

3040 - 4040

Inhaltsverzeichnis

<u>Inhalt</u>	<u>Seite</u>
1. Technische Daten	1
2. Floppy-Befehle DOS1/DOS2	2
3. DOS-Beschreibung	3 - 5
4. Diskettenorganisation	6 - 10
5. Fehlermeldung und Fehlerbehandlung	11 - 17
6. Blockschaltbild	18
7. Spannungsversorgung	19
8. Logic-Beschreibung Digitalboard	20 - 22
9. Steckerbelegung	23
10. Logicbeschreibung Analogboard	24
11. ROM-Bestückung Floppy-Disk	25

1. Technische Daten

3040 - 4040

Speicherkapazität

Gesamtkapazität	174848 Bytes pro Diskette (170180)
Serielle Dateien	168656 Bytes pro Diskette
Relative Dateien	167132 Bytes pro Diskette
	65535 Sätze pro Datei
Directory-Einträge	144 pro Diskette (152)
Sektoren pro Spur	17 bis 21
Bytes pro Sektor	256 (nutzbar: 254 bei PRG-, SEQ-, REL-Dateien bzw. 255 bei Direkt-Zugriff) (254)
Spuren	35
Blöcke	683 (davon 664 für den Benutzer verfügbar)

IC's

Controller	650.4	Microprozessor
	6530	I/O, RAM, ROM
	6522	I/O, Timer
Interface	6502	Microprozessor
	6532 (2)	I/O, RAM, Timer
	6332 (3)	ROM (2)
Gemeinsam	6114 (8)	4*1K RAM

Physikalische Eigenschaften

Material	Stahlgehäuse
Abmessungen	18 (H) * 38 (B) * 40 (T)
Gewicht	12 kg

Elektrische Eigenschaften

Spannung	220 V
Frequenz	50 or 60 Hz
Leistungsaufnahme	50 W

Speichermedium

Mini - Disketten	5 1/4 " einseitig, single density softsektoriert (siehe III.5.5)
------------------	--

Anmerkung: Die Werte in Klammern sind für 3040

2. Floppy-Befehle DOS1/DOS2

ALL BASIC VERSIONS	UNIVERSAL DOS SUPPORT	BASIC 4.0
SAVE "dr:fn",8	SAVE"dr:fn",8	DSAVE,"fn",Ddr (drive defaults to 0)
LOAD"dr:fn",8	/dr:fn (searches both drives)	DLOAD"fn",Ddr (drive defaults to 0)
LOAD"*",8 RUN	↑ dr:fn	DLOAD"fn",Ddr RUN
LOAD"dr:fn",8 RUN	↑ *	shifted RUN/STOP
LOAD"\$0",8 LIST destroys memory	>\$0 preserves memory	DIRECTORY or DI<shifted preserves memory
10 OPEN1,8,15 20 INPUT#1,A,B\$,C,D 30 PRINT A,B\$,C,D	> return	?DS\$ or ?DS (DS is number of error only)

NOTE: Assume that OPEN1,8,15 has already been typed for all of the PRINT commands in the following formats. Commands may be spelled out or abbreviated by the first letter as illustrated.

PRINT#1,"Ix"	INITIALIZE >Ix	PRINT#1,"Ix"
PRINT#1,"Vdr"	VALIDATE >Vdr	COLLECT Ddr
PRINT#1,"Sdr:fn"	SCRATCH >dr:fn	SCRATCH"fn",Ddr
PRINT#1,"Dddr=sdr"	DUPLICATE >Dddr=sdr	BACKUP Dsdr TO Dddr
PRINT#1,"Cddr=sdr"	COPY (all disk) >Cddr=sdr	COPY Dsdr TO Dddr
PRINT#1,"Cdr:dfn= dr:sfn"	COPY (single file) >Cdr:dfn=dr:sfn	COPY Ddr,"sfn" TO Ddr,"dfn"
PRINT#1,"Cdr:dfn= dr:sfn1,dr:sfn2,...	CONCANTENATE FILES >Cdr:dfn=dr:sfn1, dr:sfn2,...	CONCAT Ddr,"sfn" TO Ddr,"dfn"
PRINT#1,"Rdr:dfn=sfn"	RENAME FILES >Rdr:dfn=sfn"	RENAME Ddr,"sfn" TO "dfn"
PRINT#1,"Ndr:dname,xx"	FORMAT A DISKETTE >Ndr:dname,xx	HEADER "dname",Ddr,Ixx

3. DOS-Beschreibung

Floppy Organisation

Dieser Abschnitt soll die interne Organisation des Floppy-Disk-Laufwerks beschreiben. Dabei wird nicht auf den physikalischen Aufbau eingegangen, sondern dem Programmierer sollen hier nur die logischen Verbindungen im Floppy erklärt werden.

Dieses Wissen ist für die Behandlung von SEQ, REL usw. Dateien nicht notwendig, da man mit den komfortablen Floppy-Kommandos immer direkt Dateien oder Programme auf der Diskette anspricht und sich somit nicht darum kümmern mußte, wie das Floppy diese Kommandos ausführt und auf welchen Wegen Daten von oder zur Diskette transportiert werden.

Will man aber die tieferen Möglichkeiten des Floppy-Disk-Betriebssystems (Abkürzung DOS = Disk Operating System) nutzen, dann sollte man einen Überblick über die interne Organisation des DOS haben.

DOS mit Pipeline-Struktur

Das DOS steuert die Handhabung des gesamten Informationsaustausches zwischen den Disketten und dem IEEE-488-Bus. Über diesen Bus wird die Verbindung zum Rechner organisiert. Die Handhabung des IEEE-488-Busses sollte dem Benutzer schon bekannt sein.

Wenn im folgenden vom "Floppy" die Rede ist, ist damit nicht die Diskette gemeint, sondern die Verbindung des IEEE-488-Bus, da hier die Stelle ist, an die der Rechner direkt angeschlossen ist.

Dazu steht dem DOS ein interner Schreib-/Lesespeicher zur Verfügung. Dieser Speicher wird als Puffer benutzt, um die Information, die von oder zu der Diskette übertragen wird, zwischenspeichern. Dies ist sinnvoll, weil die Übertragung Rechner-Floppy wesentlich schneller geht als die Übertragung Floppy-Diskette.

Eine solche Struktur wird als "Pipeline-Struktur" bezeichnet. Der Pufferspeicher ist die "Pipeline" zwischen Rechner und Diskette und speichert die schnell anfallende Information vom Rechner, um sie dann an das langsamere Medium Diskette weiterzugeben.

Die Übertragung von Disk zum Rechner wird durch diese Pipeline-Struktur auch unterstützt. Wenn der Rechner die anfallende Information nicht so schnell verarbeiten kann wie sie von der Diskette gelesen wird, wird sie im Pufferspeicher zwischengespeichert und kann bei Bedarf schnell an den Rechner weitergegeben werden. Dieser muß also nicht erst warten, bis die weitere Information von der Diskette gelesen wird.

Das Floppy liest also "auf Verdacht" von der Diskette, denn es wäre natürlich auch der Fall möglich, daß der Rechner die weitere Information nicht mehr benötigt.

Pufferspeicher

Der Pufferspeicher hat eine Kapazität von 4 K-Bytes. Diese 4 K sind in 16 Bereiche zu je 256 Bytes aufgeteilt. Von diesen Bereichen werden 15 als Puffer verwendet, die mit Puffernummern von 0 bis 14 bezeichnet werden. 10 Puffer werden für die Datenübertragung verwendet, die restlichen benötigt das DOS selbst.

Kanalorganisation

Das Dateisystem ist durch sogenannte **Kanäle** organisiert. Diese Kanäle sind keine physikalischen, sondern rein logische Verbindungen im Floppy.

Man kann sie mit der Dateistruktur beim Rechner vergleichen. Wird dort eine Datei mit Logischer Adresse, Primäradresse und Sekundäradresse durch OPEN geöffnet, dann werden in einer Liste im Betriebssystem-Speicher die verschiedenen Adressen einander zugeordnet. Will der Programmierer nachher auf diese Datei zugreifen, muß er nur die logische Adresse ansprechen. Das Betriebssystem kann aus der Liste selbständig alle anderen Adressen entnehmen und damit den Datenaustausch organisieren.

Beim Floppy werden durch einen OPEN-Befehl einem Kanal ein, zwei oder drei Pufferbereiche zugeordnet. Bei OPEN auf eine Datei vom Typ SEQ oder PRG interessiert den Benutzer nicht, welcher Kanal welchem Puffer zugeordnet worden ist.

Gesamtübersicht über das DOS mit Verbindungen

Generell kann man also sagen, daß für den Programmierer alle Verbindungen im Floppy über Kanäle angesprochen werden. Der Kanal 15 als Kommandokanal hat einen Sonderstatus. Über ihn können keine Daten von oder zu der Disk übertragen werden.

Es gibt also im Floppy folgende logische Verbindungen, die auch im Bild eingetragen sind:

Für den Benutzer direkt zugreifbar:

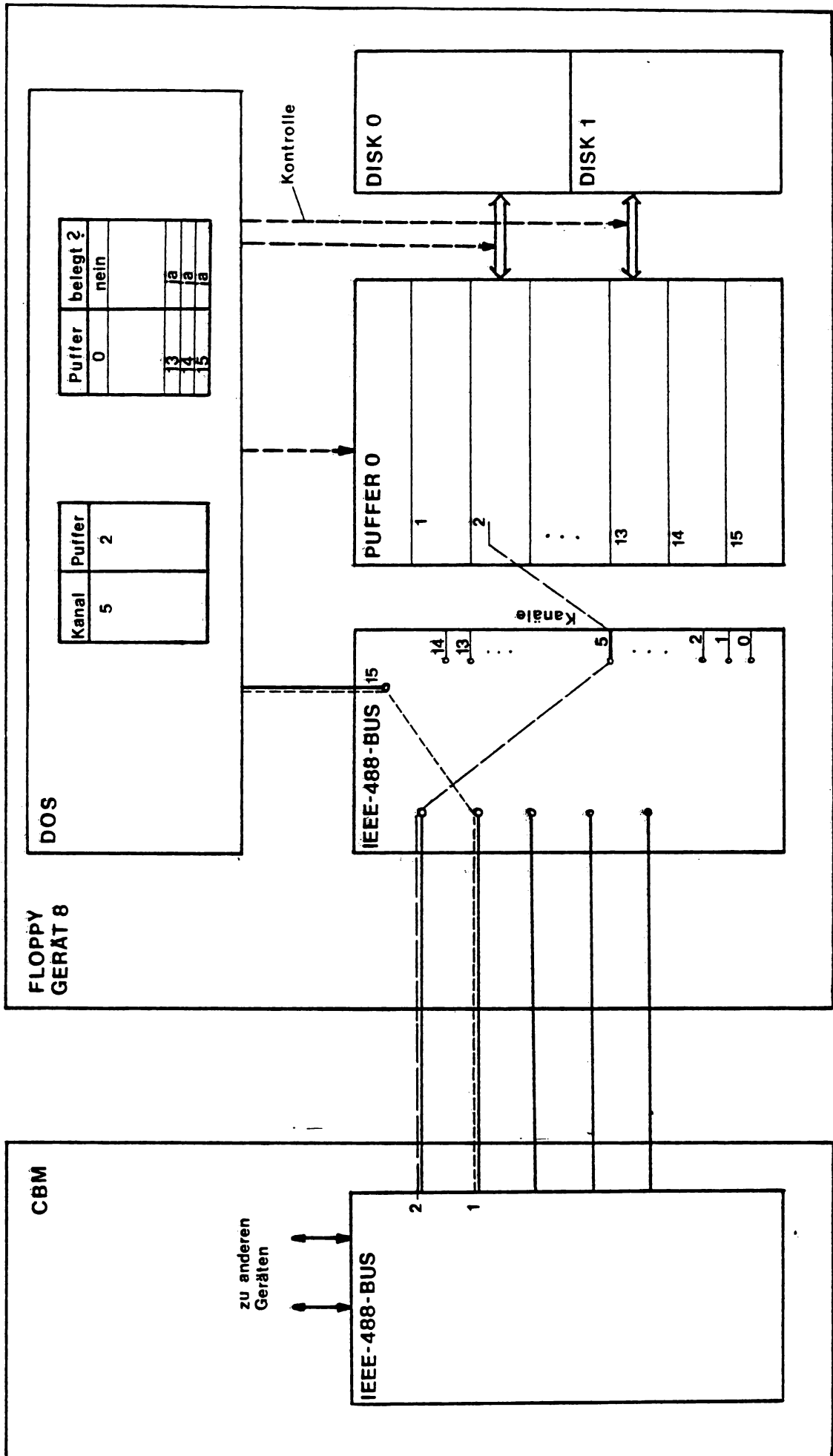
- a) Rechner - Floppy IEEE-488-Bus (durch Logische Adresse)
- b) Floppy IEEE-488-Bus - Pufferspeicher (durch Sekundäradresse = Kanal)
- c) Floppy IEEE-488-Bus - DOS (durch Kommandokanal)

Für den Benutzer nicht direkt zugreifbar:

- d) Pufferspeicher - Diskette

Diese Verbindung ist indirekt zu beeinflussen, indem man ein Kommando schickt, (über Kanal 15) in dem das DOS angewiesen wird, Information in einer gewünschten Richtung auf dieser Verbindung zu übertragen.

Schema der logischen Verbindungen im Floppy



Beispiel: OPEN 2,8,5,"#2"
OPEN 1,8,15

4. Diskettenorganisation

Einteilung in Spur und Sektor

Damit man auf der Diskette auf bestimmte Teile der abgespeicherten Information zugreifen kann, ist diese in Spuren und Sektoren eingeteilt. Diese Einteilung wird bei der Formatierung einer Diskette vorgenommen.

Die Spuren sind konzentrische Kreise um den Mittelpunkt der Diskette. Diese sind weiter aufgeteilt in Sektoren.

Im Gegensatz zu den sonst üblichen Diskettenformaten ist die Anzahl der Sektoren pro Spur nicht auf der ganzen Diskette gleich. Da auf den äußeren Spuren der Umfang größer ist als weiter innen, hat hier eine Spur auch mehr Sektoren. Diese Art der Disketteneinteilung hat den großen Vorteil, daß ohne Erhöhung der maximalen Bitdichte wesentlich mehr Information auf die Diskette gepackt werden kann.

Es gibt insgesamt 35 Spuren, die mit Nummern 1 bis 35 durchnummeriert sind. Die Nummerierung der Sektoren beginnt in jeder Spur bei 0.

Beispiel: Bei Spur 1 gibt es die 21 Sektoren 0 bis 20.

Spur		Sektoranzahl		
		pro Spur	insgesamt	
1 bis	17	21	357	
18 bis	24	19	133	
25 bis	30	18	108	
31 bis	35	17	85	
			<u>683</u>	
			- 18	für Directory
			- 1	für BAM
			<u>664</u>	Blöcke für freie Benutzung

Blockbelegung

Die kleinste auf der Diskette adressierbare Einheit, die mit Spur und Sektor angesprochen wird, nennt man **Block**. Spur/Sektor wird im folgenden als Blockadresse bezeichnet.

Ein Block enthält 256 Bytes an Information. Der Rechner kann 255 Bytes von der Diskette lesen oder auf die Diskette schreiben.

