragt:

Namen der Programmdatei (.68K) eingeben:

Danach wird das Programm abgespeichert und das Kommando beendet.

Es können unter anderem folgende Fehlersituationen eintreten.

Parameterfehler

Diese Meldung erscheint wenn in der Kommandozeile ungültige Parameter eingegeben wurden.

LADEFEHLER

Dieser Hinweis taucht auf, wenn eine Quelltextdatei nicht geladen werden konnte.

Falsche ORG-Anweisungen benutzt

Mit diesem Text wird der Anwender darauf aufmerksam gemacht, daß er im Quelltext eine ORG-Anweisung programmiert hat, die die Adreßaufteilung durcheinander gebracht hat. Die ORG-Anweisung ist zu entfernen.

Fehler beim Speichern der Programmdatei

Bei dem Versuch, den Maschinencode als Programmdatei abzuspeichern, ist ein Fehler aufgetaucht. Der Massenspeicher sollte überprüft werden.

Kein Platz mehr im Inhaltsverzeichnis

oder

Kein Platz mehr auf dem Laufwerk

Mit diesen Meldungen wird der Anwender darauf aufmerksam gemacht, daß das verwendete Laufwerk keine freie Kapazität mehr hat. Durch Löschen von nicht mehr benötigten Dateien oder Wechsel auf ein freies Laufwerk kann wieder freie Kapazität gewonnen werden.

Beispiele

ass more.asm

--> übersetzt MORE.ASM und speichert das

Programm unter dem Namen MORE.68K

ass @doslink jados.sys --> übersetzt alle in DOSLINK verzeichneten
Dateien und speichert das Programm
unter dem Namen JADOS.SYS

ass @doslink --> übersetzt alle in DOSLINK verzeichneten
Dateien und speichert das Programm
unter dem Namen DOSLINK.68K

3) bell

Bell ist der englische Ausdruck für das Klingelzeichen. Zwar ist im NDR-Klein-Computer keine echte Klingel eingebaut. Wer jedoch über die Baugruppe SOUND verfügt, kann in den Genuß eines akustischen Signals kommen. JADOS bietet dem Programmierer einige Unterprogramme an, mit der diese Baugruppe angesprochen werden kann.

Auch auf der Ebene des Kommandointerpreters wird die SOUND-Baugruppe unterstützt. Durch Eingabe des obigen Kommandos ertönt ein kurzes, einem Glockenton ähnliches Signal. In Batchdateien kann man damit z.B. die Aufmerksamkeit des Anwenders auf eine Eingabe oder das Ende des Batchlaufs erreichen.

4) `chmod r!w filespec [i]`

Chmod steht für CHANGE MODE oder auf deutsch: Ändere den Dateimodus.

Jede Datei kann einzeln schreibgeschützt oder zum Beschreiben frei gegeben werden. Durch Angabe eines Suchmusters kann auch mehr als eine Datei innerhalb des Kommandos selektiert werden.

Funktionsablauf:

Alle auf das Suchmuster passenden Dateinamen des angegebenen Laufwerks werden in einen Puffer gespeichert. Der gewünschte Dateimodus wird so für jede im Puffer verzeichnete Datei übernommen. Falls der Interaktivmodus eingestellt ist, wird jeder im Puffer vorhandene Dateiname angezeigt und kann vom Anwender bestätigt oder übergangen werden.

Parameter:

Der erste Parameter gibt den neuen Modus an. Dabei gelten die folgenden Bedeutungen:

R	-->	READONLY	= Schreib- und Löschgeschützt
W	-->	WRITE ENABLE	= Überschreib- und Löschar

Wird dieser Parameter nicht in der Kommandozeile angegeben, so wird er vom Benutzer erfragt.

Der zweite Parameter gibt das Suchmuster für die Dateinamen an. Es kann entweder eine einzelne Datei durch Angabe ihres Namens selektiert werden oder eine Gruppe von Dateien durch Angabe von Jokerzeichen. Die Verwendung von Jokerzeichen ist im Kapitel "Bedienung" ausführlich erklärt und wird hier nicht noch einmal wiederholt.

Falls der Parameter nicht in der Kommandozeile eingegeben wurde, wird er vom Anwender abgefragt.

Der dritte Parameter stellt den Interaktivmodus ein. Dieser Parameter ist nur optional und wird nicht abgefragt, falls er vergessen wurde.

Werden keine Parameter angegeben, so werden sie vom Benutzer erfragt. Die Frage nach dem neuen Dateimodus lautet:

Welcher Modus (R oder W) ?

Der Anwender muß darauf mit "R" oder "W" antworten. <ESC> führt zum Abbruch des Kommandos. Andere Zeichen werden zunächst akzeptiert, bei Ausführung des Kommandos aber abgelehnt und führen dann zum Abbruch des Kommandos.

Die Frage nach Dateisuchmuster lautet:

Welche Datei(en) im Modus ändern ?

Hierauf wird das Dateisuchmuster eingegeben. <ESC> führt zum Abbruch des Kommandos. Ist alles korrekt, so werden die selektierten Dateien bearbeitet. Dabei wird jeder Dateiname angezeigt.

Falls der Interaktivschalter gesetzt wurde, so findet ein Dialog mit dem Benutzer statt. Jeder selektierte Dateiname wird angezeigt und es wird gefragt, ob der neue Modus übernommen werden soll:

filename (J/N) ?

Durch quittieren mit <J> <RETURN> wird der Modus übernommen. Dabei wird der Text *geändert* angezeigt. Durch andere Zeichen als <J> oder nur durch <RETURN> wird *NEIN* angezeigt und die Datei übergangen. Durch <ESC> kann das Kommando komplett abgebrochen werden.

Es können die folgenden Fehlersituationen auftreten.

FEHLER: Quelldateien ungültig

Diese Meldung wird ausgegeben wenn das Inhaltsverzeichnis des gewünschten Laufwerks nicht gelesen werden konnte.

Keine Dateien gefunden

Wenn keine Dateien selektiert werden konnten, erscheint obige Meldung.

FEHLER: Kommando abgebrochen

Dieser Fehlertext wird angezeigt wenn der Modus nicht geändert werden konnte. Es liegt hier ein Zugriffsfehler auf das Laufwerk vor.

FEHLER: Modus ist falsch (R oder W)

Wenn der eingegebene Modus undefiniert ist, dann wird diese Meldung angezeigt.

Kommando abgebrochen

Wenn im Interaktivmodus <ESC> eingegeben wurde, dann erscheint dieser Text als Bestätigung.

Beispiele

chmod r *.a

--> Schützt alle Dateien des aktuellen Laufwerks, die vom Typ .A sind

chmod W c:d*.* i

--> Entriegelt alle Dateien des Laufwerks C:, die mit "D" anfangen, interaktiv

5) `cls`

`cls` steht für CLEAR SCREEN oder auf deutsch: Bildschirm "putzen".

