

TURBO-DOS XL/XE V 2.1 - (P) 2005 by ABBUC e.V.

Anleitung zum  
**Turbo-DOS**<sup>XL</sup>/<sub>XE</sub>

& UTILITIES (VERSION 2.1)

FÜR ATARI XL/XE



von :

**REITERSTADT**

computertechnik



Revised by GoodByte for ABBUC e.V. in January 2005

Inhaltsverzeichnis:

Kapitel	Seite
Bezugsquellen und Copyright .....	3
Vorwort .....	4
Es geht los (wichtige Hinweise) .....	5
Die vier Versionen des TURBO-DOS XL/XE .....	7
Allgemeine Informationen über DOS und. Diskettenstationen .....	8
DOS konfigurieren (CONFIG.COM, SETXLRD.COM, DOSINFO.COM) .....	11
Batch-Verarbeitung "AUTOCOPY" (AUTORUN.SYS, SETUP.BAT) .....	14
Batchfile-Editor (BATEDIT.COM) .....	17
Anleitung zum TUBBO-DVP XL/XE (DUP.SYS) .....	19
Diskettenkopierer (DISKCOPY.COM) .....	40
File-Kopierer (FILECOPY.COM) .....	42
Disketten retten (DISKFIX.COM) .....	45
DOS-Konvertierer (CONV234.COM) .....	48
3 Programmlader (WRGAME.COM, WRBASIC.COM, CHOOSE.CON) .....	49
Druckertreiber (PRINTER.COM) .....	50
Sektoranalyse (DISKMAP.COM) .....	50
Utilities für TURBO-BASIC XL und BIBO-DOS .....	51
Utility für das TURBO-Modul (CREATURX.COM) .....	52
Nachwort .....	52
Anhang	
Fehlerdiagnose .....	53
Technische Daten .....	56
TURBO-DOS XL/XE in der Praxis .....	59
Stichwortverzeichnis .....	61
Nachtrag zu Speedy und XF 551 .....	64
Referenzblatt zu TURBO-DOS XL/XE V. 2.1 .....	65

Bezugsquellen:

TURBO-BASIC XL : ABBUC e.V.  
TURBO-Modul 1050 : Info: ABBUC e.V.  
HAPPY Modul : Happy-Computers (nur gebraucht erhältlich)  
SPEEDY Modul : ABBUC e.V.  
TURBO-DOS XL/XE : ABBUC e.V.  
RAM-Erweiterungen: Info: ABBUC e.V.

Aus rechtlichen Gründen muss darauf hingewiesen werden, dass die Teile des DOS.SYS, die mit dem des DOS 2.5 übereinstimmen, dem Copyright der ATARI Corp., Sunnyvale, USA, unterliegen und kostenlos mitgeliefert werden. Hierzu besteht die ausdrückliche Genehmigung der Firma ATARI Deutschland GmbH, 6096 Raunheim.

TURBO-DOS XL/XE Version 2.1

Überarbeitete Neuauflage 01/2005

Anleitung & Software sind urheberrechtlich geschützt. Alle Rechte liegen bei ABBUC e.V., Wieschenbeck 45, 45699 Herten.

Die hier beschriebene Software ist als Public Domain freigegeben.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in dieser Anleitung geschieht ohne besondere Kennzeichnung. Sie berechtigt nicht zu der Annahme, dass diese Namen im Sinne der Warenzeichen- und Markenschutzbestimmungen als frei zu betrachten wären. Alle Angaben erfolgen ohne Berücksichtigung eines eventuell vorliegenden Patentschutzes. Eine Haftung oder Verantwortung seitens des Lieferanten bzw. Herstellers für Schäden gleich welcher Art, die dem Käufer, Anwender oder Dritten durch Verwendung des gelieferten Materials inklusive Beschreibung und Software entstehen, ist ausgeschlossen. Jegliche Haftung liegt einzig und allein beim Käufer, Eigentümer oder Anwender. Einsendungen an den Lieferanten haben in jedem Fall frei zu erfolgen. Sendungen seitens des Lieferanten - dies gilt für Bestellungen wie für Rücksendungen von nachgebesserten oder ausgetauschten Teilen- erfolgen grundsätzlich auf Kosten und Gefahr des Kunden. Erfüllungsort ist der Sitz des Lieferanten.

Copyright (c) 1989 Martin Reitershan Computertechnik

Publisher (p) 2005 ABBUC e.V., Wieschenbeck 45, 45699 Herten

Jedwede Reproduktion, gleich welcher Art, auch auszugsweise, nur mit schriftlicher Genehmigung.

Liebe Bit Byter und ATARI XL/XE Fans!

Vor euch liegt die Neuauflage des Handbuchs zum TurboDOS XL/XE V. 2.1. Dieses DOS mit der wohl breitesten Palette an unterstützter Hardware bei gleichzeitig sehr hoher Kompatibilität zu ATARI DOS 2.5 ist nach wie vor eines der besten DOS für den XL/XE.

Entwickelt wurde dieses hervorragende Stück Software von Martin Reitershan, Herbert Barth und Frank Bruchhäuser in den Jahren 1986 bis 1988. Ursprünglich sollte es ein ATARI-System mit 130XE und 1050Turbo optimal unterstützen, da es damals dafür kein wirkliches geeignetes DOS gab. Das hat sich bis heute nicht geändert!

Die Entwicklung von V. 1.5 über V. 1.7 und V. 2.0 zur endgültigen V. 2.1 führte zu einem Diskettenbetriebssystem, das sehr benutzerfreundlich, weitestgehend kompatibel zu der Ende der 80er Jahre für den XL/XE erhältlichen Hardware sowie schnell und stabil ist. Wer also schon immer Mal vier Floppies mit unterschiedlicher Hardwareerweiterung unter einem DOS vereint wissen wollte und dazu noch eine große RAMDisk verwendet, liegt hier genau richtig. Selbst so exotische Hardware wie das MIO werden unterstützt.

Mit dabei sind eine Anzahl hervorragender Utilities wie DOS-Converter, Game-DOS, BASIC-Lader, Druckertreiber, Disk-Copy, Speichermonitor, Diskfixer, Programmlader, Sector-Analyzer, RESET-Schutz für TurboBASIC XL, Turbo-Modul-Utility, Quad Density vom BiboDOS, etc.

Dieses ausgezeichnete DOS ist seit einigen Jahren Public Domain und in unsere Public Domain Bibliothek als PD-Disk #506 erhältlich. Daneben wird es von mir beim Aufarbeiten und Aktualisieren der PD-Disks der Bibliothek immer dort eingesetzt, wo die Programme es zulassen.

Viel Spaß mit diesem DOS

GoodByte

### Es geht los ... (wichtige Hinweise)

Wer schon über Erfahrungen im Duplizieren kompletter Disketten verfügt, der sollte sich jetzt eine Kopie der Originaldisk anfertigen. Hierzu kann auch das Programm DISKCOPY.COM verwendet werden. Da hierbei fatale Fehlbedienungen nicht auszuschließen sind, ist dem Anfänger dringend zu empfehlen, zunächst eine Masterdisk nach folgender Anleitung zu erstellen:

#### - Erstellen einer floppyspezifischen Masterdisk:

Wie bereits im Vorwort erwähnt, befinden sich auf der vorliegenden Programmdiskette vier verschiedene DOS-Versionen, die jede auf einen bestimmten Laufwerkstyp zugeschnitten ist. Genaueres über die einzelnen Auswahlmöglichkeiten findet Ihr im nächsten Kapitel. Die Euch vorliegende Diskette verfügt über ein Menü, aus dem Ihr die für Euch passende Version, abhängig von dem verwendeten Diskettenlaufwerk, auswählen könnt. Hiernach erstellt das Programm automatisch eine Diskette dieser Version inklusive aller passenden Utilities, wobei Ihr nur den Benutzerhinweisen am Bildschirm zu folgen braucht.

Hierzu legt Ihr die Originaldiskette bei ausgeschaltetem Computer in die eingeschaltete Floppy ein. Nach dem Einschalten des ATARI erscheint nach kurzer Zeit das Titelbild besagten Programms (diesen Vorgang nennt man "booten"), welches jetzt bis zum Ende durchgearbeitet werden muss.

#### - Im Arbeitsmenü (DUP)

Die so erstellte Masterdisk wird nun nach dem bekannten Muster "gebootet", wonach Ihr Euch im Arbeitsmenü (der sog. Benutzeroberfläche) des TURBO-DOS XL/XE befindet. Jetzt ist es an der Zeit, sich die Anleitung zum TURBO-DUP XL/XE zu Gemüte zu führen (aber nicht die Kurzanleitung, die als Gedächtnisstütze beim späteren Arbeiten gedacht ist).

#### - DOS konfigurieren

Als nächstes solltet Ihr das DOS an Eure Hardware anpassen, d.h. die Anzahl der angeschlossenen Diskettenlaufwerke sowie die Art der RD müssen eingegeben werden, wonach diese Einstellungen auf eure Masterdisk abzuspeichern sind und somit bei jedem Bootvorgang automatisch vorgenommen werden.

Bei der Eingabe der benötigten Parameter hilft Euch das Programm CONFIG.COM, das Ihr nun starten solltet. Bitte schaut Euch hierzu die entsprechende Anleitung an. Nun liegt also eine an Eure Gerätekonfiguration angepasste DOS-Version vor .

Abschließend ist es noch sinnvoll, das Batchfile SETUP.BAT auf Eure Wünsche einzustellen. In der Hauptsache können hiermit beim Booten automatisch Files gestartet bzw. auf die RD kopiert werden (hier- zu unbedingt das Kapitel "Batch-Verarbeitung" durchlesen).

- Erstellen von Arbeitsdisketten

Die so erstellte Version ist Eure persönliche DOS-Masterdiskette. Um nun das DOS auf andere Arbeitsdisketten zu übernehmen (z.B. BASIC, Assembler u.a.), bootet Ihr zunächst diese Masterdisk. Nachdem Ihr die gewünschte Arbeitsdisk eingelegt (und evtl. formatiert) habt, wird im DUP der Befehl INIT eingegeben, wonach sich die Files DOS.SYS und DUP.SYS auf der Diskette befinden. Hiernach muss das File AUTORUN.SYS (verantwortlich für die Batch-Verarbeitung) mit dem COPY - Befehl auf diese Diskette kopiert werden.

Nun wird noch SETUP.BAT entsprechend den Anforderungen für diese Arbeitsdiskette eingestellt. Im besonderen ist hierbei darauf zu achten, dass Autostart-Files jetzt nicht mehr AUTORUN.SYS heißen dürfen, da dieser Name ja schon benutzt wird. Ein solches File wird umbenannt und aus dem SETUP.BAT gestartet. Dies hat den Vorteil, dass man zum einen mehrere Files hintereinander starten kann, zum anderen diese an ihren eindeutigen Namen besser zu erkennen sind (z.B. lässt sich anhand des Namens TURBOBAS.COM schon eher erahnen, dass es sich hierbei um TURBO-BASIC handelt, als dies bei AUTORUN.SYS der Fall wäre).

- Alternative zu AUTOCOPY (File: RAMDISK.COM)

Dieses Programm leistet das gleiche wie RAMDISK.COM bei DOS 2.5, ist aber wesentlich kürzer und löscht - genau wie AUTOCOPY - die RD nur bei Bedarf bzw. gleichzeitigem Drücken von <START>. RAMDISK.COM dient als Ersatz für AUTOCOPY, wenn dessen Funktionen nicht benötigt werden oder sein Speicherplatzbedarf bei einer bestimmten Anwendung zu groß ist (Anm.: Autostart-Files müssen jetzt wieder AUTORUN.SYS heißen). Für einen Autostart von RAMDISK.COM beim Booten ist eine Umbenennung in RAMDISK.COM erforderlich.

Übrigens:

Wenn in den nun folgenden Kapiteln kursive Textpassagen auftreten, die mit "(? " beginnen, so richten sich diese an Leute mit Maschinensprache- bzw. Betriebssystemkenntnissen. Für den "normalen" Gebrauch können diese getrost überlesen werden.

- einzelne Tasten der Computer-Tastatur werden mit ihrer Beschriftung in spitzen Klammern gekennzeichnet (z.B. '<RETURN>')

- Hinweise auf andere Stellen der Anleitung stehen in runden Klammern mit Pfeil (z.B. '(-> AUTOCOPY)')

### Die vier Versionen des TURBO-DOS XL/XE

Im Laufe der mittlerweile zehnjährigen Geschichte der ATARI 8-Bit- Computer erschienen verschiedene Laufwerke von ATARI und Fremdherstellern auf dem Markt. Die in unseren Gefilden meistverkaufte ist sicherlich die Floppy ATARI 1050, die leider nicht mehr hergestellt wird und von der XF 551 abgelöst wurde.

Für das Modell 1050 gibt es einige sehr interessante Aufrüstsätze, von denen wir die drei wichtigsten berücksichtigt haben (Preise, Leistungsdaten und Bezugsquellen am Ende der Anleitung).

Fremdlaufwerke wie auch auferüstete 1050er haben eines gemeinsam:

Eine höhere Speicherkapazität bei meist schnellerer Datenübertragung. Vor allem letztere erforderte bisher die Verwendung eines speziellen DOS, mit dem die Verwaltung von Daten im Standardformat (ATARI DOS 2.0 bzw. 2.5) leider nicht immer möglich war. Außerdem sind diese "Dösse" zum Teil sehr speicherplatzschluckend, so dass sie mit vielen Programmen nicht mehr zusammenarbeiten. Die Arbeit mit einem DOS, das noch nicht einmal mit dem längst zum Standard gewordenen TURBO-BASIC XL funktioniert, erscheint uns wenig attraktiv; und hiervon gibt es einige.

TURBO-DOS XL/XE bietet Euch folgende Laufwerksanpassungen :

1.) 810, 1050 normal oder mit TURBO-Modul

Diese Version benötigt den kleinsten Platz im Arbeitsspeicher und kann mit Einschränkungen auch für alle anderen Floppies verwendet werden, stellt also die Standardeinstellung dar. Wer bei der Auswahl der passenden Version unsicher ist, sollte diese als erste probieren.

2.) 1050 mit HAPPY oder SPEEDY

Besitzer dieser Erweiterungen kommen hiermit in den Genuss der gut dreifachen Beschleunigung gegenüber dem Standardlaufwerk. Nur bei sehr speicherkritischen Programmen (siehe später) muss hierauf verzichtet werden, da dann 1.) gewählt werden sollte.

3.) XF 551

Hiermit wird sowohl die eingebaute Beschleunigung als auch die Doppelseitigkeit genutzt (max. 128 Files bei Quad Density).

4.) Vollversion

Diese Version vereinigt alle vorgenannten Leistungsdaten in einem was sich natürlich im Speicherplatzbedarf niederschlägt. Der Vorzug ist aber, hiermit ein DOS zu besitzen, mit dem alle Laufwerke der Punkte 1 bis 3 gleichzeitig betrieben werden können und dies bei 1 voller Ausnutzung der den einzelnen Geräten eigenen Beschleunigung und erhöhten Speicherkapazität. Auch hier ist aber bei bis zu 3 Floppies oder 2 Floppies mit RD immer noch die Zusammenarbeit mit TURBO-BASIC XL gewährleistet !

Bei gleichzeitiger Verwendung eines TURBO-Modul-Laufwerkes mit HAPPY, SPEEDY bzw. XF 551 sollte nicht (4.) gewählt werden (!); da in diesem Fall auch (2.) bzw. (3.) genügt. Wichtig ist in jedem Fall, dass das TURBO-Laufwerk als physikalisches Laufwerk 1 eingestellt wird. Bei mehr als einem TURBO-Laufwerk ist der AUTOCOPY-Befehl 'SPEED-DRIVES' zu beachten.

## Allgemeines über DOS und Diskettenstationen

### 1. - Diskettenformate

Um auf Disketten Informationen abspeichern zu können, müssen diese formatiert werden (-> TURBO-DUP), wobei die Magnetschicht der Diskettenoberfläche in nummerierte Sektoren gleicher Größe eingeteilt wird. Die Sektoren sind auf sogenannten Tracks (Spuren) angeordnet, die wie konzentrische Kreise auf der Diskette liegen.

Alle für den ATARI benutzten Formate legen 40 Tracks auf einer Diskette an, wobei aber die Größe und Anzahl der Sektoren pro Track unterschiedlich ist:

#### a) Single Density (einfache Dichte):

Dieses Format wurde von den ersten ATARI-Laufwerken (810) verwendet. Auf einem Track liegen hierbei 18 Sektoren, deren jeder 128 Bytes an Informationen aufnehmen kann.

Dies bedeutet, dass auf einer Diskette 720 (18\*40) Sektoren und somit 90 KB (720\*128/1024) Platz finden.

#### b) Enhanced Density (erweiterte Dichte):

Von ATARI irreführender Weise zunächst als "Dual Density" bezeichnet. Statt 18 werden hier 26 Sektoren pro Track aufgenommen, was einer Gesamtkapazität von 1040 Sektoren oder 130 KB entspricht. Dieses Format konnte erstmals von der 1050 erzeugt werden, die das Nachfolgemodell der 810 darstellte.

#### c) Double Density (doppelte Dichte):

Dieses Format benutzt wieder nur 18 Sektoren pro Track, die aber mit 256 Bytes exakt doppelt so viele Daten aufnehmen können wie ihre "Kollegen", womit sich eine Speicherkapazität von 180 KB ergibt. Verarbeitet werden konnte diese Dichte bisher nur von Fremdlaufwerken (die vor allem in den USA für den ATARI angeboten wurden), sowie den o.a. Erweiterungssätzen für die 1050.

Mit der XF 551 hat ATARI nun erstmals (von einem Flop in den USA abgesehen) ein Laufwerk auf den Markt gebracht, das auch diese Aufzeichnungsart beherrscht, durch einen kleinen technischen Mangel aber erst mit Hilfe eines entsprechend intelligenten DOS die verschiedenen Formate zu unterscheiden weiß.

Aus diesem Grund kann eine XF 551 mit dem ATARI DOS 2.5 oder unserer o.a. Version (1.) nicht von einem auf ein anderes Format kopieren, während alle anderen Floppies die Speicherdichte bereits beim Einlegen der Diskette erkennen.

Alle uns bekannten Laufwerke und Erweiterungen halten sich bei diesem Format glücklicherweise an den sog. PERCOM-Standard.

#### d) Quad Density (vierfache Dichte)

Die meisten Laufwerke verfügen über nur einen Schreib-/Lesekopf, so dass immer nur eine Seite der zumeist zweiseitigen Disketten ohne Wenden benutzt werden kann. Das neue Laufwerk XF 551 verfügt über zwei solcher Köpfe, weshalb gleichzeitig beide Seiten in Double Density verwaltet werden können, was einer Speicherkapazität von 360 KB entspricht. Bei dieser Dichte erlaubt unser DOS maximal 128 Files pro Disk, während es bei (a)-(c) - wie üblich - 64 sind.

## 2. - Verwaltung der Sektoren durch ein DOS

Es würde einen großen Aufwand bedeuten, müsste man seine abzuspeichernden Daten eigenhändig auf die beim Formatieren erzeugten Sektoren schreiben. Man müsste genau Buch führen, auf welcher Diskette welche Sektoren von welchen Programmen benutzt werden. Wer einmal mit dem Datenrecorder arbeiten "durfte", hat vielleicht noch Aufzeichnungen, in denen zu einem Programmnamen die Kassette und der zugehörige Bereich des Bandzählwerkes niedergeschrieben wurden.

Glücklicherweise wird uns diese Arbeit bei der Diskettenverwaltung von einem sogenannten 'DOS' (Disk Operating System) abgenommen, das jeder von Euch beim Kauf seiner Diskettenstation auf Diskette erhalten hat.

Es handelt sich hierbei um ein bootfähiges Programm, das auf der Diskette ein Inhaltsverzeichnis (Directory) anlegt, in dem alle Namen der vom Benutzer abgespeicherten Daten (z.B. BASIC-Programme) mit der Nummer des zugehörigen Anfangssektors sowie der benötigte Platz (in Sektoren) automatisch eingetragen werden.

Um später wieder alle zusammengehörenden Sektoren zu finden, steht in jedem (außer dem letzten) Sektor ein Verweis auf den nächsten (Sektor-Link). Zusätzlich gibt es noch eine Tabelle (VTOC), in der vom DOS verzeichnet wird, welche Sektoren schon benutzt und welche noch (oder wieder) frei sind. Natürlich könnte man auch eine andere Art der "Buchführung" wählen, doch haben auch wir uns aus Kompatibilitätsgründen an diesen von ATARI vorgegebenen Standard gehalten.

Jedes DOS besteht aus einer Benutzeroberfläche, DUP genannt (wo der Benutzer Eingaben vornehmen und auch entsprechende Rückmeldungen erhalten kann), sowie dem "unsichtbaren" 'File Management System' (File-Verwaltungssystem), kurz FMS genannt, welches - etwas vereinfacht ausgedrückt - dafür sorgt, dass bei einem Zugriff auf Files die richtigen Sektoren angesprochen werden.

Wir setzen voraus, dass Ihr bereits durch die Eurer Floppy beiliegende Anleitung ausreichend über 'File' und ähnliche Begriffe informiert seid.

Bei TURBO-DOS XL/XE befinden sich FMS und DUP in getrennten Files, die die Namen 'DOS.SYS' bzw. 'DUP.SYS' tragen, was Vorteile beim Speicherplatzbedarf bringt (-> TURBO-DUP).

Das FMS sorgt also dafür, dass bei dem Abspeichern eines Programms nur solche Sektoren auf der Diskette verwendet werden, die bisher noch frei waren, und dass bei einem späteren Laden des selben Programms eben diese Sektoren in den Arbeitsspeicher des Rechners eingelesen werden.

Der "Normalverbraucher" muss sich also nicht mit einzelnen Diskettensektoren herumplagen, sondern bekommt jene in sinnvollen Einheiten zusammengefasst präsentiert.

Wer aber selbst schnelle Diskoperationen programmieren oder auch nur dem FMS etwas in die Karten schauen will, der sollte sich eingehend mit der Technik des direkten Sektorenzugriffs vertraut machen und sei auf die Lektüre einschlägiger Literatur verwiesen.

Die Fortgeschrittenen unter Euch finden eine Kurzanleitung im Anhang (-> Technische Daten; Adresse \$76C).

### 3. - Verschiedene DOS Versionen

Das erste DOS, das ATARI mit der Floppy 810 auslieferte, nannte sich DOS 1.0 und wurde schon bald von dem verbesserten DOS 2.0 abgelöst. Letzteres gab es in zwei Ausführungen:

Während DOS 2.0s für die 810 ausreichend war, konnte mit 2.0d die Double Density des Doppellaufwerkes 815 (hierbei handelt es sich um den vorher erwähnten Flop) verwaltet werden, welches aber auf Grund technischer Ungereimtheiten bald wieder vom Markt verschwand.

Wer bei DOS 2.0 bei einer eingelegten Double-Density-Diskette <RESET> betätigt, wird daher die Überraschung erleben, dass dann auf diese Dichte zugegriffen werden kann, obwohl dieses DOS nur Single Density formatieren kann.

Mit der 1050 erschien dann auch ein neues DOS, das sich DOS 3 nannte und neben Single auch Enhanced Density beherrschte. Da es jedoch zu dem bisherigen Standard gänzlich inkompatibel war und darüber hinaus noch durch umständliche und aufgebauchte Benutzerführung glänzte, fand es bald einen Nachfolger:

Mit den neuen XE-Computern wurde DOS 2.5 ausgeliefert, das eine konsequente Weiterentwicklung des DOS 2.0 darstellte und zu diesem aufwärtskompatibel war. Neben Enhanced Density wurde auch die Verwaltung einer RD implementiert, womit der zusätzliche Speicher des 130 XE sinnvoll genutzt werden konnte.

Dies bedeutete das Aus für DOS 3, und man kann davon ausgehen, dass DOS 2.5 die bisher stärkste Verbreitung gefunden hat; somit ist es als Standard anzusehen. Zu erwähnen ist noch DOS 4 (auch als QDOS bekannt), das, einstmals von ATARI in Auftrag gegeben, wieder fallengelassen und dann als Public Domain verbreitet wurde. Diese sehr aufwendige Entwicklung hat aber wegen der fehlenden RD-Verwaltung und Formaterkennung sowie seiner Inkompatibilität zu DOS 2.0 keine allzu große Bedeutung erlangt.

