

K - D O S (TM)

H a n d b u c h

K-DOS ist vollstaendig kompatibel mit dem ATARI 2.0S und anderer verwandter Software. K-DOS bietet dem Anwender groessere Zuverlaessigkeit, Flexibilitaet und Kontrolle.

Die Eigenschaften des K-DOS:

- * Ein Maschinsprache-Monitor, welcher die Pruefung und Veraenderung des Speichers in Hexadezimal erlaubt und in ATASCII darstellt.
- * K-DOS arbeitet mit Kommando-Zeilen.
- * K-DOS verbleibt im Speicher.
- * DOS-Befehle koennen ausgefuehrt werden, auch wenn mit dem BASIC- oder ASSEMBLER-Modul gearbeitet wird.
- * Disk-Files koennen direkt auf Cassette ueberspielt werden.
- * Cassetten-Files koennen direkt auf Disk ueberspielt werden.
- * Die Unterdrueckung des BREAK-Befehls bringt das System nicht zum Zusammenbruch, sondern fuehrt zurueck zum K-DOS.
- * Neue, leistungsstarke Befehle reservieren und freisetzen Speicher-Platz.
- * K-DOS ermoeglicht dem Anwender, eigene Befehle zu definieren.
- * K-DOS beinhaltet einen "null-handler", beschleunigt das Testen und die Fehlersuche (debugging).
- * Die Befehle sind in englisch, mit Abkuerzungen.
- * Die Fehlermeldungen werden in englisch ausgegeben.

I n h a l t s v e r z e i c h n i s

A. Vorbereitung.....	12
B. File - Kontrolle.....	16
C. Programm - Kontrolle.....	21
1. Back: vom DOS zurueck zum ROM - Modul.....	21
2. WARM: Durchfuehren eines "Warmstarts", schliessen eines Files, zuruecksetzen der "Pointer".....	21
3. COLD: "Kaltstart" des ROM-Moduls.....	22
4. Xit: Ein anderer Weg zurueck zum ROM-Modul.....	22
5. UNLOAD: Loeschen des Speicherbereichs des ROM-Moduls.....	22
6. LCMem: Setzen der Speicher-Basis.....	23
7. DC: Definieren von Zeichen, um DDS-Befehle vom ROM-Modul aus zur Ausfuehrung zu bringen.....	23
D. Maschinensprache-Monitor.....	24
1. Starten eines Files (Object-File laden und starten).....	24
2. File in den Speicher laden.....	25
3. Speicherinhalt auf Disk-File "save".....	25
4. Programmstart ab Hexadezimal-Adresse.....	26
5. Fortsetzen des Programmablaufs nach BREAK- Befehl.....	26
6. Pruefen des Speicherinhalts.....	27
7. Versendern des Speicherinhalts.....	27
8. Pruefen und Versendern der Register.....	28
E. Kontrolle ueber die Anlage.....	28
1. "Status" der gesamten Anlage.....	29
2. Neu-schreiben und Umschreiben der "Display-Liste", Neu-oeffnen des "Source Editors".....	29
3. Schliessen offener Files, Abstellen des Sound, sowie der "Vertical Blank Vectors".....	29
4. Hilfe fuer Betriebssystemfunktionen.....	30

F. DUP Spezial-Befehle.....	30
1. vom Anwender definierte Befehle (UDC).....	30
2. Identifikation der DUP-Version.....	31
3. Der "KILL"-Befehl.....	32
4. Der "REVIVE"(Wiederherstellen)-Befehl.....	32
Liste der Befehle.....	33
Anhaenge	
A Fehlermeldungen.....	35
B Versenden von Speicherinhalten.....	38
C Woerterbuch der Begriffe.....	39

Abchnitt I - Einleitung

K-DOS, ein Befehls-orientiertes DOS, bietet staerkere und bequemere Eigenschaften als das ATARI-DOS 2.0S. Aus diesem Grunde ist es leichter zu benutzen. K-DOS erlaubt dem Anwender den direkten Zugriff auf Disk-Files und leichtes Handhaben von Daten auf zahlreiche Arten, z.B. Programme speichern oder laden, Files anhaengen oder loeschen, Speicheradressen veraendern, u.s.w.

Eine der bemerkenswertesten Eigenschaften des K-DOS ist das Vorhandensein eines Maschinensprache-Monitors, welcher dem Anwender das Ueberpruefen und Veraendern des Speichers ermoeglicht.

K-DOS verbleibt immer im Speicher, daher ist es nicht notwendig, es jedesmal zu laden, wenn es benoetigt wird. Zusatzlich unterstuetzt K-DOS die Steuerung des ATARI SCC-Interface, welches die Benutzung von Peripheriegeraeten wie Drucker und Modems ermoeglicht.

K-DOS bietet folgende Vorteile:

1. Bequemlichkeit

- (a) K-DOS muss nicht jedesmal geladen werden, wenn es benutzt werden soll.
- (b) K-DOS wiederholt das Laden eines Programms solange, bis es korrekt geladen ist.
- (c) K-DOS erkennt mehrere Moeglichkeiten von File-Namen und sog. "Wildcards" (das sind z.B. *.* in File-Namen), um somit grossere Kompatibilitaet zu anderen Systemen herzustellen.
- (d) K-DOS erlaubt das Abkuerzen von englischen Befehlen.

2. Flexibilitaet

- (a) DOS ermoeglicht dem Anwender das Definieren persoenlicher Befehle.
- (b) Diese persoenlichen Befehle bewirken die Ausfuehrung von DOS Befehlen waehrend der Arbeit mit dem BASIC- oder ASSEMBLER-Modul.

3. Sprachuebersetzbarkeit

Die Befehls-Belegungen werden in englischer Sprache angegeben.

4. Kontrolle

Neue, leistungstarke Befehle, wie z.B. COLD und LOMem, bieten mehr Kontrolle ueber das System.

5. Zuverlaessigkeit

Das Schreiben grosser Datenblöcke in den Speicher ist sicherer.

Die K-DOS Masterdiskette beinhaltet die folgenden System-Files:

DOS.SYS	FMS & DUP (sind immer im Speicher enthalten).
TRANS.SYS	TRANS-Befehl kopiert Files auf einem Single-Drive-System (UDC).
UDC.SYS	Der "UDC"-Befehl.
CHEKPR.SYS	Ein File, welches die Fehlermeldungen darstellt und dem Anwender gestattet, diese Meldungen zu veraendern (UDC).
ERRAZF.SYS	Dieses Programm beseitigt die Fehlermeldungen und ermöglicht ausserdem, das UDC-Verzeichnis aus dem K-DOS zu entfernen, um somit zusaetzlichen Speicherplatz frei zu machen.
DISKDU.P.SYS	Der "DISKDU"-Befehl wird benutzt, um ganze Disketten zu kopieren (UDC).
HELP.SYS	Der "HELP"-Befehl (UDC). Wenn dieses Programm aufgerufen wird (Eingabe: "HELP" oder nur "H") erscheint eine kurze Zusammenfassung aller moeglichen K-DOS-Befehle.
EQUATE.ASM	Ein File fuer den Assembler-Programmierer (system equate).
DIQU.ASM	ein Gleichsetzungs-File, um Adressen innerhalb des DOS zu erreichen (umfangreiche Adressen, einschliesslich vom Anwender aufrufbare Unterrouinen)
HELP1.F.SYS	Teil des "HELP.SYS".