Mit diesem Kommando wird der Bildschirm gelöscht. Gleichzeitig wird die Zeichengröße auf 80 Zeichen pro Zeile eingestellt.

6) cold [1|2|3] [drive]

Mit dem Kommando "cold" wird ein sogenannter Kaltstart ausgelöst. Dabei wird eine Grundinitialisierung durchgeführt, innerhalb der sämtliche Variablen initialisiert und alle Puffer gelöscht werden.

Das Kommando ist dann nützlich, wenn Fehlersituationen aufgetreten sind.

Funktionsweise:

Im einfachsten Fall wird die ursprüngliche Startadresse des Betriebssystems angesprungen. Optional kann dabei die Autostartdatei AUTOEXEC.BAT abgearbeitet werden. Es kann aber auch ein kompletter Umladevorgang vom Massenspeicher angewählt werden. Dies ist dann sinnvoll, wenn Teile des JADOS im Speicher zerstört worden sind.

Parameter:

Wird kein Parameter angegeben, dann wird lediglich die Startadresse von JADOS angesprungen.

Der erste Parameter gibt den Kaltstartmodus an. Hierbei wird unterschieden in:

1	-->	Normaler Kaltstart
2	-->	Kaltstart mit Abarbeitung der Datei AUTOEXEC.BAT
3	-->	Umladevorgang vom Massenspeicher

Der zweite Parameter ist nur im Zusammenhang mit dem Umladevorgang wichtig. Wird er nicht explizit angegeben, dann findet die Umladung vom Diskettenlaufwerk 1 statt. Durch Angabe eines Laufwerksbezeichners kann das Laufwerk, von dem gebootet werden soll, gekennzeichnet werden. Erlaubt sind hier die vier Diskettenlaufwerke 1 bis 4 und die Festplattenpartition A.

Fehlermeldungen werden nicht ausgegeben.

Beispiele

cold 2	--> Kaltstart mit Abarbeitung der AUTOEXEC.BAT
cold 3	--> Booten von Diskettenlaufwerk 1
cold 3 a	--> Booten von Festplattenpartition A

7) `copy filespec [destination] [i]`

Mit der Hilfe dieses Kommandos können einzelne Dateien oder ganze Dateigruppen kopiert werden. Die zu kopierenden Dateien werden mit einem Suchmuster selektiert.

Funktionsablauf:

Alle auf das Suchmuster passenden Dateinamen des angegebenen Laufwerks werden in einen Puffer gespeichert. Jede im Puffer verzeichnete Datei wird auf das Ziellaufwerk kopiert. Falls der Interaktivmodus eingestellt ist, wird jeder im Puffer vorhandene Dateiname angezeigt und kann vom Anwender bestätigt oder übergangen werden.

Parameter:

Der erste Parameter gibt das Suchmuster für die Dateinamen an. Es kann entweder eine einzelne Datei durch Angabe ihres Namens selektiert werden oder eine Gruppe von Dateien durch Angabe von Jokerzeichen. Die Verwendung von Jokerzeichen ist im Kapitel "Bedienung" ausführlich erklärt und wird hier nicht noch einmal wiederholt.

Falls der Parameter nicht in der Kommandozeile eingegeben wurde, wird er vom Anwender abgefragt.

Der zweite Parameter gibt das Ziel (destination) an. Hier wird das Ziellaufwerk inklusive dem Zeichen ":" angegeben. Wird der Parameter nicht angegeben, dann wird das aktuelle Laufwerk als Ziellaufwerk genommen; dies wird aber nur akzeptiert, wenn Ziel- und Quellaufwerk unterschiedlich sind. Falls das Suchmuster nur genau eine Datei selektiert, dann kann als Ziel auch ein gültiger Dateiname angegeben werden. Damit kann dann eine Datei kopiert werden und gleichzeitig einen anderen Namen erhalten. Dies ist auch auf demselben Laufwerk möglich.

Der dritte Parameter stellt den Interaktivmodus ein. Dieser Parameter ist nur optional und wird nicht abgefragt, falls er vergessen wurde.

Werden keine Parameter angegeben, so werden sie vom Benutzer erfragt. Die Frage nach dem Suchmuster der zu kopierenden Dateien lautet:

Welche Datei(en) kopieren ?

Hierauf wird das Dateisuchmuster eingegeben. <ESC> führt zum Abbruch des Kommandos. Ist alles korrekt, so werden die selektierten Dateien bearbeitet. Dabei wird jeder Dateiname angezeigt.

Falls der Interaktivschalter gesetzt wurde, so findet ein Dialog mit dem Benutzer statt. Jeder selektierte Dateiname wird angezeigt und es wird gefragt, ob die Datei kopiert werden soll:

filename (J/N) ?

Durch quittieren mit <J> <RETURN> wird die Datei kopiert. Dabei wird der Text *kopiert* angezeigt. Durch andere Zeichen als <J> oder nur durch <RETURN> wird *NEIN* angezeigt und die Datei übergangen. Durch <ESC> kann das Kommando komplett abgebrochen werden.

Es können die folgenden Fehlersituationen auftreten.

FEHLER: Quelldateien ungültig

Diese Meldung wird ausgegeben wenn das Inhaltsverzeichnis des gewünschten Laufwerks nicht gelesen werden konnte.

Keine Dateien gefunden

Wenn keine Dateien selektiert werden konnten, erscheint obige Meldung.

FEHLER: Mehrere Quellen auf ein Ziel

War das Ziel keine Laufwerksangabe sondern ein Dateiname und sind im Suchmuster mehr als eine Datei selektiert dann erscheint obige Meldung.

FEHLER: Zielangabe ungültig

Wenn als Ziel ein ungültiges Laufwerk angegeben wurde, wird der Anwender mit der obigen Meldung darauf hingewiesen.

FEHLER: Quell- und Zieldrive identisch

Falls kein Ziel definiert worden ist, wird das aktuelle Laufwerk als Ziel angenommen. Stimmt dieses mit der Laufwerksangabe der Quelle überein, so erscheint diese Meldung, da ein Laufwerk nicht auf sich selbst kopiert werden kann. Es ist zumindest sinnlos und wird daher nicht akzeptiert.

KOPIERFEHLER: Kommando abgebrochen

Dieser Fehlertext wird angezeigt wenn die Kopie nicht erfolgreich war. Es liegt hier ein Zugriffsfehler auf das Laufwerk vor.

Kommando abgebrochen

Wenn im Interaktivmodus <ESC> eingegeben wurde, dann erscheint dieser Text als Bestätigung.