Es gibt nun noch eine Unzahl an DOS-Sternen und -Sternchen privater wie kommerzieller Software-Entwickler am DOS-Himmel, auf die wir hier nicht eingehen können (und wollen). Soviel sei jedoch gesagt:

Wäre darunter auch nur ein DOS, das nicht wenigstens einen nach unserem Ermessen eklatanten Nachteil aufweisen würde, so hätte es unser TURBO-DOS XL/XE wohl nie gegeben.

DOS konfigurieren—(Files: CONFIG.COM, SETXLRD.COM, DOSINFO.COM)

Mit dem Programm CONFIG.COM wird, wie bereits in der Einleitung erwähnt, eine an Eure Hardwarekonfiguration angepasste DOS-Version erstellt.

Die Standardeinstellung sieht folgendermaßen aus:

- a) Connected Drives (angeschlossene Laufwerke): 2
- b) File Buffers (Maximalzahl gleichzeitig geöffneter Files)- 2
- c) Compatible with DOS 2.5 (DOS 2.5-kompatibel): Y (ja)
- d) Re-configure RAMdisk (RD neu einstellen): N (nein)
- e) 64kB RAMdisk Banks (Anzahl der 64KB-RDbänke): I (= 130 XE)
- f) RAMdisk Bytes (RD-Verwaltungsbytes hexadezimal): EO (= 130 XE)
- g) Logical Drive # (Nummer der logischen Laufwerke): 12345678. Dies ist der feste Teil einer Zuordnungstabelle, kann also nicht verändert werden.
- h) Physical Drive H (Nummer der physikalischen Laufwerke): 12348888
- i) Configuration OK ? (Einstellungen übernehmen ?): N

- Erklärung der einzelnen Punkte

Die einzelnen Punkte werden mit den CURSOR-Tasten (ohne <CONTROL>) angefahren und durch Betätigen der gewünschten Taste (z.B. <Y> oder <1>) eingestellt.

- c) a) An die ATARI-Computer können maximal 4 Laufwerke angeschlossen werden. Da das DOS aber für jede Floppy 256 Bytes reserviert, kann hier die tatsächliche Zahl vorhandener Laufwerke eingestellt werden, um den Speicherplatzbedarf des FMS so klein wie möglich zu halten. Wer nur eine Diskettenstation besitzt, sollte die Voreinstellung also unbedingt auf '1' abändern. Eine eventuell vorhandene RD wird an dieser Stelle nicht berücksichtigt .
- d) b) In einem Programm oder dem BASIC können maximal 8 Files gleichzeitig zum Lesen oder Schreiben geöffnet sein. Da aber auch hier für die Größe des DOS ähnliches gilt wie bei (a), sollte die Zahl möglichst klein gehalten werden. Die Voreinstellung von 2 ist in den meisten Fällen ausreichend.
- e) Im Normalfall sollte diese Einstellung unbedingt auf 'Y' verbleiben, da es sonst zu Datenverlusten auf Disketten mit Enhanced Density kommen kann! Wann und warum eine Abänderung sinnvoll sein kann, wird den Fortgeschrittenen im Anhang (Seite 60) erläutert.
- f) Mit <Y> wird ein RAMtest eingeleitet, der als Ergebnis die erkannte RD in (e) und (f) anzeigt. Es werden alle RD-Arten erkannt, die über Speicherstelle \$D201 (PORT B) angesprochen werden.
- g) Die Anzahl der Bänke kann vermindert werden, um einen Teil der RD vor Zugriffen des DOS zu schützen. Besitzer des TURBO-FREEZER XL (Engl) mit 256k RD z.B. können die Bank schützen, auf die der Freezer zugreift (Bank EO).
- h) Hier werden die Steuerbytes angezeigt, mit denen die einzelnen RD-Bänke eingeschaltet werden. Veränderungen sollten nur von "Profis" vorgenommen werden, da die einwandfreie Funktion der RD sonst nicht mehr gewährleistet ist.

**Achtung:** Eine Neueinstellung der RD hat den Verlust der auf der bisherigen RD befindlichen Daten zur Folge. Vor Benutzung der neuen RD muss diese mit CLR (-> DUP) initialisiert werden.

i) Hierunter verbirgt sich eine Tabelle, in der die Zuordnung der physikalischen zu den logischen Laufwerken verändert werden kann. Bei der Verwendung einer RD bietet dies interessante Möglichkeiten:

Wird unter (h) die Reihenfolge auf 81348888 geändert, so wird fortan jeder Zugriff auf das logische Laufwerk 1 (D1:) auf die RD umgeleitet. Das tatsächliche (physikalische) Laufwerk 1 kann aber immer noch über 'D2:' angesprochen werden.

Anwendungsbeispiele:

- Bei Programmen, die bisher das Ansprechen einer RD nicht erlaubten, ist dies mit diesem Trick möglich.
- Bei größeren RDs ist es sinnvoll, alle benötigten Programme beim Systemstart durch AUTOCOPY auf selbige kopieren zu lassen. Werden später aus einem Programm Daten von 'D:' nachgeladen, so geschieht dies nun von der RD!

Sicherlich gibt es weitere Möglichkeiten, die wir Eurer Phantasie überlassen möchten.

i) Alle eingestellten Daten werden unter diesem Punkt durch ein Betätigen von <Y> oder <RETURN> in das DOS übernommen. Ist dies nicht erwünscht, so kann das Programm jederzeit mit <ESC> ohne Übernahme der Veränderungen verlassen werden.

**Achtung:** Die Einstellungen wurden nun zunächst in das im Arbeitsspeicher des Computers befindliche DOS übernommen. Damit diese aber auch beim nächsten Booten wieder zur Verfügung stehen, muss das DOS.SYS mit 'INIT /D' (-> DUP) auf die gewünschte Diskette abgespeichert werden.

- Weitere RD Arten

Für den 800 XE oder XL mit nur 64k bietet TURBO-DOS XL/XE noch eine "nicht echte" RD-Version, die vom eigentlichen Arbeitsspeicher abgezweigt wird. Hierbei handelt es sich um das parallel zum Betriebssystem liegende RAM, welches z.B. vom eingebauten BASIC nicht benutzt wird und daher normalerweise brachliegt. Zu diesen Programmen zählen auch solche, die noch auf den ATARIs (400/800) laufen, da letztere nur über maximal 48k verfügen (als Beispiel sei der Assembler ATMAS-II genannt).

Etwas ausgefuchstere Programme (z.B. TURBO-BASIC XL) nutzen sehr wohl diesen Speicherbereich, weshalb die sogenannte "XL-RD" hiermit nicht funktioniert. Im Zweifelsfall muss die Verträglichkeit getestet werden. Immerhin reicht ihre Kapazität aus, um das DUP inklusive MEM.SAV ständig im Speicher zu halten (residentes DUP).

Die XL-RD wird durch das Programm SETXLRD.COM installiert.

Ein solches File existiert auch für die etwas exotische und wenig verbreitete IRATA-RD (SETIRATA.COM), sowie das ICD Multi I/O (SETMIORD.COM).

Nun können bis zu 256 KB angesprochen werden. Ist die Erweiterung größer, so kann in \$70B eine 256KB-Bank selektiert werden (0-3, insgesamt 1 MB).

- Was ist eine RAM-Disk?

Der 130 XE verfügt über 128 KB Arbeitsspeicher, obwohl eigentlich nur 64 KB benutzt werden können. Die verbleibenden 64 KB werden als RD genutzt, d.h. wie eine Diskettenstation (Laufwerk 8) angesprochen, was bereits mit DOS 2.5 möglich war.

Die Geschwindigkeit des Datenzugriffs liegt natürlich weit über der einer echten Floppy, da ja keine Mechanik vorliegt; allerdings sind alle abgespeicherten Informationen nach dem Abschalten des Computers verloren.

Mit TURBO-DOS XL/XE ist die Verwaltung einer solchen RD nicht nur wesentlich komfortabler geworden, es werden auch Arten bis zu 256 KB verwaltet, da es mittlerweile entsprechend große RAM-Erweiterungen am Markt gibt, mit denen auch XL-Computer ausgestattet werden können (-> Bezugsquellen).

- Informationen zum DOS mit DOSINFO.COM

DOSINFO.COM gibt eine Kurzübersicht über die wichtigsten Daten des im Arbeitsspeicher oder auf einer Diskette befindlichen FMS (DOS.SYS). Nach dem Start kann man mit <SPACE> (Leertaste) das PMS im Speicher oder mit <RETURN> jenes auf einer Diskette untersuchen. In letzterem Fall ist noch die Angabe des gewünschten log. Laufwerkes erforderlich. Es werden angezeigt:

- Die Versionsnummer des DOS.SYS, wie sie ab V2.0 auch mit dem VER-Kommando des DUP ausgegeben wird.
- Die Einstellung von Punkt (a) des CONFIG.COM
- dto. Punkt (b)
- dto. Punkt (c)
- Beginn der File-Puffer
- Niedrigste Adresse, die nicht mehr vom DOS benutzt wird (MEMLO). Hiermit wird also die Größe des FMS im Speicher angegeben, die mit den Einstellungen von (b) und (c) beeinflusst wird und auch von der Version abhängt, die bei der Masterdiskerstellung ausgewählt wurde (-> "Die vier Versionen des TURBO-DOS XL/XE").

Will ein Programm nicht mit unserem DOS funktionieren, so könnte dies an einer Speicher-kollision (zu hohes MEMLO) liegen, wobei der Verzicht auf zusätzliche Laufwerke und Beschleunigung helfen könnte. Die Lage eines Maschinenprogramms im Speicher könnt Ihr Euch mit dem HEAD-Befehl (-> DUP) anzeigen lassen.

- Die Adresse, die mit dem BACK-Befehl eingestellt wird (-> DUP).
- Die Größe der eingestellten RD in 64k-Schritten. Handelt es sich um die XL-RD, so wird 'XL' angezeigt.
- Die Zuordnungstabelle wie unter (g) und (h)

Batch-Verarbeitung "AUTOCOPY" (Files: AUTORUN.SYS, SETUP.BAT)

Computer mit einem Arbeitsspeicher  $\geq$  128 KB werden im folgenden kurz RD-ATARI genannt. Das hierfür Gesagte kann bedingt jedoch auch auf den XL mit nur 64k Arbeitsspeicher zutreffen, da mit der Version 1.7 ja auch eine RD für diese Computer implementiert wurde; diese nimmt jedoch (-> CONFIG.COM) eine Sonderstellung ein.

AUTOCOPY sollte als AUTORUN.SYS-File auf einer TURBO-DOS Diskette vorhanden sein. Es erfüllt folgende Aufgaben:

Für die Besitzer eines RD-ATARIs dient es als Ersatz für RAMDISK.COM. d.h. die RD wird initialisiert. DUP.SYS, sowie MEM.SAV werden nur noch bei Bedarf angelegt. Des weiteren ist es bei RD-ATARIs nun möglich, ausgewählte Files auf die RD zu kopieren, damit sie von Anfang an hier zur Verfügung stehen.

Die RD-Verwaltung wurde in einem wichtigen Punkt verbessert:

Da die RD nur gelöscht wird, wenn man den Computer ausschaltet, ist sie noch erhalten, wenn man den Computer durch gleichzeitiges Drücken von <RESET> und <SELECT> neu bootet, was bei TURBO-DOS jetzt möglich ist. In diesem Fall formatiert AUTOCOPY die RD nicht!

Dies ermöglicht es z.B. dem Assemblerprogrammierer, den Computer nach einem Crash neu zu booten, ohne dass ein File der RD verloren geht.

Die RD wird neu formatiert, wenn der Computer zuvor ausgeschaltet, die RD teilweise überschrieben wurde (z.B. durch DISKCOPY) oder während des Bootens <START> gedrückt wird, bis die Meldung "Formatting RD" erscheint.

Weiterhin können eine Reihe von Befehlen ausgeführt werden, die weiter unten beschrieben werden. Alle Befehle, sowie die Namen der zu kopierenden Files, werden aus dem File SETUP.-BAT gelesen. Dies ist ein normales Text-(ASCII-)File, das mit einem Editor erstellt werden kann.

Am besten geschieht dies mit dem speziellen Editor BATEDIT.COM. der ebenfalls zum Lieferumfang gehört und im nächsten Kapitel beschrieben wird.

Steht in diesem Batchfile ein Filename, so wird er bei ATARIs ohne RD ignoriert. Bei RD-ATARIs wird der Filename auf dem Bildschirm ausgegeben. Falls das File noch nicht auf der RD vorhanden ist, wird es dorthin kopiert. Das zu kopierende File darf beliebig groß sein, es muss nur auf die RD passen.

Lautet der Filename DUP.SYS, so wird im DOS ein Flag gesetzt, damit DUP.SYS später bei Bedarf von der RD geladen wird. Lautet er MEM.SAV, so wird ein entsprechendes File auf der RD angelegt (-> Beschreibung MEM.SAV in der DUP-Anleitung). Befehle müssen stets invers geschrieben werden<sup>1</sup>, wobei darauf zu achten ist, dass pro Zeile immer nur ein Befehl bzw. Filename stehen darf, auch wenn der Befehl zusätzlich einen Filename verlangt; dieser muss dann in der nächsten Zeile stehen.

Folgende Befehle sind möglich:

a) RUN:

Das folgende COM-File wird vom Bootlaufwerk gestartet.

---

<sup>1</sup> Invers wird mit der Taste in der rechten, unteren Ecke der Tastatur ein- bzw. Ausgeschaltet. Bei Verwendung des Programms BATEDIT.COM geschieht die Umschaltung automatisch.

(? Endet das Programm mit RTS, so wird AUTOCOPY fortgesetzt. Voraussetzung ist, dass der Speicherbereich von \$2200 bis \$2900 nicht benutzt wurde.)

Beispiel: **RUN:** (diese Zeile in invers Video eingeben !)  
PIC.COM (Maschinensprache-File)

Es ist unbedingt zu beachten, dass bei der Eingabe von Filenamen die Devicekennung 'D1:' bzw. 'D:' weggelassen wird, da es sonst zu einer Fehlermeldung kommt. Alle Filenamen beziehen sich auf das Bootlaufwerk bzw. die RD.

b) LOAD:

Bei ATARIs ohne RD wird ein RUN ausgeführt. Bei RD-ATARIs wird dieses File zuerst auf die RD kopiert und dann von dort gestartet. Ist das File noch auf der RD vorhanden, entfällt das Kopieren.

Beispiel: **LOAD:**  
TURBOBAS.COM

c) BASIC

AUTOCOPY schaltet automatisch das eingebaute BASIC aus. Dies ermöglicht es, zu jeder Zeit Programme zu starten, die ein ausgeschaltetes BASIC voraussetzen, ohne immer <OPTION> drücken zu müssen. Der Befehl 'BASIC' schaltet das eingebaute BASIC ein und fährt dann mit der Batch-Verarbeitung fort.

d) ABORT?

Trifft das Programm auf diesen Befehl, wird die <OPTION> abgefragt. Ist sie gedrückt, erscheint die Abfrage "Abort: BASIC, Dos, Continue (B.D.C)?" . Durch die Eingabe von B gelangt man in das eingebaute BASIC. D springt ins DUP, C setzt AUTOCOPY fort; statt D kann auch RETURN gedrückt werden.

e) TURBO 4th page (nur für Besitzer des TURBO-Moduls)

Das Übertragungsprogramm des TURBO-Moduls wird vom Stack oder der Sixth-page in den Kassettenpuffer (\$400) verschoben. Hier liegt es geschützt, solange kein Recorder benutzt wird. Bei Programmen, die die Sixth-Page (\$600) benutzen, ist dies sicherer als der Stack (\$100).

f) SPEED-DRIVES: x (x von 2 bis 4; ebenfalls nur für TURBO-Modul)

Dieses Kommando wird benötigt, um mehr als ein TURBO-Laufwerk mit Beschleunigung anzusprechen. Als Parameter muss das Laufwerk angegeben werden, bis zu dem alle im DOS angemeldeten Diskettenstationen mit der TURBO-Übertragungsgeschwindigkeit betrieben werden sollen.

Beispiel: Speed-Drives: 2 (Kommando und Parameter invers)  
(Laufwerk 1 und 2 werden mit, alle weiteren ohne TURBO-Beschleunigung betrieben)

g) EXEC:

Dieser Befehl ermöglicht es, eine BASIC-Zeile einzugeben, die automatisch ausgeführt wird. Hiermit können z.B. BASIC-Programme gestartet werden.

"EXEC" schaltet sofort das BASIC ein und führt die Befehlszeile aus. Sollten in SETUP.BAT noch mehr Befehle folgen, werden diese ignoriert.

Beispiel: **EXEC:**  
RUN "D:START.BAS" (diese Zeile nicht invers !)

- h) \* Text  
Kommentare oder Benutzerhinweise werden auf dem Bildschirm ausgegeben. Die Kommentare dürfen pro Zeile maximal 38 Zeichen enthalten, der Rest wird abgeschnitten.  
Beispiel: \* Dies ist ein Kommentar (Sternchen invers !)
- i) WAIT  
Dieser Befehl steht sinnvollerweise nach einem Benutzerhinweis, da er die Meldung "- hit any key to continue ..." ausgibt und erst nach einem Tastendruck mit der Batch-Verarbeitung fortfährt.
- j) PAUSE: x  
Als Parameter muss eine Zahl von 1 bis 9 folgen, womit AUTOCOPY für x mal 3 Sekunden unterbrochen wird.
- k) XL-RAMDISK  
Wie schon am Anfang dieses Kapitels erwähnt, nimmt die auf einem 64k-ATARI eingerichtete RD eine Sonderstellung ein. Bei allen anderen RD-Arten wird diese -sofern im DOS eingestellt- bei jedem Bootvorgang initialisiert. Bei o.g. RD muss jedoch zusätzlich als erster Befehl (k) stehen, da ein Einrichten dieser RD nicht immer gewünscht wird (z.B. weil sie sich mit einem bestimmten Programm nicht verträgt; siehe CONFIG.COM).
- l) DUP-BATCH:  
Es erfolgt ein Sprung ins DUP, wonach der nachstehende Filename als EXECUTE-File aufgerufen wird (-> Anleitung zum TURBO-DUP ). Ist dieses File auf der RD vorhanden, wird es auch von dort geladen; ansonsten erfolgt der Start vom Bootlaufwerk.  
Beispiel: DUP-BATCH:  
HELP.EXE
- m) CONFIG x: yy zz (nur für SPEEDY-Floppies)  
(? Hiermit kann die SPEEDY konfiguriert werden, wobei x die physikalische Laufwerksnummer, yy das zu sendende "Slow/Fast-Konfigurationsbyte" und zz das "Drive/Display-Konfigurationsbyte" in hexadezimaler Form (immer zweistellig) repräsentieren. Zwischen den Parametern muss sich immer genau ein <SPACE> befinden, wobei zz auch weggelassen werden kann. )

Achtung: Diese Befehle können nicht im DUP ausgeführt werden !  
Filenames werden nicht invers geschrieben !

Sollte ein Fehler auftreten, wird angezeigt, ob er in Zusammenhang mit der Diskettenstation oder der RD aufgetreten ist. Man gelangt dann, falls möglich, mit <RESET> ins DUP, bzw. kann mit <SELECT>&<RESET> neu booten.

AUTOCOPY wurde so programmiert, dass eine Diskette mit einem Batchfile für einen RD-ATARI problemlos von einem ohne RD geladen werden kann.

Batchfile-Editor (File: BATEDIT.COM)

Mit diesem Editor kann, wie bereits im vorangehenden Kapitel angekündigt, das Batchfile SETUP.BAT erstellt werden. Zugleich dient dieses Programm auch der Eingabe von EXECUTE-Files, wie sie im DUP Verwendung finden (-> Anleitung zum TURBO-DUP).

- Erstellen des Files SETUP.BAT für AUTOCOPY

Nach den Start des Programms erscheint am unteren Bildschirmrand eine kleine Menüleiste, deren einzelne Punkte durch Eingabe des entsprechenden Anfangsbuchstabens oder mit den Cursor-Tasten angewählt werden. Wir betätigen zunächst <E> für EDIT. Nun muss man sich für einen der vorgenannten File-Typen entscheiden. Mit <S> oder <RETURN> wählt man Setup-Batchfile, wie für AUTOCOPY benötigt.

Wir befinden uns nun im eigentlichen Editor.

Die einzelnen Kontrollfunktionen, wie z.B. <CONTROL>&<INSERT> oder <CONTROL>&<+>, funktionieren genau wie im BASIC und dürften daher hinreichend vertraut sein. Neu ist, dass man mit <SHIFT>&<=> bzw. <SHIFT>&<-> Seitenweise blättern kann.

Um die einzelnen Befehle einzutippen genügt es. <CONTROL> zusammen mit dem Anfangsbuchstaben des gewünschten Kommandos zu betätigen. Ist dies nicht eindeutig (gibt es also mehrere Befehle mit gleichem Anfangsbuchstaben), so genügt ein Wiederholen dieser Tastenkombination, um den nächsten Befehl mit dem gleichen Buchstaben zur Anzeige zu bringen.

Der Befehl wird, wie von AUTOCOPY gefordert, automatisch invers geschrieben. <RETURN> übernimmt den Befehl, springt zur nächsten Zeile und schaltet invers Video wieder aus.

Benötigte Filenamen und Kommentare müssen natürlich im Klartext eingegeben werden, wobei nur 'DUP.SYS' und 'MEM.SAV' eine Ausnahme bilden :

Besitzer einer RD geben daher als erstes

```
<CONTROL>&<D>   <RETURN>  
<CONTROL>&<M>   <RETURN>
```

ein, damit bei jedem Booten DUP.SYS und MEM.SAV auf der RD angelegt werden.

Am unteren Bildrand wird die Zeilennummer angegeben, in der sich der Cursor gerade befindet, wobei das Wort 'LINE' den momentanen Schriftstatus wiedergibt, also evtl. klein oder invers erscheint.

Das Fassungsvermögen des Editors ist auf 256 Zeilen begrenzt.

- Abspeichern und Laden von Batchfiles

Wurde das File komplett eingegeben, gelangt man mit <CONTROL>&<ESC> wieder zum Anfangsmenü, wo wir mit <S> die Funktion SAVE anwählen. Der zuletzt eingegebene Filename wird angezeigt, wobei dieser mit 'SETUP.BAT' vorbesetzt ist und nun ediert werden kann:

Wieder sind die vom BASIC bekannten Kommandos erlaubt, erweitert um

```
<SPACE>   löscht den rechten Zeilenrest ab der Cursorposition  
<ESC>     verlässt die Eingabe ohne Veränderung
```

Der angezeigte Filename wird mit <RETURN> bestätigt, worauf das Batchfile abgespeichert wird.

Wurde als Device 'P:' eingegeben, so erfolgt die Ausgabe des Files auf den Drucker, der zuvor natürlich eingeschaltet werden muss.

Mit <CONTROL>&<ESC> gelangt Ihr übrigens wieder in den zuletzt bearbeiteten Text.

Statt <E> kann zu Beginn auch <L> gewählt werden, womit ein bereits vorhandenes Batchfile geladen und angeschaut bzw. editiert werden kann.

Nach getaner Arbeit gelangt Ihr mit <Q> wieder zurück zum DUP.

- Erstellen eines EXECUTE-Files für das DUP

Die Erstellung eines EXECUTE-Files erfolgt nach gleichem Muster, wobei folgende Unterschiede zu beachten sind:

- Bei der Wahl der Batchfile-Art muss <D> (oder <SPACE>) gedrückt werden.
- Mit <CONTROL>&<'Anfangsbuchstabe'> werden jetzt natürlich die Befehle des DUP ausgegeben, wobei hier auf die Mehrfachbelegung der einzelnen Buchstaben geachtet werden muss (siehe oben).
- Der Filename muss auf '.EXE' enden.