Abschnitt II

Es wird angeregt, auf der Masterdiskette einen Schreibschutz anzubringen und mit "DISKDUP" sofort eine Arbeitskopie anzufertigen. Anschliessend kann die Masterdiskette an einem sicheren Ort aufbewahrt werden, wo sie nicht versehentlich gelöscht oder überschrieben werden kann.

Abschnitt III

Laden der Diskette

A. K-DOS wird genauso geladen wie das ATARI-DOS.

1. Schalten Sie das Fernsehgerät oder den Monitor ein.
2. Schalten Sie alle Disk-Stationen ein.
3. Schalten Sie -falls vorhanden- das ATARI-850-Interface ein, wenn Sie irgendwelche Peripheriegeräte, wie z.B. Drucker oder Modem benutzen wollen.
4. Legen Sie die K-DOS-Masterdiskette in Drive 1 ein, nachdem das "Busy"-Signal erloschen ist.
5. Schalten Sie den Computer ein. K-DOS wird nun "geladen".

Sollten Sie einen "BOOT-ERROR" erhalten, schalten Sie den Computer fuer ein paar Sekunden aus und dann wieder ein. Sollten Sie nach wie vor einen "BOOT-ERROR" erhalten, prüfen Sie, ob die Klappe des Disk-Drives geschlossen ist und alle Verbindungen ordnungsgemäss bestehen.

B. Speicherbedarf

K-DOS braucht 14 K Speicherplatz, einschliesslich 2 K fuer Run-BSS-Handler. Fuer den Gebrauch der Utilities (DISKOP, DISKUS) sollten mindestens 32 K Speicherplatz vorhanden sein.

D. Arbeitsablauf nach "booten" des K-DOS

das K-DOS

1. Lädt sich automatisch in den Speicher.
2. initialisiert sich selbst.
3. "bootet" das 850-Interface.
4. schreibt seinen Titel auf den Bildschirm.
5. beachtet ein ggf. vorhandenes AUTORUN.SYS-File
(veranlasst das automatische Laden dieses Files).
6. Springt in ein evtl. vorhandenes Modul oder lädt
weiter das DOS ein.

Um das K-Dos aufzurufen, während mit einem Modul gearbeitet wird, geben Sie vom BASIC- oder ASSEMBLER-Modul "DOS" ein oder drücken Sie die <SYSTEM RESET> - Taste gleichzeitig mit der <START> - Taste. K-DOS meldet sich daraufhin mit "DOS" und wartet auf Befehle.

Abchnitt IV

Eigenschaften

K-DOS besteht aus zwei Teilen:

- A. FMS - File Management System, das "Kontroll"-Programm.
- B. DUP - Disk Utility Program, eine Anzahl von Hilfsprogrammen zur Ausfuehrung der ueber das FMS aufgerufenen Befehle. Das DUP fuehrt also die Arbeit fuer das FMS aus.

A Das FMS in dieser Version des K-DOS bietet die folgenden Eigenschaften:

1. Die <BREAK> - Taste stoppt sofort jede Ein- und Ausgabe der Disketten-Station. Ein mehrmaliges Betaeetigen dieser Taste ist also nicht mehr notwendig.
2. Das Schreiben umfangreicher Datenblöcke in den Speicher ist sicherer, weil nicht mehr direkt aus dem Speicher heraus geschrieben wird. Betaeetigt man beim ATARI-FMS die <BREAK> - Taste oft genug, um ein "Save" vom BASIC aus abzubrechen, wird gleichzeitig das im Speicher enthaltene Programm zerstört.
3. FMS ueberwacht den Status der Operation (Ueberwachung, ob die Operation ordnungsgemaess ablaeuft oder ob ein Fehler auftritt) des Disk-Drives nur bei einem "COLD-Start". Bei <SYSTEM RESET> werden die bereits vorhandenen Daten benutzt.
4. Das erste Zeichen eines File-Namens kann auch eine Zahl sein. Das ATARI-DOS hingegen gestattet dies nicht.
5. Das Anhaengen (Appending) eines Files wurde wirkungsvoller gemacht. Der vorhergehende (alte) Programmblock wird verwendet, wodurch die Verschwendung von Speicherplatz vermieden wird.

B. Die Eigenschaften des Disk Utility Program (DUP) sind erkennbarer als die des FMS. Sie beinhalten:

1. Einen Maschinensprache-Monitor. Die "Alter/Examine"- (Verändern/Ueberprüfen-) Befehle nutzen die Vorteile des Screen Editors, indem sie dem Anwender ermöglichen, den Speicher in Hexadezimal zu listen und zu verändern, und dies im ATASCII-Code darzustellen. Der Screen Editor kann effektiv genutzt werden, weil die Syntax fuer die Veränderung der Adressen die gleiche ist, wie sie auf dem Bildschirm geschrieben wird.
2. Englische Befehle mit Abkuerzungen. Die am häufigsten gebrauchten Befehle koennen durch Abkuerzungen in Form eines einzigen Buchstabens gegeben werden: D, C, B, u.s.w. Risikoreichere Befehle, INIT, FORMAT, COLD, DELETE und LOMem benoetigen mehr als einen Buchstaben als Abkuerzung, um - so sicher zu gehen, dass nicht versehentlich ein falscher Befehl gegeben wird, welcher dann Daten unwiederbringlich loescht.
3. Mehr, leistungsstarke Befehle: LOMem, COLD. Diese Befehle gestatten eine groessere Kontrolle ueber das System. LOMem kann benutzt werden, um Speicherplatz zu reservieren, waehrend der "COLD"-Befehl, welcher einen Kaltstart des Moduls bewirkt, das BASIC veranlasst, den Speicher zu loeschen.
4. Englische Fehlermeldungen, dargestellt ueber das DUP, und die Moeglichkeit, diese zu beheben. "ERROR on" bringt eine Fehlermeldung entsprechend dem aufgetretenen Fehler (in Dezimal) zur Anzeige. Anhang A zeigt eine Liste von moeglichen Fehlermeldungen, wie sie vom "CHERROR.SYS"-File zur Ausgabe gebracht werden. Fehler-Nummern unter 128 werden fuer DUP-Fehler benutzt. Einige Fehlermeldungen sind die gleichen wie im ATARI-BASIC.
5. Die Anwendung des "BRK"-Befehls fuehrt zurueck ins DOS. Ein "BRK" im ATARI-DOS fuehrt normalerweise zum Zusammenbruch des Systems.
6. Befindet sich ein Modul im Schacht, fuehrt die gleichzeitige Betaeetigung der Tasten <SYSTEM UNIT> und <START> direkt ins DOS, ohne das Modul abzuschreiben.

7. Leicht und flexibel zu benutzende Syntax.
 - (a) Bei File-Namen kann eine Zahl als erstes Zeichen verwendet werden.
 - (b) Kommas koennen benutzt werden, wenn ihnen ein Leerfeld folgt.
 - (c) in kleinen Buchstaben geschriebene Eingaben sind moeglich.
 - (d) Die Befehle koennen als Abkuerzung eingegeben werden.

8. K-DOS ist kompatibel mit dem ATARI-DOS 2.05. Der Anwender, welcher mit dem ATARI-DOS vertraut ist, kann leicht zu den Vorteilen des K-DOS aufbauen.

9. Das DUP verbleibt im Speicher. Das K-DOS muss daher nicht jedesmal von der Diskette geladen werden, wenn es benutzt werden soll.