Beispiele

copy *.a 3:

--> Kopiert alle Dateien des aktuellen Laufwerks, die vom Typ .A sind, auf das Laufwerk 3

copy 2:*.68K

--> Kopiert alle 68K-Dateien des Laufwerks 2: auf das aktuelle Laufwerk

copy more.asm m.a

--> Kopiert die Datei MORE.ASM auf M.A

copy a:*.asm 2: i

--> Sichert alle Dateien vom Typ .ASM von der Festplattenpartition A auf das Diskettenlaufwerk 2, wobei der Interaktivmodus eingeschaltet ist

8) date [newdate]

Mit dem Kommando "date" kann das Systemdatum abgefragt oder eingestellt werden.

Funktionsweise:

Wird kein Parameter angegeben, dann werden das aktuelle Datum und die Uhrzeit eingelesen und im Klartext angezeigt. Wird ein Textstring eingegeben, der eine gültige Datums- und Uhrzeitinformation darstellt, so werden Datum und Zeit auf die im System befindliche Uhr übernommen. Zur Zeit werden die Uhrenkarte und die Smartwatch unterstützt. Wenn keine Uhr vorhanden ist, dann wird das angegebene Datum intern gespeichert. Dies kann nützlich sein, da das Datum des letzten schreibenden Zugriffs auf eine Datei im Inhaltsverzeichnis vermerkt wird.

Parameter:

Die einzustellende Datum- und Uhrzeitinformation muß folgendes Format einhalten um akzeptiert zu werden:

WO, TT.MM.JJ, HH:MM:SS

wobei folgende Bedeutungen gelten:

WO --> Wochentag in Form von genau zwei Zeichen, also SO, MO, DI, MI, DO, FR oder SA

TT --> Tag des Monats in Form von genau zwei Ziffern, also von 01 bis 31

MM --> Monat des Jahres in zwei Ziffern, also von 01 bis 12

JJ --> Jahreszahl in zwei Ziffern, also von 00 bis 99

HH --> Stunde des Tages in der 24-Stunden-Notation in zwei Ziffern, also von 00 bis 23

MM --> Minute in zwei Ziffern, also von 00 bis 59

SS --> Sekunde in zwei Ziffern, also von 00 bis 59

Wenn das eingegebene Format nicht korrekt ist, dann bleibt das Datum unverändert. Es wird dann folgende Meldung angezeigt:

Eingabe FALSCH (Format: WO, TT.MM.JJ, HH:MM:SS)

Beispiele

date --> Fragt Datum und Uhrzeit ab

date di, 07.11.89, 21:34:00 --> Setzt Datum und Uhrzeit

9) di [drive]

"Di" steht für "drive information" oder auf deutsch: Informationen über ein Laufwerk abrufen.

Funktionsweise:

Die Spurtabelle und das Inhaltsverzeichnis des gewünschten Laufwerks werden eingelesen und analysiert. Das Ergebnis der Analyse wird angezeigt. Es umfaßt folgende Angaben:

- Anzahl belegter Spuren
- Anzahl freier Spuren
- Anzahl belegter Einträge im Inhaltsverzeichnis
- Anzahl freier Einträge im Inhaltsverzeichnis
- Anzahl belegter Sektoren
- Anzahl freier Sektoren
- Anzahl freier Bytes (nur der freien Sektoren)

Parameter:

Der einzige Parameter bezeichnet die Nummer des gewünschten Laufwerks. Gültig sind die Bezeichner 0 bis 4 und A bis Z. Es wird kein Doppelpunkt verlangt.

Wird kein Parameter eingegeben, dann wird das aktuelle Laufwerk genommen.

Wenn das eingegebene Laufwerk nicht gültig ist, dann wird folgende Meldung angezeigt:

Laufwerk n ungültig

wobei "n" stellvertretend für die Nummer des Laufwerks steht.

Beispiele

di	--> Angaben über aktuelles Laufwerk abrufen
di a	--> Angaben über Festplattenpartition A abrufen

11) dpath [drive1,drive2,...]

"Dpath" steht für "Drive path" oder auf deutsch: Zugriffspfad für Laufwerke.

Funktionsweise:

Die durch Kommata getrennten Laufwerksbezeichner werden in einen internen Puffer geschrieben. Üblicherweise wird in JADOS ein Programm durch Eingabe seines Namens gestartet. Die zugehörige Programmdatei wird auf dem aktuellen Laufwerk gesucht. Wird dem Programm ein Laufwerksbezeichner gefolgt von ":" vorangestellt, dann wird die Programmdatei auf dem bezeichneten Laufwerk gesucht. Durch Definition eines Zugriffspfades wird das gewünschte Programm zusätzlich noch auf allen Laufwerken gesucht, die im Pfad verzeichnet sind. Die Suche gilt für alle Dateien des Typs .68K, .COM und .BAT.

Parameter:

Die Parameter bestehen aus einer Liste von Laufwerksnummern, die durch Kommata getrennt sind. Leerzeichen in der Liste beenden die Pfadangabe. Die Zahl der Laufwerke im Pfad ist auf acht begrenzt.

Wird kein Parameter eingegeben, dann wird der aktuelle Zugriffspfad angezeigt.

Beispiele

dpath	--> Aktuellen Zugriffspfad anzeigen
dpath 0,a,b,c	--> Ramdisk und Festplattenpartitionen A,B und C in den Zugriffspfad aufnehmen

12) `edit [textfile]`

"Edit" bedeutet, einen Text zu erstellen oder zu überarbeiten.

Funktionsweise:

Im Grundprogramm ist ein einfacher bildschirmorientierter Editor enthalten, der alle notwendigen Funktionen zum Erstellen oder Ändern eines Textes bereithält. JADOS stellt dem Anwender diesen Editor zur Verfügung. Innerhalb des Editors gelten folgende Kommandos:

<code>^E</code> Zeile auf	<code>^X</code> Zeile ab
<code>^S</code> Zeichen links	<code>^D</code> Zeichen rechts
<code>^A</code> Wort links	<code>^F</code> Wort rechts
<code>^Z</code> Scroll auf	<code>^W</code> Scroll ab
<code>^Y</code> Zeile löschen	<code>^N</code> Zeile einfügen
<code>^G</code> Zeichen löschen	<code>^T</code> Zeile rechts löschen
<code>^R</code> Seite zurück	<code>^C</code> Seite vor
<code>^V</code> Einfügemodus	<code>^P</code> Zeichensatz
<code>^U</code> Zeichen einfügen	<code>^B</code> An Zeilenanfang
<code>^QC</code> An Textende	<code>^QR</code> An Textanfang
<code>^QF</code> Suchen	<code>^QA</code> Ersetzen
<code>^KB</code> Blockanfang	<code>^KK</code> Blockende
<code>^KY</code> Block löschen	<code>^KH</code> Blockmarken löschen
<code>^KV</code> Block verschieben	<code>^KC</code> Block kopieren
<code>^KR</code> Block aus Ram lesen	<code>^KW</code> Block ins Ram schreiben
<code>^KS</code> Block auf Disk	<code>^KL</code> Block von Disk holen
<code>^KI</code> Inhaltsverzeichnis	<code>^KT</code> Text auf Disk sichern
<code>^KX</code> Beenden	<code>^KQ</code> Verlassen ohne Speichern
<code>^KA</code> Assembler aufrufen	
<code>ESC A</code> Alter Text	<code>ESC N</code> Neuer Text
<code>ESC L</code> Linker Rand	<code>ESC T</code> Autotab
<code>ESC M</code> Maus	<code>ESC S</code> Scrollmodus
<code>ESC O</code> Ausdruck ohne	<code>ESC P</code> Ausdruck mit

Für eine eingehendere Beschreibung wird auf die Dokumentation zum Grundprogramm verwiesen. Nach Verlassen des Editors mit `^KX` wird der Text auf den Massenspeicher zurückgeschrieben. Soll der Text auf gar keinen Fall zurückgeschrieben werden, dann ist der Editor mit `^KQ` zu verlassen.