Zusätzlich zu den normalen Befehlen des DUP existieren noch folgende Kommandos, die nicht über <CONTROL> erreicht werden:

- '? x' unterbricht die Abarbeitung des EXECUTE-File, wobei x für die Dauer in 50stel Sekunden steht. Ein Fragezeichen ohne nachstehende Zahl erzeugt eine Unterbrechung bis zum nächsten Tastendruck.
- '\*' (nicht invers) muss vor einem Kommentar stehen.

Achtung: Die Befehle für ein EXECUTE-File werden nicht invers eingegeben, was der Editor bereits berücksichtigt hat.

## ANLEITUNG ZUM TURBO-DUP XE Version 2.1

### Was ist DUP ?

DUP ist die Abkürzung für "Disk-Utility-Package" und heißt, grob ins Deutsche übersetzt, "Disketten-Hilfsmittel-Paket". Das DUP ist die Benutzeroberfläche des DOS, über die man Veränderungen auf Disketten vornehmen kann.

Bei TURBO-DOS ist das DUP ein Zusatzprogramm auf Diskette oder RD, das den Namen "DUP.SYS" trägt. Dieses Maschinenprogramm wird - wie bei den ATARI-DOS - nur bei Bedarf in den Speicher geladen und gestartet. Wir wählten diese Lösung, weil das DUP auf diese Weise keinen Speicherplatz im 64 KB Hauptspeicher eures Computers belegt, solange es nicht gebraucht wird. Dadurch treten keine Einschränkungen beim Betrieb mit anderen wichtigen Programmen wie BASIC, TURBO-BASIC etc. auf.

(? Auch eine Benutzung des "unsichtbaren" RAMs, das im Bereich des Betriebssystems (OS-ROM) versteckt liegt, kam für das TURBO-DUP nicht in Frage, da wichtige Programme, allen voran das TURBO-BASIC, diesen Speicherbereich benutzen. Bei TURBO-DOS ist also im Gegensatz zu manch anderem DOS die einwandfreie Zusammenarbeit mit derartiger Software gewährleistet.)

### Wie gelangt man ins DUP ?

Wenn Ihr die TURBO-Masterdisk bootet, gelangt Ihr automatisch ins DUP. Im BASIC gibt es den Befehl 'DOS', mit dem man das DUP laden und starten kann. In anderen Programmen oder Programmiersprachen gibt es zumeist einen ähnlichen Befehl oder einen Menüpunkt, um ins DUP zu gelangen. Ist dies nicht der Fall, so führt die Tastenkombination

<SHIFT>&<RESET> ins DUP.

Außerdem existiert bei der Batch-Verarbeitung des TURBO-DOS noch eine Möglichkeit zum "Quereinstieg" ins DUP. Mehr darüber erfahrt Ihr im entsprechenden Abschnitt der Anleitung.

### Arbeiten mit und ohne MEM.SAV

Wie bereits erwähnt, wird das DUP bei Bedarf von Diskette (oder RD) in den Speicher geladen und zerstört somit dort gelegene Programme oder Daten. Hat man also z.B. im BASIC ein Programm eingetippt und geht ins DUP, um sich das Inhaltsverzeichnis (->DIRECTORY) einer Diskette anzusehen, so wird man nach Verlassen des DUP seine Programmzeilen nicht mehr finden. Mit TURBO-DOS kann man diesen Nachteil umgehen. Befindet sich nämlich auf dem Laufwerk, von dem die DUP-Datei geladen wird, ein File namens "MEM.SAV", so wird vor Laden des DUP der dazu benötigte Speicherplatz auf eben dieses File abgespeichert. Nach Verlassen des DUP wird dann automatisch das "MEM.SAV" wieder in den Speicher zurückgeladen. Danach befindet sich Euer Computer wieder im gleichen Zustand wie vor dem Starten des DUP, und es geht nichts an Information (z.B. Euer BASIC-Programm) verloren.

Befindet sich auf dem Laufwerk, von dem "DUP.SYS" geladen wird, keine Datei mit dem Namen "MEM.SAV", so findet keine Sicherung der im DUP-Speicherbereich liegenden Daten statt.

Das beschleunigt zwar das Laden des DUP, doch Ihr müsst dann darauf achten, dass Ihr zuvor selbst eure Daten sichert, falls Ihr nach Benutzung des DUP damit weiterarbeiten möchtet.

### MEM.SAV erzeugen

Im TURBO-DUP gibt es keinen eigenen Befehl zur Aktivierung von MEM.SAV wie z.B. bei den ATARI-DOS. Da es nur darauf ankommt, ob ein File mit dem Namen "MEM.SAV" vorhanden ist, kann man diese Aktivierung durch Anlegen einer solchen Datei getrost mit anderen DUP-Befehlen "simulieren". Beispiele hierfür sind: 'SAVE MEM.SAV/N,0.!' (-> SAVE) oder aber 'COPY DUP.SYS, MEM.SAV (-> COPY).

Außerdem gibt es noch bei der (->) Batch-Verarbeitung des TURBO-DOS die Möglichkeit, "MEM.SAV" automatisch auf der RD anzulegen.

**ACHTUNG:** Natürlich funktioniert MEM.SAV erst nach seiner Aktivierung, d.h. erst beim nächsten Laden des DUP ! Das bedeutet, wenn Ihr einmal ohne MEM.SAV ins DUP gelangt seid, nützt eine nachträgliche Aktivierung nichts mehr. Eure zuvor bearbeiteten Daten sind verloren, denn sie wurden ja nicht vorher abgespeichert.

### DUP-Benutzeroberfläche

Zwei zusätzliche Textzeilen am oberen Bildschirmrand lassen erkennen, dass man sich im DUP befindet. Diese Kopfzeilen geben Auskunft über Verfasser und Urheberrechte des TURBO-DOS. Sie können nicht überschrieben werden und bleiben bis zum Verlassen des DUP immer sichtbar. Die Bildschirmfarbe könnt Ihr Eurem Geschmack gemäß bleibend verändern (-> "Technische Daten ...").

TURBO-DUP arbeitet ohne ein Menü, d.h. die Befehle müssen in Textform eingegeben werden. Dies erfordert zwar (vielleicht) eine etwas längere Einarbeitungszeit für den Benutzer, ist aber, sobald man einmal mit den Kommandos vertraut ist, wesentlich bequemer, schneller und vor allem nervenschonender, weil man sich nicht durch unzählige Abfragen hindurchquälen muss (siehe DOS 3 !). Außerdem benötigt ein Menü durch die langen Texte mehr Speicherplatz.

### Prompt

Am Bildschirm ist die Meldung ("Prompt") "D1>" zu sehen. "D1" bedeutet, dass das Diskettenlaufwerk mit der Nummer 1 als Bezugslaufwerk (später auch "aktuelles Laufwerk" genannt) eingestellt ist. Alle Diskettenbefehle beziehen sich jetzt zunächst einmal auf dieses Laufwerk. Der Pfeil ">" deutet dem Benutzer an, dass an dieser Stelle eine Befehlseingabe von ihm erwartet wird. Dies kann auf verschiedene Weise geschehen:

### CONSOL-Tasten

Einige wichtige Funktionen des DUP wurden auf die sogenannten "CONSOL"-Tasten des ATARI-Computers - das sind die von der Tastatur abgesetzten Tasten <HELP> bis <OPTION> - gelegt.

Jedes Mal nach der Eingabeaufforderung (= Prompt) hat man die Möglichkeit, entweder etwas über die Normaltastatur einzugeben oder eine dieser Konsolentasten zu drücken. Hat man einmal eine normale Taste gedrückt, so haben <HELP> bis <OPTION> bis zum nächsten Prompt keine Wirkung mehr. Ihre Bedeutungen sind:

<HELP> Alle verfügbaren Befehle werden auf dem Bildschirm ausgegeben (-> HELP)  
<START> Disketteninhalt anzeigen (-> DIRECTORY)  
<SELECT> Schaltet um zwischen Laufwerk 1 und der RD, d.h. zumeist Laufwerk 8 (-> 1-8:  
DRIVE NUMBER)  
<OPTION> TURBO-DUP verlassen (-> CARTRIDGE)

#### Bildschirm-Editor

Mit den CONSOL-Tasten lassen sich wegen deren beschränkter Anzahl allerdings nur einige Befehle ausführen. Alle anderen Kommandos können nur in Textform eingegeben werden. TURBO-DUP benutzt zur Befehlseingabe den eingebauten Bildschirmeditor des ATARI-Betriebssystems, den Ihr sicher schon vom BASIC her kennt. Alle von dort gewohnten Steuertasten haben auch hier die gleiche Bedeutung, z.B. Cursor-Tasten, <INSERT>, <DELETE>, usw. Am Anfang jeder Befehlszeile, d.h. hinter dem Prompt, muss das Befehlskürzel stehen, das entweder aus 3 Großbuchstaben (zumeist die Anfangsbuchstaben des Befehlsnamens) oder aus einem sonstigen Zeichen besteht. Bei den Befehlskürzeln sind nur die ersten drei Buchstaben signifikant, es können aber nach Belieben mehr eingegeben werden, so z.B. 'COPY' statt nur 'COP' oder 'RENAME' statt 'REN'. Alle weiteren Zeichen außer diesen dreien werden ignoriert.

Gibt man hinter einem Befehl noch Parameter an, so müssen diese durch Leerzeichen oder Kommas untereinander und vom Kommandokürzel abgetrennt werden. Die Eingabe einer Befehlszeile wird durch Drücken von <RETURN> abgeschlossen und dann vom DUP bearbeitet. Wurde ein Eingabefehler gemacht, so erscheint mit akustischer Untermalung die Meldung "ERROR". Dahinter steht noch mal die Zeile, die man eingegeben hat, wobei nun die Fehlerstelle invers hervorgehoben ist. Man kann dann - sofern man das Kommando überhaupt noch ausführen lassen will - entweder die Zeile neu eingeben oder die zuvor falsch eingetippte Zeile mit dem Bildschirmeditor korrigieren.

Zusätzlich zu den herkömmlichen Funktionen dieses Editors gibt es bei TURBO-DUP die Möglichkeit, die zuletzt eingegebene Zeile weiter zu editieren. Drückt man nämlich nach dem Prompt <DELETE>, so erscheint das letzte Kommando erneut in der Eingabezeile. Hier kann es genau so , als hätte man es ein zweites Mal eingetippt, nach Belieben verändert und wieder ausgeführt werden.

Drückt man <CONTROL>&<DELETE> nach Erscheinen des Prompts, so wird das letzte Kommando einfach wiederholt. Dies entspricht dem Drücken von <DELETE> gefolgt von <RETURN>.

Doch nun zu den TURBO-DUP-Befehlen im einzelnen:

Jedem Kommando ist ein eigenes kleines Kapitel gewidmet, in dem er in all seinen Variationen vorgestellt und erläutert wird. Es empfiehlt sich, diese Abschnitte in der hier aufgeführten Reihenfolge durcharbeiten, weil sie inhaltlich aufeinander aufbauen.

## 1. HELP

Nach der Eingabe von 'HEL' oder Drücken von <HELP> werden alle im TURBO-DUP verwendbaren Befehlskürzel auf dem Bildschirm aufgelistet. Habt Ihr als Benutzer einmal die Abkürzung für einen Befehl vergessen - was am Anfang bestimmt des öfteren der Fall sein wird - so könnt Ihr hier nachschauen, welche Kommandos überhaupt verfügbar sind.

Die gesuchte Funktion wird Euch anhand dieser Aufstellung sicherlich gleich ins Auge springen. Gute Dienste wird Euch in diesem Zusammenhang auch sicherlich das mitgelieferte Referenzblatt erweisen, das Ihr aus der Anleitung entnehmen und ggf. in einer Klarsichthülle unterbringen solltet.

Auf diese Weise braucht Ihr nicht erst das TURBO-DOS-Handbuch hervorzukramen und durchzublättern, wenn Euch einmal ein Kommandokürzel entfallen ist. Eine ausführliche Anleitung zur Benutzung des DUP hingegen können und sollen HELP-Befehl und Referenzblatt nicht sein; hierfür gibt es schließlich das Handbuch, in dem ihr gerade lest.

(? Da HELP lediglich die zur Interpretation der Kommando-Eingabe ohnehin benötigte Kürzeltabelle auf den Bildschirm ausgibt, wird kein zusätzlicher Speicherplatz für Help-Texte "verbraten". HELP stellt somit einen Kompromiss zwischen ausführlicher Anleitung und Speicherplatzersparnis dar.)

## 2. DIRECTORY

### a) allgemeines DIR

Der Befehl 'DIR' zeigt das Inhaltsverzeichnis ("Directory") der Diskette im aktuellen Laufwerk an. Da man sich mit ihm über die auf der Diskette befindlichen Dateien ("Files") informieren kann, ist er der wichtigste, d.h. der am meisten gebrauchte Befehl im DUP.

Aus diesem Grunde wurde auch <START> mit dieser Funktion belegt. Das Directory erscheint in folgender Form: (z.B. Masterdisk) Die Kopfzeile "D1: SINGLE DENSITY" informiert über das Format der Diskette in der angewählten Floppy. Danach folgt eine nummerierte Liste aller Files auf dieser Disk. Die einzelnen Dateien werden durch Zeichenketten identifiziert, die jeweils aus zwei Teilen bestehen:

der erste Teil wird "Filename" genannt und besteht aus bis zu 8 Buchstaben und/oder Ziffern; er stellt die eigentliche Bezeichnung der Datei dar. Der zweite Teil der Namen, den man als "Extender" bezeichnet, ist des schöneren Aussehens willen durch Leerzeichen ("Spaces") vom ersten Teil abgetrennt. Der Extender kann aus maximal 3 alphanumerischen Zeichen bestehen und gibt zumeist Aufschluss über die Art der Datei (-> eingeschränktes DIR).

Hinter Filename und Extender steht jeweils eine dreistellige Zahl, die angibt, wie viel Speicherplatz das File auf der Diskette belegt.

Diese Zahl ist nichts weiter als die Anzahl der Disketten-Sektoren, die zur Speicherung der Datei benötigt werden. Sie hängt ab vom Format, d.h. vom Fassungsvermögen der Sektoren auf dieser Diskette.

Ein File, das 10 Sektoren auf einer in Single Density formatierten Diskette belegt, wird auf einer in Double Density formatierten ungefähr nur die Hälfte, d.h. 5 Sektoren belegen, denn die Sektoren solcher Disketten nehmen die doppelte Speichergröße eines Single Density Sektors auf.

Im Directory der TURBO-DOS-Masterdisk steht vor allen Dateinamen ein Sternchen. Ein Stern an einem Dateinamen zeigt an, dass diese Datei vor Überschreiben geschützt ist (-> LOCK).

Im TURBO-DUP wird außerdem noch zur besseren Übersichtlichkeit vor jedem Dateinamen eine Nummer angezeigt.

### Seitenweise Bildschirmausgabe

Ihr werdet euch vielleicht schon gefragt haben, was der inverse Balken mit den Worten "More (Y/A)?" am unteren Bildschirmrand bedeutet. Die Frage nach mehr bezieht sich auf die Bildschirmausgabe und tritt immer dann auf, wenn vom DUP aus eine Bildschirmseite vollgeschrieben worden ist. Die Bildschirmausgabe wird solange angehalten, bis eine Taste gedrückt wird. Bei <Y> und <RETURN> wird die Ausgabe fortgesetzt und die nächste Bildschirmseite vollgeschrieben. Drückt man <SHIFT>&<RETURN>, so wird die Seitenweise Bildschirmausgabe für den Rest der Befehlsabarbeitung abgeschaltet, d.h. die Zeilen "scrollen" ganz normal über die Mattscheibe.

Beim Drücken von <A> (= Abort) oder <ESC> wird das gerade ausgeführte Kommando abgebrochen, und man befindet sich wieder im Eingabezyklus.

Beim Directory der Masterdisk tritt das More-Prompt auch auf, da sie mehr als 20 Files enthält. Drückt Ihr nun <RETURN>, so erscheint der Rest des Inhaltsverzeichnisses. Am Ende steht dann noch die Zeile "250 FREE SECTORS.", die Euch sagt, wie viel Speicherplatz auf dieser Diskette noch frei ist. (Statt "250" erscheint bei Eurem Directory natürlich die tatsächliche Anzahl freier Sektoren auf Eurer Disk.)

Alternativ zur einspaltigen Darstellung kann das Directory im TURBO-DUP auch zweiseitig ausgegeben werden. Wegen der begrenzten Bildschirmbreite ist hierbei allerdings keine Filenummerierung möglich. Durch 'DIR /A' oder 'DIR +' kann man zwischen beiden Darstellungsarten hin- und herschalten. Der so gewählte Anzeigemodus bleibt dann für alle weiteren Ausgaben des Directory bis zum nächsten Auftreten der Option "/A" oder "+" in einem DIR-Kommando aktiv. Er wird außerdem beim Schreiben des DUP-Files abgespeichert (-> INIT).

### b) eingeschränktes DIR

Da man sich oft nur für einen bestimmten Teil der Dateien auf einer Diskette interessiert, - z.B. für alle BASIC-Programme oder alle Dateien, die mit einem ganz bestimmten Buchstaben anfangen - gibt es im DUP die Möglichkeit, durch Angabe einer Dateispezifikation oder kurz "Filespec" hinter dem DIR-Befehl eine Vor-Auswahl der auf Diskette befindlichen Dateien zu treffen.

### Dateinamen

Eine Dateispezifikation ist im Prinzip ein erweiterter Dateiname. Zunächst also eine Erklärung dazu: bereits weiter oben wurde gesagt, dass eine Dateibezeichnung aus zwei Teilen besteht, nämlich aus Filename und Extender. Der Filename besteht aus einer Reihe von mindestens einem und höchstens 8 alphanumerischen Zeichen. Außer Großbuchstaben und Ziffern sind keine anderen Zeichen darin erlaubt. Beim Extender gelten die gleichen Bestimmungen; seine Länge ist jedoch auf 0-3 Zeichen beschränkt. Wird ein Extender benutzt, so muss er bei der Eingabe durch einen Punkt "." vom Filename abgetrennt werden. Der Extender hat die Aufgabe, Auskunft über den Dateityp zu geben.

Folgende Extender sind gebräuchlich:

SYS Systemeigene Datei des DOS.  
COM "Compound-File": zusammengesetztes Maschinensprache-File (-> LOAD, SAVE, HEAD)  
BAS BASIC-Programm, kann vom BASIC aus mit 'LOAD' geladen werden.  
TUR dto. in TURBO-BASIC XL.  
BAT "Batch-File" (siehe Batch-Verarbeitung).  
EXE "Execute-File": DUP-Batch-Files (siehe EXE-Befehl).  
TXT Text-Files im ATASCII-Code, können vom DUP aus mit 'TYPE' auf dem Bildschirm ausgegeben werden.  
ASM, SRC Assembler Source-Text.

Hier noch eine Liste von erlaubten und verbotenen Dateinamen:

richtig:	FILENAME.EXT	falsch:	FileName.Ext
	TEST		.TST
	Z.B		(nichts)
	PROG1.BAS		PROG_1.BAS
	3DIM.PIC		VIELZULANGER.NAME
	TEXT.012		HALLO.!!!

### Wildcards

Man kann für den Dateinamen oder Teile von ihm auch Platzhalter ("Wildcards") einsetzen. Es werden dann nur die Files angezeigt, die bis auf die durch Wildcards besetzten Buchstaben mit der Filespec übereinstimmen. Dabei ersetzt ein Fragezeichen "?" ein Zeichen, ein Sternchen "\*" den Rest der Zeichenkette bis zum Punkt oder Ende und ein Minuszeichen "-" den gesamten Rest des Dateinamens.

Wildcards können auch von anderen Programmen aus benutzt werden, wenn man sich z.B. nicht mehr an den vollen Namen einer Datei erinnern kann. Es folgen Beispiele für den Einsatz von Wildcards:

'DIR D-' zeigt alle Files an, die "D" als Anfangsbuchstaben haben.

'DIR \*.COM' zeigt alle Files mit dem Extender "COM" an.

'DIR ?!' zeigt alle Files an, deren zweiter Buchstabe ein "I" ist.

Es gibt noch eine Menge anderer Kombinationen von Wildcards, auf die wir aber aus Platzgründen nicht eingehen können. Am besten probiert Ihr selbst einmal aus, was man mit Wildcards alles "herausfiltern" kann.

Neben den Wildcards gibt es bei TURBO-DUP noch andere Möglichkeiten, eine Gruppe von Files im Directory abzugrenzen:

### "Inverse" Filespec

Stellt man unmittelbar vor die Filespec einen "Rückwärts-Schrägstrich" (= Backslash, entspricht <SHIFT>&<+> auf der Tastatur), so werden alle Files außer denen, die durch die Filespec angesprochen sind, angezeigt. Das Kommando 'DIR \D-' z.B. gibt also alle Filenamen auf dem Bildschirm aus, die nicht mit "D" anfangen. Oder 'DIR \\*.COM' präsentiert alle Dateien, die nicht den Extender "COM" haben.

### Nummernorientierte Filespec

Schreibt man vor die Filespec eine Nummer zwischen 1 und 99, so wird eine entsprechende Anzahl von Files, die mit der Filespec übereinstimmen, ignoriert, d.h. es werden nur die Files angezeigt, deren Nummer in der spezifizierten Dateigruppe größer ist als die vorangestellte Zahl. Zwischen der Zahl und der übrigen Filespec muss ein normaler Schrägstrich "/" stehen. Z.B. 'DIR 8/' zeigt von dem neunten ab alle Files auf der Diskette an.

### Device-Kennung

Alle Diskettenbefehle im TURBO-DUP gelten für das aktuelle Laufwerk, d.h. die Diskettenstation, deren Nummer im Prompt "D?>" angezeigt wird. Will man ein Kommando auf eine andere Diskettenstation anwenden, so braucht man hierzu nicht unbedingt die aktuelle Laufwerksnummer umzustellen, sondern man kann die Nummer der Diskettenstation vor die Filespec schreiben. Die Nummer muss durch einen Doppelpunkt oder ein Semikolon von der Filespec abgetrennt werden. So zeigt 'DIR D4:\*.COM' alle COM-Files der Diskette in Floppy Nummer vier an. Folgende Laufwerksnummernangaben sind gleichwertig:

"D8:" = "D8;" = "8:" = "8;" (für "8" kann eine Zahl von 1-8 stehen)  
zusätzlich kann für "D1:" auch "D:" oder "D;" geschrieben werden.

### Anhänge an die Filespec

Durch Anhängen bestimmter Buchstaben mit vorangestelltem Schrägstrich an die Filespec können im Zusammenhang mit einigen Befehlen bestimmte Optionen ausgewählt werden. Eine davon haben wir schon kennengelernt: "/A" schaltet zwischen Ein- und Zweispaltenausgabe beim DIR-Befehl um. Die anderen so einstellbaren Optionen werden in den Kapiteln über die Befehle, in denen sie vorkommen, erläutert. Es können auch mehrere Optionen gleichzeitig ausgewählt werden, indem man einfach die entsprechenden Anhänge aneinander reiht, z.B. 'SAV TEST/A/V/N 2000 2100'.

Alle Elemente der Filespec können auch beliebig kombiniert werden. So zeigt -DIR D2:4\A-' das fünfte File (und alle nachfolgenden) auf der Diskette in der zweiten Floppy an, das nicht mit "D" anfängt. Beim Einsatz aller Komponenten der Filespec ist auf die richtige Reihenfolge zu achten:

Laufwerksnummer-Filenummer-Slash-Backslash-Dateiname-Anhang

Bei TURBO-DUP-Kommandos, die als Parameter nur eine Filespec haben, kann diese auch weggelassen werden. Dies entspricht dann dem Wildcard "-", d.h. es werden alle Files auf der Diskette angesprochen. Es gilt also:

'DIR -' = 'DIR'

'DIR D2:-' = 'DIR 2;'

'DIR D3:-/A' = 'DIR 3;/A'

### 3. LOCK

Mit dem Kommando 'LOCK' kann man Dateien vor Überschreiben schützen. Diese Files können dann nicht mehr verändert oder gelöscht werden, was bei wichtigen Programmen oder Daten sicher sinnvoll ist, um deren unbeabsichtigten Verlust vorzubeugen.

