

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter:	Page:
Level: 2	Author: Stotz		Date: 05.03.80	

B E S C H R E I B U N G
S c h n i t t s t e l l e
F l o p p y - T r e i b e r



Helmut Wiertalla
Abteilungsleiter Software

Karlsruhe, den 05. März 1980
S t o t z

sks
STEINMETZ, KRISCHKE
SYSTEMTECHNIK – GMBH
Maybachstraße 10
7500 Karlsruhe 41
Telefon 07 21 / 40 30 76/77/78
Telex : 7 826 934 sksg-d

Programmier: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter:	Page:
Level: 2	Author: Stotz		Date: 05.03.80	

- 0. Einleitung
- 1. Schnittstellen-Übersicht
 - 1.1 Klassifikation
 - 1.2 Aufbau des Treibereingangs
 - 1.3 Zwischenrückgabe
 - 1.4 Aufrufparameter
 - 1.5 Rückgabeparameter
- 2. Beschreibung der Treiberfunktionen
- 3. Datenorganisation auf einer Diskette

- 3.1 Gliederung in Spuren und Sektoren
- 3.2 Form eines Sektors
- 3.3 Besonderheiten des Blockweisen Betriebs



STEINMETZ, KRISCHKE
SYSTEMTECHNIK - GMBH
Maybachstraße 10
7500 Karlsruhe 41
Telefon 07 21 / 40 30 76/77/78
Telex : 7 826 934 sksg-d

Helmut Wiertalla
Abteilungsleiter Software



Programmierer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 0	Page: 1
Level: 2	Author: Stotz		Date: 05.03.80	

0. Einleitung

Der Floppy-Treiber ist ein Kanalprogramm zum Betrieb von maximal drei Doppelkopf-Laufwerken am SKS-MC80 System mit MOS Betriebssystem.

Die Steuerung des Treibers erfolgt über Aufruf- und Rückgabeparameter, welche in den Registern und Flags übergeben werden.

Die Parameterstruktur wurde weitgehend an die MC-4 Kanalschnittstellen angepaßt.

Daraus ergibt sich eine Einteilung der Treiberfunktionen in IO- und CTRL-Befehle.

Der Floppy-Treiber ist nicht reentrant und normalerweise nicht unterbrechbar. Bei langdauernden Operationen unter Hardware-Kontrolle hat der Anwender die Möglichkeit der Zwischenrückgabe, um eigene Programme zwischenzeitlich abzuarbeiten.



Helmut Wiertalla
Abteilungsleiter Software

SKS
STEINMETZ, KRISCHKE
SYSTEMTECHNIK – GMBH
Maybachstraße 10
7500 Karlsruhe 41

Telefon 07 21 / 40 30 76/77/78
Telex : 7 826 934 sksg-d

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 1	Page: 1
Level: 2	Author: Stotz		Date: 05.03.80	

1. Übersicht über Funktion und Aufbau des Treibers

1.1 Treiberfunktionen

Folgende Funktionen lassen sich über den Treiber abwickeln

- Blockweise Lesen und Schreiben mit und ohne Wiederholungsvorgaben in Fehlerfällen
- Sektorweise Lesen und Schreiben im quasi - wahlfreien Zugriff zu Segmenten von 128 Bytes Länge (256 Bytes bei Double Density).
- Schreib/Lese Kopfpositionierung spur-und sektorweise
- Initialisierung des Interfacekontrollers und der Laufwerke
- Kennzeichnung von gelöschten oder geschützten Datenbereichen.
- Block- oder fileorientierte Positionierung.

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 1	Page: 2
Level: 2	Author: Stotz		Date: 05.03.80	

1.2 Treibereingänge

Es existieren zwei Haupteingänge

1. Select Entry (SENTRY):
Laufwerksnummer und Diskettenseite werden in einem Parameter explizit angegeben.
2. Direct Entry (DENTRY):
Laufwerksnummer und Diskettenseite ändern sich gegenüber dem vorhergehenden Aufruf nicht.

Bei Aufruf des Treibers erfolgt die Funktionsanwahl über eine Funktionsnummer. Diese wird den Haupteingängen im Akku übergeben.

Die häufig angewendeten Funktionen können auch über einen Nebeneingang aufgerufen werden. Dabei wird die Funktionsnummer der dem Nebeneingang zugeordneten Funktion vom Treiber selbst erzeugt.

Die Nebeneingänge sind:

- LOAD: Schreib-Lesekopf auf Diskettenanfang (Spur 0, Sektor 1) positionieren.
- READ: Block lesen
- WRITE: Block schreiben
- MOVE: N Filemarken vorsetzen.