Parameter:

Der Parameter gibt den Namen der neu zu erstellenden oder der zu ändernden Textdatei an. Existiert die Datei bereits, dann wird sie geladen und der Editor aufgerufen. Nach Beenden des Editierens mit `^KX` wird der Text unter demselben Namen gespeichert.

Wird kein Parameter angegeben, dann tritt das Kommando in einen kleinen Dialog mit dem Benutzer.

```

--- T E X T E D I T O R ---
1 - neuen Text erstellen
2 - ALTEN TEXT ÄNDERN
3 - Text im RAM ändern
9 - beenden

```

Durch Eingabe von 1 wird ein etwaiger alter im Ram befindlicher Text gelöscht und anschließend der Editor aufgerufen. Nach Beenden mit ^KX wird der Name der neuen Datei erfragt:

Namen der NEUEN Textdatei eingeben:

Unter dem eingegebenen Namen wird dann der Text gespeichert. Durch Eingabe von ESC kann die Speicherung verhindert werden.

Durch Eingabe von 2 oder *Return* kann ein vorhandener Text modifiziert werden. Dazu wird erst nach dem Namen der Datei gefragt:

Welche Textdatei ändern ?

Nach Eingabe des Namens wird der Text geladen und anschließend der Editor aufgerufen. Nach Beenden mit ^KX wird zuerst der Name der ursprünglichen Datei angezeigt und dann nach dem Namen der geänderten Datei gefragt:

Name der URSPRÜNGLICHEN Textdatei: nnnnnnnn.ttt
Namen der geänderten Textdatei eingeben:

Durch ^J kann der Name der ursprünglichen Datei gewählt werden. Durch ESC kann die Speicherung verhindert werden.

Durch Eingabe von 3 wird direkt der Editor aufgerufen. Änderungen können nicht zurückgespeichert werden. Dazu ist das Kommando "TSAVE" zu benutzen.

Durch Eingabe von 9 wird das Kommando beendet, ohne daß der Editor benutzt wird.

Hinweise

Der Editor arbeitet nicht unter der Kontrolle von JADOS. Insbesondere das Speichermanagement ist dem Zugriff von JADOS entzogen. Der Anwender hat selbst dafür zu sorgen, daß die Speicherbereiche des JADOS nicht von dem Text überschrieben werden.

Die Startadresse des Textes liegt dort, wo das Kommando "ASS" die Texte ablegt. Dies ist in der Regel 16 KByte hinter dem Beginn des Userbereichs. Dieser Offset kann in der Datei CONFIG.SYS verändert werden.

Mit dem Kommando "TSAVE" kann ein im Ram befindlicher Text auf den Massenspeicher geschrieben werden. Dies kann in manchen Situationen nützlich sein.

Beispiele

edit more.asm

--> Die Datei MORE.ASM wird geladen,
editiert und wieder gespeichert

13) `era filespec [exec]`

"Era" steht für ERASE oder auf deutsch: Lösche Datei.

Funktionsablauf:

Alle auf das Suchmuster passenden Dateinamen des angegebenen Laufwerks werden in einen Puffer gespeichert. Alle im Puffer verzeichneten Dateien werden vom Massenspeicher gelöscht. Es wird automatisch der sogenannte Interaktivmodus eingestellt. Dabei wird jeder im Puffer vorhandene Dateiname angezeigt und kann vom Anwender bestätigt oder übergangen werden.

Parameter:

Der erste Parameter gibt das Suchmuster für die Dateinamen an. Es kann entweder eine einzelne Datei durch Angabe ihres Namens selektiert werden oder eine Gruppe von Dateien durch Angabe von Jokerzeichen. Die Verwendung von Jokerzeichen ist im Kapitel "Bedienung" ausführlich erklärt und wird hier nicht noch einmal wiederholt.

Falls der Parameter nicht in der Kommandozeile eingegeben wurde, wird er vom Anwender abgefragt.

Der zweite Parameter stellt den Interaktivmodus ab. Dieser Parameter ist nur optional und wird nicht abgefragt, falls er vergessen wurde. Es ist wichtig, daß genau "exec" eingegeben wird.

Werden keine Parameter angegeben, so werden sie vom Benutzer erfragt. Die Frage nach dem Dateisuchmuster lautet:

Welche Datei(en) löschen ?

Hierauf wird das Dateisuchmuster eingegeben. <ESC> führt zum Abbruch des Kommandos. Ist alles korrekt, so werden die selektierten Dateien bearbeitet. Dabei wird jeder Dateiname angezeigt.

Bei gesetztem Interaktivschalter findet ein Dialog mit dem Benutzer statt. Jeder selektierte Dateiname wird angezeigt und es wird gefragt, ob die Datei gelöscht werden soll:

filename löschen (J/N) ?

Durch quittieren mit <J> <RETURN> wird die Datei gelöscht. Dabei wird der Text *gelöscht* angezeigt. Durch andere Zeichen als <J> oder nur durch <RETURN> wird *NEIN* angezeigt und die Datei übergangen. Durch <ESC> kann das Kommando komplett abgebrochen werden.

Es können die folgenden Fehlersituationen auftreten.

Keine Dateien gefunden

Wenn keine Dateien selektiert werden konnten, erscheint obige Meldung.

Kommando abgebrochen

Wenn im Interaktivmodus <ESC> eingegeben wurde, dann erscheint dieser Text als Bestätigung. Auch wenn die Datei nicht gelöscht werden konnte, wird obige Fehlermeldung ausgegeben.

Beispiele

era *.68k

--> Alle .68K-Dateien des aktuellen Laufwerks interaktiv löschen

era *.txt exec

--> Alle .TXT-Dateien des aktuellen Laufwerks löschen ohne Rückfrage beim Anwender

14) `exit`

Mit diesem Kommando wird das JADOS-Betriebssystem verlassen. Der Anwender befindet sich dann wieder in der Bedieneroberfläche des Grundprogramms.

Da JADOS den Stack nach eigenen Bedürfnissen einrichtet, erfolgt der Rücksprung nicht über einen simplen "RTS"-Befehl. JADOS sucht die Startadresse des Grundprogramms und springt diese mit "JMP" an.