BAM = Block-Verfügbarkeits-Tabelle

Die BAM (**B**lock **A**vailability **M**ap) ist eine Liste, in der für jeden Block angegeben ist, ob dieser belegt ist oder nicht.

Wird zum Beispiel eine Datei auf den Blöcken Spur 17 Sektor 0 und Spur 17 Sektor 8 abgelegt, dann werden diese Blöcke in der BAM als belegt gekennzeichnet. Dadurch wird gewährleistet, daß diese Blöcke beim Abspeichern von anderen Dateien nicht mehr benutzt werden. Beim Löschen von Dateien werden die von dieser Datei belegten Blöcke wieder freigegeben.

Die BAM steht auf der Diskette auf Spur 18 Sektor 0. Dieser Block steht nach dem Initialisieren immer im Pufferspeicher.

Inhaltsverzeichnis

Beim Abspeichern einer seriellen Datei auf eine Diskette wird der Name dieser Datei im Inhaltsverzeichnis vermerkt.

Für das Inhaltsverzeichnis ist auf der Diskette die Spur 18 reserviert. Auf diese Spur werden deshalb keine Dateien abgespeichert.

In einem Block können bis zu acht Dateinamen stehen. Die Reihenfolge der Namen hängt von der Reihenfolge der Abspeicherungen ab. Beim Löschen einer Datei wird der Name im Inhaltsverzeichnis als gelöscht gekennzeichnet. Dadurch wird er beim Lesen des Inhaltsverzeichnisses nicht mehr berücksichtigt.

Beim Abspeichern einer Datei auf die Diskette wird das Inhaltsverzeichnis von vorne nach dem ersten freien Platz durchsucht. Dieser wird dann für den Namen der abzuspeichernden Datei verwendet.

In derselben Reihenfolge, wie die gültigen Namen im Inhaltsverzeichnis abgespeichert sind, werden sie auch gelesen.

Die Reihenfolge, der Dateinamen ist wichtig, wenn man nach dem Einschalten des Systems ein bestimmtes Programm mit SHIFT/RUN oder 'LOAD"',8' laden möchte. Wie schon beschrieben muß dieses Programm als erstes im Inhaltsverzeichnis stehen.

Dies erreicht man auf zwei Arten:

Wenn noch keine Datei auf der Diskette steht, wird dieses Programm als erstes abgespeichert.

Wenn die Diskette schon beschrieben ist, muß die erste Datei gelöscht und dann das benötigte Programm abgespeichert werden. Dessen Name wird in dem freien Platz abgespeichert.

Verwaltung der seriellen Dateien

Serielle Dateien werden auf der Diskette so abgespeichert, daß das DOS von Spur 39 nach unten und oben die nächsten freien Blöcke sucht. Normalerweise werden Blöcke verwendet, die nicht benachbart sind (Zeitoptimierung). Allgemein können serielle Dateien auf beliebigen Blöcken in beliebiger Reihenfolge stehen.

Damit das DOS beim Zugriff auf eine serielle Datei diese wieder richtig finden kann, werden die Blöcke durch Verweise verkettet.

Der Zugriff auf eine serielle Datei geht in folgender Reihenfolge:

Beim Namen der Datei im Inhaltsverzeichnis stehen Spur und Sektor des ersten Blocks dieser Datei.

Auf jedem Block der Datei steht ein Verweis auf den nächsten Block in den ersten zwei Bytes, und zwar die Spur in Byte 0, der Sektor in Byte 1.

Der letzte Block einer Datei ist dadurch gekennzeichnet, daß in Byte 0 der Wert 255 steht. In Byte 1 steht hier ein Zeiger auf das Ende der Datei im Block.

Tabellen für Direktzugriff

Standard-Sprung-Tabelle

Kommando	Adresse	Kommentar
U1 oder UA		BLOCK-READ-Ersatz
U2 oder UB		BLOCK-WRITE-Ersatz
U3 oder UC	\$1300	frei verfügbare Anwender-Adressen
U4 oder UD	\$1303	"
U5 oder UE	\$1306	"
U6 oder UF	\$1309	"
U7 oder UG	\$130C	"
U8 oder UH	\$130F	"
U9 oder UI	\$10F0	NMI Vektor
U: oder UJ		POWER UP Vektor

Die Adressen U3 bis U8 liegen im Puffer-Bereich des RAM und können deshalb vom Programmierer mit eigenen Werten belegt werden.

Verteilung der Blöcke über die Spuren

Spur-Nummer	Block-Bereich
1 bis 17	0 bis 20
18 bis 24	0 bis 18
25 bis 30	0 bis 17
31 bis 35	0 bis 16

BAM-Format

Adresse: Spur 18 / Sektor 0

Byte	Inhalt	Definition
0, 1	18, 1	Spur und Sektor des ersten Directory-Blocks
2	65	'A' = 4040-Format
3	0	nicht verwendet
4- 7		Bitweises Verzeichnis der verfügbaren Blöcke auf Spur 1
8-143	*	BAM der Spuren 2-35, jeweils 4 Bytes pro Spur

Struktur der BAM für eine Spur

Byte	Definition
0	Anzahl der verfügbaren Sektoren dieser Spur
1	Bits der Sektoren 0- 7
2	Bits der Sektoren 8-15
3	Bits der Sektoren 16-21

Bit gesetzt = Sektor ist verfügbar
 Bit gelöscht = Sektor ist belegt

Vorspann des Inhaltsverzeichnisses (Directory Header)**Adresse:** Spur 18 / Sektor 0

Byte	Inhalt	Definition
144-161		Disketten-Name (mit SHIFT-SPACEs aufgefüllt)
162,163		Disk-ID
164	160	SHIFT-SPACE
165,166	50,65	'2A' = DOS-Version und Format-Typ
171-255	0	nicht verwendet

In den Stellen 180 - 191 können auf manchen Disketten ASCII-Zeichen erscheinen.

Directory-Format

Das Directory belegt die Sektoren 1 bis 18 auf der Spur 18. Der Vorspann steht im Block auf Spur 18 Sektor 0.

Byte	Definition
0, 1	Spur und Sektor des nächsten Directory-Blocks
2- 31	Datei-Eintrag 1
34- 65	Datei-Eintrag 2
66- 97	Datei-Eintrag 3
98-129	Datei-Eintrag 4
130-161	Datei-Eintrag 5
162-193	Datei-Eintrag 6
194-225	Datei-Eintrag 7
226-255	Datei-Eintrag 8

Format eines Datei-Eintrags

Byte	Definition
0	0 = gelöscht 1 = SEQ 2 = PRG 3 = USR 4 = REL Wenn Bit 7 gesetzt ist, wurde die Datei richtig geschlossen. Ist Bit 7 gelöscht, trägt die Datei einen Stern im Inhaltsverzeichnis und ist nicht mehr lesbar.
1, 2	Spur und Sektor des ersten Daten-Blocks
3-18	Datei-Name mit SHIFT-SPACEs aufgefüllt
19,20	Nur bei REL: Spur und Sektor des ersten Verwaltungs-Blocks
21	Nur bei REL: Record-Länge
22-25	nicht verwendet
26,27	Spur und Sektor der ersetzenden Datei, wenn 'ERSETZEN' arbeitet
28,29	Anzahl der Blöcke der Datei (LOW/HIGH)

Struktur der Dateitypen SEQ, PRG und USR**Byte Definition**

- 0, 1 Spur und Sektor des nächsten Blocks
oder Byte 1 = 0 und Byte 2 = letztes gültiges Byte dieses Blocks
beim letzten Block der Datei
- 2-256 254 Daten-Bytes

Daten-Block der REL-Datei**Byte Definition**

- 0, 1 Spur und Sektor des nächsten Daten-Blocks
- 2-256 254 Daten-Bytes
Leere Records enthalten im ersten Byte \$FF und in allen folgenden 0
Teilweise beschriebene Records sind mit 0 aufgefüllt

Verwaltungs-Block der REL-Datei**Byte Definition**

- 0, 1 Spur und Sektor des nächsten Verwaltungsblocks
- 2 Nummer des Verwaltungs-Blocks (0-5)
- 3 Record-Länge
- 4, 5 Spur und Sektor des 1. Verwaltungsblocks (Nummer 0)
- 6, 7 Spur und Sektor des 2. Verwaltungsblocks (Nummer 1)
- 8, 9 Spur und Sektor des 3. Verwaltungsblocks (Nummer 2)
- 10, 11 Spur und Sektor des 4. Verwaltungsblocks (Nummer 3)
- 12, 13 Spur und Sektor des 5. Verwaltungsblocks (Nummer 4)
- 14, 16 Spur und Sektor des 6. Verwaltungsblocks (Nummer 5)
- 16-256 Spur- und Sektor-Zeiger für 120 Daten-Blöcke

5. Fehlermeldung und Fehlerbehandlung**Fehlertabelle**

!	DS\$!		
DS !				
F !	F\$!	F1	F2
Keine Fehler				
00	OK	0	0	Alles in Ordnung
01	FILES SCRATCHED	A	0	F1 enthält Anzahl gelöschter Dateien
Lesefehler				
20	READ ERROR	T	S	Kein Block "header"
21	READ ERROR	T	S	Kein Synchronisationszeichen
22	READ ERROR	T	S	Datenblock nicht vorhanden
23	READ ERROR	T	S	Prüfsummenfehler im Datenblock
24	READ ERROR	T	S	Byte wurde falsch dekodiert
27	READ ERROR	T	S	Prüfsummenfehler im "header"
71	DIR ERROR	0	0	Directory Fehler
Schreibfehler				
25	WRITE ERROR	T	S	Schreib- / Prüffehler
28	WRITE ERROR	T	S	Datenblock zu lang
Syntax Fehler				
30	SYNTAX ERROR	0	0	Allgemeine Syntax
31	SYNTAX ERROR	0	0	Ungültiges Kommando
32	SYNTAX ERROR	0	0	Zu langes Kommando
33	SYNTAX ERROR	0	0	Ungültiger Dateiname
34	SYNTAX ERROR	0	0	Dateiname fehlt
39	SYNTAX ERROR	0	0	Ungültiges Kommando
Bedienungsfehler				
26	WRITE PROTECT ON	T	S	Schreibschutz "geklebt"
29	DISK ID MISMATCH	T	S	Disk ID stimmt nicht überein
50	RECORD NOT PRESENT	T	S	Record ist nicht vorhanden
51	OVERFLOW IN RECORD	T	S	Record ist zu kurz für die Daten
52	FILE TOO LARGE	T	S	Recordnummer ist zu hoch
60	WRITE FILE OPEN	0	0	Bereits Datei zum Schreiben geöffnet
61	FILE NOT OPEN	0	0	Datei ist nicht "offen"
62	FILE NOT FOUND	0	0	Datei wurde nicht gefunden
63	FILE EXISTS	0	0	Datei existiert bereits
64	FILE TYPE MISMATCH	0	0	Dateitypendurcheinander
65	NO BLOCK	T	S	Kein Block frei
66	ILLEGAL TRACK OR SECTOR	T	S	Unerlaubte Spur oder Sektor
67	ILLEGAL SYSTEM T OR S	T	S	Unerlaubte Spur oder Sektor
70	NO CHANNEL	0	0	Kein Kanal frei
71	DIR ERROR	T	S	BAM ist defekt
72	DISK FULL	T	S	Diskette voll
73	CBM DOS V2	0	0	Meldung der DOS-Version

Allgemeine Hinweise zur Fehlermeldung

Bei der Ausführung von Floppy - Kommandos kann es zu Fehlern kommen. Dabei gibt es Fehler, die zwangsläufig und vorhersehbar eintreten, wie etwa WRITE PROTECT ERROR. Daneben können aber auch Fehler auftreten, die zufälliger Natur sind, wie etwa der Fehler 25.

Die Fehler- bzw. Zustandsmeldungen des Floppy können über den Fehlerkanal abgefragt werden.

Wenn ein Fehler auftritt, wird dies durch die Fehlerlampe zwischen den beiden Laufwerken angezeigt. Dieser Hinweis dürfte allerdings nur beim Arbeiten im Direktmodus interessant sein. Sobald Floppy - Kommandos von Programmen aus gegeben werden, empfehlen wir, grundsätzlich nach dem Kommando die Fehlermeldung einzulesen, um den Benutzer möglichst schnell auf Fehler hinweisen zu können.

Da die meisten Fehlermeldungen durch ein neues Kommando gelöscht werden, muß nach jedem einzelnen Kommando die Fehlermeldung gelesen und ausgewertet werden. Dabei sind zwei Möglichkeiten zu unterscheiden:

- (1) Die Fehlermeldung wird unmittelbar **nach** dem jeweiligen Kommando gelesen.
- (2) Die Fehlermeldung wird unmittelbar **vor** dem nächsten Kommando gelesen.

Die erste Möglichkeit ist im Programm einfacher zu realisieren, hat aber den Nachteil, daß der Anwender warten muß, bis das Floppy das betreffende Kommando ausgeführt hat. INPUT\$ wartet nämlich, bis die betreffende Operation zu Ende ist.

Die zweite Möglichkeit bietet den Vorteil, daß der Dialog weiterlaufen kann, während das Floppy arbeitet. Da das Programm aber dann 'rückwirkend' auf Fehler reagieren muß, wird die Struktur des Dialogteils erheblich anspruchsvoller als im ersten Fall.

Die Bestandteile der Fehlermeldung

Die Fehlermeldung besteht aus vier Teilen. Zuerst wird die **Fehlernummer (F)** gemeldet, dann der zugehörige **englische Klartext (F\$)**. Die beiden folgenden Zahlen geben **Spur (F1)** und **Sektor (F2)** an, wo der Fehler aufgetreten ist, soweit diese Angabe sinnvoll ist.

Lesen der Fehlermeldung durch DS und DS\$ (ab BASIC-4)

Bei BASIC-4 stehen die beiden Variablen-Funktionen DS und DS\$ zur Verfügung. Jeder Aufruf einer der beiden Variablen bewirkt, daß die Floppy-Fehlermeldung gelesen wird, sofern seit dem letzten Aufruf eine Floppy-Operation durchgeführt wurde.

DS enthält die Fehlernummer und DS\$ die gesamte Fehlermeldung (einschließlich der Fehlernummer).

DS und DS\$ holen automatisch die Meldung des zuletzt angesprochenen Gerätes ein. Sie können damit ohne weiteres mehrere Floppy-Geräte mit dieser einen Funktion bearbeiten.

Lesen der Fehlermeldung mit INPUT\$ (nicht nötig ab BASIC-4)

Bei BASIC-3 kann die Fehlermeldung nicht im Direktmodus, sondern nur im BASIC Programm durch INPUT\$ gelesen werden. INPUT\$ setzt voraus, daß ein OPEN auf den Fehlerkanal (sa=15) durchgeführt wurde.

Der Klartext (2. Meldung) muß in eine Stringvariable gelesen werden, die drei anderen Meldungen können sowohl in String- als auch in Zahlenvariable gebracht werden:

```
10 INPUT$ 15, F, F$, F1, F2
```

Durch

```
20 PRINT F; F$; F1; F2
```

kann die Meldung auf den Bildschirm gebracht werden.