10. DDC (User Defined Commands) gestattet dem Anwender das Definieren eines Befehls, welcher ein Maschinenprogramm laedt und startet.

11. DC (Defined Character)- Befehl. Gestattet die Ausfuehrung von DOS-Befehlen, waehrend mit einem Modul gearbeitet wird.

12. Das ATARI-850-Interface wird "gebootet".

13. Vom Anwender aufrufbare Unterroutinen. Routinen innerhalb des DOS, mit einfache I/O (Eingabe/Ausgabe)-Routinen, um das DOS selbst zu versendern. Siehe hierzu DEOU.ASM-File.

14. Die Befehle NOTE und POINT arbeiten nun mit dem Screen Editor. POINT ist aehnlich dem BASIC-Position-Befehl und NOTE ist das Gegenteil hierzu. Dieses wird mehr zur Abstimmung beim "cleaning" als zum "poken" in den Speicherbereich benutzt.

Abschnitt V

Befehle

Dieser Abschnitt beschreibt die Befehle des K-DQS im einzelnen. Diese Befehle sind in folgende Kategorien aufgeteilt, so dass der Anwender leicht nachschlagen und diese Befehle benutzen kann. Jedem Befehl folgt ein Beispiel, um die Funktion darzustellen. Eine Liste der möglichen Befehle befindet sich auf Seite .

- A. Vorbereitung der Diskette, um Daten aufzunehmen, das Kopieren einer Diskette.
- B. File-Kontrolle.
Manipulation von Files.
- C. Programm-Kontrolle. .
Die Handhabung des Operations-Systems.
- D. Maschinsprache-Monitor.
Direkter Zugang zum Speicher.
- E. Kontrolle über die Anlage.
Kontrolle über die gesamte Anlage, einschliesslich Peripheriegeraete.
- F. GUP-Extras.
Anwender-definierte Befehle (UDC).
- G. Liste der Befehle

A. Vorbereitung

Jede Diskette muss formatiert werden, bevor sie benutzt werden kann. "Formatieren" heisst, die Oberflaeche der Diskette wird aufgeteilt in Spuren und Sektoren, so dass Daten auf die Diskette geschrieben und von der Diskette gelesen werden koennen. Auch eine Diskette, welche bereits Daten enthaelt, kann formatiert werden. Es ist jedoch dabei zu beachten, dass diese Daten hierbei unwiederbringlich geloescht werden.

Es sind zwei verschiedene Befehle moeglich, um eine Diskette zu formatieren:

- 1) INIT n
(n - mit einer vorausgehenden Leerspalte - steht fuer die Nummer einer Disketten-Station).
- 2) FORMAT n

1. INIT n

formatiert die Diskette in dem Drive n, hierbei werden alle evtl. enthaltenen Daten geloescht, mit Ausnahme des DOS.SYS.

Eingabe: INIT 1
(formatiere Diskette in Drive 1)

Bildschirm: Type Y to format Disk 1

Eingabe: Y
(druecken Sie die <RETURN> - Taste, wenn die Diskette nicht formatiert werden soll, ansonsten geben Sie "Y" ein).

Bildschirm: Saving DOS in D1:DOS.SYS

Eingabe: INIT

Bildschirm: Need 1 thru 8 for disk #
(Eingabe einer Nummer zwischen 1 und 8 fuer die Disketten-Station ist erforderlich).

2. FORMAT n
 formatiert die Diskette im Drive n, wobei
 saemtliche Daten - einschliesslich DOS -
 geloescht werden.

Eingabe: FORMAT 1
 (formatiere Diskette in Drive 1).

Bildschirm: Type Y to format disk 1

Eingabe: Y
 (druecken Sie die <RETURN> - Taste, wenn
 die Diskette nicht formatiert werden soll,
 ansonsten geben Sie "Y" ein).

Eingabe: FORMAT 2
 (formatiere Diskette in Drive 2)
 FORMAT 3
 (formatiere Diskette in Drive 3)

In allen Faellen ist die Ausgabe des
Bildschirms die gleiche wie bei "FORMAT 1".

3. WBOOT <n>
 mit diesem Befehl wird das DOS als File
 DOS.SYS auf die Diskette geschrieben. Die
 Diskette muss selbstverstaendlich vorher
 formatiert worden sein. WBOOT kann fuer
 Disketten verwendet werden, auf denen sich
 bereits Files befinden. Diese werden
 hierdurch nicht beruehrt.

Eingabe: WBOOT (um DOS auf Drive 1 zu schreiben).

Bildschirm: Saving DOS in D1:DOS.SYS

Eingabe: WBOOT 2 (um DOS auf Drive 2 zu schreiben).

Bildschirm: Saving DOS in D1:DOS.SYS

4. DISKdup <quelle<<, >ziel>>/All>
 <Write>>/Forever>>/Put>>

Der DDC-Befehl wird benutzt, um den Inhalt
einer Diskette auf eine andere Diskette zu
uebertragen. Dieser Befehl ruft ein
Programm auf, das DISKDUP.SYS, welches eine
Diskette Sektor fuer Sektor kopiert. Dies
geschieht, indem saemtlicher Speicherplatz
als Puffer benutzt wird, um den Inhalt der
Sektoren einzulesen und sie anschliessend
wieder auszuschriften. Reicht der
Speicherplatz hierfuer nicht aus, ist ein
mehrfacher Wechsel der Disketten
erforderlich, wenn nur eine Diskettenstation
verwendet wird. DISKDUP fordert in diesem
Fall zum Wechseln der entsprechenden
Disketten auf. Werden waehrend des
Kopiervorgangs die Tasten <RETURN> oder
<SPACE> betraetigt, so wird dieser Vorgang
abgebrochen.

/All

gibt die Anweisung zum Kopieren aller Sektoren, ohne Rücksicht auf das Directory.

/Write

gibt die Anweisung zur Ueberpruefung, ob die Daten korrekt geschrieben wurden.

/Put

bewirkt das Gegenteil von "Write". Ein Pruefvorgang findet somit nicht statt.

/Forever

Der Schreibvorgang wird solange wiederholt, bis er korrekt erfolgte. Wird die <BREAK> - Taste betaetigt oder ist das Schreiben eines Sektors voellig unmoeglich, unterbricht der Kopiervorgang und auf dem Bildschirm erscheint: "Type C, S, Q, A, or ? for help".

(Eingabe C, S, Q, A oder ? fuer Hilfe)

- C continue trying
(Schreibversuch fortsetzen)
- S skip
(Suecknahme des Kopier-Befehls)
- Q quit after this pass
(Abbruch nach diesem Kopier-Abschnitt)
- ? this message
(Darstellung dieser Befehlsliste auf dem Bildschirm)
- A abort immediately
(sofortiger Abbruch)

DISKDUP gibt eine Waernung aus, wenn es einen Sektor nicht lesen oder schreiben kann, und sagt ebenfalls, wieviele Sektoren bereits kopiert sind und wieviel fehlerhafte Sektoren festgestellt wurden.

Eingabe: DISK

Bildschirm: Disk duplicate V1.8

From (,to)

?

(hier ist anzugeben, mit welcher
Diskettenstation geschrieben werden soll)

Eingabe: 1

Bildschirm: Insert source disk, type <return>

?

(lege Quellen-Diskette ein, druecke
<RETURN>)

Insert destination disk, type <return>

?