Es gibt mehrere Möglichkeiten, aus dem Grundprogramm wieder in das JADOS zu kommen. Der einfachste Weg besteht darin, aus der Bibliothek den Warmstart aufzurufen, da hierbei die residenten Programme und der Kommandopuffer erhalten bleiben. Desweiteren gibt es den Kaltstart aus der Bibliothek sowie den erneuten Umladevorgang.

15) **ff**

"FF" steht für FORM FEED oder auf deutsch: Drucker auf neue Seite einstellen.

Mit diesem Kommando wird der Drucker auf den Anfang einer neuen Seite eingestellt. Fast alle Drucker kennen das hierbei von JADOS verwendete Steuerzeichen "^L". In Batchdateien ist dieses Kommando nützlich, wenn der Anwender mehrere Textdateien mit dem Kommando "PRINT" ausdrucken will, wobei jeder Text auf einer neuen Seite beginnen soll.

16) **gp**

"Gp" steht für GRUNDPROGRAMM. Damit wird das Grundprogramm als Unterprogramm aufgerufen. Es wird die neue Menüform gewählt, bei der alle Möglichkeiten des Grundprogramms auf einer Seite dargestellt sind. Im Grundprogramm kann der Anwender nun beliebig arbeiten. Im Gegensatz zu "EXIT" gelingt der Rücksprung ins JADOS mit dem Grundprogrammkommando "Z".

17) inst [program]

"Inst" steht für "Install program" oder auf deutsch: Programm installieren.

Funktionsweise:

Im Hauptspeicher können sogenannte residente Programme installiert werden. Dies sind normale Programmdateien des Typs 68K oder COM, die dann direkt aus dem Hauptspeicher abgearbeitet werden ohne daß sie vom Massenspeicher geladen werden müssen. Dabei werden die 68K-Dateien vorher aus dem residenten Bereich auf die Startadresse des Userbereichs kopiert. Die COM-Dateien werden direkt in dem Speicherbereich gestartet, in dem sie sich befinden. Dadurch besteht die Möglichkeit, Treiberprogramme zu installieren, die sich z.B. in die Trapvektoren hineinhängen. Die residenten Programme haben nichts mit der Ramdisk zu tun. Gestartet wird ein residentes Programm durch Eingeben seines Namens. Der Lader im JADOS erkennt ein residentes Programm direkt nach den internen Kommandos und vor den 68K-, COM- und BAT-Dateien.

Es können maximal 20 Programme resident geladen werden. Der nötige Speicher verkleinert den Userbereich.

Parameter:

Als Parameter wird der Programmname eingegeben. Dabei muß auch der Dateityp angegeben werden. Falsche Typen werden abgelehnt.

Wird kein Parameter eingegeben, dann wird er erfragt:

Welches Programm installieren ?

Nach erfolgreicher Installation meldet das Kommando:

*Programm installiert
Freier Speicher in KB: xxxx*

Es können die folgenden Fehlermeldungen erscheinen.

Programm nicht vorhanden

Dies erfolgt, wenn die angegebene Datei nicht existiert.

Kein Speicherplatz mehr

Diese Meldung erscheint, wenn der Userspeicher erschöpft ist. Es sollten dann nicht mehr benötigte residente Programme mit "REMOV" entfernt werden. Alternativ sollte über den Ausbau des Speichers nachgedacht werden.

Installationstabelle voll

Wenn zwar genug Speicher vorhanden ist, die Verwaltungstabelle für die residenten Programme aber voll ist, dann wird obige Meldung angezeigt. Die Grenze liegt bei 20 Programmen.

Dies ist KEIN Programm

Wurde eine Datei angegeben, die zwar existiert, aber nicht vom Typ 68K oder COM ist, dann erscheint diese Meldung.

Programm ist schon resident

Dies wird angezeigt, wenn das zu installierende Programm bereits resident ist. Soll ein residentes Programm durch ein neues vom gleichen Namen ersetzt werden, so muß das alte Programm vorher mit "REMOV" entfernt werden.

Beispiele

inst jedi.68k

--> Editor JEDI.68K resident installieren

18) `ldir [filespec] [l;p]`

"Ldir" steht für "Long directory" oder auf deutsch: Ausführliches Inhaltsverzeichnis anzeigen.

Funktionsweise:

Alle auf das Suchmuster passenden Dateinamen des gewünschten Laufwerks werden angezeigt oder ausgedruckt. Dabei werden je Datei die folgenden Angaben ausgegeben:

Dateiname, Dateityp, Dateilänge, Modus und Datum

Parameter:

Der erste Parameter gibt das Suchmuster für die Dateinamen an. Es kann entweder eine einzelne Datei durch Angabe ihres Namens selektiert werden oder eine Gruppe von Dateien durch Angabe von Jokerzeichen. Die Verwendung von Jokerzeichen ist im Kapitel "Bedienung" ausführlich erklärt und wird hier nicht noch einmal wiederholt.

Wird kein Parameter eingegeben, dann werden alle Dateien des aktuellen Laufwerk selektiert (wie *.*).

Der zweite Parameter steuert die Ausgabe. Wird er nicht angegeben, dann wird das Ergebnis angezeigt, ansonsten ausgedruckt.

Wenn das eingegebene Laufwerk nicht gültig ist, dann wird folgende Meldung angezeigt:

Laufwerk n ungültig

wobei "n" stellvertretend für die Nummer des Laufwerks steht.

Beispiele

`ldir c:`

--> Alle Dateien der Festplattenpartition anzeigen

Das Ergebnis kann z.B. wie folgt aussehen:

<i>FORMAT.ASM</i>	15	13.01.89	<i>RW</i>	<i>MORE.ASM</i>	13	17.09.88	<i>RW</i>
<i>DSAVE.ASM</i>	4	11.09.88	<i>RW</i>	<i>1COPY.ASM</i>	6	11.09.88	<i>RW</i>
<i>DISKCOPY.ASM</i>	26	13.01.89	<i>RW</i>	<i>ROMSTART.ASM</i>	12	11.09.88	<i>RW</i>
<i>SYS.ASM</i>	11	04.09.89	<i>RW</i>	<i>BRDRUCK.ASM</i>	72	04.11.89	<i>RW</i>
<i>BANNER.ASM</i>	14	28.10.89	<i>RW</i>	<i>FORMAT.68K</i>	3	13.01.89	<i>RW</i>
<i>BANNER.68K</i>	2	28.10.89	<i>RW</i>	<i>ROMSTART.68K</i>	2	13.09.88	<i>RW</i>
<i>DSAVE.COM</i>	1	13.09.88	<i>RW</i>	<i>1COPY.68K</i>	1	13.09.88	<i>RW</i>
<i>SYS.68K</i>	2	04.09.89	<i>RW</i>	<i>HELP.</i>	4	04.11.89	<i>RW</i>
<i>VERS.ASM</i>	3	30.08.89	<i>RW</i>	<i>VERS.68K</i>	2	30.08.89	<i>RW</i>
<i>DISKCOPY.68K</i>	3	13.01.89	<i>RW</i>	<i>MORE.68K</i>	2	17.09.88	<i>RW</i>
<i>KOMMANDO.JHB</i>	39	09.11.89	<i>RW</i>	<i>BRDRUCK.68K</i>	7	04.11.89	<i>RW</i>
<i>JADOS.BAN</i>	1	04.11.89	<i>RW</i>	<i>HDFORMAT.ASM</i>	14	25.06.89	<i>RW</i>
<i>HDFORMAT.68K</i>	3	25.06.89	<i>RW</i>	<i>HDINIT.ASM</i>	14	28.06.89	<i>RW</i>
<i>HDINIT.68K</i>	3	28.06.89	<i>RW</i>	<i>SRD.ASM</i>	57	30.08.89	<i>RW</i>