Fehlerbehandlung

Wenn vom Programm die weitere Fehlerbehandlung durchgeführt wird, reicht es, nur F zu lesen. Dann kann über F z.B. auf eine Tabelle mit deutschen Fehlermeldungen zugegriffen werden, oder/und in verschiedene Fehlerbehandlungsroutinen verzweigt werden.

Ursachen und Zusammenhänge

Im Folgenden sind die einzelnen Fehlermeldung beschrieben.

Die 'Fehlermeldungen' kann man aufteilen in die beiden 'Zustandsmeldungen' 0 und 1, die Lesefehler, Schreibfehler, Syntaxfehler und Bedienungsfehler.

Lesefehler

Bei Lesefehlern kann man mehrmals versuchen, die Datei zu lesen. Dabei kann hilfreich sein, die Diskette herauszunehmen, von Hand zu zentrieren und beim Einlegen sehr vorsichtig vorzugehen.

In jedem Fall sollte bei Lesefehlern sofort alles kopiert werden, was wichtig ist und noch kopiert werden kann!

Wenn beim Einlesen von Dateien Lesefehler vom Programm nicht durch eigene Prüfungen erkannt werden, sollte nach jedem INPUT oder GET die Fehlermeldung des Floppy gelesen werden.

Schreibfehler

Der Fehler 25 kann bei schlechten Disketten auftreten. Das Floppy liest einen gerade aufgezeichneten Sektor (Block) und vergleicht ihn mit dem Inhalt des Puffers im Arbeitsspeicher. Dadurch wird geprüft, ob die Aufzeichnung fehlerfrei und lesbar erfolgt ist.

Wenn also Fehler 25 gemeldet wird, sollte die betreffende Diskette vermutlich möglichst umgehend aus dem Verkehr gezogen werden!

Da nur mit dem Inhalt des Floppy-Arbeitsspeichers verglichen wird und nicht mit dem des Rechner-Speichers, werden Übertragungsfehler auf dem Weg zum Floppy durch diesen Test nicht erkannt. Wir empfehlen deshalb, wichtige Aufzeichnungen durch 'VERIFY' bei PRG-Dateien oder durch Lesen und Vergleichen bei sonstigen Dateien zu überprüfen.

Syntaxfehler (Formatfehler)

Die sechs Syntaxfehlermeldungen geben detailliert Auskunft über die Art des Fehlers.

30 und 39: Allgemeine Syntax

Hier wurde gegen die generellen Regeln des Kommandostringaufbaues verstossen. Mögliche Fälle:

- vor ':' nicht '0' oder '1'
- kein '=' zwischen Ziel- und Quellnamen
- kein ',' zwischen mehreren Quellnamen
- führendes Blank im Kommando

Siehe dazu auch die Beispiele bei den jeweiligen Kommandos!

31: Ungültiges Kommando

Hier wurde ein nicht definiertes Kommandozeichen gefunden. Die häufigste Ursache dafür dürfte sein, daß vergessen wurde, das Kommandozeichen zu schreiben und dadurch die Laufwerknummer oder der Dateiname an die Stelle des Kommandos gerückt ist.

32: Zu langes Kommando

Die Länge des Kommandostrings ist auf 40 Zeichen begrenzt.

33: Ungültiger Dateiname

Hier wurde gegen die Regeln für den Dateinamen verstoßen. Mögliche Fehler wären z.B.:

- Länge größer als 16 Zeichen
- Ein verbotenes Zeichen wurde verwendet

In manchen Fällen wird aber bei längeren Namen einfach der hintere Teil abgeschnitten (beim Lesen und Schreiben). Ebenso werden unerlaubte Zeichen im Namen entweder mißinterpretiert, oder Fehler 30 wird gemeldet.

Es ist also angebracht, in Programmen eine Syntaxprüfung des Dateinamens selbst durchzuführen, um Überraschungen zu vermeiden!

34: Kein Dateiname angegeben

Bei den Kommandos N, und S muß mindestens 1 Name angegeben werden. Bei R müssen genau 2, bei C mindestens 2 Namen angegeben werden.

Fehler bei REL-Dateien

50: Record existiert nicht

Nachdem durch RECORD bzw. P ein (noch) nicht vorhandener Record angesprochen werden sollte, erscheint diese Fehlermeldung.

Werden dann durch PRINT\$ Daten zu diesem Record gesandt, wird er eingerichtet und alles ist in Ordnung.

Dagegen darf auf keinen Fall versucht werden, nach dieser Meldung durch INPUT\$ aus diesem Record zu lesen, der Rechner 'stirbt' daran!

51: Zu viele Daten für diesen Satz

Zu einem Satz dürfen nur so viele Bytes ohne CR (CHR\$(13)) geschickt werden, wie die beantragte Satzlänge ist. Ist also Satzlänge 50 vereinbart, können nur maximal 49 Bytes + 1 CR in den Satz geschrieben werden. Das CR zählt also wie ein Datenbyte, da es mit in den Satz geschrieben wird.

Wenn 51 gemeldet wird, wurden die überzähligen Bytes 'weggeworfen'!

52: Satznummer zu groß

Wenn der verlangte Satz noch eingerichtet werden würde, würde DISK FULL auftreten. Diese Meldung kann ebenso wie 50 nach RECORD auftreten.

Bedienungsfehler

Die folgenden Fehlermeldungen werden als Bedienungsfehler bezeichnet, weil sie im Gegensatz zu (einigen) Schreib- / Lesefehlern bei entsprechender Vorgehensweise immer eintreten.

Sie sind also prinzipiell durch entsprechende Intelligenz des Programmes zu vermeiden. Natürlich kann man es bei manchen Meldungen auch 'drauf ankommen lassen' und im Fehlerfall definiert reagieren. Dies kann u.U. für den Benutzer unmerklich erfolgen.

26: Schreibschutz

Hier wurde versucht, schreibend auf eine schreibgeschützte Diskette zuzugreifen. Der Schreibschutz wird vor jedem schreibenden Zugriff überprüft.

Nachdem man den Schreibschutz entfernt hat, kann die Datei beschrieben werden.

29: Disk ID stimmt nicht überein

siehe 'Initialisieren'

60: Die Datei ist bereits zum Schreiben geöffnet

Auf eine Datei kann gleichzeitig entweder schreibend oder mehrmals lesend zugegriffen werden. Wenn also auf eine bestimmte Datei bereits schreibend zugegriffen wird (eine Datei geöffnet ist), kann nicht gleichzeitig eine zweite Datei unter dem gleichen Namen geöffnet werden. Dies schließt auch einen lesenden Zugriff ein.

61: Datei ist nicht geöffnet worden

Es ist grundsätzlich möglich, daß eine Datei zwar im CBM - Betriebssystem angemeldet ist, aber das Floppy dies durch aufgetretene Fehler wieder 'vergessen' hat. Dies ist z.B. immer der Fall nach CLOSE 15 oder nach dem Herausnehmen von Disketten.

62: Datei wurde nicht gefunden

Dies bedeutet, daß auf eine Datei lesend zugegriffen werden sollte, die im Inhaltsverzeichnis (unter dem angegebenen) Namen nicht enthalten ist.

Meistens ist ein Schreibfehler beim Dateinamen die Ursache für diese Fehlermeldung.

63: Datei existiert bereits

Hier sollte eine Datei unter einem Namen abgelegt werden, (schreibender Zugriff) der bereits im Inhaltsverzeichnis eingetragen ist.

Entweder gibt man einen anderen Namen an, oder man löscht die vorhandene Datei oder benennt sie um.

64: Dateitypen - Durcheinander

Bei Kommandos mit mehreren Dateinamen müssen sich alle Namen auf Dateien des gleichen Typs beziehen (PRG oder SEQ).

65: Kein Block frei

Meldung nach B-A, wenn der gewünschte Block nicht frei ist. Der 3. und 4. Parameter der Meldung enthält einen freien Block, der in der Regel der nächste freie Block ist.

70: No Channel

Mehr als 5 Datenkanäle stehen nicht zur Verfügung!

Hier sollte ein 6. Kanal geöffnet werden.

66 und 67: Nicht mögliche Spur und Sektor

Bei einem Direktzugriffkommando wurde eine nicht mögliche Spur-/Sektor-Kombination angegeben.

71: BAM zerstört

Die BAM im Arbeitsspeicher ist nicht in Ordnung. Versuchen Sie, durch Initialisieren nochmal die BAM der Diskette in den Arbeitsspeicher zu laden. Wenn der Fehler auch dann noch auftaucht, ist die Diskette nicht mehr lesbar.

Durch diese 'Rettungsaktion' werden die Blöcke offener serieller Schreibdateien wieder freigegeben, diese Datei-Inhalte sind also verloren.

72: Diskette voll

Alle Blöcke der Diskette sind bereits belegt, oder 144 Dateinamen sind bereits im Inhaltsverzeichnis.

73: Meldung der DOS-Version

Nach dem Einschalten des Floppy (Reset) steht diese Meldung im Fehlerkanal. Außerdem wird sie gemeldet, wenn versucht wird, auf eine DOS1-Diskette schreibend zuzugreifen. Hier bedeutet diese Meldung 'Falsche DOS-Version'.

7. Spannungsversorgung

(Logic Seite 3 von 3)

Die vom Gerät benötigten Gleichspannungen (2 x +12V, +5V) werden von integrierten Spannungsreglern erzeugt:

<u>Spannungsregler</u>	<u>Position</u>	<u>Spannungen Testpunkte</u>	
		<u>Eingang/Ausgang</u>	<u>Eingang/Ausgang</u>
7812	VR1	+ 20 V /+ 12 V	C29+ / P3/5
7812	VR2	+ 20 V /+ 12 V	C31+ / P3/3
LM	VR3	+ 9,6V /+ 5 V	C33+ / P3/1

Wechselspannungen vom Transformator

Stecker P4 Spannung (AC)

zwischen Pin

6, 7	45 Vss
1, 2	45 Vss

8. Logic-Beschreibung Digitalboard

<u>Bezeichnung</u>	<u>Beschreibung</u>
UN1	6502 Microprozessor; steuert Datenübertragung zum IEC-Bus; Daten werden über 4 K RAM dem zweiten Prozessor übergeben
UN1/37	1 MHz Takt
UH1/UL1	2332 ROM 4 K x 8 Bit; Betriebssystem Adressen: UH1: 7000 - 7FFF UL1: 6000 - 6FFF UJ1: 5000 - 5FFF (4040)
UB3	BCD zu Dezimaldecoder generiert aus den Adressen IA13 - IA15 die Signale IK1 - IK4 zum selektieren eines 1 K-Memory-Blocks.
UE1	6532 RAM, I/O, Timer Erzeugung der IEC-Steuersig- nale, IEC-Adress-Einstellung, Ansteuerung der LED;
UC1	6532 RAM, I/O, Timer Datenpuffer zum IEC-Bus, Ansteuerung von 2 x 8 Daten- leitungen, jeweils 8 für In- put und 8 für Output
UB1	Leistungstreiber für IEC-Bus- Daten DIO1 - DIO4
UB2	Leistungstreiber für IEC-Bus- Daten DIO5 - DIO8
UD2	Leistungstreiber für IEC-Bus- Steuersignale, Data Valid, End or Identify, Not Ready for Data, No Data Accept

UM2	Timer zur Erzeugung des Reset-Signals
Y1	Quarz 16 MHz
UB6	4 Bit-Binär-Zähler zur Frequenzteilung

Seite 2 von 3

<u>Chip/Pin</u>	<u>Beschreibung</u>
UH3	6504 Microprozessor, steuert Laufwerkmechanik, Datenübertragung zum Analog-Board
UH3/37	1 MHz, Clock des Prozessor UN1 negiert. Es werden nur 13 Adreßleitungen benötigt.
UA3	BCD-Dezimal-Decoder, generiert aus den Adressen FA10 - FA12, vier Signale, FK1 - FK4, welche je einen 1 K Memory-Block selektieren
UC3/UD3/UE3/UF3	Vier 2 nach 1 Multiplexer, Takt I/O selektiert, welcher Prozessor die RAM's adressiert Pin 1 = high: Adressierung durch Prozessor UN1
UC4/UC5/UD4/UD5	4 K RAM-Memory
UE4/UE5/UF4/UF5	8 x 1 K x 4 Bit RAM's 2114 Adresse: 0400 - 13FF
UM3	6522 VIA, Laufwerksteuerung, Zwischenspeicherung der gelesenen Daten
UK3	6530, 64 Byte RAM, 1 K Byte ROM, I/O, Timer Betriebssystem zur Laufwerksteuerung, Zwischenspeicherung der Schreibdaten

Seite 3 von 3

<u>Chip/Pin</u>	<u>Beschreibung</u>
UJ5/UK5/UL5	Drei 4 x 1 nach 1 Multiplexer
UJ5/1	Das Signal R/W Select schaltet entweder die Schreibdaten oder die Lesedaten zum ROM UK6 durch
UC6/UD6	Erzeugung des Schreib-/Lesetaktes
Signale DS0/DS1	über diese Signale wird eine Frequenzverstellung der Schreib- und Lesetakte in Abhängigkeit der Schreib-/Lesekopf-Position vorgenommen
UP5/UN5	2 x Shift-Register, Seriell in, Parallel out, Umwandlung der seriellen Lesedaten
UK6	ROM, Datencodeumsetzung 8 Bit nach 8 Bit + 2 Controlbit
UN6/UM6	Shift-Register, Parallel in, Seriell out, Schreibdatenumwandlung

9. SteckerbelegungSignale vom Digital- zum Analogboard -

Stecker P7 (nur TTL-Regel, Verbindung zum Digitalboard)

<u>Pin</u>	<u>Signal</u>
1	Ground
2	Head Select 1
3	Kopfmotor 0 on
4	Write Protect 0
5	Activ LED 0
6	Write Protect 1
7	Kopfmotor 1 on
8	Activ LED 1
9	S1B Steuerphase Kopfsteppermotor 1
10	Ground
11	Ground
12	Head Select 0
13	Write Enable
14	Write Data
15	Read Data
16	Read Inhibit
17	SOA Steuerphase Kopfsteppermotor 0
18	SOB Steuerphase
19	S1A Steuerphase Kopfsteppermotor 1
20	Ground

10. Logicbeschreibung Analogboard

Steppermotor-Ansteuerung

Stecker P2/17,18

zwei Phasen zur Ansteuerung von Drive 0. Durch Invertierung ergeben sich vier Steppermotor-Steuersignale. Über die Transistoren Q8, Q7, Q6, Q5 werden + 12 V Impulse auf die Wicklungen des Motors geschaltet (Stecker P8)

P2/9,19

zwei Phasen zur Ansteuerung von Drive 1 Steppermotor

Schreibsignal

P2/14

Die seriellen Schreibdaten werden durch das Flip Flop B3 und die Transistoren C3 in Spannungspegel umgewandelt, mit welchen die Schreibleseköpfe angesteuert werden

P2/2;12

Head Select 1/Head Select 2
Durch diese beiden Signale wird entweder das Signal Common 1 oder Common 2 auf Ground gelegt, wodurch ein Schreiben auf Diskette ermöglicht wird.