Die Adressen der Eingänge liegen relativ zu DENTRY:

DENTRY+0, LOAD+3, READ+8, WRITE+13, MOVE+18, SENTRY+20

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 1	Page: 3
Level: 2	Author: Stotz		Date: 05.03.80	

1.3 Zwischenrückgabe Option

Einige Funktionen des Treibers dauern gegenüber den Programm-
laufzeiten lang und laufen ausschließlich unter Hardware- und
Interface Kontrolle (überwiegend Positioniervorgänge).

In diesen Fällen wird der Treiber über eine Zwischenrückgabe
verlassen.

Der Anwender hat dann die Möglichkeit, andere Programme abzuar-
beiten (Hintergrundprogramme, Interruptserviceroutinen usw.).

Alle Register und Flags stehen zur Verfügung. Der Prozessor be-
ginnt mit gesperrtem Interrupt.

Keine der Hardware-Funktionen ist zeitkritisch. Es liegt im Be-
lieben des Anwenders, wie lange das Zwischenprogramm dauert.

Die Adresse des Zwischenprogramms muß mindestens einmal, bei
Initialisierung des Treibers, übergeben werden, kann aber be-
liebig oft durch Neuinitialisierung geändert werden. Die Rück-
kehr in den Treiber erfolgt mit einem Return-Befehl (Rückkehr-
adresse liegt im Stack!).

Programmier: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 1	Page: 3
Level: 2	Author: Stotz		Date: 05.03.80	

1.3 Zwischenrückgabe Option

Einige Funktionen des Treibers dauern gegenüber den Programm-
laufzeiten lang und laufen ausschließlich unter Hardware- und
Interface Kontrolle (überwiegend Positioniervorgänge).

In diesen Fällen wird der Treiber über eine Zwischenrückgabe
verlassen.

Der Anwender hat dann die Möglichkeit, andere Programme abzuar-
beiten (Hintergrundprogramme, Interruptserviceroutinen usw.).

Alle Register und Flags stehen zur Verfügung. Der Prozessor be-
ginnt mit gesperrtem Interrupt.

Keine der Hardware-Funktionen ist zeitkritisch. Es liegt im Be-
lieben des Anwenders, wie lange das Zwischenprogramm dauert.

Die Adresse des Zwischenprogramms muß mindestens einmal, bei
Initialisierung des Treibers, übergeben werden, kann aber be-
liebig oft durch Neuinitialisierung geändert werden. Die Rück-
kehr in den Treiber erfolgt mit einem Return-Befehl (Rückkehr-
adresse liegt im Stack!).

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 1	Page: 4
Level: 2	Author: Stotz		Date: 05.03.80	

1.4 Allgemeine Aufrufparameter

Die Parameterübergabe bei Treiberaufruf geschieht funktionspezifisch.

Die folgenden Parameter sind jedoch bei allen Funktionen gemeinsam und müssen in jedem Fall übergeben werden:

Eingang SENTRY: Funktionsnummer in AC
Laufwerk, Seite in RL

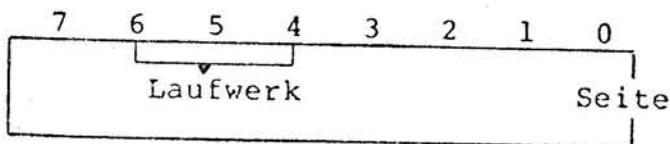
Eingang DENTRY: Funktionsnummer in AC
(Laufwerk, Seite wie bisher)

Nebeneingänge:

LOAD	:	Laufwerk/Seite in RL
READ	:	" " " "
WRITE	:	" " " "
MOVE	:	" " " "

Die Funktionsnummern sind in Tabelle I zusammengefaßt.

Der Parameter Laufwerk/Seite in RL hat die Form



Die funktionspezifischen Parameter werden bei der Beschreibung der einzelnen Funktionen behandelt.