`ldir *.asm p`

--> Alle ASM-Dateien des aktuellen Laufwerks ausdrucken

19) **mem**

"Mem" steht für "Memory" oder auf deutsch: Speicherplatz anzeigen.

Der von JADOS verwaltete Speicher ist in mehrere Bereiche eingeteilt. Da gibt es die Ramdisk, das JADOS-Programm nebst Datenspeicher, zwei Stacks und den Bereich der residenten Programme. Der Rest des Speichers ist der Userbereich. Hier laufen die 68K-Programme ab. Der hierfür verbliebene Speicher wird von "MEM" angezeigt.

Freier Speicher in KB: xxx

20) park

Dieses Kommando ist für den Betrieb mit Festplatte nützlich. Die von JADOS unterstützte Festplatte arbeitet mit einem SCSI-Controller. Dieser Controller besitzt eigene Intelligenz. Unter anderem führt er auch eine Statistik über die Bewegungen und aufgetretenen Fehler der Festplatte. Das vorliegende Kommando ruft diese Statistik ab und zeigt sie an. Anschließend werden die Schreib-/Leseköpfe auf die sogenannten Parkspuren bewegt, sodaß sich die Platte gefahrlos abschalten läßt. Es ist zwar nicht schlimm, wenn dies einmal vergessen wird, besser ist aber die regelmäßige Benutzung des Kommandos nach getaner Arbeit.

Die Statistik umfaßt die folgenden Angaben:

```
----- Festplattenstatistik -----  
Anzahl gelesener Sektoren.....: xxxx  
Anzahl der Suchvorgänge.....: xxxx  
Anzahl nicht korrigierter Fehler:   x  
Anzahl korrigierter Fehler.....:   x  
Anzahl der Suchfehler.....:       x
```

Danach folgt die Meldung, daß die Platte geparkt wurde.

Festplatte ist geparkt

Falls keine Festplatte vorhanden ist, erfolgt die Meldung

Es ist keine Festplatte vorhanden. Eventuell DIL-Schalter einstellen

Wenn die Platte zwar vorhanden ist, der Zugriff aber fehlschlägt, dann wird

Fehler beim Festplattenzugriff. Programm abgebrochen.

angezeigt.

21) `pause [...text...]`

"Pause" ist ein Befehl, der für den Einsatz in Batchdateien gedacht ist. Es ist eine interaktive Eingriffsmöglichkeit in den Ablauf der Batchprogramme. Damit ist es möglich, den Anwender darauf hinzuweisen, daß er z.B. eine Diskette wechseln oder den Drucker einschalten soll. Durch Betätigung einer beliebigen Taste läuft anschließend der Batchbetrieb weiter.

Die Meldung des Kommandos lautet

Wenn fertig, bitte eine Taste betätigen ...

22) `print [filename]`

Mit dem Kommando "PRINT" kann eine Textdatei ausgedruckt werden.

Funktionsweise:

Die gewünschte Textdatei wird in den Userbereich geladen. Anschließend wird sie auf den an der Parallelschnittstelle angeschlossenen Drucker ausgegeben. Dabei werden die gedruckten Zeilen mitgezählt. Alle 60 Zeilen erfolgt ein automatischer Seitenvorschub, damit der Text nicht über die Perforation des Endlospapiers gedruckt wird.

Parameter:

Der einzige Parameter ist der Name der zu druckenden Textdatei. Wird der Parameter nicht angegeben, so wird danach gefragt:

Welche Textdatei ausgeben ?

Beispiel

```
print update.doc
```

```
--> Die Datei UPDATE.DOC wird ausge-  
druckt
```

23) `rem [comment]`

"Rem" steht für "Remark" oder auf deutsch: Anmerkung, Kommentar.

Dieser Befehl ist dafür gedacht, in Batchdateien Kommentarzeilen einzustreuen. Der hinter "REM" stehende Text kann beliebig sein.

24) remov [program]

"Remov" steht für "Remov program" oder auf deutsch: residentes Programm entfernen.

Funktionsweise:

Im Hauptspeicher können sogenannte residente Programme installiert werden. Dies sind normale Programmdateien des Typs 68K oder COM, die dann direkt aus dem Hauptspeicher abgearbeitet werden ohne daß sie vom Massenspeicher geladen werden müssen. Ein solches residentes Programm belegt einen Teil des Userbereichs. Außerdem ist die Zahl residenter Programme auf zwanzig begrenzt. Um wieder mehr freien Userbereich zu erhalten, können nicht mehr benötigte residente Programme mit "REMOV" entfernt werden. Dabei werden die eventuell entstehenden Lücken im Speicher durch verschieben wieder geschlossen.

Parameter:

Als Parameter wird der Programmname eingegeben. Dabei muß auch der Dateityp angegeben werden. Falsche Typen werden abgelehnt.

Wird kein Parameter eingegeben, dann wird er erfragt:

Welches Programm DE-Installieren ?

Nach erfolgreicher Entfernung meldet das Kommando:

*Programm entfernt
Freier Speicher in KB: xxxx*

Es können die folgenden Fehlermeldungen erscheinen.

Programmname falsch

Dies erfolgt, wenn der angegebene Name fehlerhaft ist.

Programm nicht resident

Dies wird angezeigt, wenn das zu entfernende Programm gar nicht resident ist.

HINWEIS

Residente Programme werden hauptsächlich als Treiberprogramme benutzt. Wenn hierbei Interruptvektoren "verbogen" wurden, so kann es nach dem Entfernen des Treibers zu Problemen kommen. Die Interruptvektoren zeigen dann in Programmbereiche, die mit dem ursprünglichen Treiber nichts mehr zu tun haben. Ein Rechnerabsturz ist dann sehr wahrscheinlich. Der Treiber sollte daher einen Deinstallationsteil haben, der die Interruptvektoren wieder "gerade biegt".