P2/13

Write enable = high für Schreiben

Lesesignal

P2/15

Read Data, serielle Lesedaten
Lesesignale von Kopf 1 und Kopf 2 werden zusammengeführt. Über das Signal Head Select 0,1 wird ausgewählt, von welchem Kopf gelesen werden soll. Die Differenzverstärker A4, A5 wandeln die analogen Lesesignale in bitserielle digitale Daten.
(TP3, TP4-Testpunkte für Alignment)

P2/16

Read Inhibit = low: verhindert das Einlesen von Daten

11. ROM-Bestückung Floppy-Disk

FLOPPY-TYPE:		3040		4040 (modif.3040)		8050	
BETRIEBSSYSTEM:		DOS 1.0		DOS 2.0		DOS 2.5	
PLATZ	Teilenummer	Type	Teilenummer	Type	Teilenummer	Type	
UK6	901467-01	6316	901467-01	6316	901467-01	6316	
UK3	901466-02	6530	901466-04	6530	901483-03 *)	6530	
UL9	901468-06	6332	901468-12/	6332	901482-01/03/07	2364	
UH1	901468-07	6332	901468-13	6332	901482-02/04/06	2364	
UJ1	- - -	- -	901468-11	6332	- - -	- -	
					*) 901483-04 für Tendron-Laufwerke		

8250

DOS 2.7

	Teilenummer	Type
UK 6	901467-01	6316
UK 3	901885-04	6530
UL 1	901887-01	2364
UH 1	901888-01	2364

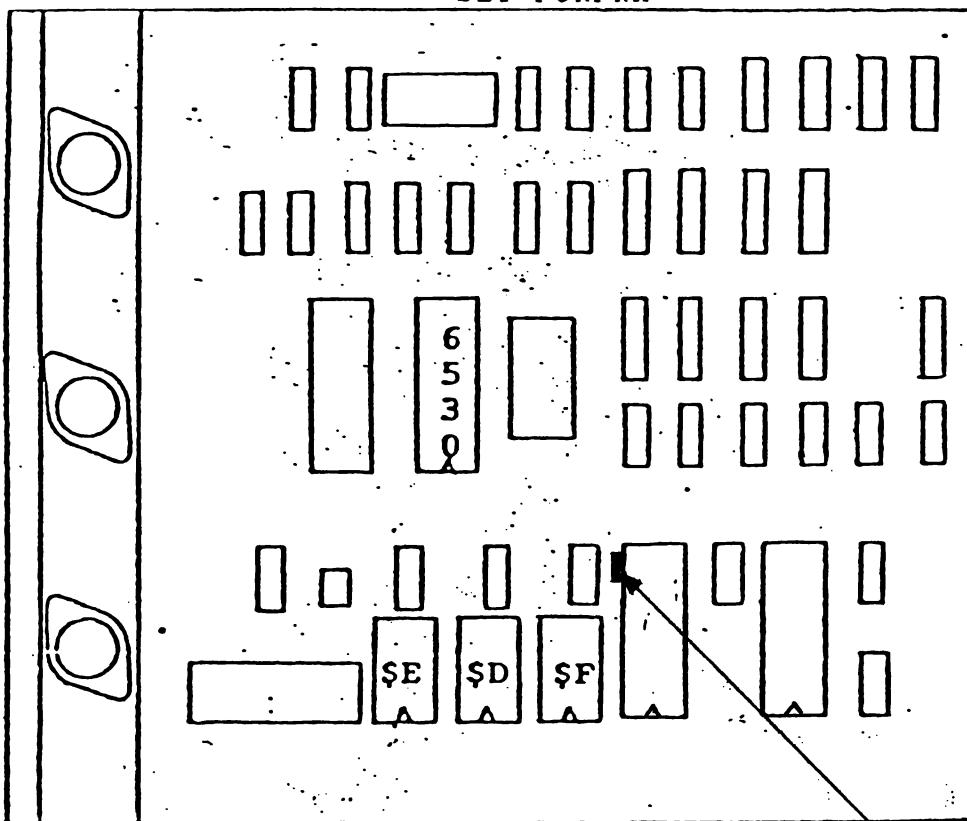
DOS 2.1

Monteringsvejledning for DOS 2.1 i CBM 3040
SAMT ÆNDRING AF DEVICE NR PÅ 3040-4040-8050

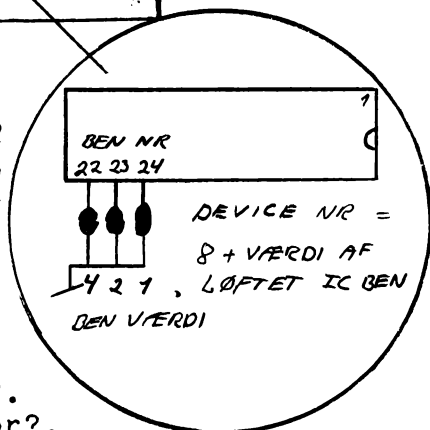
For at installere DOS 2.1 i CBM 3040 floppy disk åbnes floppy disken ved at fjerne de to skruer, en i hver side. Vip låget op og hold dette oppe med stangen, som er monteret i venstre side af låget.

Monter derefter DOS 2.1 i printet som vist her.

SET FORFRA



pos:	nummer:	pos:	nummer:
	GL.DOS1.0		NY.DOS2.1
\$D	TOM	\$D	901468-11
\$E	901468-06	\$E	901468-12
\$F	901468-07	\$F	901468-13
6530	901466-02	6530	901466-04



Kontroller: Er rommerne placeret rigtigt?
Er rommerne vendt som vist på figuren?
Er alle ben i soklerne og ikke uden for?.

Hvis disse betingelser er i orden kan maskinen tændes. Maskinen skulle nu vise en ny opstart ved at drive 0 i floppy disken starter med at initializere.

4040 DOS 2.1

POWER UP DIAGNOSTICS

Antal blink	Område	Komponent	Lokation
1	ZERO PAGE \$00-FF	6532	E1, C1
2	ROM \$F000-FFFF	6332	H1
3	ROM \$E000-EFFF	6332	L1
4	ROM \$D000-DFFF	6332	J1
5	CONTROLLER	6504	H3
	ZEROPAGE \$00-3F		
	ROM \$FC00-FFFF	6530	K3
6	RAM \$1000-13FF	2114	C4, C5
7	RAM \$2000-23FF	2114	D4, D5
8	RAM \$3000-33FF	2114	E4, E5
9	RAM \$4000-43FF	2114	F4, F5
10	ROM \$FC00-FFFF	6530	K3
	CONTROLLER	6504	H3

810815

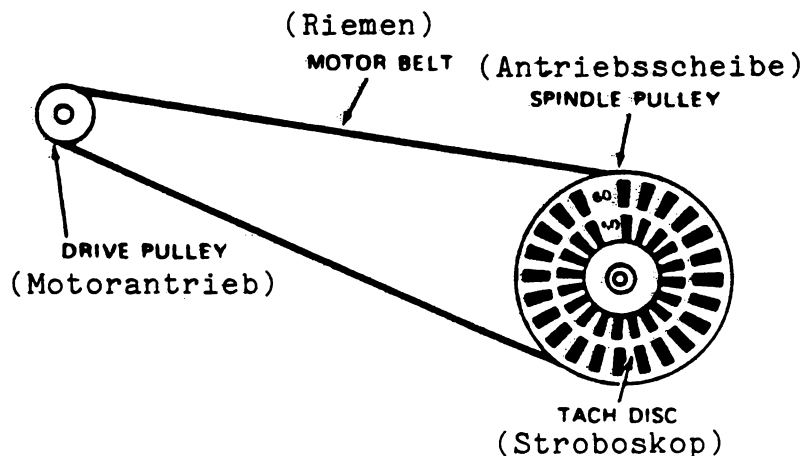
Knud Hansen

Test der Commodore-Floppy 3040 (4040)

Um die Floppy 3040 (4040) zu testen, benötigen Sie die Testdiskette "3040-V1" oder die amerikanische Ausführung "2040 System Test". Für den Test des Schreibschutzschalter (Write protect switch), den LED-Test, den Compatibilitäts-Test und den Motor-Geschwindigkeitstest (Motor speed test) befolgen Sie bitte die Anweisungen des Programmes.

Der Schreibschutzschalter sitzt auf der linken Seite des Drives und wird mittels eines kleinen Inbus-Schlüssels justiert.

Die Motorgeschwindigkeit wird am Poti auf dem Motorcontrolboard des Drives eingestellt. Dazu wird mit einer Leuchtstoffröhre auf das Stroboskop (Tach disk) der Antriebsscheibe (Spindel pulley) geleuchtet. Bei korrekter Motorgeschwindigkeit muß das Stroboskop stehen.



Allignment des Shugart-Drives SA390

Für das Allignment der SA390 Drives benötigen Sie die Dysan Allignmentdiskette Dysan 224, sowie ein 2-Strahl Oszilloskop mit ext. Trigger und Addition.

Anschluß des Oszilloskop's

Analogboard:

Kanal 1 auf Testpunkt 3 (TP3) - R30

Kanal 2 auf Testpunkt 4 (TP4) - R37

Digitalboard:

ext. Trigger an UM5 - Pin 9 (74 LS 133)

Einstellung des Oszilloskop's

Kanal 1 invertiert

Kanal 1 + 2 addiert

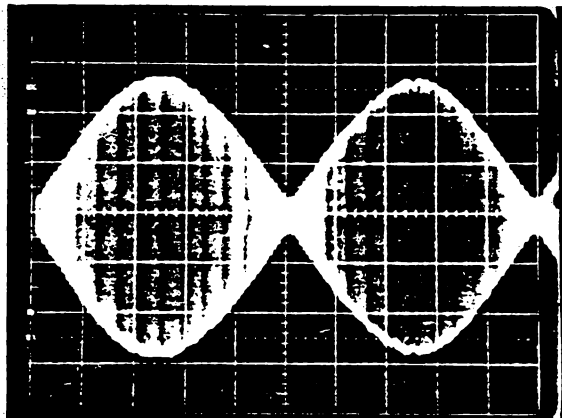
ext. Trigger ein

20 ms/cm

50 mv/cm

AC

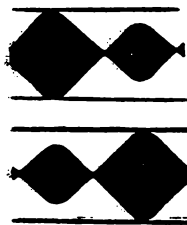
Nach Abruf des Allignmentprogramms und nach Einlegen der Allignmentdiskette 224 sollte ein sogenanntes "Katzenauge" auf dem Oszilloscope erscheinen.



Optimal eingestelltes
"Katzenauge"

Um das "Katzenauge" einzustellen, werden die beiden Befestigungsschrauben des Stepermotors gelöst. Nun drehen wir den Stepermotor so lange, bis das "Katzenauge" optimal eingestellt ist. Nun werden die Hysteresissteps nach innen und außen durchgeführt (Programmablauf beachten) und eventuell der Stepermotor nachjustiert. Nachdem alle Durchgänge beendet sind, werden die Befestigungsschrauben des Stepermotors wieder angezogen und dieser Programmablauf wiederholt.

schlecht eingestellter Drive



muß nachjustiert werden

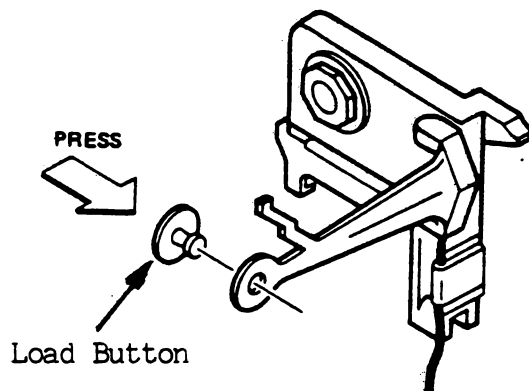
muß nachjustiert werden

gut eingestellter Drive



optimale Einstellung

Austausch des Load Button (Andruckfilz)



Bei Abnutzung oder Vibration (der Drive "singt") muß der Andruckfilz ausgetauscht werden. Mit einer Zange wird die Halteklammer des Andruckfilz zusammen gedrückt und herausgeschoben. Neuer Andruckfilz wird nur in die Halterung gedrückt.

8050 - 8250 - 8250LP 8296D

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

<u>Kapitel</u>	<u>Seite</u>
1. Technische Daten	1 + 2
2. Floppy-Befehle DOS1/DOS2	3
3. DOS2.5 Beschreibung	4 - 6
4. Diskettenorganisation	7 - 10
5. Fehlermeldung und Fehlerbe- handlung	11 - 18
6. Blockschalbild	19
7. Spannungsversorgung	20
8. Logic-Beschreibung Digitalboard	21 - 23
9. Steckerbelegung	24 + 25
10.- Logic-Beschreibung Analog Board	26 + 27
11. Lese-Routine	28
12. Micropolis Drive Alignment	29 - 33

1. Technische Daten

Speicherkapazität

Gesamtkapazität	533248 Bytes pro Diskette
Serielle Dateien	521208 Bytes pro Diskette
Relative Dateien	464312 bis 517398 Bytes pro Diskette abhängig von der Satzgröße
	182880 Bytes pro Datei
	65535 Sätze pro Datei
Directory-Einträge	224 pro Diskette
Sektoren pro Spur	23 bis 29
Bytes pro Sektor	256 (nutzbar: 254 bei PRG-, SEQ-, REL-Dateien bzw. 255 bei Direkt- Zugriff)
Spuren	77
Blöcke	2083 (davon 2052 für den Benutzer verfügbar)

IC's

Controller	6502	Microprozessor
	6530	I/O, RAM, ROM
	6522	I/O, Timer
Interface	6502	Microprozessor
	6532 (2)	I/O, RAM, Timer
	6564 (2)	ROM
Gemeinsam	2114 (8) 4*	1K RAM

Physikalische Eigenschaften

Material	Stahlgehäuse
Abmessungen	18 (H) * 38 (B) * 40 (T)
Gewicht	12 kg

Elektrische Eigenschaften

Spannung	220 V
Frequenz	50 or 60 Hz
Leistungsaufnahme	50 W

2. Floppy-Befehle DOS1/DOS2

ALL BASIC VERSIONS	UNIVERSAL DOS SUPPORT	BASIC 4.0
SAVE "dr:fn",8	SAVE"dr:fn",8	DSAVE,"fn",Ddr (drive defaults to 0)
LOAD"dr:fn",8	/dr:fn (searches both drives)	DLOAD"fn",Ddr (drive defaults to 0)
LOAD"*",8 RUN	↑ dr:fn	DLOAD"fn",Ddr RUN
LOAD"dr:fn",8 RUN	↑ *	shifted RUN/STOP
LOAD"\$0",8 LIST destroys memory	>\$0 preserves memory	DIRECTORY or DI<shifted preserves memory
10 OPEN1,8,15 20 INPUT#1,A,B\$,C,D 30 PRINT A,B\$,C,D	> return	?DS\$ or ?DS (DS is number of error only)

NOTE: Assume that OPEN1,8,15 has already been typed for all of the PRINT commands in the following formats. Commands may be spelled out or abbreviated by the first letter as illustrated.