(lege Ziel-Diskette ein, druecke <RETURN>)

***Diese beiden Anweisungen wiederholen
sich solange, bis die Diskette vollstaendig
und korrekt kopiert ist.***

Bildschirm: # sectors copied
(Anzahl der kopierten Sektoren)

Eingabe: DISK 1/F

Bildschirm: Insert source disk, type <return>

?

Insert destination disk, type <return>

?

(Instructions repeated)

sectors copied

*** wiederholte Ausfuehrung der Anweisung***

B. File-Kontrolle

Die folgenden acht Befehle koennen benutzt werden, um Files zu manipulieren:

1. Direct
2. Copy
3. DELEte
4. LOCK
5. UNlock
6. REName
7. APPend
8. TRansfer

1. Direct <File-Bezeichnung><,Ausgabe>

Dieser Befehl bewirkt das Listen des Disk-Directory fuer das angegebene File. Der Filename, der Extender (NAME.xxx) und die Anzahl der von diesem File belegten Sektoren werden gelistet. Ohne besondere Angabe geht der Computer davon aus, dass als File-Bezeichnung "Di:*.*)" und als Ausgabeart die ueber den Bildschirm gewünscht wurde.

Flagset: D

bewirkt die Ausgabe aller auf Disk 1 vorhandener Files auf den Bildschirm.

D2

bewirkt die Ausgabe aller auf Disk 2 vorhandener Files auf den Bildschirm.

D H*

Listet alle Files, deren Filename mit dem Buchstaben "H" beginnen. "*" ist hierbei eine sog. "Wildcard", ersetzt also jede beliebige Kombination von Zeichen.

D ,P:

Print das Disk-Directory ueber Drucker.

2. Copy input <,Ausgabe>

Dieser Befehl kopiert das im Speicher befindliche File auf das mit "Ausgabe" bezeichnete Geræat (Screen Editor, Drucker, u.s.w.). Wurde keine Angabe zur <Ausgabe> gemacht, geht der Computer davon aus, dass die Ausgabe ueber den Screen Editor (E:) gewünscht wird. Wollen Sie sich ein File lediglich ansehen, geben Sie ein: "C Filename" und druecken Sie <RETURN>.

Eingabe: C TEST.TXT

Das File TEST.TXT wird auf dem Bildschirm gelistet. Die Zeichen von BASIC-Programmen werden im allgemeinen als "garbadge" gelistet (unsinnige Zeichen).

Eingabe: C FILE 1 FILE 2

File 1 wird auf der gleichen Diskette als File 2 kopiert.

Eingabe: C GELD D2:AUSGABEN

Kopiert das File mit dem Namen "GELD" von Disk 1 auf Disk 2 und benennt dieses File dort um in "AUSGABEN".

Eingabe: C E:, SCHWIMM

Ein Textfile wird geoeffnet, indem man eingibt: "C E:,FILENAME" und <RETURN> drueckt (Im vorliegenden Fall das Textfile "SCHWIMM". Nun koennen Sie Ihren Text eingeben, z.B.:

Brustschwimmen

Rueckenschwimmen

Butterfly

Freistil

<CTRL 3>

Das druecken von <CTRL 3> kennzeichnet das Ende des Files. Vergessen Sie nicht das Druecken der <RETURN> - Taste vor der Eingabe von <CTRL 3>, denn der Text in der Zeile mit <CTRL 3> wird, wie das <CTRL 3>, nicht auf Diskette uebernommen.

Eingabe: C SCHWIMM
listet das File "SCHWIMM" auf dem
Bildschirm.

Eingabe: C FILENAME
wenn das File weniger als 20 Zeilen lang
ist, kann der Screen Editor als Text-Editor
verwendet werden.

Listen des Files: Benutzen Sie die
Cursor-Tasten, um das File zu editieren.
Fliegen Sie "E:" an das "C" an, dann steuern
Sie mit dem Cursor die zu editierenden
Zeilen an. Zum Beenden dieses Vorgangs
druecken Sie <CTRL 3>. Das File ist nun
editiert und entsprechend geaendert.

Eingabe: C PRETTY.ASM,P:
Das File mit dem Namen "PRETTY" wird ueber
den Drucker ausgegeben.

Eingabe: C PRETTY.OBJ,N:
"N" ist ein sog. "Dummy". Es hat keinerlei
Funktion. Die Ausgabe des Files ueber "N"
ermoeglicht -lediglich, sicherzugehen, ob
das File in Ordnung ist und gelesen werden
kann.

4. DFL file Filebezeichnung </Noquery>

Mit diesem Befehl kann ein File von der
Diskette geloescht werden. Nach Eingabe
dieses Befehls erfolgt die Abfrage, ob das
File tatsaechlich geloescht werden soll, es
sei denn, hinter der Filebezeichnung wird
"/N" eingegeben. In diesem Fall unterbleibt
die Abfrage.

Eingabe: DFL PIC2
Das File "PIC2" soll geloescht werden.

Bildschirm: Type Y to delete
D1:PIC2 Press (return) to keep file

Eingabe: Y <RETURN>

Das File "PIC2" wird nun geloescht.

Eingabe: DFL NAME/N
Das File "NAME" wird ohne vorherige Abfrage
sofort geloescht.

Bildschirm: DFL

4. LOCK Filebezeichnung
Mit diesem Befehl werden Files vor versehentlichem ueberschreiben oder loeschen gesichert.

Eingabe: LDC DRIVER.ASM

Nach Eingabe dieses Befehls wird das File "DRIVER.ASM" gesichert. Dies ist auf dem Bildschirm daran zu erkennen, dass im Directory vor dem Filenamen ein "*" steht. Der Versuch, ein gesichertes File zu loeschen oder zu ueberschreiben verursacht einen "ERROR 167", File Locked (File gesichert).

Es sei nochmals darauf hingewiesen, dass zur Eingabe eines Befehls die entsprechende Abkuerzung (gross geschriebene Buchstaben) ausreichend ist.

5. UNlock Filebezeichnung
Das (die) gesicherte(n) File(s) werden entsichert.

Eingabe: UN DRIVER.ASM

Der UNlock-Befehl macht den LOCK-Befehl rueckgaengig, sodass das nunmehr entsicherte File ueberschrieben oder geloescht werden kann. Im Directory wird das entsprechende File nicht mehr durch "*" gekennzeichnet.

6. RENAME File, Filename
Hiermit kann ein Filename im Directory geaendert werden.

Eingabe: REN SCHECK,BARGELD

Auf Drive 1 wird der Name des Files "SCHECK" in "BARGELD" geaendert.

Eingabe: REN D2:WALTER,WILLI

Auf Drive 2 wird der Name des Files "WALTER" in "WILLI" geaendert.

ACHTUNG: Es sollte vermieden werden, auf einer Diskette zwei Files mit dem gleichen Namen zu benennen.

7. Append <Quellenfile,> Zielfile
Hiermit koennen Daten (z.B. Files) an ein bereits bestehendes File "angehaengt" werden.

Eingabe: AP DRIVER.ASM,MAZER.ASM
Das File "DRIVER.ASM" wird an das File "MAZER.ASM" angefuegt.

Eingabe: AP LAENDER.TXT

BESSEN

NIEDERSACHSEN

NORDRHEIN-WESTFALEN

BADEN-WUERTEMBERG

<CTRL 3>

Da nicht Quellen- und Zielfile, sondern nur ein einziges File eingegeben wurde, geht der Computer davon aus, dass das Quellenfile "E:" (Screen Editor) ist. Das bearbeitete File erhaelt somit den Namen "LAENDER", die nachfolgenden Eingaben werden in dieses File geschrieben. Die Betaeufigung der Tasten <CTRL 3> beendet die Eingabe und wird nicht auf Diskette geschrieben.