Gibt man im TURBO-DUP einfach "LOCK" ein, so werden nacheinander die Namen aller Files auf der Diskette im aktuellen Laufwerk angezeigt. Hinter jedem Namen folgt das Prompt "Yes, No or Abort ?", mit dem das DUP nachfragt, ob die angezeigte Datei geschützt werden soll.

Drückt man <Y> oder <RETURN>, wird die Operation ausgeführt. Bei <N> oder <SPACE> wird die Datei nicht bearbeitet und bei <A> oder <ESC> die Kommandoausführung abgebrochen.

Im TURBO-DUP tritt dieses Quittierverfahren immer dann auf, wenn die angegebene Filespec (^eingeschränktes DIR) mehr als eine der tatsächlich auf Diskette befindlichen Dateien repräsentiert.

Ist nur ein File vorhanden, auf das die Filespec zutrifft, wird die Operation ohne Nachfragen an diesem ausgeführt. Diese Methode hat sich als sehr bequem erwiesen, vor allem wenn man mehrere Dateien bearbeiten möchte, die sich nicht mit einer gemeinsamen Filespec von den anderen Files auf der Diskette abgrenzen lassen.

Ist man sich sicher, dass man seine Anweisung auf alle spezifizierten Dateien anwenden will, so kann man das Quittieren durch Anhängen von "/N" an die Filespec unterbinden. Die Operation wird dann ohne Nachfrage des DUP bei all diesen Files ausgeführt. Es werden dann lediglich die Namen aller bearbeiteten Files am Bildschirm angezeigt.

Hängt man bei den Befehlen LOCK.UNLOCK und DELETE ein "/D" an die Filespec an, so wird das Kommando direkt ans FMS übergeben und dann ohne Nachfrage ausgeführt. Dies geht schneller als die Ausführung mit "/N", allerdings kann man in der Filespec nur die vom FMS zugelassenen Wildcards "?", "\*" und "-" verwenden, und es erfolgt außerdem keine Anzeige der bearbeiteten Files.

Vor jeder Ausführung eines Diskettenkommandos liest TURBO-DUP das (eingeschränkte) Directory ein, um dem Benutzer die Auswahl zu ermöglichen bzw. um die Namen der tatsächlich bearbeiteten Files anzeigen zu können. Bei "/D" werden keine Filenamen eingelesen, daher kann auch keine Rückmeldung auf dem Bildschirm erfolgen.

Der Anhang "/V" an die Filespec schaltet bei allen Befehlen des DUP den "Verify"-Modus ein. Hierbei wird jedes Mal, wenn das FMS Daten auf einen Diskettensektor schreibt, nachgeprüft, ob die Daten auch richtig auf der Diskette angekommen sind. Dies dauert natürlich länger als das Schreiben ohne Verifikation. Verify lohnt sich nur, falls man seiner Floppy oder seinen Disketten nicht mehr traut, denn Datenfehler treten erfahrungsgemäß selbst bei Billig-Disketten nur selten auf, sofern diese nicht schon beim Formatieren als unbrauchbar erkannt wurden.

Wird von anderen Programmen aus versucht, eine geschützte Datei zu verändern, so tritt ein Fehlersituation ein, auf die das Programm irgendwie reagiert (oder auch nicht). Das TURBO-DUP gibt in einer solchen Situation hinter dem Dateinamen die Meldung "is locked!" aus.

Anschließend folgt die Frage "Unlock file ? Yes, No or Abort ?". Durch deren Beantwortung entscheidet der Benutzer, ob der Dateischutz automatisch vor Ausführung des eigentlichen Kommandos aufgehoben werden soll. Die Tastenbelegung beim Yes/No-Prompt ist im TURBO-DUP einheitlich, wie oben bereits erläutert. Sagt man "No", bleibt die Datei unangetastet, und es wird versucht, das Kommando an der nächsten Datei auszuführen.

Doch zurück zu unserem LOCK-Kommando: Die ge-"lock"-ten Dateien werden im Directory mit einem Sternchen vor dem Namen ausgegeben. Ungeschützte Files tragen keinen Stern.

#### 4. UNLOCK

Geschützte Dateien können mit dem UNLOCK-Befehl wieder freigegeben werden. Dabei verschwindet das Sternchen vor dem Dateinamen im Directory. Die Eingabe bei 'UNL' erfolgt nach dem gleichen Prinzip wie bei LOCK.

#### 5. RENAME

Mit diesem Kommando kann man Namen von bestehenden Dateien verändern. Die Syntax bei der Befehlseingabe ist hier: 'REN fspec.fname' (statt des Kommas kann auch ein Leerzeichen stehen). Der erste Parameter "fspec" hinter dem Befehlskürzel ist eine Filespec und legt fest, welche Datei umbenannt werden soll. Als zweiten Parameter "fname" muss man dahinter eine Dateibezeichnung angeben, die den neuen Namen der Datei bestimmt. Werden durch "fspec" mehrere Dateien angesprochen, so erfolgt das Umbenennen mit Quittierung. Der neue Name für die Dateien kann auch Wildcards enthalten; die so ersetzten Buchstaben des alten Dateinamens bleiben dann unverändert. Beispiele:

```
'REN DUP.SYS,TEST.COM' : benennt "DUP.SYS" in "TEST.COM" um.  
'REN D-,W-'           : wandelt die Anfangsbuchstaben "D" in "W" um.  
'REN *.OBJ,*.COM'     : wandelt die Extender "OBJ" in "COM" um.
```

Wenn man einmal aus Versehen durch RENAME mehrere Files mit demselben Namen versehen hat, kann man entweder nur noch auf das erste File oder aber auf alle Files dieses Namens im Directory zugreifen. Um die anderen Files wieder zugänglich zu machen, muss man sie der Reihe nach mit RENAME einzeln auf verschiedene Namen "umtaufen".

## 6. DELETE

Mit DELETE kann man Dateien löschen. Auf diese Weise gelöschte Files erscheinen vom DUP aus im Directory nicht mehr. Der zurückgewonnene Speicherplatz schlägt sich in der vergrößerten Anzahl der freien Sektoren nieder. Die Befehlseingabe erfolgt im Standardformat, d.h. in der Form 'DEL fspec', wobei "fspec" die zu löschenden Dateien bezeichnet. Auch hier wird bei mehreren angesprochenen Files einzeln nachgefragt, ob diese gelöscht werden sollen oder nicht.

Mit DELETE versehentlich gelöschte Dateien können unter bestimmten Umständen mit dem Programm "DISKFIX.COM" wieder verfügbar gemacht werden (-> "Disketten retten"),

## 7. CLEAR

Mit dem Befehl 'CLR' werden alle Files (also auch die ge"lock"ten) auf einer Diskette gelöscht. Nach der Eingabe des Befehlskürzels 'CLR' folgt hinter der Frage "Sure ?" das bereits bekannte Yes/No-Prompt. Mit dieser Sicherheitsabfrage soll sichergestellt werden, dass allzu "destruktive" Operationen nicht aus Versehen ausgeführt werden. Als Parameter kann man hinter 'CLR' die Nummer des Laufwerks angeben, in der sich die zu löschende Diskette befindet. z.B. 'CLR 2;' löscht die Diskette in Laufwerk #2.

CLEAR ist um ein Vielfaches schneller als das Löschen aller Files mit DELETE (z.B. durch 'DEL / D') und auch schneller als das Formatieren mit den nachfolgend erklärten Befehlen 'FMS', 'FME' oder 'FMD'.

(? Bei CLEAR werden einfach Directory und VTOC gelöscht, wodurch die Diskette für das FMS wieder wie "frisch formatiert" aussieht. )

Hat man einmal aus Versehen eine Disk mit 'CLR' gelöscht, kann man die Daten mit der Funktion "Recover Directory" des DISKFIX-Programms wiederherstellen. Mehr dazu im entsprechenden Kapitel.

## 8. Formatierbefehle

Im Handel erhältliche, "frische" Leerdisketten können von der Floppy nicht einfach so gelesen oder beschrieben werden; sie müssen zuvor formatiert werden. Hierbei wird die Magnetschicht auf der Disk in verschiedene Speicherbereiche (Spuren und Sektoren) aufgeteilt.

Näheres hierzu entnehmt Ihr bitte dem Kapitel über die Diskettenformate im Artikel "Allgemeine Informationen über DOS ..." auf Seite 7 .

Die Formatierbefehle im TURBO-DUP lauten:

'FMS': formatiert in Single Density

'FME': formatiert in Enhanced Density (auch Medium oder Dual Density genannt)

'FMD': formatiert in Double Density

'FMQ': formatiert in Quad Density

Da beim Formatieren einer Diskette alle Daten unwiederbringlich gelöscht werden, erfolgt auch bei diesen Befehlen eine Sicherheitsabfrage wie beim 'CLR'-Kommando. Die Befehlseingabe erfolgt ebenfalls nach dem gleichen Muster.

Falls die Diskettenstation die Diskette nicht ordnungsgemäß formatieren kann, erscheint die Meldung "ERROR \*AD=.173". Sollte sich diese Diskette nach einem zweiten Formatierungsversuch weiterhin als unformatierbar erweisen, solltet Ihr sie am besten sofort wegwerfen.

## 9. INIT

Damit man DOS und DUP benutzen kann, müssen diese natürlich auf Diskette vorhanden sein. Mit anderen Worten: DOS und DUP müssen zuvor einmal auf die Diskette geschrieben worden sein. Dieses genau bewerkstelligt der INIT-Befehl.

Die Eingabe erfolgt wie bei 'CLR' und den Formatierbefehlen, d.h. optional unter Angabe der Laufwerksnummer. Es werden dann die gerade im Speicher befindlichen Versionen von DOS und DUP abgespeichert.

Zweckmäßigerweise führt man dieses Kommando direkt nach dem Formatieren oder "Clear"-en einer Disk aus. 'INIT' kann aber auch dann ausgeführt werden, wenn vorher schon andere Dateien auf der Diskette vorhanden sind (auch "DOS.SYS" oder "DUP.SYS"). Da man auf vielfältige Weise DOS und DUP seinen eigenen Belangen anpassen kann (-> BACK, MODE), ist es möglich, das aktuelle DOS und DUP auch getrennt auf Diskette zu schreiben: 'INI /D' schreibt das "DOS.SYS", 'INI /N' das "DUP.SYS".

## 10. COPY

Mit dem COPY-Befehl kann man Dateien umkopieren (oder allgemeiner: Dateien zwischen verschiedenen Geräten austauschen). Das Eingabeformat bei diesem Befehl lautet:

'COP source,dest', wobei, ähnlich wie beim RENAME-Befehl, der erste Parameter (source) eine Filespec und der zweite (dest) ein Dateiname sein muss. "Source" spezifiziert die zu bearbeitende(n) Datei(en), wie Ihr das schon von anderen Befehlen her kennt; d.h. hier könnt Ihr mit Hilfe von Laufwerksnummer, Filenummer, Wildcards usw. angeben, welche Files Ihr kopieren möchtet.

Der zweite Parameter "dest" legt fest, wohin die zuvor spezifizierten Dateien kopiert werden sollen. Gibt man hier nichts oder "-" an, so werden die Files unter Beibehaltung ihres Namens auf das aktuelle Laufwerk kopiert. Das Ziellaufwerk kann man in gewohnter Weise auch durch Angeben der Drive-Nummer vor dem Dateinamen "dest" bestimmen. Durch Angabe eines solchen Dateibezeichners kann man die Files unter anderem Namen abspeichern. Enthält "dest" Wildcards, so werden bei jeder kopierten Datei alle Buchstaben, außer den durch Wildcards ersetzten, in gewünschter Weise abgeändert. So kopiert z.B. der Befehl:

'COP D1:\*.COM,D2:\*.BAK' Files mit dem Extender "COM" aus Floppy #1 auf das zweite Laufwerk, wo sie dann mit dem neuen Extender "BAK" erscheinen. Soviel zu den Eingabeparametern.

Beim Kopiervorgang selbst werden zunächst die zu kopierenden Files (evtl. mit Quittierung) nacheinander in den Speicher geladen. Erst danach werden die nun im Speicher befindlichen Dateien (evtl. Unter anderem Namen) auf die Zieldiskette abgespeichert. Falls Quell- und Ziellaufwerk identisch sind, folgt zuvor das Prompt "Insert destition & hit key !". Dadurch ist es möglich, mit nur einem Laufwerk zwischen zwei verschiedenen Disketten zu kopieren. Nach dem Prompt nämlich hat man Gelegenheit, die Zieldiskette einzulegen. Ein beliebiger Tastendruck teilt dann dem DUP mit, dass das Schreiben der Daten beginnen kann. Quell- und Zieldiskette können natürlich auch gleich sein, wenn man zum Beispiel Sicherheitskopien unter anderem Namen auf ein- und derselben Diskette anfertigen oder aber zur Platzersparnis (-> COPY-Option "/A") Files auf sich selbst kopieren möchte.

Da das DUP aber nicht "riechen" kann, ob Ihr auf dieselbe oder eine andere Diskette kopieren möchtet, folgt das Insert-Prompt auf jeden Fall, also auch dann, wenn die Zieldisk mit der Quelledisk identisch und daher bereits eingelegt ist. Wenn nicht alle zu kopierenden Files in den Speicher passen, erfolgt das Kopieren in mehreren Durchgängen. Bei Benutzung nur eines Laufwerks wird Euch dabei angezeigt, wann Ihr die Diskette wechseln sollt ("Insert destination"="Zieldiskette einlegen"; "Insert source" = "Quelldiskette einlegen").

Alle Lese- und Schreibvorgänge werden im TURBO-DUP durch Ausgabe des momentan bearbeiteten Filenamens dokumentiert.

Beim COPY-Befehl gibt es eine Reihe von Optionen, die man durch Anhängen von Buchstaben an den Dateinamen auswählen kann. Neben "/N" zum Ausschalten des Quittierens und "/V" zum Einschalten von Verify sind hier noch die Anhänge "/A" und "/D" von Bedeutung.

Beim Anhängen von "/A" oder "+" erfolgt das Schreiben der Dateien im "Append"-Modus, d.h. sie werden an ein bereits bestehendes File angehängt, dessen Name durch den zweiten COPY-Parameter "dest" gegeben ist. Z.B. das Kommando 'COP \*.TXT,ALLE.TXT/A' kopiert alle Files mit dem Extender "TXT" auf die Datei "ALLE.TXT" zusammen. Da das FMS beim Anhängen von Daten an eine Datei auf jeden Fall einen neuen Sektor "anbricht", wird auf der Diskette dadurch ein bisschen Speicherplatz vergeudet. Aus diesem Grunde erscheint zumeist die Kopie eines mit "Append" zusammengesetzten Files mit verringerter Sektoranzahl.

Beim Anhängen von "/D" erhalten die Wildcards im "dest"-Parameter eine andere Bedeutung: Vorher wurden bei den Namen der zu schreibenden Dateien die durch Wildcards vertretenen Buchstaben einfach aus den Quelldateinamen übernommen. Bei "/D" als Anhang jedoch erfolgt das Kopieren ohne Umbenennen, stattdessen wird der Zielname mit den Wildcards ans FMS übergeben. Die Quelldateien werden somit auf eine (evtl. bereits bestehende) durch die Filespec "dest" festgelegte Datei auf der Zieldiskette kopiert. 'COP TEST,2;R-/D' z.B. kopiert die Datei "TEST" aus dem aktuellen Laufwerk auf das erste File von Drive #2, das mit "R" anfängt. Der gleiche Befehl ohne "/D" bewirkt das Kopieren auf die Datei "REST". Alle Optionen können auch beliebig gemischt werden:

'COPY \*.COM/N.AL-/D/A/V bewirkt, dass ohne Nachfragen alle COM-Files mit Verify an die erste mit "AL" anfangende Datei der Zieldiskette angehängt werden.

Arbeitet man mit "MEM.SAV", so wird zum Kopieren nur der geschützte Speicherbereich als Puffer benutzt. Wegen dessen geringer Größe erfordert der Kopiervorgang bei großen Dateien mehrere Durchgänge, was länger dauert und außerdem bei Benutzung nur eines Laufwerks ein ständiges Wechseln von Ziel- und Quelldiskette bedeutet. Für das Kopieren von vielen oder großen Files ist es daher ratsam, den Puffer durch Desaktivieren von "MEM.SAV" zu vergrößern. Dieses erreicht man am einfachsten durch Eingabe der Befehlszeile '>711 0' (-> Monitor-Befehle). Weil durch dieses Rücksetzen des MEM.SAV-Flags im FMS dem DUP der gesamte Rest des Speichers zur Verfügung steht, geht der Kopiervorgang erheblich schneller vonstatten. Natürlich werden dabei evtl. in diesen Speicherbereichen befindliche Daten (z.B. BASIC-Programm) zerstört. Man kann das DUP auch so einstellen, dass beim Kopieren generell der gesamte Speicher benutzt wird (-> MODE).

Der COPY-Befehl kann allerdings noch mehr als bloß Files zwischen Diskettenstationen austauschen. Bei den Parametern "source" und "dest" nämlich kann man jedes beliebige Gerät angeben, vorausgesetzt, es ist angeschlossen und dem Betriebssystem bekannt. Für das Betriebssystem werden die Ein- und Ausgabegeräte durch sogenannte Handler repräsentiert, die ihrerseits durch jeweils einen bestimmten Buchstaben identifiziert werden (z.B. "E"=Editor, "S"=Screen, "K"=Keyboard, "P"=Printer, "C"=Cassettenlaufwerk, "D"=Diskettenstation).

Um diese Geräte von Disketten-Files zu unterscheiden, muss man bei der Eingabe vom DUP aus hinter das Handler-Kennzeichen einen Doppel- oder Strichpunkt stellen. Es folgen einige Beispiele zum Datenaustausch zwischen verschiedenen Geräten:

'COPY D2:HELP.TXT.P:' gibt die Datei "HELP.TXT" aus Laufwerk 2 auf den Drucker (sofern vorhanden) aus. Steht "S:" statt "P:", so erfolgt die Ausgabe auf den Bildschirm.

'COPY C:,BACKUP.CAS' kopiert einen Datensatz aus der Cassettenstation auf das File "BACKUP.CAS" der aktuellen Floppy.

'COPY E:,D8:SETUP.BAT' kopiert Textzeilen aus dem Bildschirmeditor auf das RD-File "SETUP.BAT". Nach Eingabe dieses Befehls erscheint zunächst ein leerer "Graphics-0"-Bildschirm in gewohntem ATARI-Blau. Jetzt kann man über die Tastatur Textzeilen eingeben, wie man das vom eingebauten ATARI-BASIC her kennt. Wichtig ist es, am Ende jeder Zeile <RETURN> zu drücken, damit diese in den Kopierpuffer aufgenommen wird.

Um die Texteingabe abzuschließen, muss man die Taste <CONTROL>&<3> drücken, wodurch der Bildschirmeditor geschlossen, das Einge Tipperte ans DUP übergeben und von dort aus schließlich auf das gewünschte File abgespeichert wird.

Bei der Beantwortung der während des COPY-Vorgangs angezeigten Prompts ist zu beachten, dass bei einem Abbruch mit <BREAK>, <A> oder <ESC> die eingelesenen Daten evtl. noch nicht vollständig auf das Zielgerät geschrieben worden sind. diese Dateien also nicht komplett kopiert

wurden! Sicherheitshalber solltet ihr daher alle Prompts mit Ja oder Nein beantworten, damit nichts schief gehen kann.

## 11. LOAD

Mit dem Kommando LOAD werden Maschinenprogramme geladen und gestartet. Die Syntax lautet 'LOAD fspec', wobei "fspec" das zu ladende Maschinensprache-File spezifiziert. Erweist sich dieser Parameter als mehrdeutig, so fragt das DUP in gewohnter Manier nach, welches Programm geladen werden soll. Nachdem man sich für eine Datei entschieden hat, wird diese in den Speicher geladen und ggf. initialisiert und / oder gestartet. Bei Anhängen von "/D" and den Parameter "fspec" unterbleiben beide letzteren Aktionen, d.h. das Programm wird lediglich in den Computer geladen.

Nach Ablauf des gestarteten Maschinenprogramms bzw. nach Beendigung des Ladevorgangs bei nicht gestarteten Programmen übernimmt normalerweise das DUP wieder die Kontrolle über das Geschehen im Computer. Wurden Teile des DUP durch das Programm zerstört, ist ein sicheres Funktionieren der Benutzeroberfläche natürlich nicht mehr gewährleistet. Hier hilft dann nur noch <RESET>.

Da das Laden und Starten von Maschinenprogrammen eine häufig gebrauchte Funktion ist, gibt es bei TURBO-DUP eine Variante, die die Tipparbeit bei der Befehlseingabe ein wenig erleichtert: Ihr könnt nämlich das Befehlskürzel 'LOA' vor dem Programmnamen weglassen.

Stattdessen müsst Ihr durch Abschluss der Befehlszeile mit <SHIFT>&<RETURN> dem DUP mitteilen, dass es sich um einen Ladebefehl handelt.

Anm.: Die meisten anderen DOS mit Kommandozeile lassen ebenfalls diese Abkürzung des Ladebefehls zu, allerdings ohne Benutzung von <SHIFT>. Dieses hat zum einen den Nachteil, dass jedes Mal, wenn man sich bei der Befehlseingabe vertippt, sogleich versucht wird, ein File dieses vermeintlichen Namens zu laden.

Außerdem kann man mit dieser Methode keine Files laden, die den Namen eines DUP-Befehls tragen, wie z.B. "DIR.COM". Bei TURBO-DUP werden beide Nachteile durch den Einsatz von <SHIFT> auf simple Art umgangen.

Als weitere Tipperleichterung wird beim Abschluss der Kommandozeile mit <SHIFT>&<RETURN> automatisch die Zeichenkette "\*.COM" an den Filenamen angehängt, was den Vorteil hat, dass man zum Laden von Maschinenprogrammen mit dem Extender "COM" nur noch einen signifikanten Teil des Namens angeben muss. Z.B. 'DISKF <SHIFT>&<RETURN>' startet bei eingelegter TURBO-DOS-Masterdisk das Utility "DISKFIX.COM" zum "Reparieren" von "software-geschädigten" Disketten.

Wie aber sehen Dateien aus, die man mit dem LOAD-Befehl laden und starten kann? Im Prinzip sind solche Files nichts anderes als auf Diskette fixierte Speicherbereiche des Computers. Da Maschinenprogramme für den 8-Bit-ATARI im allgemeinen an einen festen Ort im Speicher gebunden sind, muss zusätzlich zu dem Inhalt dieses Bereiches auch noch festgehalten werden, an welcher Stelle er im Speicher liegt.

Konkret sieht das so aus: am Anfang eines ladbaren Files stehen vier Bytes, von denen die ersten beiden die Anfangs- und die anderen beiden die Endadresse des Speicherbereiches darstellen. Im Anschluss an diesen sogenannten Header folgen dann Byte für Byte die Inhalte aller Speicherzellen zwischen diesen beiden Grenzen. Ein kleines Beispiel soll Klarheit schaffen (alle Zahlenangaben hexadezimal): Nehmen wir an, im Speicher läge ab Adresse AB12 das aus fünf Bytes bestehende Maschinenprogramm:

2C 12 34 EA 60. Das entsprechende "Speicher-File" würde sich dann wie folgt zusammensetzen:

12 AB	16 AB	2C 12 34 EA 60	(2+2+5=9 Bytes)
Anfangs- Adresse	End- adresse	Inhalt des Speichers von AB12 bis einschließlich AB16	

Wir kennen nun die Struktur eines abgespeicherten Maschinenprogramms. Um es nun auch ausführen zu können, muss dem Computer die Speicherstelle, an der das Programm gestartet werden soll, mitgeteilt werden. Es gibt bei den 8-Bit-ATARI's eine Konvention, nach der diese sog. RUN-Adresse in speziell dafür vorgesehene Speicherstellen (\$2E0-\$2E1) eingetragen werden muss. Diese zwei Speicherzellen können nämlich einfach durch Laden eines entsprechenden Speicherfiles der oben dargestellten Art besetzt werden. Ein Maschinenprogramm wird also durch Laden der (zuvor abgespeicherten) RUN-Adresse in die Speicherzellen \$2E0 und \$2E1 gestartet. In der Praxis muss man diese zwei Ladevorgänge nicht getrennt in die Wege leiten. Das DOS lädt nämlich selbständig Files, die (z.B. durch Kopieren im Append-Modus) aus mehreren Speicherfiles zusammengesetzt sind, nacheinander in den Speicher.