PRINT#1,"Ix"	INITIALIZE >Ix	PRINT#1,"Ix"
PRINT#1,"Vdr"	VALIDATE >Vdr	COLLECT Ddr
PRINT#1,"Sdr:fn"	SCRATCH >dr:fn	SCRATCH"fn",Ddr
PRINT#1,"Dddr=sdr"	DUPLICATE >Dddr=sdr	BACKUP Dsdr TO Dddr
PRINT#1,"Cddr=sdr"	COPY (all disk) >Cddr=sdr	COPY Dsdr TO Dddr
PRINT#1,"Cdr:dfn= dr:sfn"	COPY (single file) >Cdr:dfn=dr:sfn	COPY Ddr,"sfn" TO Ddr,"dfn"
PRINT#1,"Cdr:dfn= dr:sfn1,dr:sfn2,...	CONCANTENATE FILES >Cdr:dfn=dr:sfn1, dr:sfn2,...	CONCAT Ddr,"sfn" TO Ddr,"dfn"
PRINT#1,"Rdr:dfn=sfn"	RENAME FILES >Rdr:dfn=sfn"	RENAME Ddr,"sfn" TO "dfn"
PRINT#1,"Ndr:dname,xx"	FORMAT A DISKETTE >Ndr:dname,xx	HEADER "dname",Ddr,Ixx

3. DOS2.5 Beschreibung

Floppy Organisation

Dieser Abschnitt soll die interne Organisation des Floppy-Disk-Laufwerks beschreiben. Dabei wird nicht auf den physikalischen Aufbau eingegangen, sondern dem Programmierer sollen hier nur die logischen Verbindungen im Floppy erklärt werden.

Dieses Wissen ist für die Behandlung von SEQ, REL usw. Dateien nicht notwendig, da man mit den komfortablen FloppyKommandos immer direkt Dateien oder Programme auf der Diskette anspricht und sich somit nicht darum kümmern mußte, wie das Floppy diese Kommandos ausführt und auf welchen Wegen Daten von oder zur Diskette transportiert werden.

Will man aber die tieferen Möglichkeiten des Floppy-Disk-Betriebssystems (Abkürzung DOS = Disk Operating System) nutzen, dann sollte man einen Überblick über die interne Organisation des DOS haben.

DOS mit Pipeline-Struktur

Das DOS steuert die Handhabung des gesamten Informationsaustausches zwischen den Disketten und dem IEEE-488-Bus. Über diesen Bus wird die Verbindung zum Rechner organisiert. Die Handhabung des IEEE-488-Busses sollte dem Benutzer schon bekannt sein.

Wenn im folgenden vom "Floppy" die Rede ist, ist damit nicht die Diskette gemeint, sondern die Verbindung des IEEE-488-Bus, da hier die Stelle ist, an die der Rechner direkt angeschlossen ist.

Dazu steht dem DOS ein interner Schreib-/Lesespeicher zur Verfügung. Dieser Speicher wird als Puffer benutzt, um die Information, die von oder zu der Disk übertragen wird, zwischenzuspeichern. Dies ist sinnvoll, weil die Übertragung Rechner-Floppy wesentlich schneller geht als die Übertragung Floppy-Diskette.

Eine solche Struktur wird als "Pipeline-Struktur" bezeichnet. Der Pufferspeicher ist die "Pipeline" zwischen Rechner und Diskette und speichert die schnell anfallende Information vom Rechner, um sie dann an das langsamere Medium Diskette weiterzugeben.

Die Übertragung von Disk zum Rechner wird durch diese Pipeline-Struktur auch unterstützt. Wenn der Rechner die anfallende Information nicht so schnell verarbeiten kann wie sie von der Diskette gelesen wird, wird sie im Pufferspeicher zwischengespeichert und kann bei Bedarf schnell an den Rechner weitergegeben werden. Dieser muß also nicht erst warten, bis die weitere Information von der Disk gelesen wird.

Das Floppy liest also "auf Verdacht" von der Diskette, denn es wäre natürlich auch der Fall möglich, daß der Rechner die weitere Information nicht mehr benötigt.

Pufferspeicher

Der Pufferspeicher hat eine Kapazität von 4 K-Bytes. Diese 4 K sind in 16 Bereiche zu je 256 Bytes aufgeteilt. Von diesen Bereichen werden 15 als Puffer verwendet, die mit Puffernummern von 0 bis 14 bezeichnet werden. 10 Puffer werden für die Datenübertragung verwendet, die restlichen benötigt das DOS selbst.

Kanalorganisation

Das Dateisystem ist durch sogenannte **Kanäle** organisiert. Diese Kanäle sind keine physikalischen, sondern rein logische Verbindungen im Floppy.

Man kann sie mit der Dateistruktur beim Rechner vergleichen. Wird dort eine Datei mit Logischer Adresse, Primäradresse und Sekundäradresse durch OPEN geöffnet, dann werden in einer Liste im Betriebssystem-Speicher die verschiedenen Adressen einander zugeordnet. Will der Programmierer nachher auf diese Datei zugreifen, muß er nur die logische Adresse ansprechen. Das Betriebssystem kann aus der Liste selbständig alle anderen Adressen entnehmen und damit den Datenaustausch organisieren.

Beim Floppy werden durch einen OPEN-Befehl einem Kanal ein, zwei oder drei Pufferbereiche zugeordnet. Bei OPEN auf eine Datei vom Typ SEQ oder PRG interessiert den Benutzer nicht, welcher Kanal welchem Puffer zugeordnet worden ist.

Gesamtübersicht über das DOS mit Verbindungen

Generell kann man also sagen, daß für den Programmierer alle Verbindungen im Floppy über Kanäle angesprochen werden. Der Kanal 15 als Kommandokanal hat einen Sonderstatus. Über ihn können keine Daten von oder zu der Disk übertragen werden.

Es gibt also im Floppy folgende logische Verbindungen, die auch im Bild eingetragen sind:

Für den Benutzer direkt zugreifbar:

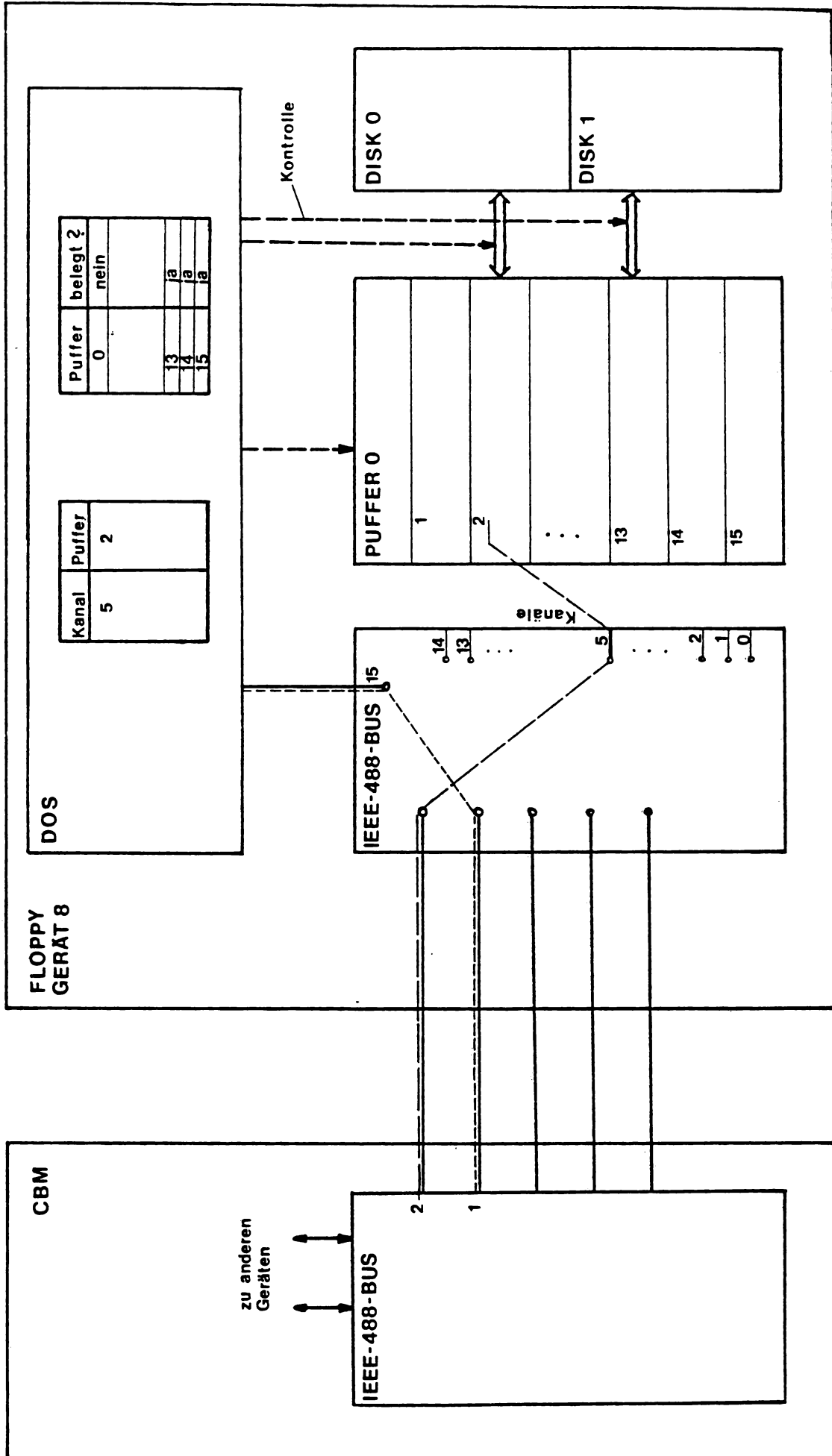
- a) Rechner - Floppy IEEE-488-Bus (durch Logische Adresse)
- b) Floppy IEEE-488-Bus - Pufferspeicher (durch Sekundäradresse = Kanal)
- c) Floppy IEEE-488-Bus - DOS (durch Kommandokanal)

Für den Benutzer nicht direkt zugreifbar:

- d) Pufferspeicher - Diskette

Diese Verbindung ist indirekt zu beeinflussen, indem man ein Kommando schickt, (über Kanal 15) in dem das DOS angewiesen wird, Information in einer gewünschten Richtung auf dieser Verbindung zu übertragen.

Schema der logischen Verbindungen im Floppy



Betspiel: OPEN 2.8.5" #2"
OPEN 1.8.15

4. Diskettenorganisation

Einteilung in Spur und Sektor

Damit man auf der Diskette auf bestimmte Teile der abgespeicherten Information zugreifen kann, ist diese in Spuren und Sektoren eingeteilt. Diese Einteilung wird bei der Formatierung einer Diskette vorgenommen.

Die Spuren sind konzentrische Kreise um den Mittelpunkt der Diskette. Diese sind weiter aufgeteilt in Sektoren.

Im Gegensatz zu den sonst üblichen Diskettenformaten ist die Anzahl der Sektoren pro Spur nicht auf der ganzen Diskette gleich. Da auf den äußeren Spuren der Umfang größer ist als weiter innen, hat hier eine Spur auch mehr Sektoren. Diese Art der Disketteneinteilung hat den großen Vorteil, daß ohne Erhöhung der maximalen Bitdichte wesentlich mehr Information auf die Diskette gepackt werden kann.

Es gibt insgesamt 77 Spuren, die mit Nummern 1 bis 77 durchnummeriert sind. Die Nummerierung der Sektoren beginnt in jeder Spur bei 0.

Beispiel: Bei Spur 1 gibt es die 29 Sektoren 0 bis 28.

Spur	Sektoranzahl	Pro Spur Insgesamt 8050/8052	
1 bis 39	+ 78 bis 116 = 29	1131	2262
40 "	53 + 117 " 130 = 27	378	756
54 "	64 + 131 " 141 = 25	275	550
65 "	77 + 142 " 154 = 23	299	598
		<u>2083</u>	<u>4166</u>
Für Directory		- 29	- 29
Für BAM		- 2	- 4
Blöcke für freie Benutzung		<u>2052</u>	<u>4133</u>

Blockbelegung

Die kleinste auf der Diskette adressierbare Einheit, die mit Spur und Sektor angesprochen wird, nennt man **Block**. Spur/Sektor wird im folgenden als Blockadresse bezeichnet.

Ein Block enthält 256 Bytes an Information. Der Rechner kann 255 Bytes von der Diskette lesen oder auf die Diskette schreiben.

BAM = Block-Verfügbarkeits-Tabelle

Die BAM (Block Availability Map) ist eine Liste, in der für jeden Block angegeben ist, ob dieser belegt ist oder nicht.

Wird zum Beispiel eine Datei auf den Blöcken Spur 17 Sektor 0 und Spur 17 Sektor 8 abgelegt, dann werden diese Blöcke in der BAM als belegt gekennzeichnet. Dadurch wird gewährleistet, daß diese Blöcke beim Abspeichern von anderen Dateien nicht mehr benutzt werden. Beim Löschen von Dateien werden die von dieser Datei belegten Blöcke wieder freigegeben.

Die BAM steht auf der Diskette auf Spur 38 Sektor 0 und 3. Ein Block steht nach dem Initialisieren immer im Pufferspeicher und wird bei Bedarf gegen den anderen ausgetauscht. BAM bei 8250 auf Spur 38 Sektor 0,3,6,9

Inhaltsverzeichnis

Beim Abspeichern einer seriellen Datei auf eine Diskette wird der Name dieser Datei im Inhaltsverzeichnis vermerkt.

Für das Inhaltsverzeichnis ist auf der Diskette die ganze Spur 39 reserviert. Auf diese Spur werden deshalb keine Dateien abgespeichert.

In einem Block können bis zu acht Dateinamen stehen. Die Reihenfolge der Namen hängt von der Reihenfolge der Abspeicherungen ab. Beim Löschen einer Datei wird der Name im Inhaltsverzeichnis als gelöscht gekennzeichnet. Dadurch wird er beim Lesen des Inhaltsverzeichnisses nicht mehr berücksichtigt.

Beim Abspeichern einer Datei auf die Diskette wird das Inhaltsverzeichnis von vorne nach dem ersten freien Platz durchsucht. Dieser wird dann für den Namen der abzuspeichernden Datei verwendet.

In derselben Reihenfolge, wie die gültigen Namen im Inhaltsverzeichnis abgespeichert sind, werden sie auch gelesen.

Die Reihenfolge, der Dateinamen ist wichtig, wenn man nach dem Einschalten des Systems ein bestimmtes Programm mit SHIFT/RUN laden möchte. Wie schon beschrieben muß dieses Programm als erstes im Inhaltsverzeichnis stehen.

Dies erreicht man auf zwei Arten:

Wenn noch keine Datei auf der Diskette steht, wird dieses Programm als erstes abgespeichert.

Wenn die Diskette schon beschrieben ist, muß die erste Datei gelöscht und dann das benötigte Programm abgespeichert werden. Dessen Name wird in dem freien Platz abgespeichert.

Verwaltung der seriellen Dateien

Serielle Dateien werden auf der Diskette so abgespeichert, daß das DOS von Spur 39 nach unten und oben die nächsten freien Blöcke sucht. Normalerweise werden Blöcke verwendet, die nicht benachbart sind (Zeitoptimierung). Allgemein können serielle Dateien auf beliebigen Blöcken in beliebiger Reihenfolge stehen.

Damit das DOS beim Zugriff auf eine serielle Datei diese wieder richtig finden kann, werden die Blöcke durch Verweise verkettet.

Der Zugriff auf eine serielle Datei geht in folgender Reihenfolge:

Beim Namen der Datei im Inhaltsverzeichnis stehen Spur und Sektor des ersten Blocks dieser Datei.