8. Transfer: Filename </SIRG>
<,Filename></SIRG>

Hiermit koennen Files auf einer Diskettenstation kopiert werden. Nach diesem Befehl wird das bezeichnete File von einer Diskette gelesen, im Speicher abgelegt, und auf eine andere Diskette geschrieben. Der Speicher wird als Puffer verwendet, so dass das File u.U., je nach Laenge, in einem Zug gelesen werden kann. Dieses ist ein "DDC"-Programm in dem File "TRANS.SYS".

Eingabe: TR PRETTY.ASM

Das File "PRETTY.ASM" wird von einer Diskette auf eine andere Diskette kopiert. Je nach Laenge des Files ist es notwendig, Quellen- und Zieldiskette mehrere Male auszutauschen.

9. Prompt: Set up source, (return)
Set up destination, (return)

>>> source = Quelle..., destination = Ziel... <<<

10. Prompt: TR PRETTY,C:/SIRG

"C:/SIRG" wird eingegeben, wenn Daten auf die Zieldiskette uebertragen werden sollen.

C. Programm-Kontrolle

Dieses Befehls-Paket ermöglicht die Kontrolle ueber
das Operations-System. Z.B. Ruecksprung zum ROM-Modul,
Rueckkehr ins DOS, u.s.w.

- 1) Back
- 2) WARM
- 3) COLD
- 4) Xit
- 5) UNLOAD
- 6) LDMem
- 7) DC <Zeichen>

1. Back
Ruecksprung ins ROM-Modul (BASIC oder
ASSEMBLER).

Bildschirm: DDS

Eingabe: B

Bildschirm: READY (wenn BASIC-Modul im Schacht)

ODER

Bildschirm: EDIT (wenn ASSEMBLER-Modul im Schacht)

ODER

Bildschirm: No Cartridge (wenn kein Modul im Schacht)

zur Erinnerung: um zurueck ins DOS zu kommen, entweder
Eingabe "DOS" oder gleichzeitiges
Druecken der Tasten <SYSTEM RESET> und
<START>.

2. WARM
um einen "Warmstart" durchzufuehren,
reinitialisieren, ohne den Speicherinhalt
zu veraendern, Files zu schliessen, Pointer
zurueckzusetzen, ohne den Speicherinhalt zu
loeschen. Nur zu benutzen, wenn das DOS
moeglicherweise mit dem benutzten ROM-Modul
kollidieren koennte. Dieser Befehl ist
nuetzlich nach einem "RESET"-Befehl, wenn
man sicher ist, dass der Speicherinhalt des
Moduls in Ordnung ist.

Bildschirm: DOS DOS
Eingabe: WARM WARM
Bildschirm: READY (BASIC) READY (ASSEMBLER)

3. COLD

"Kaltstart" des Moduls. Das gleiche wie "NEW" im BASIC oder im EDIT/ASM., jedoch wirkungsvoller, weil der Anwender-Bereich im Speicher gelöscht wird.

Bildschirm: DOS
Eingabe: COLD
Bildschirm: Type Y if okay to coldstart cartridge?
Eingabe: Y
Bildschirm: EDIT (ASSEMBLER)
 DBLR
Bildschirm: READY (BASIC)

4. Xit

Ruecksprung vom DOS dorthin, von wo aus es aufgerufen wurde; eine andere Moeglichkeit, zurueck zum Modul zu kommen. Wurde das DOS beispielsweise waehrend des Abarbeiten eines Programms von diesem aufgerufen, wird das Programm dort fortgesetzt, wo es verlassen wurde.

Bildschirm: DOS
Eingabe: X
Bildschirm: READY (BASIC)

5. UNLOAD

Fuehrt zum Versuch, den Speicherbereich des Moduls zu loeschen. Entlaedt alle auf RAM basierenden Module, setzt LOMem zurueck ans Ende von DOS. Das Programm, welches sich zwischen dem DOS und dem LOMem-Bereich befindet, wird gelöscht.

Bildschirm: DOS

Eingabe: UNLOAD

Bildschirm: Type Y if ok to coldstart cartridge?

Eingabe: Y

Bildschirm: DOS

6. LOMem <hhhh>

Bestimmt die untere Speicheradresse fuer ein Modul. Dies kann als ein "Reservespeicher" fuer eine Unterroutine in Maschinensprache benutzt werden, auf welche das Modul keinen Zugriff haben soll. Weil das bedeutet, dass der Speicherbereich des Moduls verschoben wurde, ist dies gleichbedeutend mit einem Kaltstart des Moduls.

Eingabe: LOM

Bildschirm: DOS Bottom Low High Top
2F58 31DB 31DB 31DB 9C1F

Eingabe: LOM 2F58

>>> Der untere Speicherbereich darf nicht geringer als 31DB sein. Der Speicherbereich 2F58-31D7 wird vom Disk-Puffer benutzt! <<<

Bildschirm: LOMem out of range

7. DC <Zeichen>

Erlaubt dem Anwender, ein Zeichen, wie z.B. "/", zu definieren, und wenn dieses Zeichen definiert ist, koennen die DOS-Befehle mittels dieses Zeichens vom ROM-Modul aus aufgerufen werden.

Eingabe: DC/

Bildschirm: DOS

Eingabe: B

Bildschirm: READY

/Noinit

Das File wird in den Speicher geladen, jedoch nicht initialisiert. Somit kann das Programm nicht nach einem normalen LOAD-Befehl gestartet werden.

Eingabe: R PRETTY.OBJ/N

/Patch

Speicherbereich-Fehler werden nicht beachtet. Pointers werden dorthin geladen, wo es vom File spezifiziert ist. Auch ein Ueberschreiben des DGS ist dann moeglich.

Eingabe: R D2:HERE/M/N

Startadressen: 6000-6090
02E0-02E3
6010 INIT
6020 GO
BRK at 6020

load file </Map></Noinit><Patch>

Das bezeichnete File wird in den Speicher geladen. Es kann mit dem "GO"-Befehl gestartet werden, falls es eine Startadresse hat (bei #2E0).

/Map

Beim Ladevorgang werden die vorhandenen Aufzeichnungen auf dem Bildschirm dargestellt.

/Noinit

Das File wird in den Speicher geladen, jedoch nicht initialisiert.

/Patch

Speicherbereich-Fehler werden nicht beachtet.

Eingabe: L D2:HERE/M/N

Startadressen: 6000 6090
02E2-02E3
6010 INIT
6020 GO

save file <Append> anfang ende
<<init> start>

Der Speicherinhalt wird als File auf Diskette geschrieben. Die Adressen #2E2 und #2E3 werden als Startadressen fuer den GO-Befehl gesetzt. Alle Adressenangaben sind in Hexadezimal zu geben.

Eingabe: /C PRETTY

Wurde - entgegen dem vorliegenden Fall "/" - kein eigenes Zeichen definiert, muss das vom Computer vorgegebene Zeichen "," verwendet werden. Hier wurde jedoch vorher das Zeichen "/" definiert, so dass mit obigem Befehl AUS DEM BASIC HERAUS das File "PRETTY" geladen wird. Nach Definieren eines eigenen Zeichens kann das DOS mit WBOOT neu geschrieben und so das eigene Zeichen erhalten werden. Wird versehentlich das DC-Zeichen ohne naehere Angaben eingegeben, "springen Sie" irgenwohin zwischen BASIC und DOS. Um hier wieder herauszukommen, druecken Sie die Tasten <CTRL 3>, was Sie ins BASIC zurueckfuehrt.