Solche "komponierten" Dateien bezeichnet man als "compound-" oder kurz "COM-Files", woher dann auch der Extender "COM" für selbststartende Maschinenprogramme rührt. Damit COM-Files für das DOS als solche erkennbar sind, tragen sie am Anfang eine Kennung, und zwar zwei Bytes mit dem Wert 255=\$FF. Die durch das Zusammenkopieren von COM-Files auftretende Häufung dieser Kennbytes innerhalb der so entstandenen neuen Datei bringt das DOS nicht in Verwirrung, weil sie

einfach ignoriert werden. Die zwei \$FF-Bytes am Anfang des Files sind jedoch unverzichtbar. Ich möchte noch auf ein weiteres Merkmal von COM-Files hinweisen:

Neben der RUN-Adresse zum Starten gibt es noch die sog. INIT-Adresse, mit der man ebenfalls Maschinenprogramme aufrufen kann. Während die RUN-Adresse in \$2E0 und \$2E1 erst nach dem Laden aller Teile des COM-Files "angesprungen" wird, erfolgt beim Laden der INIT-Adresse nach \$2E2 und \$2E3 dieser Programmaufruf sofort. Falls das so gestartete Programm ordnungsgemäß abgelaufen ist, fährt das DOS mit der Bearbeitung des COM-Files fort, um dann an dessen Ende zur eigentlichen Startadresse in \$2E0/\$2E1 zu springen. (Werden im Verlauf eines COM-Files mehrfach RUN-Adressen nach \$2E0/\$2E1 geladen, so ist nur die zuletzt eingetragene maßgebend.)

Auf diese Weise kann man eine nur durch den Speicherplatz begrenzte Kette von Maschinenprogrammen automatisch nacheinander laden und ausführen lassen.

## 12. SAVE

Der Befehl "SAVE" dient zum Abspeichern von Speicherblöcken. Wie im Kapitel über den LOAD-Befehl bereits ausgeführt wurde, gibt es zur Speicherung von Maschinenprogrammen bzw. Speicherbereichen eine spezielle Form von Dateien, die als COM-Files bezeichnet werden. Das Kommando 'SAVE fspec,adr1,adr2' speichert nun den Speicherblock zwischen den beiden Adressen "adr1" (untere Grenze) und "adr2" (obere Grenze) auf die durch "fspec" festgelegte Datei in Gestalt eines COM-Files ab. Die Adressen können wie alle Zahleneingaben im TURBO-DUP entweder in hexadezimaler oder dezimaler Schreibweise angegeben werden, wobei zur Unterscheidung vor Dezimalzahlen ein Punkt "." stehen muss (-> Monitorbefehle).

Nach korrekter Eingabe des SAVE-Kommandos (auch hier können alternativ zu den Kommas Leerzeichen stehen !) hat man die Möglichkeit, dem COM-File gleich eine RUN- und INIT-Adresse hinzufügen zu lassen. Hinter den nacheinander erscheinenden Prompts "INIT:" und "RUN:" erwartet das DUP jeweils eine Zahleneingabe. Bei Eingabe einer Leerzeile wird die entsprechende Adresse nicht an das File angehängt.

Nachdem man beide Prompts beantwortet hat, wird der Speicherblock ggf. mit RUN/INIT-Adresse auf das File "fspec" geschrieben. Diese Datei kann von nun an mit LOAD wieder geladen (und gestartet) werden .

TURBO-DUP bietet auch die Möglichkeit, einen Speicherblock unter einer anderen Bestimmungsadresse abzuspeichern, was aber nur für Assemblerprogrammierer interessant sein dürfte. Gibt man hinter den beiden Blockgrenzen "adr1,adr2" eine weitere Adresse an, werden die Headerbytes entsprechend verändert, so dass das File beim späteren Laden an eben diese neue Adresse geladen wird.

Fügt man an den SAVE-Parameter "fspec" "/"A" oder "+" an, so werden die Daten an eine bereits bestehende Datei dieses Namens angehängt. "/"V" schaltet wie gewohnt Verify ein, während "/"N" eine andere Bedeutung als bei anderen DUP-Kommandos hat: ein dem "fspec" beigefügtes "/"N" bewirkt, dass der Speicherbereich nicht als COM-File, sondern als einfaches Datenfile, d.h. ohne die Headerbytes (Anfangs- und Endadresse) abgespeichert wird. Bei dieser Option sind INIT- und RUN-Adresse unsinnig, weshalb sie auch in diesem Falle vom DUP nicht abgefragt werden. Ein mit "/"N" abgespeichertes File ist kein COM-File und kann folglich auch nicht wieder mit dem LOAD-Befehl geladen werden!

### 13. HEAD

Das Kommando 'HEAD fspec' zeigt die Ladebereiche (Header) des durch "fspec" angesprochenen COM-Files an. Bei zusammengesetzten COM-Files werden die einzelnen Teilblöcke durchnummeriert. Ferner erfolgt eine Anzeige von RUN/INIT-Adressen, sofern diese beigefügt wurden. Alle Adressen werden in der hierfür üblichen hexadezimalen Schreibweise ausgegeben. Dieses Kommando ist wohl in erster Linie interessant für Leute, die in Maschinensprache etwas bewandert sind.

### 14. RUN

Mit dem Befehl 'RUN adr' wird ein Maschinenprogramm an der durch "adr" bestimmten Speicheradresse gestartet. Endet das Programm regulär (mit dem Maschinenbefehls "RTS" bei aufgeräumtem Stack) und wurde das DUP nicht zerstört, so befindet man sich hiernach wieder im TURBO-DUP. Lässt man den Parameter "adr" weg, so wird das zuletzt mit LOAD geladene Maschinenprogramm wieder-gestartet, d.h. es erfolgt ein indirekter Sprung über die COM-File-Startadresse in \$2E0/\$2E1.

### 15 SUM

Der Befehl 'SUM fspec' gibt Länge und Prüfsumme der durch "fspec" spezifizierten Datei(en) aus. Bei der Ausführung dieses Befehls zählt das DUP alle in dem File enthaltenen Bytes und summiert deren Werte. Anschließend werden die so ermittelten Zahlen hexadezimal auf dem Bildschirm ausgegeben: hinter "LEN:" erscheint die Anzahl und hinter "SUM:" die Summe der Bytes. Das SUM-Kommando hat sich als hilfreich erwiesen, um Files miteinander zu vergleichen.

### 16. TYPE

Durch das Kommando 'TYPE fspec' wird das File "fspec" in Form von ATASCII-Zeichen auf dem Bildschirm ausgegeben. Dabei werden CONTROL-Codes wie z.B. Bildschirmlöschen nicht ausgeführt, sondern durch die dafür vorgesehenen Sonderzeichen dargestellt. Wie bei allen anderen Befehles des TURBO-DUP erfolgt auch hier die Bildschirmausgabe Seitenweise, d.h. nach Ausgabe einer Seite folgt das schon erwähnte More-Prompt. Bei den drei Eile-Kommandos HEAD, SUM und TYPE hat die <SPACE> eine besondere Bedeutung: nach Betätigung wird die Ausgabe

des momentan bearbeiteten Files abgebrochen und ggf. das nächste im Quittierzyklus in Angriff genommen.

Fügt man "/D" an den Parameter "fspec" an, so werden die CONTROL-Zeichen wie beim Kopieren von Files auf den "E:"-Handler ausgeführt.

### 17. BON/BOFF

Diese beiden Befehle zum Ein- bzw. Ausschalten des eingebauten BASIC bedürfen wohl keiner weiteren Erläuterung, bis auf den Hinweis, dass bei eingeschaltetem BASIC der vom DUP zum Kopieren als Puffer benutzbare Speicherbereich um 8 KB kleiner ist als ohne BASIC.

### 18. CARTRIDGE

Der Befehl 'CAR' verlässt das DUP und übergibt die Kontrolle an ein ggf. eingestecktes Modul oder das eingebaute BASIC, falls dieses eingeschaltet ist. Liegt keiner dieser beiden Fälle vor, wird an eine durch den Benutzer bestimmbare Speicheradresse gesprungen (-> BACK). Optional kann man hinter dem Befehlskürzel "CAR" noch ein Kommando an das kontrollübernehmende System übergeben. Voraussetzung hierzu ist, dass dieses System seine Eingaben über den Bildschirmeditor vornimmt. Ein Beispiel hierzu: 'CAR RUN "D:DEMO.BAS"' geht ins BASIC (sofern eingeschaltet) und startet von dort aus sofort das Programm "DEMO.BAS" aus Floppy #1.

Falls man mit "MEM.SAV" arbeitet, wird beim Verlassen des DUP der auf diese Datei gesicherte Speicherbereich wieder in den Computer zurückgeladen, so dass der Computerspeicher wieder wie vor Aufruf des DUP aussieht.

Statt 'CAR' kann man auch einfach die Taste <OPTION> drücken.

### 19. BACK

Dieser Befehl arbeitet mit dem CAR-Kommando zusammen. Findet dieses nämlich kein Modul oder BASIC vor, so wird die in den Speicherstellen S12FC und S12FD stehende Adresse angesprungen.

Mit 'BACK' kann man sich nun diese Adresse anzeigen lassen. 'BACK adr' hingegen trägt die Zahl "adr" in diese Speicherzellen ein.

Durch den BACK-Befehl kann man das DOS speziell auf Programme einstellen. Wenn man diese Einstellung einmal vorgenommen hat, pflanzt sie sich bei jedem Schreiben des DOS auf Diskette (-> INIT) fort und ist auch nach deren nächstem Booten noch wirksam.

Hier nun eine Liste der Einsprungadressen einiger bekannter Programme.

TURBO-BASIC XL: \$2080; ATMAS-II: \$2800; DESIGN-MASTER: S1D00.

Die hier erwähnten Programme können mit 'CAR' natürlich nur dann wieder-gestartet werden, wenn sie vor Aufruf des DUP im Speicher lagen und wenn mit "MEM.SAV" gearbeitet wird.

Ein DOS mit spezieller BACK-Adresse sollte sich nur auf Disketten befinden, die auch wirklich etwas mit der von ihm unterstützten Software zu tun haben. Arbeitet man nämlich nicht mit dem vom momentanen DOS unterstützten Programm, so führt der CAR-Befehl zumeist zum System-Absturz, da sich an der BACK-Adresse nicht der richtige Objekt-Code befindet. Zur Erstellung eines allgemeinen, unbedenklich verwendbaren DOS sollte man 'BACK 0' eingeben. Bei diesem Wert nämlich kann man im Falle des nicht eingeschalteten Moduls das DUP mit Hilfe des CAR-Befehls nicht mehr verlassen.

### 20. MODE

Mit diesem Befehl ist es möglich, das TURBO-DUP eigenen Belangen anzupassen. Er beeinflusst ein im DUP-Programmspeicher liegendes Byte, durch dessen Wert einige Verhaltens-

weisen der Benutzeroberfläche bestimmt werden. Hier zunächst die Bedeutung der 8 Bits dieses Modus-Bytes :

Bit #0 (1): Falls dieses Bit gesetzt ist, wird das Quittieren ausgeschaltet. Spricht also eine Filespec mehr als nur eine Datei an, so wird die Operation ohne Nachfrage an all diesen Files ausgeführt.

Das DUP verhält sich dann so, als hätte man ein "/N" an die Filespec angehängt. Bei den Formatierkommandos kann man die Nachfrage des DUP allerdings nicht unterdrücken.

Bit #1 (2): Mit diesem Bit wird der Verify-Modus eingeschaltet. Ist es gesetzt, so erfolgen alle Schreiboperationen mit sofortiger Überprüfung. Dies entspricht dem Anhang "/V" an alle Filenamen.

Bit #2 (4): Durch Setzen dieses Bits wird beim COPY-Kommando der gesamte freie Speicherplatz als Puffer benutzt. Bei Benutzung von "MEM.SAV" wird dieses aber nur dann deaktiviert, wenn man mit Hilfe des COPY-Befehls Daten kopiert. Alle anderen Operationen greifen nach wie vor nur auf den mit "MEM.SAV" geschützten Speicherbereich zu.

Bit #3 (8): Falls dieses Bit gleich eins ist, werden mit LOCK geschützte Dateien vor dem Beschreiben ohne Nachfrage selbständig "ent"schützt.

Bit #4 (16) : Durch Setzen von Bit #4 wird die akustische Untermalung des Diskettenzugriffs (Beep-Beep) ausgeschaltet.

Bit #5 (32): Durch Setzen dieses Bits unterbleibt der DUP-ERROR-Ton.

Bit #6 (64): Ist Bit #6 auf eins, so erfolgt die Bildschirmausgabe im TURBO-DUP nicht mehr Seitenweise. Stattdessen "scrollt" der Text durch, wie man das vom BASIC her gewohnt ist.

Bit #7 (128) ist unbenutzt.

Gibt man 'MODE' ein, so wird das Modus-Byte hexadezimal auf dem Bildschirm angezeigt. Das Kommando 'MODE byte' setzt den Modus auf "byte". Der Wert "byte" ergibt sich folgendermaßen: Für jedes gesetzte Modus-Bit müsst ihr die im obigen Text jeweils hinter den Bit-Nummern in Klammern befindliche Dezimalzahl aufschreiben und anschließend alle diese Zahlen addieren. Die Summe ist dann das einzugebende Modus-Byte für eure gewählte Konfiguration. Achtet aber bitte darauf, dass ihr bei dezimaler Zahleneingabe eine Punkt "." direkt vor die Zahl setzen müsst. Z.B. 'MODE .34' bedeutet: Verify ein und ERROR-Ton aus. Das Modus-Byte auf eurer Masterdisk ist auf null gesetzt, da diese Kombination für "Uneingeweihte" wohl am günstigsten ist. Beim Schreiben des "DUP.SYS"-Files mit dem Befehl 'INIT /N' wird das momentane Modus-Byte mitabgespeichert und ist dann beim nächsten Laden des DUP von dieser Diskette von vornherein auf diesen Wert gesetzt.

## 21. VERSION

Das Kommando 'VER' zeigt an, um welche Version es sich bei dem momentan im Speicher befindlichen DOS (FMS) handelt (-> Seite 6).

Dabei gilt folgende Nomenklatur:

VER 2.1 : Normalversion.

VER 2.1 HS : Version für 1050 mit HAPPY-Enhancement oder SPEEDY 1050.

VER 2.1 XF : Version für XF 551.

VER 2.1 EX : Vollversion (alle drei Versionen).

## 22. 1-8: DRIVE NUMBER

Durch Eingabe einer Ziffer zwischen "1" und "8" kann die Nummer des aktuellen Laufwerks bestimmt werden. Ist dies geschehen, erscheint das Prompt "Dn>" mit der neu eingegebenen Num-

mer, wodurch sich für das DUP alle Filenamen ohne Laufwerksangaben auf dieses Drive beziehen.

### 23. Monitor-Befehle

Zusätzlich zu den normalen Kommandos ist in das TURBO-DUP noch ein kleiner Maschinensprache-Monitor integriert, mit dem man bequem vom DUP aus Speicherinhalte anschauen und verändern kann. Er ist vor allen Dingen für die Leute gedacht, die in Maschinensprache schon etwas bewandert sind. Doch zunächst etwas über die Eingabe von Zahlen im TURBO-DUP: Alle Zahlen können dezimal, hexadezimal oder als ATASCII-Zeichen eingegeben werden. Hexadezimalzahlen bedürfen keiner besonderen Kennzeichnung, da diese Schreibweise für die im DUP häufig dargestellten Speicheradressen üblich ist. Dezimalzahlen müssen durch einen Punkt "." vor der ersten Ziffer gekennzeichnet werden. ATASCII-Zeichen müssen von einem Apostroph "' ' " angeführt werden. Es folgen Beispiele für Zahleneingaben im TURBO-DUP:

Hexadezimal : A3 FFFD  
Dezimal : .13 .255 .19834  
ATASCII-Code : 'a '8 '&

Weiterhin kann man Zahlen auch durch einen Offset bezüglich der zuletzt eingegebenen Zahl angeben, indem man vor die Zahl ein Plus "+" schreibt.

Beispiel: ' SAVE TEST 2000 2004' = ' SAVE TEST 2000 +4'

Widmen wir uns aber nun den Monitor-Befehlen:

#### a) Zahlenumrechnung

Gibt man als Kommando hinter einem Gleichheitszeichen "=" eine Zahl in beliebiger Schreibweise ein, so wird diese Zahl hexadezimal, dezimal und als ATASCII-Zeichen ausgegeben.

#### b) Hexdump

'> adr' zeigt den Inhalt der 8 Speicherzellen ab Adresse "adr" in hexadezimaler Schreibweise an. So erscheint z.B. nach Eingabe von '>680' die Zeile: ">0680 00 00 00 00 00 00 00" auf dem Bildschirm. Gibt man, durch Leerzeichen oder Kommas getrennt, hinter '> adr' ein Byte ein, so wird dieses in die Speicherzelle "adr" eingetragen. Gibt man mehrere Bytes an, so werden diese der Reihe nach in die darauffolgenden Speicherzellen eingetragen. Zahlen, die größer als 255 sind, werden als Zwei-Byte-Wort im LO-HI-Format (d.h. zuerst das nieder-, dann das höherwertige Byte) in den Speicher geschrieben.

Selbstverständlich können alle Zahlen in beliebiger Schreibweise eingegeben werden. Wieder ein Beispiel zur Verdeutlichung: '>680 D8 .2048 'a 1234 .100 FF' ; Nach Eingabe dieser Zeile sieht der Speicher ab Adresse 680 (hex.) wie folgt aus (Hexdump mit '>'): ">0680 D8 00 08 61 34 12 64 FF"

Der Hexdump wird in dieser speziellen Form ausgegeben, weil man die so erscheinenden Zeilen mit Hilfe des Bildschirmditors manipulieren und somit selbst als Kommando zum Verändern dieser Speicherstellen verwenden kann.

#### c) ATASCII-Dump

Gibt man '< adr' ein, so wird der Inhalt der 8 Speicherzellen ab Adresse "adr" als ATASCII-Zeichen angezeigt. Ansonsten verhält es sich bei diesem Kommando wie bei der soeben erklärten Hexdump-Funktion.

#### d) Dump-Adresse verschieben

Durch '+' bzw. '-' wird der Dump mit um 8 Bytes nach oben bzw. Unten verschobener Anfangsadresse angezeigt. Die Anzeigart (Hex/ATASCII) richtet sich dabei nach der zuletzt benutzten. Diese kann man übrigens einfach durch Eingeben von '>' bzw. '<' wechseln.

#### 24. Ausgabeumleitung

Bei TURBO-DUP besteht die Möglichkeit, alle Bildschirmausgaben zusätzlich auf ein zweites Gerät auszugeben. Um dieses zu erreichen, müsst Ihr einfach am Anfang der Kommandozeile ein Ausrufezeichen und dahinter das Ausgabegerät, das auch ein File sein kann, angeben. Dadurch werden alle Bildschirmausgaben des nachfolgenden Kommandos zusätzlich auf das so festgelegte Gerät geschrieben. Diese Zusatz-Ausgabe bleibt bis zum Erscheinen des nächsten Prompts bestehen. Einige Beispiele:

'! P: DIR' gibt das Inhaltsverzeichnis des aktuellen Laufwerks auf den Drucker aus, sofern dieser dazu bereit ist.

'! D8:TEST HEAD D:CONV234.COM' zeigt die Ladebereiche des Programms "CONV234.COM" an, und gibt diesen Text auf das File "TEST" auf der RD aus.

#### 25. EXECUTE

Ähnlich wie bei der Bildschirmausgabe kann man auch die Eingabe im TURBO-DUP umleiten. Genauer gesagt: alles, was dem DUP mittels des Bildschirmditors vom Benutzer mitgeteilt wird, kann stattdessen von einer Datei eingelesen werden. Eine solche Datei bezeichnen wir als "Execute-File", da die in ihm enthaltenen Kommandozeilen vom DUP ausgeführt werden können.

Durch das Kommando 'EXE fspec' wird das durch "fspec" repräsentierte File ausgeführt. Ähnlich wie bei dem LOAD-Befehl gibt es auch hier eine Abkürzung: Bei Abschluss der Eingabezeile mit <CONTROL>&<RETÜRN> wird das Eingegebene als Filename interpretiert und ".EXE" angehängt. Anschließend versucht das DUP, eine Execute-Datei dieses Namens auszuführen.

Execute-Files sind nichts anderes als Textdateien, die beliebig viele Zeilen enthalten können. Diese Zeilen werden vom DUP genauso behandelt, als hättet ihr sie selbst über die Tastatur eingegeben; sie werden nacheinander abgearbeitet. Falls Fehler auftreten, so werden sie wie im Direktbetrieb angezeigt, es wird aber danach noch versucht, den Rest des EXE-Files abzuarbeiten. Mit <SHIFT>&<BREAK> kann man jedoch auf jeden Fall das Execute-File abbrechen, um danach die Kommandoingaben wieder selbst vorzunehmen.

Zum Erstellen von EXE-Files dient der Batch-File-Editor (-> BATEDIT.COM). Es wurde bereits erwähnt, dass bei EXECUTE alle vom DUP geforderten Eingaben statt über den Bildschirmditor aus dem EXE-File erfolgen. Dies bezieht sich jedoch nicht auf Eingaben im DUP, die nur einen Tastendruck erwarten, also nicht mit <RETURN> abgeschlossen werden. Aus diesem Grunde müssen z.B. die Quittierabfragen nach wie vor durch Drücken der entsprechenden Tasten von Euch beantwortet werden. Die Eingaben der INIT/RUN-Adressen beim SAVE-Befehl hingegen werden im Execute-Betrieb aus dem EXE-File gelesen.

Auf Eurer TURBO-DOS-Masterdisk befinden sich einige Demos für Execute-Files, die am Extender "EXE" erkennbar sind und auf jeden Fall von einer RD gestartet werden sollten. Besitzer eines ATARI mit nur 64k können hierfür die XL-RD (-> CONFIG.COM) benutzen.

Man kann eine Execute-Datei auch von außerhalb des TURBO-DUP aufrufen. Ist nämlich beim Starten vom DUP der Ein-/Ausgabekanal Nummer sieben (IOCB #7) zum Lesen auf ein File geöffnet, so holt sich die Benutzeroberfläche ihre Eingaben aus diesem Kanal.

Es folgt ein Beispiel für ein BASIC-Programm zum Starten eines EXE-Files:

```
10 REM BASIC-Programm zum Starten des Execute-Files "DEMO.EXE".
```

20 REM Zuerst sicherheitshalber Kanal 7 schließen.

30 CLOSE #7

40 REM Jetzt Kanal 7 zum Lesen von "D:DEMO.EXE" öffnen.

50 OPEN #7,4,0,"D:DEMO.EXE"

60 REM Über OS-Vektor \$A=10 ins DUP gehen; da der Befehl "DOS" alle Kanäle schließt, ist er hier nicht anwendbar.

70 Q=USR(PEEK(10)+256\*PEEK(11))

## 26. Kommentar und Warteschleife

Um EXE-Files etwas komfortabler gestalten zu können, bietet TURBO-DUP noch zwei weitere kleine Kommandos an, die allerdings für den Direktbetrieb nicht besonders sinnvoll sind und daher am Ende dieser DUP-Beschreibung stehen:

Hinter dem Sternchen kann man Kommentare angeben, die bei Ausführung eines EXE-Files einfach auf dem Bildschirm erscheinen, sonst aber nichts weiter bewirken. Z.B. \* Dies ist ein Kommentar'

Der Befehl '? zahl' bewegt das DUP zu einer Pause, wobei der Parameter "zahl" die Pausendauer in Einheiten von Fünfzigstelsekunden angibt.

Fehlt die Angabe "zahl", so wird vom DUP der größtmögliche Pausenwert genommen (ca. 21'50.42").

Durch einen beliebigen Tastendruck kann man die Pause auch vorzeitig beenden.