Auf jedem Block der Datei steht ein Verweis auf den nächsten Block in den ersten zwei Bytes, und zwar die Spur in Byte 0, der Sektor in Byte 1.

Der letzte Block einer Datei ist dadurch gekennzeichnet, daß in Byte 0 der Wert 255 steht. In Byte 1 steht hier ein Zeiger auf das Ende der Datei im Block.

Standard-Sprung-Tabelle

Kommando	Adresse	Kommentar
U1 oder UA		BLOCK-READ-Ersatz
U2 oder UB		BLOCK-WRITE-Ersatz
U3 oder UC	\$1300	frei verfügbare Anwender-Adressen
U4 oder UD	\$1303	"
U5 oder UE	\$1306	"
U6 oder UF	\$1309	"
U7 oder UG	\$130C	"
U8 oder UH	\$130F	"
U9 oder UI	\$10F0	NMI Vektor
U: oder UJ		POWER UP Vektor

Die Adressen U3 bis U8 liegen im Puffer-Bereich des RAM und können deshalb vom Programmierer mit eigenen Werten belegt werden.

Verteilung der Blöcke über die Spuren

Spur-Nummer	Block-Bereich
1 bis 39	0 bis 28
40 bis 53	0 bis 26
54 bis 64	0 bis 24
65 bis 77	0 bis 22

Vorspann des Inhaltsverzeichnisses (Directory Header Block)

Adresse: Spur 39 / Sektor 0

Byte	Inhalt	Definition
0, 1	38, 0	Spur und Sektor des ersten BAM-Blocks
2	67	'C' = 8050-Format
3- 5	0	noch nicht verwendet
6- 21		Disketten-Name (mit SHIFT-SPACEs aufgefüllt)
22, 23	160	SHIFT-SPACEs
24, 25		Disk-ID
26	160	SHIFT-SPACE
27, 28	50, 67	'2C' = DOS-Version und Format-Typ
29- 32	160	SHIFT-SPACEs
33-255	0	nicht verwendet

Erster BAM-Block

Adresse: Spur 38 / Sektor 0

Byte	Inhalt	Definition
0, 1	38, 3	Spur und Sektor des zweiten BAM-Blocks
2	67	'C' = 8050-Format
3	0	nicht verwendet
4	1	Niedrigste Spur-Nummer, die in diesem BAM-Block vertreten ist
5	51	Höchste Spur-Nummer+1, die in diesem BAM-Block vertreten ist
6		Anzahl ungenutzte Blöcke der Spur 1
7- 10		Bitweises Verzeichnis der verfügbaren Blöcke auf Spur 1
11-255	*	BAM der Spuren 2-50, jeweils 5 Bytes pro Spur

Zweiter BAM-Block

Adresse: Spur 38 / Sektor 3

Byte	Inhalt	Definition
0, 1	39, 1	Spur und Sektor des ersten Directory-Blocks
2	67	'C' = 8050-Format
3	0	nicht verwendet
4	51	Niedrigste Spur-Nummer, die in diesem BAM-Block vertreten ist
5	78	Höchste Spur-Nummer+1, die in diesem BAM-Block vertreten ist
6		Anzahl ungenutzte Blöcke der Spur 51
7- 10		Bitweises Verzeichnis der verfügbaren Blöcke auf Spur 51
11-140	*	BAM der Spuren 52-77, jeweils 5 Bytes pro Spur
141-255		Ungenutzt

Struktur der BAM für eine Spur

Byte	Definition
0	Anzahl der verfügbaren Sektoren dieser Spur
1	Bits der Sektoren 0- 7
2	Bits der Sektoren 8-15
3	Bits der Sektoren 16-23
4	Bits der Sektoren 24-31

Bit gesetzt = Sektor ist verfügbar

Bit gelöscht = Sektor ist belegt

5. Fehlermeldung und Fehlerbehandlung

Directory-Format

Das Directory belegt die ganze Spur 39. Sektor 0 enthält den Vorspann mit dem Namen der Diskette und ab Sektor 1 stehen die Dateinamen.

Byte	Definition
0, 1	Spur und Sektor des nächsten Directory-Blocks
2- 31	Datei-Eintrag 1
34- 65	Datei-Eintrag 2
66- 97	Datei-Eintrag 3
98-129	Datei-Eintrag 4
130-161	Datei-Eintrag 5
162-193	Datei-Eintrag 6
194-225	Datei-Eintrag 7
226-255	Datei-Eintrag 8

Format eines Datei-Eintrags

Byte	Definition							
0	<table border="0"> <tr> <td>0 = gelöscht</td> <td rowspan="4">Wenn Bit 7 gesetzt ist, wurde die Datei richtig geschlossen. Ist Bit 7 gelöscht, trägt die Datei einen Stern im Inhaltsverzeichnis und ist nicht mehr lesbar.</td> </tr> <tr> <td>1 = SEQ</td> </tr> <tr> <td>2 = PRG</td> </tr> <tr> <td>3 = USR</td> </tr> <tr> <td>4 = REL</td> <td></td> </tr> </table>	0 = gelöscht	Wenn Bit 7 gesetzt ist, wurde die Datei richtig geschlossen. Ist Bit 7 gelöscht, trägt die Datei einen Stern im Inhaltsverzeichnis und ist nicht mehr lesbar.	1 = SEQ	2 = PRG	3 = USR	4 = REL	
0 = gelöscht	Wenn Bit 7 gesetzt ist, wurde die Datei richtig geschlossen. Ist Bit 7 gelöscht, trägt die Datei einen Stern im Inhaltsverzeichnis und ist nicht mehr lesbar.							
1 = SEQ								
2 = PRG								
3 = USR								
4 = REL								
1, 2	Spur und Sektor des ersten Daten-Blocks							
3-18	Datei-Name mit SHIFT-SPACEs aufgefüllt							
19,20	Nur bei REL: Spur und Sektor des ersten Verwaltungs-Blocks							
21	Nur bei REL: Record-Länge							
22-25	nicht verwendet							
26,27	Spur und Sektor der ersetzenden Datei, wenn 'ERSETZEN' arbeitet							
28,29	Anzahl der Blöcke der Datei (LOW/HIGH)							

Struktur der Dateitypen SEQ, PRG und USR

Byte	Definition
0, 1	Spur und Sektor des nächsten Blocks oder Byte 1 = 0 und Byte 2 = letztes gültiges Byte dieses Blocks beim letzten Block der Datei
2-256	254 Daten-Bytes

Fehlertabelle

! DS\$!
 DS !
 F ! F\$! F1 F2

Keine Fehler

00 OK 0 0 Alles in Ordnung
 01 FILES SCRATCHED A 0 F1 enthält Anzahl gelöschter Dateien

Lesefehler

20 READ ERROR T S Kein Block 'header'
 21 READ ERROR T S Kein Synchronisationszeichen
 22 READ ERROR T S Datenblock nicht vorhanden
 23 READ ERROR T S Prüfsummenfehler im Datenblock
 24 READ ERROR T S Byte wurde falsch dekodiert
 27 READ ERROR T S Prüfsummenfehler im 'header'
 71 DIR ERROR 0 0 Directory Fehler

Schreibfehler

25 WRITE ERROR T S Schreib- / Prüffehler
 28 WRITE ERROR T S Datenblock zu lang

Syntax Fehler

30 SYNTAX ERROR 0 0 Allgemeine Syntax
 31 SYNTAX ERROR 0 0 Ungültiges Kommando
 32 SYNTAX ERROR 0 0 Zu langes Kommando
 33 SYNTAX ERROR 0 0 Ungültiger Dateiname
 34 SYNTAX ERROR 0 0 Dateiname fehlt
 39 SYNTAX ERROR 0 0 Ungültiges Kommando

Bedienungsfehler

26 WRITE PROTECT ON T S Schreibschutz 'geklebt'
 29 DISK ID MISMATCH T S Disk ID stimmt nicht überein
 50 RECORD NOT PRESENT T S Record ist nicht vorhanden
 51 OVERFLOW IN RECORD T S Record ist zu kurz für die Daten
 52 FILE TOO LARGE T S Recordnummer ist zu hoch
 60 WRITE FILE OPEN 0 0 Bereits Datei zum Schreiben geöffnet
 61 FILE NOT OPEN 0 0 Datei ist nicht 'offen'
 62 FILE NOT FOUND 0 0 Datei wurde nicht gefunden
 63 FILE EXISTS 0 0 Datei existiert bereits
 64 FILE TYPE MISMATCH 0 0 Dateitypendurcheinander
 65 NO BLOCK T S Kein Block frei
 66 ILLEGAL TRACK T S Unerlaubte Spur oder Sektor
 AND SECTOR
 67 ILLEGAL SYSTEM T OR S T S Unerlaubte Spur oder Sektor
 70 NO CHANNEL 0 0 Kein Kanal frei
 71 DIR ERROR T S BAM ist defekt
 72 DISK FULL T S Diskette voll
 73 CBM DOS V2.5 0 0 Meldung der DOS-Version
 74 DRIVE NOT READY 0 0 Laufwerk enthält keine Diskette
 75 FORMAT SPEED ERROR C 0 UMLAUFZEIT weicht mehr als 2ms ab
 NUR 8250

Allgemeine Hinweise zur Fehlermeldung

Bei der Ausführung von Floppy - Kommandos kann es zu Fehlern kommen. Dabei gibt es Fehler, die zwangsläufig und vorhersehbar eintreten, wie etwa WRITE PROTECT ERROR. Daneben können aber auch Fehler auftreten, die zufälliger Natur sind, wie etwa der Fehler 25.

Die Fehler- bzw. Zustandsmeldungen des Floppy können über den Fehlerkanal abgefragt werden.

Wenn ein Fehler auftritt, wird dies durch die Fehlerlampe zwischen den beiden Laufwerken angezeigt. Dieser Hinweis dürfte allerdings nur beim Arbeiten im Direktmodus interessant sein. Sobald Floppy - Kommandos von Programmen aus gegeben werden, empfehlen wir, grundsätzlich nach dem Kommando die Fehlermeldung einzulesen, um den Benutzer möglichst schnell auf Fehler hinweisen zu können.

Da die meisten Fehlermeldungen durch ein neues Kommando gelöscht werden, muß nach jedem einzelnen Kommando die Fehlermeldung gelesen und ausgewertet werden. Dabei sind zwei Möglichkeiten zu unterscheiden:

- (1) Die Fehlermeldung wird unmittelbar nach dem jeweiligen Kommando gelesen.
- (2) Die Fehlermeldung wird unmittelbar vor dem nächsten Kommando gelesen.

Die erste Möglichkeit ist im Programm einfacher zu realisieren, hat aber den Nachteil, daß der Anwender warten muß, bis das Floppy das betreffende Kommando ausgeführt hat. INPUT\$ wartet nämlich, bis die betreffende Operation zu Ende ist.

Die zweite Möglichkeit bietet den Vorteil, daß der Dialog weiterlaufen kann, während das Floppy arbeitet. Da das Programm aber dann 'rückwirkend' auf Fehler reagieren muß, wird die Struktur des Dialogteils erheblich anspruchsvoller als im ersten Fall.

Die Bestandteile der Fehlermeldung

Die Fehlermeldung besteht aus vier Teilen. Zuerst wird die **Fehlernummer (F)** gemeldet, dann der zugehörige **englische Klartext (F\$)**. Die beiden folgenden Zahlen geben **Spur (F1)** und **Sektor (F2)** an, wo der Fehler aufgetreten ist, soweit diese Angabe sinnvoll ist.

Lesen der Fehlermeldung durch DS und DS\$ (ab BASIC-4)

Bei BASIC-4 stehen die beiden Variablen-Funktionen DS und DS\$ zur Verfügung. Jeder Aufruf einer der beiden Variablen bewirkt, daß die Floppy-Fehlermeldung gelesen wird, sofern seit dem letzten Aufruf eine Floppy-Operation durchgeführt wurde.

DS enthält die Fehlernummer und DS\$ die gesamte Fehlermeldung (einschließlich der Fehlernummer).

DS und DS\$ holen automatisch die Meldung des zuletzt angesprochenen Gerätes ein. Sie können damit ohne weiteres mehrere Floppy-Geräte mit dieser einen Funktion bearbeiten.

Lesen der Fehlermeldung mit INPUT\$ (nicht nötig ab BASIC-4)

Bei BASIC-3 kann die Fehlermeldung nicht im Direktmodus, sondern nur im BASIC Programm durch INPUT\$ gelesen werden. INPUT\$ setzt voraus, daß ein OPEN auf den Fehlerkanal (sa=15) durchgeführt wurde.

Der Klartext (2. Meldung) muß in eine Stringvariable gelesen werden, die drei anderen Meldungen können sowohl in String- als auch in Zahlenvariable gebracht werden:

```
10 INPUT$ 15, F, F$, F1, F2
```

Durch

```
20 PRINT F; F$; F1; F2
```

kann die Meldung auf den Bildschirm gebracht werden.

Fehlerbehandlung

Wenn vom Programm die weitere Fehlerbehandlung durchgeführt wird, reicht es, nur F zu lesen. Dann kann über F z.B. auf eine Tabelle mit deutschen Fehlermeldungen zugegriffen werden, oder/und in verschiedene Fehlerbehandlungsroutinen verzweigt werden.

Ursachen und Zusammenhänge

Im Folgenden sind die einzelnen Fehlermeldung beschrieben.

Die 'Fehlermeldungen' kann man aufteilen in die beiden 'Zustandsmeldungen' 0 und 1, die Lesefehler, Schreibfehler, Syntaxfehler und Bedienungsfehler.

Lesefehler

Bei Lesefehlern kann man mehrmals versuchen, die Datei zu lesen. Dabei kann hilfreich sein, die Diskette herauszunehmen, von Hand zu zentrieren und beim Einlegen sehr vorsichtig vorzugehen.

In jedem Fall sollte bei Lesefehlern sofort alles kopiert werden, was wichtig ist und noch kopiert werden kann!

Wenn beim Einlesen von Dateien Lesefehler vom Programm nicht durch eigene Prüfungen erkannt werden, sollte nach jedem INPUT oder GET die Fehlermeldung des Floppy gelesen werden.

Schreibfehler

Der Fehler 25 kann bei schlechten Disketten auftreten. Das Floppy liest einen gerade aufgezeichneten Sektor (Block) und vergleicht ihn mit dem Inhalt des Puffers im Arbeitsspeicher. Dadurch wird geprüft, ob die Aufzeichnung fehlerfrei und lesbar erfolgt ist.

Wenn also Fehler 25 gemeldet wird, sollte die betreffende Diskette vermutlich möglichst umgehend aus dem Verkehr gezogen werden!

Da nur mit dem Inhalt des Floppy-Arbeitsspeichers verglichen wird und nicht mit dem des Rechner-Speichers, werden Übertragungsfehler auf dem Weg zum Floppy durch diesen Test nicht erkannt. Wir empfehlen deshalb, wichtige Aufzeichnungen durch 'VERIFY' bei PRG-Dateien oder durch Lesen und Vergleichen bei sonstigen Dateien zu überprüfen.

Syntaxfehler (Formatfehler)

Die sechs Syntaxfehlermeldungen geben detailliert Auskunft über die Art des Fehlers.