Beispiel

remov jedi.68k

--> Editor JEDI.68K wird entfernt

25) ren file1 file2

"Ren" steht für "Rename" oder auf deutsch: Datei umbenennen.

Funktionsweise:

Der Name einer bereits existierenden Datei wird geändert. Dieses Kommando gilt nur für einzelne Dateien. Jokerzeichen werden nicht akzeptiert. Auch kann die Änderung des Namens nur auf demselben Laufwerk erfolgen, da lediglich das Inhaltsverzeichnis geändert wird.

Parameter:

Der erste Parameter gibt den ursprünglichen Namen der Datei an. Der zweite Parameter gibt den neuen Namen an.

Nicht angegebene Parameter werden abgefragt.

Welche Datei umbenennen ?

Hierauf muß der ursprüngliche Name eingegeben werden.

Neuer Name ?

Jetzt kann der geänderte Name angegeben werden. Nach erfolgreicher Änderung erscheint die Bestätigung

Datei umbenannt

Fehlermeldungen können die folgenden sein.

Datei nicht gefunden

Die zu ändernde Datei existiert überhaupt nicht.

Dateiname existiert bereits

Es gibt bereits eine Datei, deren Name dem neuen Namen entspricht. Das Kommando wird in diesem Fall abgebrochen.

Umbenennung nur auf GLEICHEM Laufwerk !

Eine Datei kann nur auf demselben Laufwerk umbenannt werden.

Beispiel

```
ren more.asm m.a
```

--> Die Datei MORE.ASM wird in M.A umbenannt

26) `sdir [filespec] [!p]`

"Sdir" steht für "Short directory" oder auf deutsch: Kurzes Inhaltsverzeichnis anzeigen.

Funktionsweise:

Alle auf das Suchmuster passenden Dateinamen des gewünschten Laufwerks werden angezeigt oder ausgedruckt. Dabei werden je Datei die folgenden Angaben ausgegeben:

Dateiname und Dateityp

Parameter:

Der erste Parameter gibt das Suchmuster für die Dateinamen an. Es kann entweder eine einzelne Datei durch Angabe ihres Namens selektiert werden oder eine Gruppe von Dateien durch Angabe von Jokerzeichen. Die Verwendung von Jokerzeichen ist im Kapitel "Bedienung" ausführlich erklärt und wird hier nicht noch einmal wiederholt.

Wird kein Parameter eingegeben, dann werden alle Dateien des aktuellen Laufwerks selektiert (wie *.*).

Der zweite Parameter steuert die Ausgabe. Wird er nicht angegeben, dann wird das Ergebnis angezeigt, ansonsten ausgedruckt.

Wenn das eingegebene Laufwerk nicht gültig ist, dann wird folgende Meldung angezeigt:

Laufwerk n ungültig

wobei "n" stellvertretend für die Nummer des Laufwerks steht.

Beispiele

`sdir c:` --> Alle Dateien der Festplattenpartition C: anzeigen

Das Ergebnis kann z.B. wie folgt aussehen:

<i>FORMAT.ASM</i>	<i>MORE.ASM</i>	<i>DSAVE.ASM</i>	<i>1COPY.ASM</i>	<i>DISKCOPY.ASM</i>
<i>ROMSTART.ASM</i>	<i>SYS.ASM</i>	<i>BRDRUCK.ASM</i>	<i>BANNER.ASM</i>	<i>FORMAT.68K</i>
<i>BANNER.68K</i>	<i>ROMSTART.68K</i>	<i>DSAVE.COM</i>	<i>1COPY.68K</i>	<i>SYS.68K</i>
<i>HELP.</i>	<i>VERS.ASM</i>	<i>VERS.68K</i>	<i>DISKCOPY.68K</i>	<i>MORE.68K</i>
<i>KOMMANDO.JHB</i>	<i>BRDRUCK.68K</i>	<i>JADOS.BAN</i>	<i>HDFORMAT.ASM</i>	<i>HDFORMAT.68K</i>
<i>HDINIT.ASM</i>	<i>HDINIT.68K</i>	<i>SRD.ASM</i>	<i>SRD.COM</i>	<i>HILFE.68K</i>
<i>HILFE.HLF</i>	<i>JADOS.HLF</i>			

`sdir *.asm p` --> Alle ASM-Dateien des aktuellen Laufwerks ausdrucken

28) `table [!p]`

Mit dem Kommando "TABLE" kann das Verzeichnis der residenten Programme angezeigt oder ausgedruckt werden.

Funktionsweise:

Im Hauptspeicher können sogenannte residente Programme installiert werden. Dies sind normale Programmdateien des Typs 68K oder COM, die dann direkt aus dem Hauptspeicher abgearbeitet werden ohne daß sie vom Massenspeicher geladen werden müssen. Ein solches residentes Programm belegt einen Teil des Userbereichs. Außerdem ist die Zahl residenter Programme auf zwanzig begrenzt. Alle residenten Programme werden in einer Tabelle verwaltet. Diese Tabelle kann mit "TABLE" sichtbar gemacht werden.

Parameter:

Der einzige Parameter dient nur dazu, die Ausgabe des Kommandos auf den Drucker umzulenken.

Wird kein Parameter angegeben, dann wird die Ausgabe auf den Bildschirm gelenkt.

Die Anzeige ist z.B. wie folgt:

```
Verzeichnis der residenten Programme
--Name--  Adresse  Typ
DISKCOPY  $69000  68K
DISASS    $66000  68K
```

Beispiel

```
table p
```

```
--> Tabelle der residenten Programme
ausdrucken
```

29) `tload [address] [filename]`

"Tload" steht für "Text load" oder auf deutsch: Eine oder mehrere Textdateien in den Hauptspeicher laden.

Funktionsweise:

Auf die angegebene Hauptspeicheradresse wird eine Textdatei geladen. Werden keine Parameter angegeben, dann können auch mehrere Texte hintereinander geladen werden. Auf diese Weise können mehrere Dateien zu einer einzigen zusammengebunden werden. Mit "TSAVE" kann diese große Datei wieder auf den Massenspeicher gebracht werden. Vorteilhaft ist "TLOAD" dann, wenn ein Programmierer sein Programm mit der Debugoption im Einzelschritt austesten möchte. Er lädt dann die Quelltexte in den Hauptspeicher und ruft dann mit "GP" das Grundprogramm auf. Dort kann er die Debugoption einstellen, assemblieren und in die Einzelschrittverarbeitung gehen.

Parameter:

Der erste Parameter gibt die Hauptspeicheradresse an, ab der die Texte geladen werden sollen. Der zweite Parameter enthält den Namen der Textdatei.

Werden keine Parameter angegeben, dann wird der Interaktivmodus eingeschaltet, der mehr Möglichkeiten bietet.