## Übrigens...

Bei der Datenein- und Ausgabe auftretende Fehler werden durch Ausgabe der üblichen ERROR-Codes in hexadezimaler und dezimaler Form protokolliert (-> Anhang: Fehlerdiagnose).

Dabei werden auf jeden Fall das laufende Kommando abgebrochen und alle offenen Datenkanäle ordnungsgemäß geschlossen, wonach Ihr Euch in der Eingabezeile mit dem "Dn>"-Prompt wiederfindet.

Wenn man also nicht gerade <RESET> während eines Schreibvorganges drückt, sollten keine Dateien durch Nicht-Schließen des Kanals von der Diskette verschwinden (-> DISKFIX.COM).

Alle DUP-Operationen können durch Drücken von <BREAK> während der Bildschirmausgabe oder bei Erwartung eines Tastendrucks abgebrochen werden. Es erscheint dann die Fehlermeldung "ERROR #80=.128".

Der Abbruch eines Lesevorganges durch <BREAK> ist unproblematisch. Beim Drücken dieser Taste während eines Schreibvorganges müsst Ihr beachten, dass dadurch natürlich ein Teil der Daten nicht geschrieben wird und daher das so entstehende File unvollständig ist.

Bei Erscheinen einiger TURBO-DUP-Prompts könnt Ihr durch Drücken von <SHIFT> zusätzlich zur gewünschten Taste letztere verriegeln. Dadurch werden alle nachfolgenden Prompts vom DUP automatisch mit der gewählten Taste beantwortet, so als würdet Ihr sie selbst eintippen. Diese Verriegelung wird durch das Ende des Befehls oder durch Drücken einer beliebigen Taste aufgehoben. Ferner wird sie bei Erscheinen des "Insert"-Prompts (-> COPY) abgeschaltet.

## Schlussbemerkung

Ein Lob für all diejenigen, die sich bis hierher durch die Anleitung zum TURBO-DUP hindurchgearbeitet (oder aber -gequält ?) haben !

Eine Benutzeroberfläche ist schließlich das wichtigste Werkzeug eines jeden Disk-Operating-Systems. Damit Ihr Eure Disketten gescheit verwalten könnt, solltet Ihr Euch im DUP wie in Eurer

Rock- oder Hosentasche auskennen. Ich hoffe, dass Euch diese Anleitung als Einstiegs-, Begleit- und Nachschlagelektüre einigermaßen hilfreich war, ist und sein wird.

Klarheit bis ins letzte Detail kann Euch jedoch allein die Praxis mit dem TURBO-DUP XL/XE beschaffen.

Diskettenkopierer (File: DISKCOPY.COM)

Mit diesem Sektorkopierer kann, im Gegensatz zur COPY-Funktion des DOS, eine gesamte Diskette kopiert werden, auch wenn kein Directory vorhanden ist (wie der Name schon sagt, wird hier einfach Sektor für Sektor kopiert), daher lassen sich hiermit auch sog. BOOT-Disketten vervielfältigen. Voraussetzung ist natürlich, dass die Diskette nicht über einen Kopierschutz verfügt. Hierbei ist darauf hinzuweisen, dass urheberrechtlich geschützte Software nur zum eigenen Gebrauch (Sicherheitskopie) vervielfältigt werden darf. Sogenanntes "Raubkopieren" wird strafrechtlich verfolgt !

Das Programm ist auf allen XL/XE-Modellen mit mindestens 64 KB RAM lauffähig, wobei bei evtl. vorhandenen RD der erweiterte Speicherbereich bis 256k genutzt wird (Anzeige in der oberen Zeile je nach Gerätekonfiguration zw. 64k und 320k), wobei zu beachten ist, dass die Verwaltungsbytes bei Programmstart aus dem FMS übernommen werden, also Größe und Art der RD auch hier von CONFIG.COM bestimmt sind.

Das Laden des Programms erfolgt wie bei jedem COM-File, also aus dem DOS heraus mit dem LOAD-Befehl oder der Eingabe des Filenamens ohne Extender mit <SHIFT>&<RETURN> (-> Anleitung zum TURBO-DUP):

DISKCOPY <SHIFT>&<RETURN>

Nach dem Start präsentieren sich einige Menüpunkte sowie ein Benutzerhinweis in der letzten Zeile. Dieser gibt Auskunft darüber was als nächstes getan werden muss, bzw. welche Tätigkeit der Rechner gerade ausführt.'

Durch Drücken von <RESET> bei gleichzeitigem Festhalten von <SELECT> kann der Computer neu "gebootet" werden.

In das Menü gelangt man mit <ESC>, während des Kopiervorgangs mit <START>. Die einzelnen Menüpunkte werden dann mit den Cursor(Pfeil-)tasten (ohne <CONTROL>) angefahren, An-/Ausschalten erfolgt mit <RETURN>.

Das Menü wird mit <ESC> wieder verlassen, wobei das Programm an der Stelle fortgesetzt wird, an der es unterbrochen wurde.

Die Menüpunkte im einzelnen:

a) SOURCE/DESTINATION-DRIVE

Hiermit können Quell- und Ziellaufwerk gewählt werden. Die Grundeinstellung ist für Systeme mit einem Laufwerk gedacht. Sind zwei Laufwerke vorhanden, so wird üblicherweise 'DESTINATION' auf 2 gesetzt und die Zieldiskette in Laufwerk 2 eingelegt. Bei Verwendung von 2 Laufwerken wird die RD nicht benutzt.

b) VERIFY WRITTEN SECTORS

Erst wenn der gesamte Speicher auf die Diskette geschrieben worden ist, wird diese mit dem Speicherinhalt verglichen. Trifft das Programm auf eine Unstimmigkeit, so wird die Meldung "Abort, Ignore, Retry" ausgegeben:

- Abort: Abbruch - Ignore: Fehler ignorieren und fortfahren
- Retry: Fehlerhaften Sektor erneut schreiben und prüfen

- c) **TURBO-MODULE**  
(Bei nicht vorhandenem TURBO-Modul bzw. Übertragungsprogramm ist dieser Menüpunkt ohne Wirkung.)  
Das TURBO-Modul kann, falls vorhanden, ein- bzw. ausgeschaltet werden. Ein Ausschalten ermöglicht es, Disketten bei eingebautem Modul so zu formatieren, dass sie von einer "normalen" Floppy mit maximaler Geschwindigkeit bearbeitet werden können. Umgekehrt wird bei aktiviertem Modul die Kopie einer Standarddiskette im TURBO-Format erstellt.
- d) **CONTINUE ON ERROR**  
Ist diese Option eingeschaltet, so werden auftretende Fehler zwar angezeigt, das Programm jedoch ohne Unterbrechung fortgesetzt.
- e) **SINGLE --> ENHANCED CONVERSION**  
Liegt die Quelldiskette in Single Density vor, so wird die Zieldiskette in Enhanced Density formatiert. Hierdurch lassen sich z.B. Bootdisketten schneller laden.
- f) **ENHANCED --> SINGLE CONVERSION**  
Die Umkehrung von Punkt (e) ist sinnvoll, um Enhanced Disketten auch 810-Besitzern zugänglich zu machen. Funktionieren kann dies natürlich nur, wenn auf der Quelldiskette kein Sektor über 720 benutzt wurde.
- g) **QUIT ON EMPTY SECTOR**  
Das Kopieren wird bei Auftreten eines leeren Sektors abgebrochen, da bei DOS-Disketten jetzt nur noch leere folgen würden.  
(? Nur die relevanten Sektoren von Directory und VTOC werden noch kopiert. )  
Achtung: Diese Methode funktioniert nicht bei DOS 3, DOS 4 (QDOS) und Bootdisketten, einwandfrei jedoch bei DOS 2.0-kompatiblen (DOS 2.5, OSS A+, HAPPY-DOS, TURBO-DOS XL/XE, etc.).
- h) **MULTICOPY MODE**  
Nach jedem Schreibvorgang wird gefragt, ob eine weitere Diskette beschrieben werden soll, was das Anfertigen mehrerer Kopien in einem Arbeitsgang ermöglicht.
- i) **DIRECTORY**  
Zeigt das Directory (Inhaltsverzeichnis) einer DOS-Diskette (DOS 2.x-kompatibel) an.  
Achtung: Da hierfür Speicherplatz benötigt wird, wird ein eventuell laufender Kopiervorgang abgebrochen.

Eine versehentlich angewählte Funktion kann mit der BREAK-Taste oder mit <RESET> wieder verlassen werden. Sollte ein Fehler auftreten, so wird der Fehlercode als Hexadezimalzahl ausgegeben, gefolgt von der weiter oben beschriebenen Abfrage "Abort, Ignore, Retry", wobei auf der Tastatur der Anfangsbuchstabe der gewählten Funktion gedrückt werden muss.

(? Ein eventuell vorhandenes Übertragungsprogramm des TURBO-Moduls (in der 4- oder 6-Page) wird in den Stack verschoben, da der Speicher ab \$240 benutzt wird.)

File-Kopierer (File: FILECOPY.COM)

Im Gegensatz zum vorhergehenden Programm werden hiermit nicht ganze Disketten, sondern nur ausgewählte Files kopiert. Zwar ist dies auch recht gut im DUP zu bewerkstelligen, doch ist bei aufwendigeren Kopierarbeiten ein verstärktes Maß an Komfort wünschenswert. Ein sinnvolles Arbeiten ist hierbei jedoch nur mit 2 Floppies oder einer Floppy mit RD möglich.

Nach Laden des Files "FILECOPY.COM" von Eurer Masterdisk (am einfachsten vom DUP aus durch Eingabe von 'F'. gefolgt von <SHIFT>&<RETURN>) erscheint ein Konfigurationsfenster, mit dem Ihr festlegen könnt, welche Laufwerke von FILECOPY verwendet werden sollen.

Bereits hier werden drei grundlegende Eigenschaften der Benutzerführung von FILECOPY deutlich: Zum einen markiert stets ein blinkender Cursor die Stelle auf dem Bildschirm, an der vom Benutzer eine Eingabe erwartet wird. Dieser Cursor kann (in den meisten Fällen) durch Betätigen der Pfeiltasten bewegt werden, wobei das zusätzliche Drücken von <CONTROL> nicht unbedingt notwendig ist.

Zweitens könnt ihr zu jedem Zeitpunkt (außer beim Daten-I/O) durch Drücken der Taste <HELP> ein spezielles Help-Window aufrufen, das alle in dieser Situation möglichen Eingaben auflistet und kurz kommentiert. Durch einen beliebigen Tastendruck verschwindet der Text wieder und Ihr könnt dann eines der zuvor angezeigten Zeichen eingeben.

Alle Texte werden übrigens in "Computer-Englisch" ausgegeben, was im allgemeinen kürzer und eindeutiger ist als deutsche Übersetzungen wie z.B. "Entschützen". Die verwendeten Vokabeln, zumeist sowieso aus dem Angelsächsischen entlehnte Fachausdrücke, sind Euch sicher geläufig.

Als weiteres Merkmal von FILECOPY ist die Möglichkeit zu nennen, jede Operation oder. im weiteren Sinne, jedes Fenster durch Drücken von <ESC> oder <BREAK> abzubrechen bzw. zu verlassen.

Doch zurück zum Konfigurationsfenster. Hier könnt Ihr durch Drücken einer der Ziffern-Tasten "1" bis "8" die Laufwerksnummern der beiden Diskettenstationen auswählen, die Ihr benutzen möchtet. Wie oben bereits erwähnt, wird der Eingabe-Cursor mit den Pfeiltasten bewegt. "LEFT" und "RIGHT" stehen für die Seiten der Bildschirmhälften, auf denen dann später der Disketteninhalt der entsprechenden Laufwerke dargestellt wird. Da die beiden Seiten verschiedene Drives repräsentieren sollen, ist es nicht erlaubt, für beide Fenster die gleiche Diskettenstation anzumelden. Beim Starten von FILECOPY ist für das linke Fenster Drive #1 und für das rechte Drive #8 (RD) eingestellt.

Bevor Ihr mit <RETURN> das Konfigurationsfenster abschließt, müsst Ihr die dadurch angemeldeten Laufwerke mit (für TURBO-DOS lesbaren) Disketten bestücken, weil hiernach sofort deren Inhaltsverzeichnisse eingelesen werden. Währenddessen erfolgt dann auch der Aufbau des Hauptbildschirms von FILECOPY, der aus drei Teilen zusammengesetzt ist. Die oberste Zeile enthält lediglich Programmtitel und -version.

Den Mittelteil bilden die stets auf aktuellem Stand gehaltenen Inhaltsverzeichnisse der bearbeiteten Disketten, während die Menüleiste im unteren Teil des Bildschirms die verfügbaren Diskettenoperationen darbietet.

Die Inhaltsverzeichnisse (Directories) sind jeweils mit Laufwerksnummer und Dichte der Diskette überschrieben. Das momentan in Arbeit befindliche Directory-Fenster wird durch eine inverse Kopfzeile gekennzeichnet. Die bearbeitete Seite kann jederzeit durch <TAB> gewechselt werden. In den Fenstern selbst erscheinen dann die Namen der auf der jeweiligen Diskette befindlichen Dateien, wie wir das vom DUP her kennen, d.h. mit Größenangabe und ggf. mit "LOCK-Stern-

chen". Sie sind zur besseren Übersichtlichkeit mit Nummern versehen. Am Ende jeder Directory steht die Anzahl der freien, also noch verfügbaren Sektoren auf der Diskette.

Jedes Directory-Fenster verfügt über einen File-Cursor (kleiner nach rechts weisender Pfeil), den man mit Hilfe der Pfeil-Tasten verschieben kann. Unter Zuhilfenahme von <SHIFT> und <CONTROL> werden die Bewegungsschritte vergrößert. Auf diese Weise könnt Ihr das Sichtfenster bequem über alle Einträge im Inhaltsverzeichnis bewegen.

Drückt man den Anfangsbuchstaben eines der im unteren Bildschirmteil liegenden Menüpunkte, so wird die entsprechende Operation auf dasjenige File angewendet, vor dem sich der blinkende File-Cursor befindet. So kann man gezielt einzelne Dateien bearbeiten.

Möchtet ihr in einem Durchgang gleich mehrere Dateien bearbeiten, so könnt Ihr diese durch <RETURN> oder <SPACE> im Directory-Fenster markieren. Nochmaliges Betätigen dieser Taste macht die durch Invertierung des Dateinamens angezeigte Markierung wieder rückgängig. Zur Manipulation aller Files gleichzeitig dienen <CLEAR>, <INSERT> und <DELETE>.

Sind alle gewünschten Dateien markiert, so könnt Ihr durch <ESC> oder <BREAK> auf die Menüleiste wechseln. (Die gleichen Tasten bewirken dann später auch wieder das Umschalten aufs Directory-Fenster.) Hier könnt Ihr nun entweder durch Bewegen des Balken-Cursors mit anschließendem <RETURN> oder direkt durch Eingeben des Anfangsbuchstabens einen Menüpunkt starten. Die so ausgewählte Operation wird hiernach an allen markierten Dateien im aktuellen Fenster (das mit der inversen Kopfzeile) vorgenommen.

Es folgt nun eine Liste der in der Menüleiste vorkommenden Funktionen:

- a) INDEX: Diese Funktion liest einfach das Directory der Diskette im aktuellen Fenster erneut ein. Damit dies geschehen kann, muss zuvor natürlich die Diskette in das entsprechende Laufwerk eingelegt werden. Die Laufwerksnummer des aktuellen Directory-Fensters könnt ihr jederzeit einfach durch Eingeben der gewünschten Ziffer ("1"-8) verändern, wonach sofort das neue Inhaltsverzeichnis eingelesen wird.
- b) DELETE löscht Dateien.
- c) COPY kopiert Dateien aus dem aktuellen Directory-Fenster auf die Diskette im anderen Laufwerk.
- d) RENAME: Bei Aufruf dieser Funktion hat man Gelegenheit, im Directory-Fenster direkt die Namen von Files zu editieren. Die Editier-Tasten entnehmt ihr bitte dem dazugehörigen Help-Window. <RETURN> bewirkt, dass der geänderte Name auf Diskette geschrieben wird. Mit <BREAK> könnt ihr ein Schreiben des Namens verhindern.
- e) LOCK versieht einzelne Dateien mit einem Software-Schutz. Auf diese Weise geschützte Files erscheinen im Directory mit einem Sternchen.
- f) UNLOCK hebt diesen Schutz wieder auf.
- g) FORMAT: Durch Anwählen dieser Option erscheint ein Untermenü, in dem neben den vier Formatierdichten Single, Medium (=Enhanced), Double und Quad aus Platzgründen noch zwei weitere Funktionen untergebracht sind: "CLEAR" löscht das Directory; "WRDOS" schreibt das FMS in Form des "DOS.SYS"-Files auf Diskette. Die Funktionen des FORMAT-Menüs werden in gleicher Weise gestartet wie die Punkte der Menüleiste; <ESC> oder <BREAK> verlassen das FORMAT-Menü.

- h) TYPE gibt Dateien als ATASCII-Zeichen auf dem gegenüberliegenden Directory-Fenster aus. Nach Ausdruck einer Seite wird auf die Bestätigung des Benutzers zum Fortfahren gewartet. Zur Erklärung der möglichen Beantwortungstasten verweise ich erneut auf <HELP>.
- i) START lädt und startet Maschinenprogramme. Nach deren Rückkehr bzw. bei Auftreten eines Fehlers wird FILECOPYY, sofern noch vorhanden, neu gestartet.
- j) QUIT verlässt FILECOPYY nach einer Sicherheitsabfrage. Bei <RETURN> wird das DUP geladen. <ESC> oder <BREAK> startet FILECOPYY durch Aufruf des Konfigurationsfensters neu. Jede andere Taste verlässt die QUIT-Funktion.

Da FILECOPYY seine Directory-Fenster immer auf dem neuesten Stand halten möchte, wird nach jeder Funktion, die das Directory verändert, dieses erneut eingelesen. Dabei werden alle Markierungen in diesem Fenster gelöscht. Ferner wird der File-Cursor auf die erste Datei gesetzt.

Falls ein I/O-Fehler auftritt, wird der entsprechende Error-Code angezeigt und die Funktion abgebrochen. Nach Drücken von <RETURN> liest FILECOPYY dann sicherheitshalber das Directory neu ein. Tritt dabei wieder ein Fehler auf, so müsst Ihr überprüfen, ob die richtigen Disketten eingelegt sind. Habt Ihr aus Versehen (oder Absicht) durch Drücken einer der Ziffern "1" bis "8" das aktuelle Fenster für ein nicht angeschlossenes Laufwerk geöffnet, führt <BREAK> oder <ESC> beim ERROR-Prompt zurück zum Konfigurationsfenster, von wo aus Ihr dies korrigieren könnt. Wenn alles nichts mehr hilft, kann vielleicht <RESET> wieder Ordnung ins System bringen.

Da man mit FILECOPYY jederzeit Überblick über alle Dateien der eingelegten Disketten hat, eignet sich dieses Utility besonders dann zum Kopieren, Umbenennen und Löschen von Dateien, wenn man ob der Menge und Unterschiedlichkeit der Files vom DUP aus den Überblick verloren hat.

Leider ist es einem durch die geringe Größe des normalen Bildschirms der 8-Bit-ATARIs versagt, genügend Informationen gleichzeitig und dabei übersichtlich darzubieten, wenn man den vom Betriebssystem unterstützten Text-Modus benutzen möchte.

Auch bei FILECOPYY kommt dieser Umstand einer fehlenden, vernünftigen standardmäßigen 80-Zeichen-Darstellung bei den "kleinen" ATARI-Computern zum Tragen: Die Inhaltsverzeichnisse von zwei Diskettenstationen sind nur unter einem gewissen Verlust an Übersichtlichkeit darstellbar. sofern man nicht auf essentielle Informationen verzichten möchte.

### Disketten retten (File: DISKFIX.COM)

Mit diesem Programm seid Ihr im Besitz eines enorm starken Utility, das euch bei einer Vielzahl von Software-"Unfällen" mit euren Disketten helfen kann. Das gleichnamige Zusatzprogramm zu DOS 2.5 wird hiervon klar geschlagen, da auch Disketten mit vollkommen zerstörter Directory restauriert werden können.

Selbstverständlich wurden auch die Besitzer der XF 551 mit einer speziellen Version bedacht.

#### - Allgemeines

- + Bei Ja/Nein-Entscheidungen kann statt <Y> / <N> auch <RETURN> / <SPACE> verwendet werden
- + Das Editieren eines Filename erfolgt wie beim Batch-Editor
- + Mit <ESC> gelangt man aus einem Untermenü in das übergeordnete Menü. beim Hauptmenü ist dies das DUP - ein versehentlich angewähltes Fenster kann mit <BREAK> jederzeit verlassen werden
- + der blinkende Cursor kann mit <CONTROL>&<ESC> aus- oder angeschaltet werden
- + <HELP> gibt Hilfsmeldungen aus

#### - Erklärung der einzelnen Punkte

Nach dem Laden präsentiert sich das Hauptmenü, dessen Punkte mit den CURSOR-Tasten (ohne <CONTROL>) angefahren werden können und mit <RETURN> aktiviert werden können.

#### a) DISK DRIVE: x

Hier wird das physikalische Laufwerk gewählt, in dem sich die zu bearbeitende Diskette befindet. Wiederholtes drücken von <RETURN> bringt alle möglichen Einstellungen (x von 1 bis 4) zur Anzeige.

#### b) VERIFY DISK

- eine fehlerhafte VTOC wird restauriert
- nicht geschlossene Files werden gelöscht
- ein fehlerhafter Längeneintrag eines Files im Directory wird korrigiert
- ein fehlerhafter Sektor-Link wird erkannt und kann auf verschiedene Weise bearbeitet werden

Nach der Anwahl dieser Funktion erscheint das Fenster "Redirect output", womit man das Überprüfungsprotokoll zusätzlich zum Bildschirm auch auf dem Drucker oder einem File (späteres Anschauen im DUP mit 'TYPE') ausgeben kann. Mit <N> übergeht man diese Möglichkeit. Bei <Y> erscheint ein weiteres Fenster, das die Eingabe des Device mit Filename erlaubt und mit 'P:' (=Drucker) bzw. dem zuletzt eingegebenen Filename vorbesetzt ist.

Alsdann beginnt das Programm mit der Untersuchung aller im Directory eingetragenen Files, wobei diese mit ihrer fortlaufenden Nummer, ihrem Namen und den von ihnen belegten Sektoren angezeigt werden.

Einträge mit der Markierung 'not closed' werden aus dem Directory entfernt.

Findet das Programm einen defekten Sektor-Link (Anzeige: "ERROR 164"), so kann man sich entscheiden ob man das File löschen, zum Hauptmenü zurückkehren oder den Link reparieren will (nur möglich, wenn nicht schon der erste Sektor-Link eines Files schadhaf ist).

(? Letztere Funktion arbeitet mit folgendem Algorithmus:

Nach Auftreten eines schadhafte Link wird der nächste zu diesem File gehörende Sek-

tor gesucht und der defekte Link auf diesen "umgebogen".

Hierbei wird bei Erreichen des Diskettenendes auch der Raum zwischen Sektor 4 und dem fraglichen File berücksichtigt. Wird kein Sektor mehr gefunden, wird das File an der defekten Stelle geschlossen.)

Am Ende der Analyse wird angezeigt, ob die gefundenen Daten mit der bestehenden VTOC übereinstimmen. Ist dies nicht der Fall, wird diese nach einem Tastendruck neu geschrieben oder mit <BREAK> unverändert belassen.

#### c) RECOVER DIRECTORY

Dies ist die leistungsstärkste und aufwendigste Funktion. Mit ihrer Hilfe kann eine vollständig zerstörte (überschriebene) Directory wieder hergestellt werden.

Nach dem Aufruf wird zunächst die gesamte Diskette gelesen, wobei der gerade eingeleseene Sektor hexadezimal angezeigt wird. Der Lesevorgang wird bei leeren Sektoren nicht abgebrochen werden (->DISKCOPY), da bei einer defekten Diskette auch ein File-Sektor leer sein könnte. Diese Entscheidung wird dem Benutzer überlassen. Anhand der gefundenen Daten versucht das Programm nun, die ehemalige Zuordnung der Sektoren in Files vorzunehmen. Bei Ausführung dieser Funktion wird auch eine Analyse der entdeckten Files vorgenommen. Eindeutig erkannt werden dabei Maschinensprache- (.COM) und BASIC-Files (.BAS), was mit dem entsprechenden Extender angezeigt wird. War das File keiner dieser beiden Arten zuzuordnen, werden die ersten 10 Bytes untersucht. Handelt es sich dabei um reinen ASCII-Code, erhält das File die Extension ".TXT". Alle anderen Files erhalten den Extender ".REC".