D. Maschinensprache-Monitor

Die folgenden Befehle ermoeglichen dem Anwender die direkte Arbeit im Speicher; ueberpruefen, aendern des Speicherinhalts u.s.w.:

- 1) Run
- 2) load
- 3) Save
- 4) Go
- 5) Proceed
- 6) Examine
- 7) Alter
- 8) Register

1. Run file </Map></Noinit></Patch>

Laedt und startet ein Object-File. Sollte das Programm ueber den Programmbereich hinaus laden, erfolgt die Abfrage, ob ein Kaltstart des Moduls gewuenscht wird.

/Map

Beim Ladevorgang werden die vorhandenen Aufzeichnungen auf dem Bildschirm dargestellt.

Eingabe: R PRETTY.OBJ/M

7 Append

Hiermit koennen Daten an das Object-File angefuegt werden, ohne einen anderen "Object-file-header" zu schreiben.

Eingabe: S D2:HERE 6000 6090 6010 6020

4. Go <hhhh>

Programmstart ab vorgegebener Hexadezimal-Adresse (oder ab letzter geladener oder gespeicherter File-Startadresse). Es ist zu beachten, dass dieser Befehl den "CLOSE"-Befehl beinhaltet, und die Register-Inhalte nicht laedt. Eine Ruecksprung-Adresse verbleibt im Stack, sodass RTS die Kontrolle zurueckgibt an das DOS. Dies kann benutzt werden, um "UDC" neu zu starten.

Eingabe: G 5000

5. Proceed <hhhh>

Fortsetzung des Programmablaufs nach einem "BRK"-Befehl. Die Adresse, ab welcher fortgefahren werden soll, wird durch <hhhh> angegeben. Dieser Befehl veraendert nicht den Registerinhalt, und fuehrt nicht zum schliessen offener Files. Dieser Befehl als solcher kann im Zusammenhang mit dem "BRK"-Befehl angewendet werden, und der veraenderte Befehl - "hhhh" - kann beim "debugging" eines Maschinensprache-Programms nuetzlich sein.

Eingabe: Brk at 0402

6. Examine (<beginn>(<ende>))

Der Speicherinhalt kann in Hexadezimal und in ATASCII gesichtet werden.

Eingabeformat: addr<h1 h2 h3 h4 h5 h6 h7 h8
"12345678, kompatibel mit dem
"ALTER"-Befehl. Der "Examine"-Befehl folgt
vorgegebenermassen nach dem vorhergehenden
"Examine"-Befehl, dem letzten
"ALTER"-Befehl, dem letzten geladenen
Programm oder der Startadresse, falls
vorhanden. Der Befehl "Examine n" fuehrt
zur Ausgabe von 8 Bytes, beginnend bei "n".

Eingabe: E 5000,5010 oder

E 700 oder

E

7. Alter (hhhh)<(hh)(<,>)(hh)(<,>)....

garbadge

Alter (hhhh)<"ascii

oder (eingeschlossener Modus)<hhhh>

(das gleiche wie oben)

BEACHTEN: Aendern des Speicherinhalts in Hexadezimal oder in ATASCII. #60, ein auf der spitze stehendes Quadrat oder eine betonte Hervorhebung (nicht darstellbares Zeichen) auf dem Drucker kann nicht im Speicher abgelegt werden, weil der Examine-Befehl dies benutzt, um darauf hinzuweisen, dass ein Byte nicht im ATASCII-Code dargestellt werden kann.

Eingabe: A 600<0 oder 600<0 oder 600<0"##?3T

B. REgister

REgister (r<h)

Überprüfen und verändern des gespeicherten 6502-Registers. "RE" überprüft alle Register, "RE r<h" verändert die Register. "r" steht fuer A, X, Y, S, C, P wie folgt:

A, X, Y sind die Arbeitsregister

S ist der stack pointer

C ist das flag register

P ist der program counter

h ist die Hexadezimale Anzahl in Byte

WICHTIG: dieser Befehl sollte nur von Anwendern benutzt werden, welche Kenntnisse in Maschinensprache besitzen.

Beispiel: RE PKYARD

RE AK9B, X<ED oder

RE

A. Befehle über die Anlage

Diese Befehle regeln die Funktion der Anlage sowie der angeschlossenen Geräte, wie z.B. Screen Editor, Eingabemaschine, Disk-Drives, Drucker und/oder Interface Modul.

1. RESET
2. -Text
3. -Close
4. -ERR or inn

1. RESET
Dieser Befehl führt zum "Reset" aller Peripheriegeräte, welche das DOS kennt. Er führt ausserdem zum Kaltstart des Moduls. Er kann gegeben werden zur Bezeichnung der Disk-Drives und der Anzahl der File-Puffer. Dies erfolgt durch Eingabe:

Alter 709K# Puffer,Drives
RESET

"Drives" ist die Markierung der gewünschten Drives in Bit und "#Puffer" die maximale Anzahl von I/O (Eingabe/Ausgabe) - Kanälen, welche zur gleichen Zeit zur Diskettenstation geöffnet sein sollen. Definierte Anwender-Befehle werden hierdurch nicht berührt. Der grösste Nutzen ergibt sich beim Wechsel der Puffer.

ACHTUNG: Der "RESET"-Befehl ist nicht mit der Taste <SYSTEM RESET> zu verwechseln.

Eingabe: RESET

2. `ready`: Type Y if ok to coldstart cartridge?

Eingabe: Y

3. `PS.000`: READY

2. Text
Dieser Befehl führt zum neuschreiben der Display-List und veranlasst den Computer, einen leeren Bildschirm darzustellen. Er öffnet erneut den Screen Editor im Modus 0 und ist das Äquivalent zum "SR.0" in BASIC.

Eingabe: T

Eingabe: DOS

3. Close
Alle geöffneten Files werden geschlossen, alle Sound-Kanäle werden abgeschaltet, die "vertical blank vectors" werden zurückgesetzt und die Player Missile Grafik abgeschaltet. Dieser Befehl ist funktionsstärker als der BASIC-Befehl END. BASIC schliesst automatisch sämtliche Files, bevor das DOS aufgerufen wird.

4. Error nn

"nn" steht fuer eine Dezimalzahl. Dieser Befehl stellt die zu "nn" gehoerende Fehlermeldung erklarend in englischem Text dar. Nummern unter 128 werden fuer K-DOS-Fehler benutzt.

Eingabe: ER 144

Beispielformat: ERROR 144, DEVICE ERROR

Eingabe: ER 38

Beispielformat: Incompatible disk drive

BEHUTUNG: Eine Liste der moeglichen Fehlermeldungen befindet sich im Anhang A.

5. DUF - Diskal-Befehle

(Disk Utility Program)

(DUF - Disk Utility Program)

Diese vier Befehle bieten eine besondere Ergaenzung fuer das DUF.

1. UDC
2. IDENT
3. KILL
4. REVIVE

UDC - User Defined Command
(vom Anwender definierter Befehl)

Dieser Befehl gibt dem Anwender die Moeglichkeit, einen Befehl zu definieren, welcher ein Maschinesprachenprogramm laedt und startet.