30 und 39: Allgemeine Syntax

Hier wurde gegen die generellen Regeln des Kommandostringaufbaues verstossen. Mögliche Fälle:

- vor ':' nicht '0' oder '1'
- kein '=' zwischen Ziel- und Quellnamen
- kein ',' zwischen mehreren Quellnamen
- führendes Blank im Kommando

Siehe dazu auch die Beispiele bei den jeweiligen Kommandos!

31: Ungültiges Kommando

Hier wurde ein nicht definiertes Kommandozeichen gefunden. Die häufigste Ursache dafür dürfte sein, daß vergessen wurde, das Kommandozeichen zu schreiben und dadurch die Laufwerksnummer oder der Dateiname an die Stelle des Kommandos gerückt ist.

32: Zu langes Kommando

Die Länge des Kommandostrings ist auf **40 Zeichen** begrenzt.

33: Ungültiger Dateiname

Hier wurde gegen die Regeln für den Dateinamen verstoßen. Mögliche Fehler wären z.B.:

- Länge größer als 16 Zeichen
- Ein verbotenes Zeichen wurde verwendet

In manchen Fällen wird aber bei längeren Namen einfach der hintere Teil abgeschnitten (beim Lesen und Schreiben). Ebenso werden unerlaubte Zeichen im Namen entweder mißinterpretiert, oder Fehler 30 wird gemeldet.

Es ist also angebracht, in Programmen eine Syntaxprüfung des Dateinamens selbst durchzuführen, um Überraschungen zu vermeiden!

34: Kein Dateiname angegeben

Bei den Kommandos N, und S muß mindestens 1 Name angegeben werden. Bei R müssen genau 2, bei C mindestens 2 Namen angegeben werden.

50: Record existiert nicht

Nachdem durch RECORD bzw. P ein (noch) nicht vorhandener Record angesprochen werden sollte, erscheint diese Fehlermeldung.

Werden dann durch PRINT§ Daten zu diesem Record gesandt, wird er eingerichtet und alles ist in Ordnung.

Dagegen darf auf keinen Fall versucht werden, nach dieser Meldung durch INPUT§ aus diesem Record zu lesen, der Rechner 'stirbt' daran!

51: Zu viele Daten für diesen Satz

Zu einem Satz dürfen nur so viele Bytes ohne CR (CHR\$(13)) geschickt werden, wie die beantragte Satzlänge ist. Ist also Satzlänge 50 vereinbart, können nur maximal 49 Bytes + 1 CR in den Satz geschrieben werden. Das CR zählt also wie ein Datenbyte, da es mit in den Satz geschrieben wird.

Wenn 51 gemeldet wird, wurden die überzähligen Bytes 'weggeworfen'!

52: Satznummer zu groß

Wenn der verlangte Satz noch eingerichtet werden würde, würde DISK FULL auftreten. Diese Meldung kann ebenso wie 50 nach RECORD auftreten.

Bedienungsfehler

Die folgenden Fehlermeldungen werden als Bedienungsfehler bezeichnet, weil sie im Gegensatz zu (einigen) Schreib- / Lesefehlern bei entsprechender Vorgehensweise immer eintreten.

Sie sind also prinzipiell durch entsprechende Intelligenz des Programmes zu vermeiden. Natürlich kann man es bei manchen Meldungen auch 'drauf ankommen lassen' und im Fehlerfall definiert reagieren. Dies kann u.U. für den Benutzer unmerklich erfolgen.

26: Schreibschutz

Hier wurde versucht, schreibend auf eine schreibgeschützte Diskette zuzugreifen. Der Schreibschutz wird vor jedem schreibenden Zugriff überprüft.

Nachdem man den Schreibschutz entfernt hat, kann die Datei beschrieben werden.

29: Disk ID stimmt nicht überein

siehe 'Initialisieren'

60: Die Datei ist bereits zum Schreiben geöffnet

Auf eine Datei kann gleichzeitig entweder schreibend oder mehrmals lesend zugegriffen werden. Wenn also auf eine bestimmte Datei bereits schreibend zugegriffen wird (eine Datei geöffnet ist), kann nicht gleichzeitig eine zweite Datei unter dem gleichen Namen geöffnet werden. Dies schließt auch einen lesenden Zugriff ein.

61: Datei ist nicht geöffnet worden

Es ist grundsätzlich möglich, daß eine Datei zwar im CBM - Betriebssystem angemeldet ist, aber das Floppy dies durch aufgetretene Fehler wieder 'vergessen' hat. Dies ist z.B. immer der Fall nach CLOSE 15 oder nach dem Herausnehmen von Disketten.

62: Datei wurde nicht gefunden

Dies bedeutet, daß auf eine Datei lesend zugegriffen werden sollte, die im Inhaltsverzeichnis (unter dem angegebenen) Namen nicht enthalten ist.

Meistens ist ein Schreibfehler beim Dateinamen die Ursache für diese Fehlermeldung.

63: Datei existiert bereits

Hier sollte eine Datei unter einem Namen abgelegt werden, (schreibender Zugriff) der bereits im Inhaltsverzeichnis eingetragen ist.

Entweder gibt man einen anderen Namen an, oder man löscht die vorhandene Datei oder benennt sie um.

64: Dateitypen - Durcheinander

Bei Kommandos mit mehreren Dateinamen müssen sich alle Namen auf Dateien des gleichen Typs beziehen (PRG oder SEQ).

65: Kein Block frei

Meldung nach B-A, wenn der gewünschte Block nicht frei ist. Der 3. und 4. Parameter der Meldung enthält einen freien Block, der in der Regel der nächste freie Block ist.

70: No Channel

Mehr als 5 Datenkanäle stehen nicht zur Verfügung!

Hier sollte ein 6. Kanal geöffnet werden.

66 und 67: Nicht mögliche Spur und Sektor

Bei einem Direktzugriffkommando wurde eine nicht mögliche Spur-/Sektor-Kombination angegeben.

71: BAM zerstört

Die BAM im Arbeitsspeicher ist nicht in Ordnung. Versuchen Sie, durch Initialisieren nochmal die BAM der Diskette in den Arbeitsspeicher zu laden. Wenn der Fehler auch dann noch auftaucht, ist die Diskette nicht mehr lesbar.

Durch diese 'Rettungsaktion' werden die Blöcke offener serieller Schreibdateien wieder freigegeben, diese Datei-Inhalte sind also verloren.

72: Diskette voll

Alle Blöcke der Diskette sind bereits belegt, oder 224 Dateinamen sind bereits im Inhaltsverzeichnis.

73: Meldung der DOS-Version

Nach dem Einschalten des Floppy (Reset) steht diese Meldung im Fehlerkanal.

74: Laufwerk enthält keine Diskette

Wenn entweder die Klappe nicht geschlossen ist, oder eine nicht formatierte Diskette oder eine nicht auf 8050 formatierte Diskette enthalten ist, wird diese Meldung abgesetzt.

75: Umlaufzeit der Diskette falsch (nur 8250)

Bei 8250 wird vor dem Formatieren die Umlaufzeit der Diskette auf + 2ms überprüft

7. Spannungsversorgung

Logic-Seite 3 von 3

<u>Spannungsregler</u>	<u>Position</u>	<u>Spannungen Testpunkte</u>	
		<u>Eingang/Ausgang</u>	<u>Eingang/Ausgang</u>
7812	VR1	+ 20 V /+ 12 V	C29+ / P3/5
7812	VR2	+ 20 V /+ 12 V	C31+ / P3/3
LM	VR3	+ 9,6V /+ 5 V	C33+ / P3/1

Wechselspannungen vom Transformator

Stecker P4 Spannung (AC)

zwischen Pin

6, 7	45 Vss
1, 2	45 Vss

8. Logic-Beschreibung Digitalboard

Die folgende Beschreibung bezieht sich auf die Logic-Blätter der Nummer 8050001 (P.C.B. ASSY, Floppy, Digital) und 8050006 (P.C.B. ASSY, Floppy, Analog)

Seite 1 von 3

<u>Chip/Pin</u>	<u>Beschreibung</u>
UN1	6502 Microprozessor; steuert Datenübertragung zum IEC-Bus; Daten werden über 4 K RAM dem zweiten Prozessor übergeben
UH1/UL1	6364 ROM 8 K x 8 Bit; Betriebssystem
UB3	BCD zu Dezimaldecoder generiert aus den Adressen IA12 - IA15 die Signale IK1 - IK4 zum selektieren eines 1 K-Memory-Blocks.
UE1	6532 RAM, I/O, Timer Erzeugung der IEC-Steuersignale, IEC-Adress-Einstellung, Ansteuerung der LED;
UC1	6532 RAM, I/O, Timer Datenpuffer zum IEC-Bus, Ansteuerung von 2 x 8 Datenleitungen, jeweils 8 für Input und 8 für Output,
UB1	Leistungstreiber für IEC-Bus-Daten DIO1- DIO4
UB2	Leistungstreiber für IEC-Bus-Daten DIO5-DIO8
UD2	Leistungstreiber für IEC-Bus-Steuersignale, Data Valid, End or Identify, Not Ready for Data, No Data Accept

UM2	Timer zur Erzeugung des- Reset-Signals
Y1	Quarz 12 MHz
UB6	Vier D-Flip-Flops zur Frequenz- teilung

Seite 2 von 3

<u>Chip/Pin</u>	<u>Beschreibung</u>
UH3	6502 Microprozessor, steuert Laufwerkmechanik, Datenübertragung zum Analog- Board
UH3/37	1 MHz, Clock des Prozessor UN1 negiert. Es werden nur 13 Adreßleitungen benötigt.
UA3	BCD-Dezimal-Decoder, generiert aus den Adressen FA10 - FA12, vier Signale, FK1 - FK4, welche je einen 1 K Memory-Block selektieren
UC3/UD3/UE3/UF3	Vier 2 nach 1 Multiplexer, Takt IO selektiert, welcher Prozessor die RAM's adressiert Pin 1 = high: Adressierung durch Prozessor UN1
UC4/UC5/UD4/UD5	4 K RAM-Memory
UE4/UE5/UF4/UF5	8 x 1 K x 4 Bit RAM's 2114
UM3	6522 VIA, Laufwerksteuerung, Zwischenspeicherung der ge- lesenen Daten
UK3	6530, 64 Byte RAM, 1 K-Byte ROM, I/O, Timer Betriebssystem zur Laufwerk- steuerung, Zwischenspeicherung der Schreibdaten

Seite 3 von 3

<u>Chip/Pin</u>	<u>Beschreibung</u>
UJ5/UK5/UL5	Drei 4 x 1 nach 1 Multi-plexer
UJ5/1	Das Signal R/W Select schaltet entweder die Schreibdaten oder die Lesedaten zum ROM UK6 durch
UI5/3	Schreibtakt
UI5/2	Lesetakt
UC6	Erzeugung des Schreibtaktes
UN4/UN3/UP6	Erzeugung des Lesetaktes
Signale DS0/DS1	über diese Signale wird eine Frequenzverstellung der Schreib- und Lesetakte in Abhängigkeit der Schreib-/Lesekopf-Position vorgenommen.
UP5/UN5	2 x Shift-Register, Seriell in, Parallel out, Umwandlung der serillen Lesedaten
UK6	ROM, Datencodenumsetzung 8 Bit nach 8 Bit + 2 Control-bit
UN6	Shift-Register, Parallel in, Seriell out, Schreibdatenumwandlung
UN6/9	Serielle Schreibdaten
UM6	4 Bit Schieberegister, Schreibdatenumwandlung, Signale zur Schreibtakterzeugung
UA4, UA5, UD6	Write-precompensation Mittelwert des magn.Flusses muß ausgeglichen sein über kleine Bereiche

9. Steckerbelegung

Signale am Analogboard

Stecker P7 (nur TTL-Regel, Verbindung zum Digitalboard)

<u>Pin</u>	<u>Signal</u>
1	Head Select
2	Drive Select 0
3	Kopfmotor 0 on
4	Write Protect 0
5	Activ LED 0
6	Write Protect 1
7	Kopfmotor 1 on
8	Activ LED 1
9	S1B Steuerphase Kopfsteppermotor 1
10	Ground
11	Ground
12	DriveSelect 1
13	Write Enable
14	Write Data
15	Read Data
16	Read Inhibit
17	SOA Steuerphase Kopfsteppermotor 0
18	SOB Steuerphase
19	S1A Steuerphase Kopfsteppermotor 1
20	Ground

Stecker P8 (Kopfmotor, Drivemotor, Schalter für Drive 0)

<u>Pin</u>	<u>Signal</u>
1	Drive Motor Input
2	Drive Motor Control
3	Drive Motor Control
4	Drive Motor Input
5	Door Switch
6	Ground
7	Ground
8	Write Protect Switch
9	0A, Phase Kopfsteeper
10	0C Kopfsteeper
15	0BKopfsteeper
16	0D Kopfsteeper
11	-
12	+ 12 V
13	+ 12 V
14	Ground

Stecker P5 (Kopfmotor, Drivemotor, Schalter für Drive 1)

<u>Pin</u>	<u>Signal</u>
1	Drive Motor Input
2	Drive Motor Control
3	Drive Motor Control
4	Drive Motor Input
8	Door Switch
6	Ground
7	Ground
5	Write Protect Switch
9	OD Phase für Kopfstepper
10	OB Phase für Kopfstepper
15	OC Phase für Kopfstepper
16	OA Phase für Kopfstepper
11	Ground
12	+ 12 V
13	+ 12 V
14	-

P1/P3 Stecker für Schreib-/Leseköpfe Drive 0/1
(siehe Floppy Analog Seite 1 von 3)

P6 Versorgungsspannungen

<u>Pin</u>	<u>Spannung</u>
1	+ 5 V
2	Ground
3	+ 12 V
4	Ground
5	+ 12 V

10. Logic-Beschreibung Analog Board

Die folgende Beschreibung bezieht sich auf die Logic-Blätter der Nummer 8050005 C

Seite 1 von 3

<u>Chip/Pin</u>	<u>Beschreibung</u>
U6/U9	D-Flip-Flop, TTL-CMOS Interface Umwandlung der Schreibdaten in, vom Schreibkopf benötigte, Spannungspegel
U8	Timer zur Erzeugung des Löschimpuls
U5/Q4	Power up safety-circuit Schützt die Disketten vor Zerstörung bei Ausschalten des Gerätes
<u>Signal</u>	
Head Select, Drive Select 0 Drive Select 1	Selection der Schreib-/Leseköpfe. (unten/oben,high=unten). Diese beiden Signale werden aus dem Signal Drive Select generiert
Write Enable	wird aus den Signalen R/W-Select und Write Protect erzeugt
Write Data	Schreibdaten
U2/14, 1	gelesene Daten vom Schreib-/Lesekopf werden durch nachfolgende Schaltungen in TTL-Signale umgewandelt
P7/15	Lesedaten mit TTL-Pegel
TP1/TP2 oder TP3/TP4	Testpunkte zur Kopfjustage

Seite 2 von 3

<u>Signal</u>	<u>Beschreibung</u>
W.D.S. (aktiv low)	Door Switch; dieses Signal sorgt dafür, daß bei Schließen der Lauf- werksklappe der Drive-Motor kurz anläuft.
Motor on (aktiv low)	bewirkt das Einschalten des Drive Motors
Motor	dient zur Geschwindigkeits- steuerung des Drive Motors, wird mit wachsender Umdre- hungszahl größer
N.C.W.P.	Write Protect Switch (high: Write Protect on)
Potentiometer R50/R61	Einstellung der Drive-Motor- Geschwindigkeit (R61 Drive 0; R50 Drive 1)

Seite 3 von 3

U18	2 x 2 Bit Binärdecoder generiert aus den Signalen SOA, SOB (Drive 0) und S1A, S1B (Drive 1) je 4 Phasen zur Steuerung der Steppmotoren der Schreib-/Leseköpfe
<u>Signal</u> MTR on 0/1 /aktiv low)	Aktiviert Antriebs-und Steppermotor von Laufwek 0 bzw.1

11. Lese Routine

Die Positionierung des Schreib-/Lesekopfes geschieht durch eine Vierphasen-Schrittmotor. Um von einer Spur zur nächsten zu fahren, benötigt der Schrittmotor vier Schritte.

kann beim Lesen kein Sync erkannt werden,
so reagiert DOS 2,5 und DOS 2,7 unterschiedlich.