Programmier: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 1	Page: 5
Level: 2	Author: Stotz		Date: 05.03.80	

1.5 Allgemeine Rückgabeparameter

Folgende Parameter werden von allen Funktionen zurückgegeben:

(Ausnahme: Funktion CTRL 3 siehe dort)

Frühwarnung: ZERO = 1

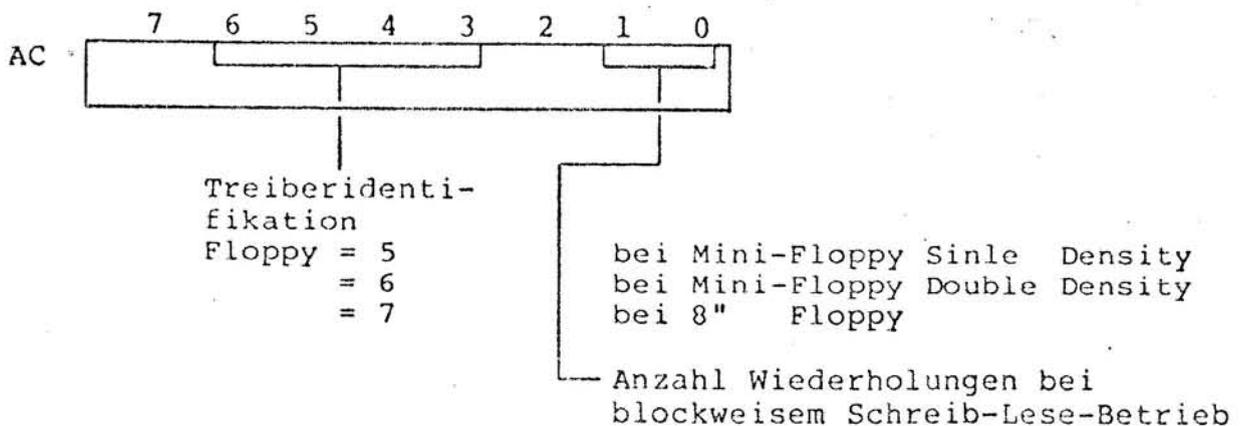
Fehler: CARRY = 1

Status: in RB/RC (siehe Tabelle II)

Identifikation/Wiederholung in AC

Alle Register und Flags, außer RH/RL werden verändert.

Der Parameter Identifikation/Wiederholung hat die Form:



Anmerkung: Die Funktionen "Treiberstatus lesen" (CTRL F) und Adressmarken einstellen (CTRL 3) bilden bezüglich Aufruf- und Rückgabeparameter Ausnahmen. Siehe Funktionsbeschreibungen.

Befehlsklasse IO		Befehlsklasse CTRL	
Nr.	Bit 2 ⁷ = 1	Nr	Bit 2 ⁷ = 0
80	Blockweise Lesen ohne Wiederholung	-	
81	Blockweise Schreiben Wiederholung	01	Initialisieren, Laufwerk/ Spur selektieren
82	Sektorweise Lesen bis zur angegebenen Pufferlänge	-	
83	Sektorweise Schreiben bis zur angegebenen Pufferlänge	03	Adressmarke einstellen
84	Positionieren auf Spur= TN; Sektor = SN (nur für formatierte Disk)	-	
-		05	N Filemarken vorsetzen
86	Wie 80, aber mit 3-maliger Wiederholung bei Lesefehlern	-	
87	Wie 81, aber mit 3-maliger Wiederholung bei Schreib- oder Prüfllesefehlern	07	N Blöcke vorsetzen
		08	N Blöcke rücksetzen
		0A	Schreib-Lesekopf über Spur 0, Sektor 1 positionieren (BOD)
		0C	Positionieren über Spur = TN, Sektor = 1 (für unformatierte Disketten)
		0F	Treiberstatus lesen

Tabelle I: Funktionsnummern der Floppy Treiber

Programmier: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 1	Page: 7
Level: 2	Author: Stotz		Date: 05.03.80	

Reg. Bit	Bedeutung	Flag
7	S/L-Kopf steht über letzter Spur, letzter Sektor	
6	S/L-Kopf steht über erster Spur, erster Sektor	
5	Gelöschte Daten gefunden	
4*	Laufwerk Interface unklar, Diskette fehlt	
3*	Treiberstatus aus "Deleted Data" gesetzt	
2*	Schreibschutz	
1	-	
0	-	
7	CRC - Fehler	SIGN
6	Frühwarnung	ZERO
5	Pufferüberlauf	
4	Sektor nicht gefunden	
3	-	
2	Filemarke gefunden	PARITY
1	Unzulässiger Vorgang oder Para- meter	
0	Fehlersummenmeldung (Anforderung nicht vollständig ausgeführt)	CARRY

Tabelle II Zusammenfassung der Statusrückmeldungen.

* siehe Funktion Treiberstatus lesen.

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 2	Page: 1
Level: 2	Author: Stotz		Date: 05.03.80	

2. Beschreibung der Treiberfunktionen

2.1 Funktion Block-Lesen (80)

Es wird ab momentaner Position bis Blockende gelesen. Ist der Block länger als der vorgegebene Puffer, so wird Pufferüberlauf gemeldet (Bit 2⁵ in RC = 1). Nach dem Lesen steht der Schreib-Lesekopf hinter dem gelesenen Block.