Die UDCs auf der Masterdiskette sind folgende:

TRANS,	D:TRANS.SYS
HELP,	D:HELP.SYS
UDC,	D:UDC.SYS
CHERROR	D:CHERROR.SYS
DISK Duplizierte,	D:DISK DUP.SYS

Links von ">" stehen die Buchstaben, welche als Abkuerzung zumindest einzugeben sind. Soll ein UDC gelöscht werden, kann die Abkuerzung benutzt werden, jedoch selbstverständlich nicht das ">".

Ein UDC kann verlassen werden mit dem "BRK"-Befehl oder einem "RTS", wenn das Stack erhalten geblieben ist. Mit WBOOT oder INIT sollte eine Kopie des DOS mit dem UDC-Verzeichnis hergestellt werden.

DOS-Befehle ueberlagern UDCs.

Eingabe: UDC

Bildschirm: UDC manager V1.2
List, Add, Delete, INIT, Stop
?

L	Listet das UDC-Verzeichnis
INIT	loescht das UDC-Verzeichnis
DEL cmd	loescht den Befehl (cmd) aus dem Verzeichnis
A cmd <, > file	fuegt den Befehl (cmd) dem Verzeichnis hinzu

2. Mont

Auf dem Bildschirm wird gezeigt, welche Version des DUP in Gebrauch ist, das K-DOS-Titelbild wird dargestellt.

Eingabe: I

Bildschirm: K-DOS By K-Byte

(das ist das gleiche wie nach dem Booten des DOS)

Copyright 1981
KAY ENTERPRISES Co.

3. KILL

KILL löscht die DOS E: (Screen Editor)-
Vektoren und die seriellen
Eingabe/Ausgabe-Stellen. Dies ist
hilfreich, wenn ein Maschinensprache-
Programm versehentlich das DUP gelöscht hat.

Das DOS unterbricht den Screen-Editor und
die Ein-/Ausgabe. Kollidiert ein Programm
mit dem DUP und läuft daher nicht
ordnungsgemäss, verhindert der
"KILL"-Befehl die Benutzung dieses Programms.

Bildschirm: DOS

Eingabe: E28A

Bildschirm: 020AK11 EB E3 27 D1 EA B2 E7
DOS

Eingabe: KILL

Bildschirm: DOS

Eingabe: E28A

Bildschirm: 070AK11 EB 90 EA D1 EA B2 E7

4. REVIVE

REVIVE ist das Gegenteil von KILL. Alle
Fehlermeldungen, welche im DUP benutzt
werden, werden im File EQUATE.ASM
gleichgesetzt. REVIVE ermöglicht dem DOS,
den Screen Editor und serielle Ein-/Ausgabe
neben das Ablesen hinaus entgegenzunehmen.

Liste der Befehle

Vorbereitung der Diskette	INIT n
	FORMAT n
	WBOOT <n>
	DISKdup <scr<<, >dest>> </W></F></P>>
File-Kontrolle	Direct <filespec><,>output>
	Copy input <,>output>
	DELEte filespec </N>
	LOCK filespec
	UNlock filespec
	REName file, filename
	APpend <sourcefile,> destfile
	TRansfer filename </SIRG> <,>filename></SIRG>
Programm-Kontrolle	Back
	WARM
	COLD
	Xit
	UNLOAD
	LOMem
	DC <character>

Maschinensprache-Monitor

Run file </M></N></P>

Load file </M></N></P>

Save file beg end
<<init>start>

Go <hhhh>

Proceed <hhhh>

Examine (<first>(<last>))

Alter (adr)(<)hex....or "ascii

Register (r<h)

Kontrolle ueber
die Anlage

RESET

Text

Close

ERRor nn

UP - Extras

UDC

Ident

KILL

REVIVE

Anhang A

Fehlermeldungen

Fehler-Nr.	Fehler-Name	Ursache und Abhilfe
1	Illegal command	DOS versteht den Befehl nicht. Geben Sie "HELP" fuer Hilfe ein.
2	Not enough memory	Das Maschinensprache-Programm laedt in zu hohen Adressen, oder das UDC-Verzeichnis ist uebergelaufen, oder TRANS hat nicht genug Speicherplatz, um das gesamte File zu laden.
8	Number?	Es wurde eine nicht zulaessige Zahl eingegeben. Perioden und "+" oder "-" Zeichen sind niemals zulaessig im DOS. Die Buchstaben A - F sind nur fuer Hexadezimale Zahlen zulaessig.
12	Go where?	Die erforderliche Eingabe einer Adresse ist unterblieben und es ist keine Startadresse fuer das letzte geladene File vorhanden. Anmerkung: <SYSTEM RESET> zerstoert die Startadresse!
21	Bad load File	Es wurde das Laden eines Files versucht, welches kein zulaessiges "LOAD"-File ist.
32	Syntax?	Es liegen nicht zulaessige oder Extra-Parameter vor.
33	Switch?	Eine unkorrekte Umstellungsbezeichnung wurde verwendet.
35	Filename too long	Ein unzulassiger Filename wurde eingegeben. Vgl. "filename" im Inhaltsverzeichnis.
36	Not a disk file	Es koennen nur Files geloescht werden, die auf der Diskette enthalten sind.
37	No cartridge	"B" - und "X" - Befehle koennen nur dann ausgefuehrt werden, wenn sich im ROM-Schacht ein Modul befindet, zu welchem zurueckgekehrt werden kann.
38	Incompatible disk drives	Die benutzten Diskettenstationen sind nicht kompatibel.

Fehler-Nr.	Fehler-Name	Ursache und Abhilfe
39	Need 1 thru 8 for disk #	Die Diskettenstationen sind mit 1-8 zu bezeichnen.
40	Illegal User Def'd command	Es wurde das Loeschen eines UDC-Befehls versucht, welcher sich nicht im Verzeichnis befindet.
41	Not Basic- use Back	Basic-Programme koennen nicht ueber DOS geladen oder gestartet werden. DOS kennt nur das in ihm enthaltene Format von Maschinensprache-Files und solchen, die ueber DOS oder ASM/ED - Modul abgespeichert wurden.
42	LOMEM out of range	Es wurde eine nicht zulaessige Adresse fuer den "LOMem"-Befehl gegeben.
43	Can't overlay DOS	Es soll ein File in den Speicher- platz geladen werden, wo sich das DOS befindet.
44	Can't proceed	Die Adresse, ab welcher nach werden soll, ist nicht vorhanden.
128	*Break	Die <BREAK>-Taste wurde betaetigt. Der Programmablauf wird unterbrochen.
130	No such device	Das Peripheriegeraet, welches angesprochen werden soll, existiert nicht.
136	End of file	In dem bearbeiteten File existieren keine Daten mehr.
138	Device timeout	Es wurde eine unkorrekte Geraetenummer angegeben oder ein falsches Geraet angesprochen. Pruefen Sie alle Anschluesse. Ueberpruefen Sie auch noch einmal den gegebenen Befehl und wiederholen Sie ihn ggf.
139	Device NAK	Keine Reaktion aufgrund fehler- hafter Parameter. Das angesprochene Geraet hat evtl. vom Computer fehlerhafte Daten erhalten.
141	Cursor out of range	Fuer den gewaehlten Grafik- Modus befindet sich der Cursor nicht innerhalb des zulaessigen Bereichs.