(? Hierzu werden zunächst die Startsektoren der Files gesucht, die dadurch identifiziert werden können, dass kein Link auf sie weist (während dieses Vorgangs wird "Checking links ..." angezeigt). Anschließend wird ein Verify wie unter (b) durchgeführt.)

Dies geht sehr schnell, da die Routinen allesamt in Maschinensprache erstellt wurden. Nach geantener Arbeit wird die neu erstellte Directory angezeigt, in der alle gefundenen Files den Namen 'FILExy.REC' tragen, wobei xy eine fortlaufende Zahl repräsentiert. Später können diese Filenames natürlich umbenannt werden. Man kann sich nun entscheiden, ob die neue Directory auf Diskette übernommen werden soll.

Das Programm kann natürlich nicht erkennen, ob ein wiederhergestelltes File eventuell bereits gelöscht war. Waren Teile eines solchen Files schon überschrieben, ist dieses jetzt nicht mehr vollständig (-> "Undelete" in Punkt (d) ).

Zur Identifizierung der wiederhergestellten Files empfiehlt sich eine Untersuchung mit,\* den DUP-Befehlen HEAD (Entdeckung von COM-Files) und TYPE (zeigt Kommentare in Programmen oder Texte in Datenfiles).

#### d) DIRECTORY

- alle Einträge der Directory werden mit ihrem Status angezeigt
- <HELP> erklärt die Bedeutung der Status-Bytes
- gelöschte Files können u.U. wieder hergestellt werden
- Files können umbenannt werden

Hiermit werden alle Einträge der Directory angezeigt, auch wenn diese bei einem normalen DIR-Aufruf (z.B. im DUP) nicht sichtbar wären.

In jeder Zeile erscheint zunächst die Eintragsnummer, dann das Status-Byte dessen Bedeutung mit <HELP> angezeigt werden kann, gefolgt von dem Filename. Hiernach stehen noch Startsektor und Länge des Files in Sektoren. Nicht geschlossene Files sind klein geschrieben Existieren mehr als 14 Einträge, so kann mit den CURSOR-Tasten gescrollt werden, um weitere Einträge sichtbar zu machen.

Ein bereits gelöscht File ist invers hervorgehoben. Wird dieses mit dem Cursor angefahren, so kann es mit <U> "undeleted", also in den ungelöschten Status zurück gebracht werden. Stellt das Programm jedoch fest, dass Teile des Files schon überschrieben worden sind. ist diese Möglichkeit nicht mehr gegeben, was mit der Meldung "can't undelete" angezeigt wird.

Weiterhin ist es möglich, ein mit dem Cursor markiertes File mit <R> (= rename) umzubenennen, wobei der Filename einfach mit den erlaubten Zeichen überschrieben wird. Die Änderung kann mit <ESC> abgebrochen oder mit <RETURN> auf Diskette übernommen werden.

e) HELP

Ein Aktivieren dieser Funktion zeigt die gleichen Hilfsmeldungen wie bei einem Betätigen von <HELP> im Hauptmenü.

f) QUIT

Die gleiche Funktion wie bei <ESC> im Hauptmenü, also eine Rückkehr zum DUP.

Nur bei XF 551:

g) SET DOUBLE/QUAD DENSITY

Da dieses Laufwerk nicht eigenständig zwischen Quad und Double Density unterscheiden kann, ist es notwendig, auf jeder Diskette die jeweilige Dichte zu verzeichnen. Dies wird durch ein Bit in der VTOC bewerkstelligt. Stimmt dieses Bit in Folge eines unkontrollierten Schreibvorgangs nicht mit dem tatsächlichen Format überein hat der Benutzer im Bedarfsfall die Möglichkeit, mit diesem Menüpunkt nachzuhelfen.

DOS-Konvertierer <File: CONV234.COM>

Dieses Programm ermöglicht es, Files zwischen den DOS-Versionen 2.5, 3 und 4 (QDOS) auszutauschen. Es wird aus dem TURBO-DUP gestartet. (Das vorliegende Programm kann nicht bei gleichzeitig installierter XL-RD verwendet werden !)

Bevor einer der weiter unten beschriebenen Befehle ausgeführt wird, stellt das Programm automatisch die DOS-Version der eingelegten Diskette fest.

Es kann auf folgende DOS-Versionen zugegriffen werden:

DOS 2.0s & kompatible

DOS 2.5 alle Formate

TURBO-DOS XL/XE alle Formate

DOS 3 alle Formate

DOS 4 Single & Enhanced Density

DOS 4 Double Density, single sided

Als Befehle stehen die DOS-Kommandos DIR, LOCK, UNLOCK, RENAME und DELETE zur Verfügung, deren Bedeutung sicherlich klar ist. Files können mit LOAD geladen werden; mit SAVE werden sie auf eine andere Diskette, die mit einer der oben genannten DOS-Versionen formatiert sein muss, gespeichert. Dieses File darf maximal 315 Sektoren lang sein. Bei der Befehlseingabe sind nur die ersten drei Buchstaben signifikant.

Folgende Varianten sind zulässig:

- a) LOAD TEST
- b) LOAD D:TEST
- c) LOAD "D8-.TEST" (auch die RD kann angesprochen werden)

Mit DOS gelangt man zurück ins DUP. Muss es von der Diskette geladen werden, so sollte vorher eine entsprechende TURBO-DOS-Diskette eingelegt werden. Alle Befehle, außer LOAD, verändern das File im Speicher nicht. Sollte ein Fehler auftreten, wird die normale Fehlernummer ausgegeben.

GAME-DOS (File: WRGAME.COM)

Dieses File schreibt ein sog. GAMEDOS auf eine Diskette mit COM-Files, auf der sich kein DOS.SYS befindet. Es ist kompatibel zu:

DOS 2.0s, DOS 2.5 (Single sowie Enhanced Density), TURBO-DOS XL/XE (alle Formate), Happy-DOS, sowie all die anderen kompatiblen DOS-Versionen (OSS A+, My-DOS, usw.). GAMEDOS ermöglicht es COM-Files zu laden, ohne vorher ein DOS booten zu müssen.

Wird eine Diskette, auf der sich GAMEDOS befindet, gebootet, so werden alle darauf befindlichen Files (max. 56) angezeigt. Über die CURSOR-Tasten gefolgt von <RETURN> wird ein File ausgewählt und geladen. Der Spielefan kann die Auswahl auch mit Hilfe eines Joysticks vornehmen, wobei die Triggertaste <RETURN> ersetzt. Das Programm ist für die Verwendung mit OLD-RUNNER oder TRANSLATOR-Disk geeignet.

Nach dem Start des Programms wird die fragliche Diskette in Laufwerk 1 eingelegt, wonach GAMEDOS mit <RETURN> hierauf geschrieben wird. Danach kann man den Schreibvorgang für weitere Disketten wiederholen. Ist dies nicht mehr gewünscht, kann das Programm mit <ESC> verlassen werden.

BASIC-Lader (File: WRBASIC.COM)

Dieses Programm hat die gleichen Leistungsdaten wie das vorhergehende, nur werden hiermit BASIC-Programme (nicht TURBO-BASIC !) geladen.

Programmmlader (File: CHOOSE.COM)

Programmmlader (File: CHOOSE.COM)

Wird dieses Programm aus dem AUTOCOPY gestartet, so werden alle COM-Files von Laufwerk I aufgelistet. Nach der Auswahl mit den CURSOR-Tasten wird ein Programm (bzw. BASIC oder DUP) ausgewählt, das mit <RETURN> gestartet wird. Bei einigen Programmen empfiehlt sich der Start mit <SHIPT>&<RETURN>, da dann bei Programmende auf einen Tastendruck gewartet wird, bevor der Rücksprung zu CHOOSE.COM erfolgt (z.B. DOSINFO.COM).

Druckertreiber (File: PRINTER.COM)

Leider verfügt der 8-Bit-ATARI über keine genormte Schnittstelle zur Außenwelt. Besonders schmerzlich macht sich dies bemerkbar, wenn man sich zum Kauf eines Druckers entschließt. Die passenden Geräte von ATARI sind wirklich nur bedingt zu empfehlen, die meisten anderen verfügen über eine sog. CENTRONICS-Schnittstelle. Mit diesem Programm wird eine solche Schnittstelle simuliert, wobei nur noch ein spezielles Kabel an die Joystickports angeschlossen werden muss. Dieses erhaltet Ihr für 38.-DM bei mir.

Nun lassen sich so schöne Drucker wie EPSON, STAR. NEC u.v.a. anschließen. Mit diesem Kabel könnt Ihr übrigens auch aus dem Textverarbeitungsprogramm STARTEXTER (Firma SYBEX) ausdrucken (siehe Handbuch), da hier ein ähnliches Programm wie PRINTER.COM integriert ist. Das Programm kann, wie gehabt, aus dem DUP gestartet werden. Mit RESET wird es deaktiviert, wobei es mit RUN 681 aus dem DUP. bzw. aus dem eingebauten BASIC mit Q=USR (I664) wieder eingeschaltet werden kann.

(? Das Programm befindet sich nach dem Start in der zweiten Hälfte der 6-page, also ab \$680. Es blendet sich in den Printerhandler des Betriebssystems ein und erzeugt die benötigten Druckersignale auf den 8 Ausgabeleitungen von Port A. Da eine dieser Leitungen für das STROBE-Signal verwendet werden muss, stehen für Daten nur noch 7 BIT zur Verfügung. Daher wird im Kabel das achte Datenbit (BIT 7) auf Masse gelegt. )

Ich möchte jedoch darauf hinweisen, dass es sich hierbei nicht um ein vollwertiges Druckerinterface handeln kann, wie es z.B. das um ein Spezialkabel erweiterte TURBO-Modul darstellt.

Sektoranalyse (File: DISKMAP.COM)

Dieses kleine Programm leistet folgendes:

Nach der Auswahl eines physikalischen Laufwerkes werden für die darin befindliche Diskette die freien, belegten und defekten Sektoren angezeigt. Alle weiteren Arbeitshinweise erfolgen im Programm.

Sauberer RESET-Schutz für TURBO-BASIC XL (File: RESET.TUR)

Natürlich hat Frank Ostrowski sein TURBO-BASIC mit einem RESET-Schutz versehen. Dieser wurde allerdings (ausnahmsweise) nicht sehr sauber programmiert. Folge:

Benutzt man eine Floppy, in die das TURBO-1050-Modul eingebaut ist, so wird das zugehörige Übertragungsprogramm bei einem RESET nicht mehr initialisiert. Programmen, die den gleichen RESET-Schutz verwenden, ergeht es ebenso.

(? Der Grund ist folgender:

Das Initialisierungsprogramm von TURBO-BASIC schreibt in Speicherstelle 9 eine 1. Hierdurch wird Bit 1 gelöscht, welches auf jeden Fall gesetzt sein muss, wenn bei einem RESET auch ein Sprung über CASINI erfolgen soll. Nach Anwendung dieses Programms wird nur noch BIT 0 gesetzt, während alle anderen BITS unverändert bleiben.)

Das TURBO-BASIC-File wird direkt auf der Diskette verändert. Nach dem Starten des Programms muss man den Filename angeben, unter dem das BASIC abgespeichert ist. Dieses File wird sofort (!! ) umgewandelt.

Die Quad Density des BIBO-DOS (Files: BIBOEIN.COM, BIBOAUS.COM)

Wer seine XF 551 bisher mit dem BIBO-DOS (ABBUC e.V. PD Bibliothek) betrieben hat, wird die auf Quad Density (QD) abgespeicherten Files auch mit unserem DOS benutzen wollen. Während dies bei allen anderen Dichten unproblematisch ist, verwenden wir bei QD ein etwas anderes Aufzeichnungsformat, welches einige Vorteile bietet.

Wer sich in der Version XF oder EX befindet, kann nach Aufruf des Programms BIBOEIN.COM auf QD-Files des BIBO-DOS zugreifen (z.B. auf die RD kopieren). Beim Schreiben wird jedoch wieder unser Format verwendet, womit ein Umkopieren (Formatanpassung) ermöglicht wird. Das im Speicher befindliche DOS ist nun aber nicht mehr unser Original-DOS, weshalb es nicht auf Disk geschrieben werden darf (-> INI /D). Die Rückkehr zum "normalen" DOS bewirkt das Programm BIBOAUS.COM.

(? Durch die erhöhte Sektorenzahl bei QD reicht die von DOS 2.x vorgesehene Kapazität der Link-Bits nicht mehr aus. BIBO-DOS benutzt nun einfach die Bits, die eigentlich für die Zuordnung des Sektors zu seinem File bestimmt sind, wodurch z.B. die Funktion "Recover Directory" des Programms DISKFIX.COM unmöglich gemacht wird. Aus diesem Grund reserviert unser DOS bei QD 4 (statt 3) Bytes eines Sektors für das FMS, damit keine Informationen verloren gehen. Wir halten dies in jedem Fall für die bessere Lösung, zumal es von ATARI noch keinen Standard für QD gibt.)

Utility für das TURBO-Modul (File: CREATURX.COM)

Dieses Programm ist nur für Besitzer des TURBO-Moduls von Bedeutung. Es dient zur Erstellung spezieller Maschinensprache-Files, die das TURBO-Übertragungsprogramm im Computerspeicher installieren, was z.B. sinnvoll ist, wenn dieses durch ein Programm ausgeschaltet wurde.

Da wir aus urheberrechtlichen Gründen das Übertragungsprogramm nicht mitliefern können, muss dieses beim Start von CREATURX.COM bereits im Speicher vorliegen (Installation beim Booten mit <START> / <SELECT>).

Im Programmmenü kann man nun entscheiden, wohin das zu erstellende COM-File das TURBO-Übertragungsprogramm legen soll (1st, 4th oder 6th-page). Zusätzlich gibt es die Möglichkeit ein Programm zu erzeugen, mit dem das TURBO-Modul ausgeschaltet wird.

(? Nach Drücken von <X> könnt Ihr auch jede beliebige Page für das Übertragungsprogramm eingeben.)

Nach Auswahl eines Menüpunktes müsst Ihr den Namen bestimmen, unter dem das File abgespeichert werden soll. Mit <RETURN> schließlich wird das Programm als COM-File abgespeichert.

Dieses kann nun jederzeit nachgeladen werden (aus dem DUP mit LOAD oder <SHIFT>&<RETURN>). wenn das Übertragungsprogramm aus irgendeinem Grund zerstört worden sein sollte.

Nachwort

Wer diese Worte liest, hat hoffentlich die gesamte Anleitung gründlich durchgearbeitet, bevor er den ABBUC mit Fragen bombardiert.

Eine Beschreibung der in dieser Anleitung erwähnten Laufwerkserweiterungen findet ihr bei den auf Seite 2 genannten Quellen.

- Anhang -

Fehlerdiagnose

Es folgt nun eine Liste aller Fehlermeldungen, die bei Betrieb des TURBO-DOS möglicherweise vorkommen können. Fehlersituationen, die bei Benutzung der mitgelieferten Utilities eigentlich nicht auftreten dürften, sind bei dieser Aufstellung in Klammern gesetzt.

- \$17=21      BAD LOAD FILE: Dieser Fehlercode tritt auf, wenn vom TURBO-DUP aus versucht wurde, mit dem Befehl 'LOAD' eine Datei zu laden, die nicht im COM-File-Format vorliegt. (-> LOAD)
- \$80=128     BREAK ABORT: Während einer Ein-/Ausgabeoperation wurde die Taste <BREAK> gedrückt und damit das Kommando abgebrochen. (-> DUP/Übrigens)
- \$81=129     PREVIOUS OPEN: Es wurde versucht, einen bereits offenen Ein-/Ausgabekanal (IOCB) erneut zu öffnen.)
- \$82=130     NONEXISTENT DEVICE: Es wurde versucht, ein Gerät anzusprechen. das in der Handlertabelle des Betriebssystems nicht verzeichnet und diesem somit unbekannt ist.  
Dies passiert, wenn man z.B. aus Versehen bei Angabe eines Gerätenamens (-> DUP/Device-Kennung) einen falschen Buchstaben eintippt, wie 'd:' statt 'D:'.
- \$83=131     WRITE ONLY: Es wurde versucht, Daten von einem Gerät zu lesen, das nur zur Ausgabe dient, wie z.B. der Drucker. Tritt z.B. auf bei: 'SUM P:' oder 'COPY P.:D:TEST'.
- \$84=132     ILLEGAL COMMAND: Das übergebene IOCB-Kommando ist dem Gerätetreiber (Handler). auf den der benutzte Kanal geöffnet ist, unbekannt und kann daher nicht ausgeführt werden.)
- \$85=133     NOT OPEN: Der benutzte IOCB-Kanal ist noch nicht geöffnet.)
- \$86=134     BAD IOCB NUMBER: Die übergebene IOCB-Nummer ist ungültig, d.h. entweder kein Vielfaches von 16 oder aber zu groß.)
- \$87=135     READ ONLY: Es wurde versucht, auf ein Gerät Daten zu schreiben, das nur zur Eingabe dient, wie z.B. Keyboard (Tastatur). Tritt z.B. auf bei: 'COPY D:TEST,K:' oder '! K: DIR'
- \$88=136     END OP FILE: Das Ende einer Datei wurde erreicht.)
- \$89=137     TRUNCATED: Bei einem Zeilenlesekommando wurde eine Zeile empfangen, die länger als der zur Verfügung stehende Puffer ist. Im DUP: Die eingegebene Kommandozeile ist zu lang.
- \$8A=138     TIMEOUT: Das angesprochene Gerät hat sich nicht innerhalb der vom Handler festgesetzten Zeitspanne zurückgemeldet. Mögliche Ursachen:  
- Gerät nicht eingeschaltet.  
- Gerät nicht ordnungsgemäß angeschlossen. Bei Diskettenstationen die dort eingestellte Laufwerksnummer (DIP-Schalter) überprüfen.

- \$8B=139      DEVICE NACK: Die negative Rückmeldung des angesprochenen Geräts bedeutet, dass die evtl. mit dem Kommando übergebenen Parameter nicht sinnvoll interpretiert werden können. Dies ist z.B. der Fall, wenn die beim Diskettenzugriff übergebene Sektornummer ungültig (gleich null oder zu groß) ist.
- \$8C=140      SERIAL FRAME ERROR: Fehler bei der seriellen Datenübertragung. Tritt z.B. auf, wenn die Übertragungsraten von Sender und Empfänger nicht übereinstimmen. Wenn man beim Betrieb einer der TURBO-DOS-Spezialversionen eine beschleunigt betriebene Floppy durch eine mit anderer Übertragungsrate ersetzt, ohne das DOS entsprechend zu abzuändern. Bei neuer Gerätekonfiguration sollte man am besten das TURBO-DOS neu booten.
- \$8D=141      CURSOR OVERRANGE: Der Cursor wurde außerhalb des ihm zustehenden Bildbereichs gesetzt.)
- \$8E=142      SIO OVERRUN: Der Computer konnte die von einem externen Gerät übersandten Daten nicht schnell genug verarbeiten. Falls dieser Fehler öfters auftritt, sollte man seinen Computer durch-"checken" lassen.
- \$8F=143      CHECKSUM ERROR: Die beim Datenaustausch zu Kontrollzwecken empfangene Prüfsumme stimmt nicht mit der vom Computer errechneten überein, d.h. die Daten wurden vom Gerät nicht richtig verarbeitet. Gerät überprüfen lassen.
- \$90=144      DEVICE DONE: Das Gerät kann ein reguläres Kommando nicht korrekt ausführen. Mögliche Ursachen bei Diskettenstationen:
- Diskette schreibgeschützt
  - Motordrehzahl falsch
  - Sektor kann nicht gelesen/geschrieben werden:
    - + Diskette defekt
    - + Diskettenstation defekt
- \$91=145      BAD SCREEN MODE: Bildschirmmodus unbekannt.)
- \$92=146      FUNCTION NOT IMPLEMENTED: Das übermittelte IOCB-Standard-kommando ist in dem angesprochenen Handler nicht vorgesehen. Z.B. Schreibkommando auf Tastatur o.ä.)
- \$93=147      INSUFFICIENT RAM: Nicht genügend Speicher zum Betrieb einer bestimmten Grafikstufe vorhanden.)
- \$A0=160      DRIVE NUMBER ERROR: Eine ungültige Laufwerksnummer wurde eingegeben. Die Nummer lag entweder nicht zwischen 1 und 8 oder aber im DOS (FMS) steht kein Puffer für diese Diskettenstation bereit. (-> CONFIG.COM)
- \$A1=161      TOO MANY OPEN FILES: Es wurde versucht, mehr Files zu öffnen, als im DOS (FMS) Puffer dafür bereitstehen. Entweder die maximale Anzahl gleichzeitig offener Files im DOS vergrößern (-> CONF1G.COM) oder mit weniger offenen Files arbeiten.
- Dieser Fehler tritt z.B. auf, wenn vom DUP aus der COPY-Befehl mit Ein- oder Ausgabeumleitung benutzt wird und im FMS nur zwei offene Files vorgesehen sind.

- \$A2=162      DISK FÜLL: Beim Schreiben von Daten wurde festgestellt, dass die Speicherkapazität der Diskette erschöpft ist. Die zu schreibenden Daten konnten folglich nicht vollständig abgespeichert werden. Abhilfe: überflüssige Dateien auf der Diskette löschen oder andere Diskette verwenden.
- \$A3=163      SYSTEM I/O ERROR: Durch das Einwirken von Fremdsoftware ist ein Fehler im DOS entstanden, das sich derzeit im Speicher befindet. Falls nach dem Neu-Booten der Fehler erneut auftritt, ist das DOS auf der Diskette defekt. In diesem Falle TURBO-DOS-Masterdisk booten und dann das defekte DOS mit 'INIT' aus dem DUP neu schreiben.
- \$A4=164      FILE NUMBER MISMATCH: Die Speicherstruktur einer Datei auf der Diskette wurde zerstört. Ein Fall für DISKFIX.
- \$A5=165      FILE NAME ERROR: Ein illegaler Dateiname wurde verwendet. (-> DUP/Dateinamen).
- \$A6=166      POINT DATA LENGTH ERROR: Der beim XIO-Kommando POINT Übergebene Byte-Zähler hat einen zu großen Wert. Die Maximalwerte lauten: Single und Enhanced Density: 125; Double Density: 253; Quad Density: 252.)
- \$A7=167      FILE LOCKED: Es wurde versucht, schreibend auf eine Datei zuzugreifen, die mit dem LOCK-Schreibschutz versehen ist. Falls die Operation wirklich ausgeführt werden soll, muss man das File zuvor mit dem DUP-Kommando 'UNLOCK' "entschützen" .
- \$A8=168      INVALID DEVICE COMMAND: Es wurde ein Spezialkommando abgeschickt, das im angesprochenen Handler nicht implementiert ist. Tritt z.B. auf, wenn man bei einer TURBO-DOS-Version, die nicht die speziellen Formate der XF-551 Floppy unterstützt, versucht, mit 'FMQ' aus dem DUP eine Diskette in Quad Density zu formatieren.
- \$A9=169      DIRECTORY FÜLL: Auf die Diskette können keine neuen Files mehr abgespeichert werden, da deren Inhaltsverzeichnis voll ist. Mit TURBO-DOS formatierte Disketten bieten Platz für maximal 64-Files, in Quad Density für 128!
- \$AA=170      FILE NOT FOUND: Es wurde ein File angesprochen, das auf dieser Diskette nicht vorhanden ist. Möglicherweise wurde eine falsche Diskette eingelegt oder ein Tippfehler bei der Namenseingabe gemacht.
- \$AB=171      POINT INVALID: Es wurde versucht, das IOCB-Kommando POINT auf ein File anzuwenden, das nicht im UPDATE-Modus (Lesen und Schreiben: zweites IOCB-Hilfsbyte gleich zwölf) geöffnet wurde.)
- \$AC=172      ILLEGAL APPEND: Es wurde versucht. Daten an ein File auf einer Diskette anzuhängen, die von DOS I beschrieben wurde. Datei bitte auf TURBO-DOS-Format umkopieren.