Aufruf:	Pufferadresse	RB/RC
	max. Pufferlänge	RD/RE
Rückgabe:	Anzahl gelesener Bytes	RD/RE
	Filemark gelesen	PARITY = 1
	Lesefehler	SIGN = 1, CARRY = 1

2.2 Funktion Block-Schreiben (81)

Es wird ab momentaner Position bis Pufferende ein Block geschrieben. Bei Blocklänge 0 wird ein Block mit 0 Datenbytes, d. h. eine Filemarke geschrieben. Nach dem Schreiben steht der Schreib-Lesekopf hinter dem gelesenen Block.

Aufruf:	Pufferadresse	RB/RC
	Anzahl Datenbytes	RD/RE
Rückgabe:	Position Sektor/Spur	RD/RE
	Schreibfehler	SIGN = 1, CARRY = 1
	Frühwarnung	ZERO = 1

Programmierer:		FLOPPY-TREIBER SCHNITTSTELLE	Chapter:	Page:
Stotz			2	2
Level:	Author:		Date:	
2	Stotz	05.03.80		

2.3 Funktion Sektorweise Lesen (82)

Es wird ab der momentanen Position die im Aufruf angegebene Anzahl Datenbytes gelesen. Der Schreib-Lesekopf steht nach dem Lesen auf dem nächsten Sektoranfang.

Aufruf:	Pufferadresse	RB/RC
	Anzahl Datenbytes	RD/RE
Rückgabe:	Position Sektor/Spur	RD/RE
	Lesefehler	SIGN = 1, CARRY = 1

2.4 Funktion Sektorweise Schreiben (83)

Es wird ab momentaner Position die im Aufruf angegebene Anzahl Datenbytes geschrieben. Der Schreib-Lesekopf steht nach dem Schreiben auf dem nächsten Sektoranfang.

Aufruf:	Pufferadresse	RB/RC
	Anzahl Datenbytes	RD/RC
Rückgabe:	Position Sektor/Spur	RD/RE
	Schreibfehler	SIGN = 1, CARRY = 1

Programmier: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 2	Page: 3
Level: 2	Author: Stotz		Date: 05.03.80	

2.5 Funktion absolutes Positionieren auf Spur und Sektor (84)

Der Schreib-Lesekopf wird auf die angegebene Position gebracht.

Aufruf:	Sektor	RD
	Spur	RE
Rückgabe:	Position Sektor/Spur	RD/RE

2.6 Funktion Block-Lesen mit Wiederholung (86)

Wie 2.2, jedoch mit 3 Wiederholungen im Fehlerfalle.

2.7 Funktion Block-Schreiben mit Wiederholung (87)

Wie 2.3, jedoch mit 3 Wiederholungen im Fehlerfalle.

2.8 Funktion Initialisieren, Laufwerk/Spur selektieren (01)

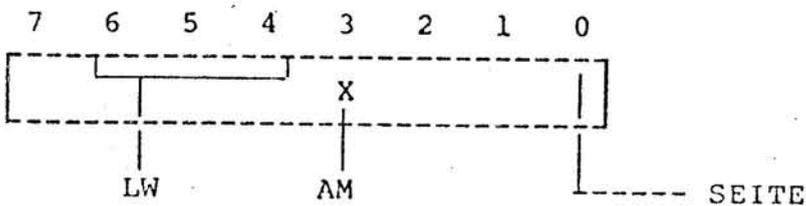
Es wird die Zwischenrückgabeadresse eingestellt und an Laufwerk und Spur selektiert.

Aufruf:	Adresse eines Zwischenprogrammes	RB/RC
	Laufwerk/Seite (Belegung siehe 1.4)	RE

Rückgabe: siehe Allgemeine Rückgabeparameter (1.5).

2.9 Funktion Adressmarke einstellen (03)

Es besteht die Möglichkeit, Daten mit verschiedenen Adressmarken auf die Floppy zu schreiben. Welche Adressmarke verwendet wird, kann durch diese Funktion bestimmt werden. Dazu dient das Bit 2^3 in RL.