Fehler Nr.	Fehler-Name	Ursache und Abhilfe
144	Device Error	Das angesprochene Geraet kann einen zulaessigen Befehl nicht ausfuehren. Evtl. ist die Diskette schreibgeschuetzt.
146	Funct. not implemented	Diese Funktion ist im 'handler' nicht vorgesehen. Es wurde der Versuch unternommen, nicht zur Verfuegung stehende Befehle zu geben oder nicht kompatible Peripheriegeraete anzusprechen.
154	Concurrent mode I/O not active	Vgl. hierzu das "850-Handler-Manual".
160	Bad drive num	Es wurde eine nicht angeschlossene Diskettenstation angesprochen.
162	Disk full	Auf der Diskette befinden sich keine freien Sektoren mehr. Benutzen Sie eine andere Diskette.
164	File over-written	Der gelesene Sektor enthaelt keine zu diesem File gehoerenden Daten.
165	Bad file name	Die verwendete File-Bezeichnung enthaelt nicht zulaessige Zeichen.
167	File locked	Ein gesichertes File kann nicht geloescht oder ueberschrieben werden, und es kann an dieses auch kein anderes File oder Daten "angehaengt" werden.
169	Directory full	Der fuer das Directory reservierte Platz ist ausgeschoeppt.
170	File not found	Das aufgerufene File existiert nicht auf dieser Diskette.
172	Incompatible DOS format	Das File wurde nicht ueber DOS 2.05 oder K-DOS erzeugt.
173	Can't format disk	Die Diskette ist unbrauchbar, da sich einer oder mehrere Sektoren nicht formatieren lassen.

Anhang C

Wörterbuch der Begriffe

Addr	Abkürzung fuer "Speicheradresse".
Arguments	Variablen in Filenamen und in <> nach dem Filenamen; alles, was nach dem Befehl steht.
ASCII	The American Standard Code for Information Interchange.
Byte	8 Bits; Basic-Masseinheit.
Boot	Das Programm wird mit dem Einschalten des Computers automatisch in den Speicher geladen und gestartet.
Buffer	Bereich, in welchem Daten voruebergehend abgelegt werden, bevor sie weiter verwendet werden. Das K-DOS beinhaltet einen 256-Byte-Buffer (Puffer) fuer bestimmte Befehle (Copy, DElete, Direct).
FD	AS82-Programm Status-Byte; das "Flags Register".
I/O	Das zentrale Eingabe/Ausgabe - Untersystem (Central input/output subsystem).
D:	Bezeichnung des Disk-Drives.
Defaults	(Vorgaben) - Automatischer Ausgleich fuer nicht vollstaendige Angaben; K-DOS hat eine Anzahl von "Defaults" fuer haeufig benutzte Eingaben: nichts ---> D: n ---> Dn: Filename ---> D:Filename [Filename ---> D:Filename d:Filename ---> d:Filename dn:Filename ---> dn:Filename Wo "n" eine einzelne Zahl ist, ist "d" ein einzelner Buchstabe fuer einen Seriaetenamen, und "Filename" steht fuer einen Namen bis zu 8 alphanumerischen Zeichen und eine Erweiterung (Extender) bis zu 3 Zeichen (z.B. durch einen Punkt getrennt).

Bestimmte Befehle, Direct, Load, Run, DElete (ausser mit "/NOQUERY") haben Vorgaben, bestehend aus "Wildcards".

ds".

D:	---	D: *.*	alle Files
D:	---	D:Name.*	alle Files mit Erweiterung (Extender)
D:Name.	---	D:Name.	nur Erweiterungen
D:Name.Ext	---	D:Name.Ext	nur dieses File

- Dest Abkuerzung von "destination", z.B. "Destination file (Ziel-File)", das zu schreibende File waehrend eines Datentransfers (z.B. Kopiervorgang).
- DUP Disk Utility Program.
- E: Ansprechbefehl fuer den 'Screen Editor'.
- Filename Name eines Files. Alphanumerische Zeichen, um ein File zu benennen; bis zu 8 Zeichen plus zusaetzlich 3 Zeichen fuer die Erweiterung.
- Filespec File-Spezifikation. 1 Zeichen fuer das anzusprechende Gerat, ggf. eine Geratenummer, ein Doppelpunkt, ein Filename bis zu 8 Zeichen und ggf. - getrennt durch einen Punkt - zusaetzlich 3 Zeichen fuer die Erweiterung.
- Flags Das 6502-Status-Register register
- FMS File Management System
- K Kilobyte; 1024 Bytes Speicherumfang
- Lower case 'kleine' Buchstaben; K-DOS akzeptiert auch die Eingabe 'kleiner' Buchstaben.
- N: Ein nicht vorhandenes Gerat (dummy device), welches von K-DOS angesprochen werden kann; Alles, was hier hinein geschrieben wird, verschwindet ohne Wirkung; es fuehrt zum File Ende.

R	steht fuer eine einzelne Zahl.
RD	steht fuer eine Dezimalzahl, z.B. ERror nn.
NOTE	Findet die Eingabe/Ausgabe-Punkte von Geräeten.
Object code	Ein anderer Name fuer Maschinensprache.
Object file	Ein File, welches den 'Object code' beinhaltet; DOS kann Files laden und generieren welche mit DOS 2.05 und ASSEMBLER-Modul arbeiten.
Parameter	Variablen, welche zu Befehlen zusaetzliche Informationen geben, z.B. Filenamen, Disk-Drive-Nummern u.s.w.
PC	6502-"Program counter". Bezeichnet die Stelle im Speicheran welcher der Computer das Programm abarbeitet.
POINT	Kennzeichnet bestimmbare Stellen innerhalb von Ein-/Ausgabe-Geräeten.
RJS	ASSEMBLER-Befehl; Ruecksprung von Unterroutine.
SR	Abkuerzung von "source (Quelle)", z.B. 'Source file' = 'Quellenfile'.
SIO	Serial input/output (serielle Eingabe/Ausgabe).
SIRG	"Short interrecord gaps". Findet Verwendung bei Kopieren eines Files von Diskette auf Cassette (vgl. Transfer).
Stack pointer	(6502 stack pointer). Deutet auf den Beginn einer in sich geschlossenen Information.
Syntax	Die Regeln von Kommata, Zeichen und Kennzeichnungen. Notwendig, um die Ordnungsgemaesse Ausfuehrung eines Programms zu gewaehrleisten.
Text files	Einfachen von Informationen, z.B. Listen, Resultate, Kopien, welche ein Programm darstellen koennen oder auch nicht.
Tokenizing	Vorgang, waehrend welchem Basic-Befehle in La-Symbole konvertiert werden; z.B.: "Run" wird reduziert auf 1 Symbol oder Byte.

- LDC User Defined Commands (vom Anwender definierte Befehle). Ermöglicht dem Anwender, Befehle zu definieren, welche ein Maschinensprache - Programm vom BASIC oder ASSEMBLER aus startet.
- Upper case Grossgeschriebene Buchstaben in einem Befehl bezeichnen den Teil des Befehls, der notwendig ist, um diesen zur Ausführung zu bringen. Z.B. in dem Befehl "Copy" ist das "C" der einzige Buchstabe, der eingegeben werden muss. Bei DELETE sind es die Buchstaben "DEL".
- VTOC Volume Table Of Contents: ein "Bit"-Verzeichnis der verfügbaren Sektoren.
- Wildcards "?" und "*" sind sog. "Wildcards".
- "?" ersetzt einen einzelnen Buchstaben.
- "*" ersetzt mehrere Zeichen (z.B. den Rest eines Filenamens).