- \$AD=173      FORMAT ERROR: Fehler beim Formatieren der Diskette:
- Diskette schreibgeschützt
  - Diskette konnte nicht ordnungsgemäß formatiert werden
- Falls der Fehler nicht behebbar ist, diese Diskette nicht verwenden.

### TECHNISCHE DATEN

Hier nun eine Zusammenstellung der wichtigsten Adressen und Systemvariablen von TURBO-DOS und -DUP.

Die Hinweise richten sich an Leute, die schon etwas Erfahrung in Maschinensprache haben und sind daher entsprechend knapp gehalten.

a) File Management System (FMS) : File "DOS.SYS"

Lage des FMS bei der Einstellung "2 open Files". "1 Drive & RD":

Normalversion	\$700 - \$1C23	Happy/Speedy	\$700 - \$1E6D
XF-Version	\$700 - \$1D64	Vollversion	\$700 - \$1F74

Folgende Adressen sind bei allen Versionen von TURBO-DOS ab Ver. 2.1 einheitlich:

- \$700    Anzahl der 64k-RD-Bänke (0-4).
- \$709    In dieser Speicherzelle steht die Anzahl der Files, die das DOS gleichzeitig offen halten kann. Sie ist durch die Anzahl der acht IOCB-Kanäle beschränkt. Das DUP fordert hier mindestens den Wert 2, bei exotischen Anwendungen sogar 3 (bei COPY aus einem EXE-File oder COPY mit Ausgabeumleitung) oder 4 (COPY mit Ausgabeumleitung aus EXE-File).
- \$70A    In diesem Byte ist verzeichnet, welche Diskettenstationen das DOS ansprechen kann. Für jede angemeldete Diskettenstation ist jeweils ein Bit gesetzt. Das niederwertigste Bit entspricht dabei Drive #1. das höchste Drive #8 (RD). Z.B. bedeutet der Wert \$83. dass die physikalischen Laufwerke mit den Nummern 1, 2 und 8 vom DOS ansprechbar sind.
- \$70B    Durch diese vier Bytes werden die
- \$70C    64k-RD-Bänke ausgewählt.
- \$70D    Die hier tabellierten Werte werden durch OR-Verknüpfung
- \$70E    in das MMU-Register \$D301 (PORTB der PIA) hinein-"maskiert".
- \$70F    Vektor zur Zurückkladeroutine von MEM.SAV und Sprung zum Modul
- \$710    wird vom DUP bei 'CAR' indirekt angesprungen.
- \$711    MEM.SAV-Flag: Steht hier \$FF, so ist MEM.SAV aktiv, d.h. es enthält den geschützten Speicherbereich, der nach Verlassen des DUP wieder zurückgeschrieben wird. Durch Eintragen von \$00 wird MEM.SAV bis zum nächsten DUP-Start deaktiviert.
- \$712    In diesen beiden Speicherzellen liegt die Anfangsadresse für die DOS-Puffer.
- \$713    Durch Verschieben nach oben kann man Speicherplatz für eigene Anwendungen reservieren.

- \$76C Mit der hier beginnenden DOS-Routine zum Schreiben und Lesen von Diskettensektoren werden die je nach Version implementierten hohen Datenübertragungsraten unterstützt. Von Anwendermaschinenprogrammen kann sie also zum schnellen Sektor-I/O benutzt werden. Vorbedingungen hierzu sind:
- die physikalische Laufwerksnummer muss in \$301 und \$21 eingetragen sein; \$08 steht für die RD.
  - die Sektorlänge muss in \$308/\$309 stehen.
  - die Sektornummer wird im Accu und Y-Register übergeben:  
A: HI-Byte, Y: LO-Byte.
  - das CARRY-Flag bestimmt die I/O-Richtung:  
CARRY clear (CLC) = Lesen; CARRY set (SEC) = Schreiben.
- Sind diese Vorbedingungen erfüllt, kann die DOS-Routine einfach mit "JSR \$76C" aufgerufen werden. Danach sind die CPU-Register wie nach Aufruf der SIO-Routine (\$E459) des ATARI-Betriebssystems gesetzt (Fehlerbyte im Y-Register usw.).
- \$779 Hier steht das SIO-Kommando zum Schreiben von Diskettensektoren. Es sind zwei Werte möglich: \$50 bedeutet Schreiben ohne, \$57 mit Verify.
- \$7E0 DOS-Puffer-Initialisierungsroutine; wird nach Laden des DOS und bei jedem System-RESET aufgerufen. Entsprechend den DOS-Variablen \$709, \$70A, \$712 und \$713 werden hierbei die Puffer für Laufwerke und Files bereitgestellt und danach der OS-Vektor in \$2E7/\$2E8 (LOWMEM) festgesetzt, der die Adresse der unteren Speichergrenze für Benutzeranwendungen enthält. Durch Einlesen eines ungültigen Directory kann es vorkommen, dass alle folgenden Directory-Aufrufe für dieses Laufwerk eine falsche Anzahl freier Sektoren anzeigen, die zumeist utopisch hoch liegt. In diesem Fall darf man keine Operationen auf DOS-Disketten mehr vornehmen, bevor man nicht durch <RESET> oder aber durch 'RUN 7E0' die Puffer neu initialisiert hat. Dann ist alles wieder ok.
- \$12FC BACK-Adresse: wird bei 'CAR' ohne Cartridge angesprochen. (-> DUP-Kommando 'BACK')
- \$153F In dieser Speicherzelle steht die Nummer der Diskettenstation, von der DUP.SYS und MEM.SAV geladen werden. Zulässig sind Werte zwischen \$31 (D1:) und \$38 (D8:).
- \$159E Dieses Flag legt fest, ob COM-Files nach dem Laden über DOS initialisiert und gestartet werden (\$00=ja, \$FF=nein).
- \$156C DOS-Routine zum Laden von COM-Files. Der Zeiger auf den Puffer, der den Namen des zu ladenden Files in der Form "Dn:FILENAME.EXT" enthält, muss der Routine in den CPU-Registern (X: HI-Byte, A: LO-Byte) übergeben werden.
- \$15C8 Ein weiterer Einsprung zum COM-File-Laden, der oft vom BASIC aus benutzt wird. Hierbei muss vor Aufruf der Routine IOCB-Channel #1 auf das zu ladende File zum Lesen geöffnet sein. Abhängig vom Flag \$159E wird das Programm dann initialisiert und/oder gestartet, wie das auch beim Einsprung \$156C der Fall ist. Wird übrigens innerhalb eines COM-Files von einer durch die INIT-Adresse angesprochenen Routine der IOCB #1 benutzt, kann das FMS das File nicht mehr zu Ende laden, da es hierzu selbst diesen Kanal verwendet.

b) Disk-Utility-Package (DUP): File DUP.SYS

Mit Hilfe der folgenden Adressen könnt Ihr Euer TURBO-DUP individuell gestalten (z.B. Befehlskürzel. Farbe ändern etc.). Die so entstehende persönliche Version pflanzt sich über 'INIT /N' fort. Wir weisen jedoch darauf hin, dass jede solche Version genauso dem Urheberrecht untersteht wie alle anderen TURBO-DUP-Utilities, da ja diese Möglichkeit zur Manipulation quasi mitgeliefert wird und es sich folglich dabei nicht um eine von Dritten erstellte Software handelt.

- \$2203 Directory-Ausgabe ein- (\$FF) oder zweispaltig (\$00).
  - \$2205 DUP-Bildschirmfarbe; Default: \$50
  - \$2206 Modus-Byte (-> MODE).
  - \$2207 Laufwerksnummer, auf die bei <SELECT> umgeschaltet wird.
  - \$2208 Herstellungsdatum (2 Bytes), dezimal codiert.
  - \$220A "\*.COM": Aus 5 Bytes bestehende Zeichenkette, die bei <SHIFT>&<RETURN> an den Dateinamen angehängt wird (-> LOAD).
  - \$220F "».EXE": w.o. mit <CONTROL>&<RETURN> (-> EXECUTE).
  - \$2214 ATASCII-Tabelle der vier möglichen Option-Appendices "NAVD".
  - \$2218 Tabelle der Ein-Zeichen-Kommandos mit zugehörigem Software-Stack.
  - \$2230 Tabelle der DUP-Befehlskürzel mit RTS-Adressen.
- Liste der zusätzlichen XIO-Kommandos (siehe BASIC-Manual):
- |          |     |          |     |          |     |
|----------|-----|----------|-----|----------|-----|
| \$22=34  | CLR | \$FF=255 | FMS | \$FE=254 | FME |
| \$FD=253 | FMD | \$FC=252 | FMQ |          |     |

### TURBO-DOS XL/XE in der Praxis

Viele von Euch werden zumindest bei der intensiveren Arbeit mit unserem DOS dessen Vorzüge kennenlernen und wünschen, dieses bei möglichst vielen Gelegenheiten einzusetzen. Aus diesem Grund haben wir versucht, das DOS mit umfangreichen Einstellmöglichkeiten zu versehen, um es für jede Anwendung individuell gestalten zu können. Mit der nötigen Erfahrung und Phantasie ist dies sicherlich nicht schwer zu bewerkstelligen.

Letzteres kann man auch mit der besten Anleitung nicht ersetzen, bei Mangel an Ersterem aber möchten wir Euch noch eine kleine Hilfestellung geben.

Grundsätzlich kann TURBO-DOS XL/XE mit jeder Software zusammenarbeiten die dies bisher mit DOS 2.0 oder DOS 2.5 getan hat (Ausnahme: Programme, die mit "ihrem" FMS sehr unsauber umgehen, d.h. ungebräuchliche Programmiertechniken und/oder Adressen benutzen).

Wie man seine Software auf unser DOS umstellt, wurde allgemein bereits auf Seite 5 kurz erklärt. Für den Anfänger wollen wir dies an einem Beispiel erläutern:

Durch seine einfache Bedienung und den günstigen Preis hat das Zeichenprogramm "DESIGN MASTER" eine weite Verbreitung gefunden. Wer dieses Programm nicht besitzt, sollte trotzdem die nachfolgend beschriebene Vorgehensweise studieren, da eine Anpassung immer nach dem gleichen Verfahren verläuft.

- die TURBO-DOS-Masterdisk wird gebootet
- die BACK-Adresse wird auf \$1D00 eingestellt (RUN-Adresse oder Programmbeginn des DESIGN MASTER-Start-File, die wir mit Hilfe des HEAD-Befehls erhalten)
- Da das Programm START.COM bereits bei Speicherzelle \$1D00 beginnt darf nur die Grundversion des DOS mit 1 Laufwerk und 2 offenen Files (-> CONFIG.COM) verwendet werden.
- eine Arbeitsdiskette, auf der sich das Programm später befinden soll, wird formatiert (je nach Laufwerk auch in Double oder Quad Density !)
- DOS.SYS und DUP.SYS werden mit INIT auf die Arbeitsdisk geschrieben
- hierauf werden nun alle Files (außer DOS.SYS) von der DM-Originaldiskette kopiert (COPY-Befehl oder FILECOPY.COM)
- das File AUTORUN.SYS wird in START.COM umbenannt
- AUTORUN.SYS wird von der TURBO-DOS-Masterdisk auf die Arbeitsdisk kopiert
- mit dem Programm BATEDIT.COM wird das File SETUP.BAT für die Arbeitsdisk erstellt:

```
DUP.SYS      ( <CONTROL>&<D> )
MEM.SAV     ( <CONTROL>&<M> )
RUN:        ( <CONTROL>&<R> )
START.COM   ( im Klartext eingeben )
```

<CONTROL>&<ESC>

<S>

<RETURN>. womit das File SETUP.BAT auf die Arbeitsdisk abgespeichert wird (die zuvor natürlich in die Floppy eingelegt werden muss).

Die so erstellte Diskette kann jetzt gebootet werden. Im Programm gelangt man nun jederzeit mit <SHIFT>&<RESET> ins DUP, das man mit CAR wieder verlassen kann, ohne dass evtl. vorhandene Daten gelöscht werden (sofern man MEM.SAV nicht zerstört; -> DUP). Beim Sprung ins DUP sollte man beachten, dass im bearbeiteten Bild kein Fenster geöffnet ist, da dieses sonst als Bildinformation abgespeichert wird .

- Zusammenarbeit mit ROM-Modulen

ROM-Module, die das Booten eines DOS' erlauben, können meist mit unserem DOS zusammenarbeiten. Als Beispiel sei ATARIARTIST (Maltafel) genannt, wobei das DOS ohne TURBO-Beschleunigung gebootet werden muss. Aus dem Malprogramm gelangt man jederzeit mit <SHIFT>&<RESET> ins DUP, wo man z.B. Bilder von RD auf Diskette oder umgekehrt kopieren kann. Im Programm selbst kann man im Menüpunkt "Storage" auf die RD zugreifen, wenn bei (L)oad oder (S)ave als Device 'D8:' eingegeben wird.

Anhang zu Seite 10. Punkt (c)

(? Bei Enhanced Density ist eine erweiterte VTOC notwendig, die bei DOS 2.5 (und ohne Änderung von (c) auch bei unserem DOS) in Sektor 1024 liegt, also auf dem letzten Track, während sich der erste Teil bekanntlich auf einem mittleren Track befindet. Bei einem Zugriff auf die VTOC wird daher der Schreib-/Lesekopf zwischen beiden Teilen hin- und hergeschoben. Die Lage der erweiterten VTOC wird mit der Einstellung 'N' bei Punkt (c) auf Sektor 268 verlegt, womit die erwähnte "Kopfarbeit" umgangen wird, die Maximalzahl der Directory-Einträge jedoch auf 56 (statt 64) sinkt. Mit einem so eingestellten DOS dürfen aber keine herkömmlichen Enhanced Density Disketten mehr beschrieben werden (und auch umgekehrt nicht), weshalb das Verlassen der Kompatibilität zu DOS 2.5 nur sehr bedingt zu empfehlen ist (z.B. wenn man überwiegend mit Enhanced Density arbeitet und wenig Disketten mit anderen Usern austauscht). )

## Alphabetisches Verzeichnis der Fachbegriffe

Die nachstehende Zahl verweist auf die Seite(n), wo der fragliche Begriff verwendet bzw. erklärt wird.

- Abort (= Abbruch) 14
- Abspeichern (-> SAVE) 32
- ATASCII (= der im ATARI verwendete ASCII-(Zeichen-)Code)
- Backslash (= Rückwärtsschrägstrich '\') 24
- Batch-File 4, 13
- Benutzeroberfläche 4, 8, 19
- Betriebssystem 18
- Booten 4
- CAR Cartridge 34
- CLR Clear 27
- COM-File Compound-File (= Maschinensprache-File) 23
- COP Copy (= Kopieren) 28
- Cursor (= Zeiger auf Bildschirmposition, an der Tastatureingaben erscheinen)
- Datei (-> File) 21
- Dateiname (-> Filename) 23
- Dateispezifikation 23
- DEL Delete (= Löschen) 26.2?
- Density (= Aufzeichnungsdichte) 7
- Destination (= Ziel(lauf werk)) 29, 40
- Device (= Gerät) 24
- DIR Directory (= Inhaltsverzeichnis) 21
- Diskette (= Speichermedium für Diskettenlaufwerke)
- DOS (= Disk Operating System) 8
- Double Density (= doppelte Dichte) 7
- Drive (= Laufwerk)
- DUP (= Disk Utility Package) 18
- Editor 16, 20
- Enhanced Density (= erweiterte Dichte) 7
- Error (= Fehler) 53

EXE Execute(-File) 37  
Extender (= Anhang des Dateinamens) 21, 23  
File (= Datei; zusammenhängendes Datensegment auf Diskette) 8, 21  
Filename (= Dateiname) 21, 23  
Filespec (= Dateispezifikation) 22, 24  
Floppy (= Diskettenlaufwerk)  
FMD, FME, FMQ, FMS Formatierbefehle 27  
FMS (= File Management System) 8  
Format (= Art der Formatierung)  
Formatieren 27  
Geschützt (-> LOCK)  
HEA Head 33  
HEL Help (= Hilfe; Konsolentaste) 19, 21  
Inhaltsverzeichnis 1 21  
INI Init (= schreibe DOS-Files) 28  
Invers 13  
Laden (-> LOAD)  
Laufwerksnummer 35 (21, 24)  
LOA Load (= Laden) 30  
LOC Lock (= Schützen) 25  
Locked (= geschützt)  
Löschen (-> Delete)  
Masterdisk (unterscheidet sich bei uns von der Originaldisk !) 4  
Medium Density (= Enhanced Density ->)  
Meldung (-> Prompt)  
MEM.SAV 18, 19  
MOD Mode 34  
More (= mehr) 22  
OPTION (= Konsolentaste; -> CAR) 19  
Prompt (= Meldung) 19  
Quad Density (= vierfache Dichte) 7  
RAM (= Random Access Memory; Schreib-/Lesespeicher)

RD (= RD) 11, 12  
Recover (= Löschen/Zerstörung rückgängig machen) 46  
REN Rename (= Umbenennen) 26  
Reset (= Computerchips in definierten Anfangszustand zurücksetzen)  
ROM (= Read Only Memory; nicht flüchtiger Nur-Lesespeicher)  
SAV Save (= Abspeichern) 32  
Schützen (-> LOCK)  
Scrollen (= Verschiebung des Bildschirminhaltes) 22  
Sektoren (= Bestandteile (Unterteilung) der Diskettenspuren) 7  
SELECT (= Konsolentaste; -> Laufwerksnummer) 19  
Single Density (= einfache Dichte) 7  
Source (= Quelle(niaufwerk)) 29, 40  
Speichermedium (= Diskettenlaufwerk, Datenrekorder, etc.)  
START (= Konsolentaste; -> DIR) 19  
SUM  
Prüfsumme und Länge eines Files 33  
Track (= Diskettenspur) 7  
TYP Type 33  
UNL Unlock (= Schutz aufheben) 26  
Utility (= Software-Werkzeug)  
Verify (= Überprüfung der geschriebenen Daten); -> MODE)  
VTOC (= Volume Table Of Contents) 8  
Wildcards (= Platzhalter in Dateinamen) 23

#### Nachtrag zum SPEEDY-Enhancement

- Beim Formatieren mit 'FMS /A' bzw. 'FMD /A' meldet sich die Floppy sofort wieder zurück, obwohl noch formatiert wird (siehe Befehl \$20 "automatisches Formatieren" in der SPEEDY-Anleitung). ( ? Hierzu wird \$24A,X auf \$FF gesetzt, sonst \$00. )
- Bekanntlich verfügt die SPEEDY über einen Track- und Cache-Puffer, der bei der HS-Version unseres DOS nach jedem Schreibvorgang auf Diskette geschrieben wird (Kommando \$51). In der EX-(Voll-)Version ist dies nicht der Fall, weshalb vor Entnahme der Diskette auf das Erlöschen der Busy-Lampe gewartet werden muss.
- Damit bei obigem Kommando nicht jedes Mal der Motor abgeschaltet wird (default), wurde im SETUP.BAT der HS-Version das "Drive/Display-Byte" auf \$08 gesetzt. Dies sollte man also auch bei eigenen Disketten vorsehen. Wird die Diskettenstation aus- und eingeschaltet, geht eine vorgenommene Konfigurierung verloren. Für diesen Fall und nachträgliche Änderungen der Konfiguration aus dem DUP existiert das Programm SEND.COH, mit dem direkt die beiden Kontrollbytes gesendet werden können. Die im Programm eingestellten Bytes werden mit <RETURN> abgeschickt; mit <ESC> kann es unverrichteter Dinge verlassen werden.

#### Nachtrag zur XF 551

Sowohl das FMS (in der Version XF und EX) als auch DISKCOPY.COM prüfen selbsttätig, ob XF-Laufwerke vorhanden sind (was die gemischte Verwendung mit anderen Laufwerken erlaubt). Aus diesem Grund ist darauf zu achten, dass beim Booten bzw. Programmstart alle XF 551 eingeschaltet sind.

Referenzblatt zu TURBO-DOS XL/XE Version 2.1

CONSOL-Tastenbelegung

<HELP> TURBO-DUP-Befehle ausgeben  
 <START> Directory anzeigen  
 <SELECT> Laufwerksnummer umschalten  
 <OPTION> CARtridge  
 <SELECT>&<RESET> Neu Booten  
 <SHIFT>&<RESET> Ins DUP springen

Diskettenbefehlsliste

DIR file Directory anzeigen  
 DIR + 1/2-spaltig umschalten  
 REN f1,f2 Umbenennen von f1 auf f2  
 DEL file Datei löschen  
 LOC file Datei sichern  
 UNL file Dateisicherung aufheben  
 CLR Disk löschen  
 FMS Single formatieren  
 FME Enhanced formatieren  
 FMD Double formatieren  
 FMQ Quad formatieren  
 INI DOS/DUP schreiben  
 INI /D Nur DOS.SYS schreiben  
 INI /N Nur DUP.SYS schreiben  
 COP f1,f2 Kopieren von f1 auf f2  
 LOA file COM-File laden und starten  
 file <SHIFT>&<RETURN> dto.  
 LOA f/D dto. ohne starten  
 SAV f a1 a2 Abspeichern von a1 bis a2 als COM-File  
 SAV f a1 a2 a3 dto., aber beim Laden nach a3 legen  
 SAV f/n a1 a2 Abspeichern als normales Datenfile  
 HEA file COM-Header anzeigen  
 SUM file Länge und Prüfsumme  
 TYP file ASCII-File ausgeben

Erläuterungen zu den Abkürzungen:

A1, a2, a3 Speicheradressen  
 file, f, f1, f2 Filespec

Filespezifikationen

Präfixe (vor dem Dateinamen):  
 D:, 8, etc. = Laufwerksnummer  
 \ (Backslash) es werden alle Files angesprochen, die nicht den folgenden Namen haben  
 0-99/ gibt an, wieviele Files dieses Namens übergangen werden sollen

Wildcards (Platzhalter):

? steht für einen Buchstaben  
 \* steht für mehrere Buchstaben  
 - steht für den Rest des Namens  
 <RETURN> steht für alle Files

Suffixe (hinter dem Dateinamen):

/N Quittiermodus unterdrücken  
 /D Direktkommando ans DOS  
 /A (Append) Anhängen an Datei  
 /V Verify aktivieren

Weitere DUP-Befehle

BON BASIC ein  
 BOF BASIC aus  
 CAR DUP verlassen (evtl. mit BASIC- Befehl als Übergabeparameter)  
 EXE f EXE-File starten  
 f <CONTROL>&<RETURN> dto.  
 1-8 Laufwerksnummer ändern  
 HEL Anzeigen aller DUP-Kommandos  
 RUN Sprung zur letzten RUN-Adresse  
 RUN a Sprung nach Adresse a  
 BAC CAR-Adresse anzeigen  
 BAC a neue CAR-Adresse eintragen  
 MOD Modus-Nummer anzeigen  
 Bit 0 Quittiermodus aus  
 Bit 1 Verify einschalten  
 Bit 2 Bei COP MEM.SAV benutzen  
 Bit 3 Auto-Unlock ein  
 Bit 4 Lade-Ton aus  
 Bit 5 ERROR-Ton aus  
 Bit 6 MORE-Modus aus  
 Bit 7 unbenutzt  
 MOD a Modus-Nummer ändern  
 VER Ausgabe der Version

Monitor-Befehle

>a Hex-Dump ab Adresse a  
 <a ASCII-Dump  
 + / - Dump 8 Bytes vor/zurück  
 =zahl Zahl in allen Basen ausgeben  
 >a zahl zahl ... : Ab a die nachfolgenden Zahlen byte-weise eintragen  
 ! Dev Zusätzliche Ausgabe der Bildschirmdaten auf Gerät Dev

Zahleneingaben:

Dezimal .13 .255 .19834  
 Hexadezimal A3 FFFD  
 ASCII-Code 'a 'C '8 '&  
 Offset bzgl. letzter Zahl: +2F0 +.123