DOS 2,5 = 8050

Der Lesekopf wird zunächst um einen Schritt nach innen und außen, dann um zwei Schritte nach innen und außen positioniert.

Wird bei allen vier Leseversuchen kein Sync erkannt,
erfolgt die Meldung " Drive not ready "

DOS 2,7 : 8250

Der Lesekopf bleibt auf der Stelle stehen.

Kann nach mehrmaligen Leseversuchen kein Sync erkannt werden,
erfolgt ein Bump und darauf ein weiterer Versuch.

Bei weiterem Mißerfolg erscheint " Drive not ready . "

12. Micropolis Drive Alignment

Die Kopfjustage der 8050 Micropolis drive erfolgt in der gleichen Art wie bei dem 3040 Shugart drive: Der Kopf wird auf die Alignment-Spur positioniert, um dann die Amplituden auf dem Oszilloscope zu justieren, so daß man ein korrektes "Cat-eye's-Muster" sieht.

Der wesentliche Unterschied der 8050 zur 3040 ist die höhere Spur- und Bitdichte. Diese höhere Dichte erschwert die Kopfjustage, so daß diese nur dann ausgeführt werden soll, wenn einwandfrei sichergestellt ist, daß ein Fehler in der Floppy an der Verstellung des Kopfes liegt.

Zur Justage wird folgendes Werkzeug benötigt:

8050 Alignment Diskette
Test-Diskette
Kreuzschlitz Schraubendreher
3/32 und 7/64 Imbusschlüssel
1/4 Inch Maulschlüssel
Zweikanal-Oszilloscope
Verlängerungskabel für Stepper-,
Drivemotoren, Türschalter, Kopf

Justage Programm:

Diskette einlegen und Programm laden.

Step 2 : Türschalter-Justage

Das Schließen der Tür hat zur Folge, daß der Drive-Motor für kurze Zeit anläuft, damit die Diskette zentriert wird.

Beim Einlegen von Disketten ist darauf zu achten, daß eine exakte Zentrierung nur dann erfolgt, wenn die Tür zügig und nicht langsam, wie bei der 3040, geschlossen wird.

Der Schalter muß so justiert werden, daß der Motor anläuft, wenn beim Schließen der Tür der obere Rand des Diskettenhalters die Höhe des unteren Rand des Einführschlitzes erreicht.

Step 3 : Schreibschutzschalter-Justage

Ist keine Diskette im Laufwerk, so leuchtet die Fehlerlampe grün. Beim Einschieben einer Diskette dagegen muß die Fehlerlampe rot leuchten.

Step 4: Motorgeschwindigkeits-Justage und An-
triebsriemen-Test

Die Motorgeschwindigkeit kann an folgenden Potentiometern auf dem Analogboard eingestellt werden:

R61: Laufwerk 0

R50: Laufwerk 1

Beim Antriebsriemen-Test werden Geschwindigkeitsunterschiede zwischen zwei Umdrehungen untersucht, die als Folge von Schlupf zwischen Antriebsriemen und Antriebsrad auftreten können. Dieser Schlupf hat Formatierungsfehler zur Folge.

Es ist darauf zu achten, daß die "glänzende" oder schwarze Seite des Antriebsriemen innen ist.

Fällt der Antriebsriemen-Test negativ aus, so muß der Riemen erneuert werden.

5?

Step 6 : Bewegung des Kopfes

Dieser Test bewegt den Kopf zwischen den Spuren 10 und 70, um den festen Sitz der Mechanik zu prüfen. Achten Sie auf die Gleichförmigkeit der Bewegung. Der Test dauert 20 Sekunden:

Step 5 : Kopfjustage-Voreinstellung

Die Spur 37 der Alignment-Diskette enthält ein Bit-Muster, welches es ermöglicht, den Kopf mit Hilfe eines Zweikanal-Oszilloscopes, genau auf die Mitte der Spur einzustellen. Dazu muß der Stop-Ring auf der Stepperwelle gelöst und mit dem Stift nach oben an den Kopfträger geschoben werden.

Step 7: Kopfjustage-Überprüfung

Alignment-Diskette einlegen

Oszilloscope: Kanal 1 an TP1 (C3-)
Kanal 2 an TP2 (C4-)
Ex. Trigger an Pin 21 von UK 3

Einstellungen: Kanal 1 invertiert
Kanal 1 und 2 addiert
ex. Trigger
20 ms/cm
50 mV/cm
AC

Die auf dem Scope erscheinenden Amplituden (Cat Eye's) müssen eine Höhendifferenz von weniger als 15 % haben. Ist dies nicht der Fall, so muß der Steppermotor verdreht werden, bis die Amplitudendifferenz im Toleranzbereich liegt.

Sind die Cat Eyes nicht zu sehen, so muß der Steppermotor durch Eintippen von "in" oder "ot" gedreht werden, um so die Alignment-Spur zu finden.

Durch Eintippen von "bump" wird erneut versucht, die Alignment-Spur zu finden.

Durch Eintippen von "hyst" erfolgt ein Hystenese-Step. Es muß kontrolliert werden, daß die Alignment-Einstellung nach diesen Steps immer noch im Toleranzbereich liegt.

Nach Festziehen der Schrauben muß die Einstellung ein weiteres Mal überprüft und gegebenenfalls korrigiert werden.

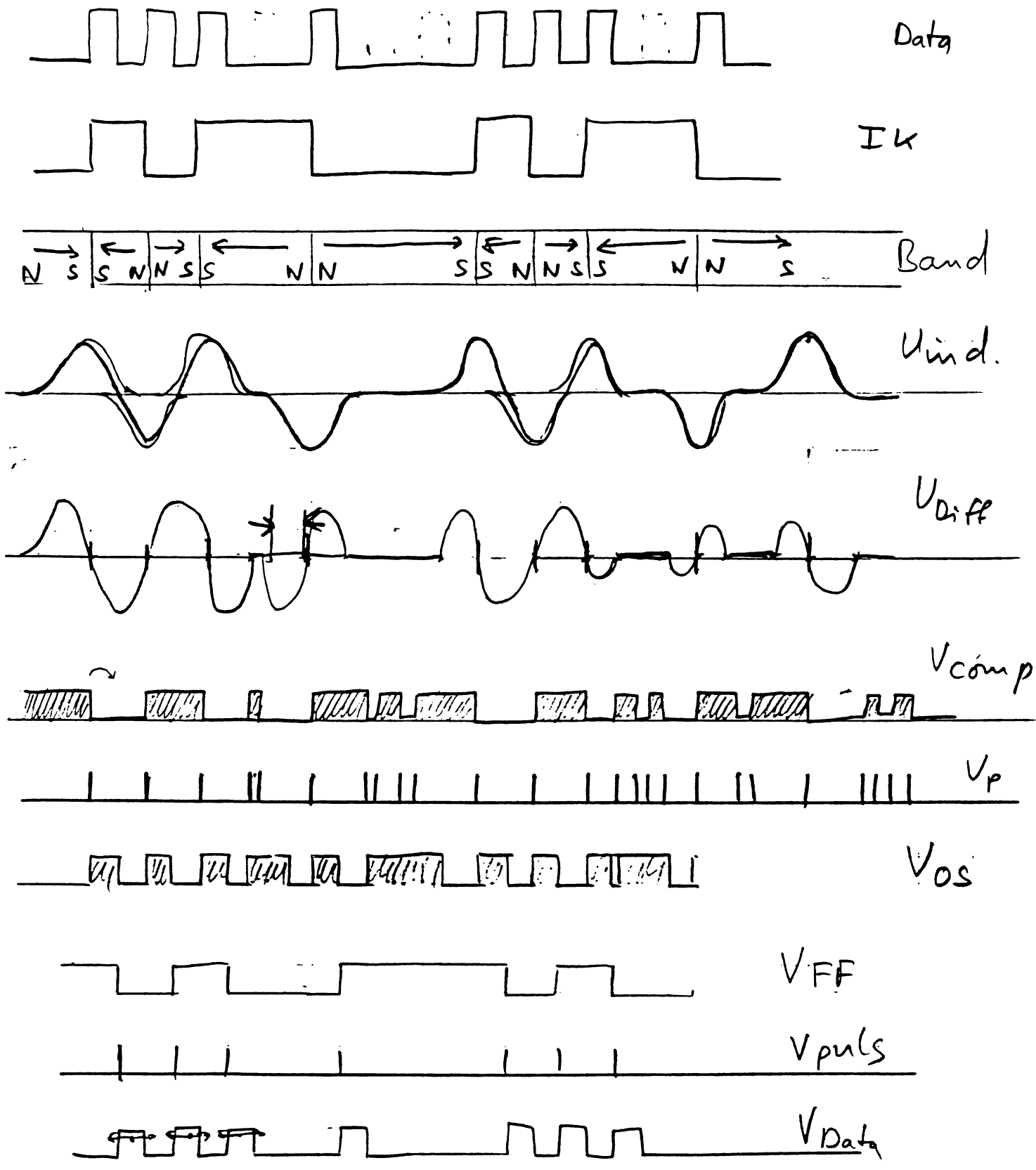
Step 8: Stop-Ring-Justage

Das Testprogramm fährt den Kopf erst auf Spur 1 zurück und dann einen viertel Schritt nach außen. In dieser Position muß der Stop-Ring auf die Lasche des Kopfträgers gelegt werden und zwar so, daß der Stift des Stop-Rings diese Lasche mit der Hälfte seiner Länge überlappt. Nun wird der Ring festgeschraubt.

Zum Schluß fährt der Kopf noch einmal auf Spur 37, so daß die "Cat Eye's" noch einmal kontrolliert werden können.

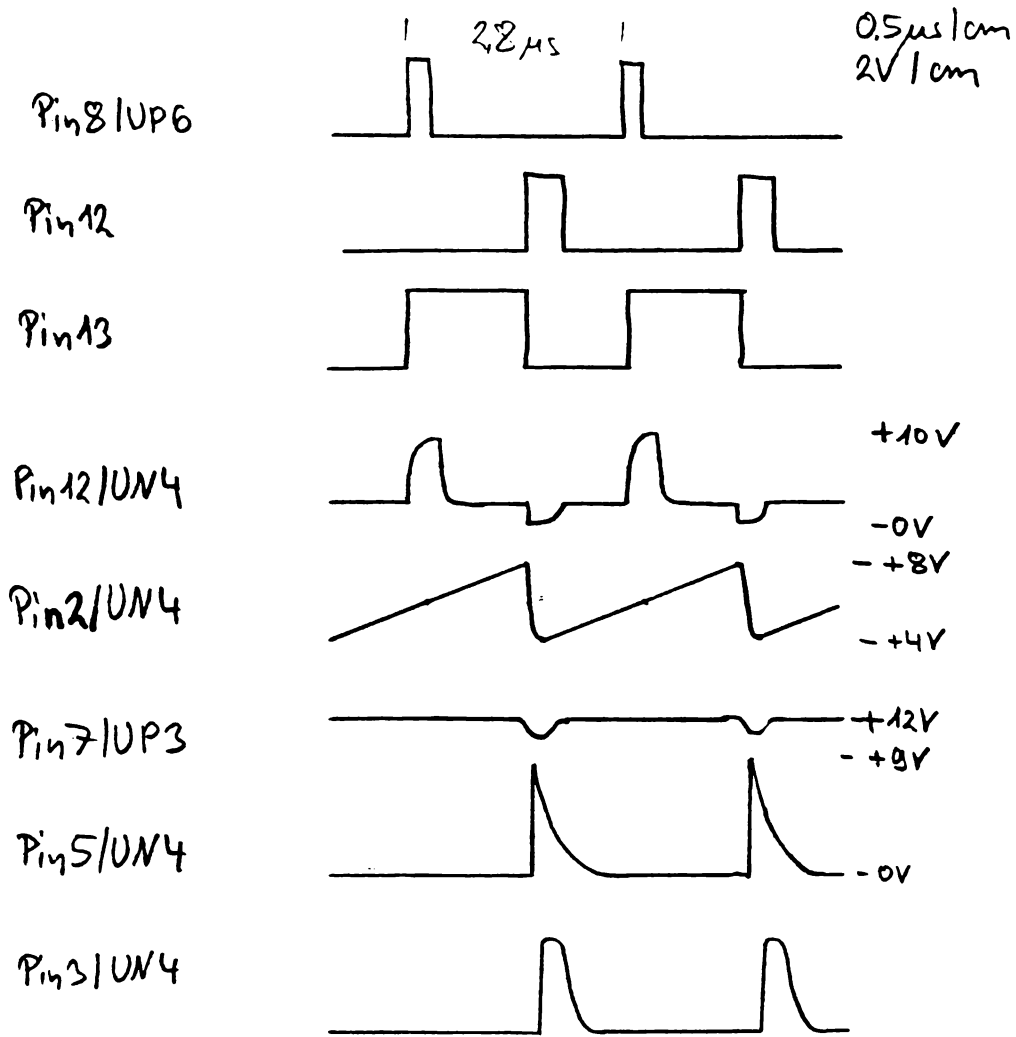
Step 10: Micropolis-Justage-Prüfung

Die Diskette wird initialisiert, beschrieben, die geschriebenen Daten werden wieder gelesen und gelöscht.



Signale auf dem Analogboard

Signale im PLL-Kreis, gemessen beim PLLSET



Spur	UP6 / Pin 6 DS 0	UP6 / Pin 3 DS 1	Zone	t [µs] (UN4 / Pin 3)
1-39	1	1	außen	2,166
40-53	0	1	↓	2,333
54-64	1	0		2,500
65-77	0	0	innen	2,666

8050 DOS 2.5POWER UP DIAGNOSTICS

Number of flashes	Error cause	Component	Location
1	ZERO PAGE	6532	E1, C1
2	ROM \$E000-FFFF	2364	H1
3	ROM \$C000-DFFF	2364	L1
4	RAM \$1000-13FF	2114	C4, C5
5	RAM \$2000-23FF	2114	D4, D5
	CONTROLLER	6502	H3
	ZEROPAGE \$00-3F	6530	K3
	ROM \$FC00-FFFF		
6	RAM \$3000-33FF	2114	E4, E5
7	RAM \$4000-43FF	2114	F4, F5

810815

Knud Hansen



This was brought to you

from the archives of

<http://retro-commadore.eu>