Als erstes wird nach der Adresse im Hauptspeicher gefragt:

Welche Startadresse ? (DEFAULT = \$xxxxxx)

Die eingestellte Adresse kann mit <RETURN> bestätigt werden oder es wird eine andere Adresse eingegeben. Danach wird der Anwender aufgefordert, die Quelltexte zu laden. Dies geschieht über ein kleines Menü:

- 1 - TEXT LADEN
- 2 - automatisch
- 9 - beenden

Durch Eingabe einer 1 wird genau ein Text geladen. Dazu wird der Name der Datei abgefragt:

WELCHE TEXTDATEI LADEN ?

Nach Eingabe des Namens erscheint erneut das obige Menü.

Durch Eingabe einer 2 wird eine Antwortdatei geladen, die die Namen der zu ladenden Quelltextdateien enthält. Der Name der Antwortdatei wird abgefragt:

Name der Antwortdatei ?

Nach Eingabe des Namens werden alle in der Antwortdatei verzeichneten Dateien geladen. Anschließend erscheint wieder obiges Menü.

Durch Eingabe einer 9 wird das Textlademenü beendet.

Es können unter anderem folgende Fehlersituationen eintreten.

LADEFEHLER

Dieser Hinweis taucht auf, wenn eine Quelltextdatei nicht geladen werden konnte.

Kein Platz mehr im Inhaltsverzeichnis

oder

Kein Platz mehr auf dem Laufwerk

Mit diesen Meldungen wird der Anwender darauf aufmerksam gemacht, daß das verwendete Laufwerk keine freie Kapazität mehr hat. Durch Löschen von nicht mehr benötigten Dateien oder Wechsel auf ein freies Laufwerk kann wieder freie Kapazität gewonnen werden.

HINWEIS

Mit "TLOAD" kann auch eine binäre Datei in den Hauptspeicher gebracht werden. Da binäre Dateien aber den Code Null beinhalten können, ist es nicht möglich, mehrere binäre Dateien zu koppeln.

Beispiele

tload	--> Textladen im Interaktivmodus aufrufen
tload \$10000 more.asm	--> Textdatei MORE.ASM auf die Adresse \$10000 laden.

30) `tsave [filename] [adress]`

"Tsave" steht für "Text save" oder auf deutsch: Einen Text, der im Hauptspeicher steht, auf den Massenspeicher sichern.

Funktionsweise:

Ein Text, der mit einer binären Null abgeschlossen ist, wird aus dem Hauptspeicher auf den Massenspeicher gebracht. Dies gilt nur für Texte, da das Kommando nach der binären Null sucht. Ein Programm kann auf diese Weise nicht auf den Massenspeicher gebracht werden, da die Null sehr häufig vorkommt. Es gibt aber eine Ausnahme. Da JADOS immer Blöcke von 1 KB verwaltet, kann ein Programm dieser Größe gespeichert werden.

Parameter:

Der erste Parameter enthält den Namen der Textdatei. Der zweite Parameter gibt die Hauptspeicheradresse an, ab der der Text gespeichert werden soll.

Werden keine Parameter angegeben, dann werden sie erfragt.

Namen der Textdatei eingeben

und

Welche Startadresse ? (DEFAULT = \$xxxxxx)

Die eingestellte Adresse kann mit <RETURN> bestätigt werden oder es wird eine andere Adresse eingegeben.

Nach erfolgreichem Speichern erscheint die Meldung:

Text abgespeichert

Nach etwaigen Fehlern erscheint der Hinweis:

Text wegen Schreibfehler nicht gespeichert

Beispiele

<code>tsave</code>	--> Text speichern im Interaktivmodus
<code>tsave text.a \$10000</code>	--> Einen Text ab der Speicheradresse \$10000 unter dem Namen TEXT.A auf das aktuelle Laufwerk sichern

31) `type [filename]`

Mit dem Kommando "TYPE" kann eine Textdatei angezeigt werden.

Funktionsweise:

Die gewünschte Textdatei wird in den Userbereich geladen. Anschließend wird sie auf den Bildschirm ausgegeben. Die Ausgabe kann mit `<^S>` angehalten und mit `<^Q>` fortgesetzt werden. Mit `<ESC>` kann die Ausgabe abgebrochen werden, aber nur wenn die Ausgabe nicht vorher gestoppt wurde.

Parameter:

Der einzige Parameter ist der Name der zu druckenden Textdatei. Wird der Parameter nicht angegeben, so wird danach gefragt:

Welche Textdatei ausgeben ?

Beispiel

`type update.doc`

--> Die Datei UPDATE.DOC wird angezeigt

ALPHABETISCHE KOMMANDO – ÜBERSICHT

Kommando	Parameter	Bedeutung	Seite
0:		Umschaltung auf Ramdisk	2
1:..4:		Umschaltung auf Diskettenlaufwerk 1..4	2
A:..Z:		Umschaltung auf Plattenlaufwerk A..Z	2
ASS	FILE [PROG]	Assembler aufrufen	3
BELL		Akustisches Signal	7
CHMOD	R W FILESPEC [I]	Dateimodus ändern	8
CLS		Bildschirm löschen	10
COLD	[1 2 3] [drive]	Kaltstart oder Booten	11
COPY	FILESPEC [DEST] [I]	Dateien kopieren	12
DATE	[NEWDATE]	Datum ansehen oder einstellen	15
DI	[DRIVE]	Laufwerksinformationen	16
DIR	[FILESPEC]	Inhaltsverzeichnis anzeigen	17
DIRP	[FILESPEC]	Inhaltsverzeichnis ausdrucken	17
DPATH	[DRIVE1,DRIVE2,...]	Laufwerkspfad angeben	18
EDIT	[TEXTFILE]	Bildschirmeditor aufrufen	19
ERA	FILESPEC [EXEC]	Dateien löschen	21
EXIT		JADOS verlassen	23
FF		Seitenvorschub am Drucker	24
GP		Grundprogramm aufrufen	25
INST	[PROGRAM]	Programm resident installieren	26
LDIR	[FILESPEC] [L P]	Ausführliches Inhaltsverzeichnis	28
MEM		Freien Userspeicher anzeigen	30
PARK		Festplattenstatistik und Köpfe parken	31
PAUSE	[TEXT]	Batchverarbeitung anhalten	32
PRINT	[FILENAME]	Textdatei ausdrucken	33
REM	[COMMENT]	Kommentarzeile in Batchprogramm	34
REMOV	[PROGRAM]	Residentes Programm entfernen	35
REN	FILE1 FILE2	Dateinamen umbenennen	36
SDIR	[FILESPEC] [L P]	Kurzes Inhaltsverzeichnis	37
STEP	[STEPRATE]	Steprate für Diskettenlaufwerke ändern	38
TABLE	[L P]	Verzeichnis der residenten Programme	39
TLOAD	[ADRESS] [FILENAME]	Textdatei(en) in Hauptspeicher laden	40
TSAVE	[FILENAME] [ADRESS]	Text aus Hauptspeicher sichern	42
TYPE	[FILENAME]	Textdatei anzeigen	43