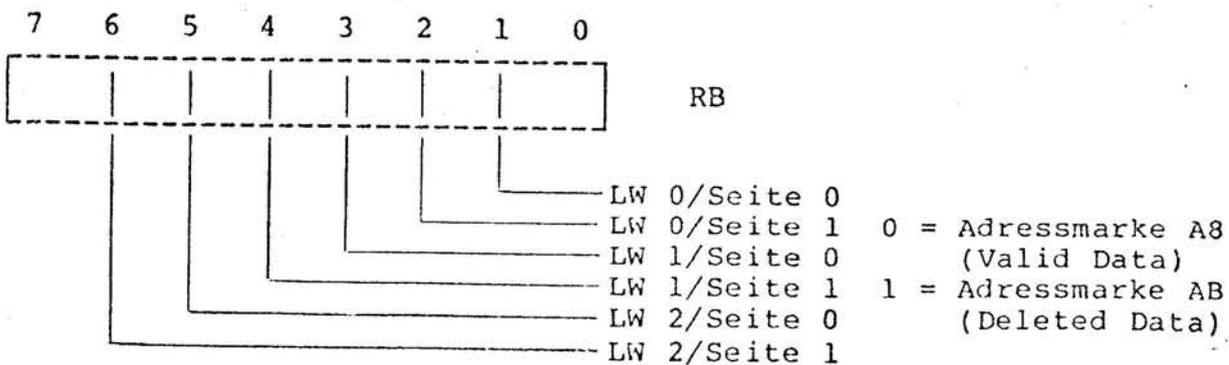
Bit $2^3 = 0$ Adressmarke = A8 = "Valid Data"

Bit $2^3 = 1$ Adressmarke = AB = "Deleted Data"

Die Funktion 03 muß über den Eingang SENTRY aufgerufen werden.

Rückgabe: Es gelten nicht die allgemeinen Rückgabeparameter

Es wird in RB eine Übersicht über die Adressmarken aller Laufwerke/Seiten zurückgegeben.



Werden Daten mit der Adressmarke AB (Deleted Data) gelesen, so wird im Treiberstatus das Bit 2^5 in RB gesetzt.

Programmier: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 2	Page: 5
Level: 2	Author: Stotz		Date: 05.03.80	

2.10 Funktion N Filemarken vorsetzen (05)

Es wird ab momentaner Position um die gegebene Anzahl Filemarken vorgesezt. Der Schreib-Lesekopf steht nach der Operation hinter der letzten Filemarke.

Aufruf: Anzahl Filemarken RE

Rückgabe: siehe Allgemeine Rückgabeparameter (1.5).

2.11 Funktion N Blöcke vorsetzen (07)

Es wird ab momentaner Position um gegebene Anzahl von Blöcken vorgesezt.

Aufruf: Anzahl Blöcke RE

Rückgabe: siehe Allgemeine Rückgabeparameter (1.5).

2.12 Funktion N Blöcke zurücksetzen (08)

Es wird ab momentaner Position um gegebene Anzahl von Blöcken zurückgesezt.

Aufruf: Anzahl Blöcke RE

Rückgabe: siehe Allgemeine Rückgabeparameter (1.5).

Programmier: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 2	Page: 6
Level: 2	Author: Stotz		Date: 05.03.80	

2.13 Funktion Restore (0A)

Der Schreib-Lesekopf wird auf Spur 0, Sektor 1 positioniert.

Aufruf: siehe Allgemeine Aufrufparameter (1.4).

Rückgabe: siehe Allgemeine Rückgabeparameter (1.5).

2.14 Funktion Positionieren-Spur (0C)

Es wird die angegebene Spur anpositioniert und das Sektorregister mit 1 geladen. Diese Funktion kann zum Positionieren unformatierter Disketten verwendet werden.

Aufruf: Spurnummer in RE

Rückgabe: siehe allgemeine Rückgabeparameter (1.5)

2.15 Funktion Treiberstatus lesen (0F)

Es wird der momentane Treiberstatus gelesen. Dabei werden nur die in Tabelle II mit * gekennzeichneten Bits zurückgemeldet.

Aufruf: siehe allgemeine Aufrufparameter (1.4)

Rückgabe: RB (siehe Tabelle II)

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 3	Page: 1
Level: 2	Author: Stotz		Date: 05.03.80	

3. Datenorganisation auf einer Diskette

3.1 Gliederung in Spuren und Sektoren

Eine Diskette hat 40 (77) Spuren zu je 16 (26) Sektoren.

1 Sektor hat eine Länge von 128 (256) Bytes.

Die Nummerierung der Spuren beginnt bei 0, die der Sektoren bei 1.

Alle Ein/Ausgabeoperationen beginnen an der Sektorkante, die durch einen vorangegangenen IO- oder Steuerbefehl erreicht wurde.

3.2 Die Form eines Sektors

Identifizier Marke	TN	SN	CRC	GAP	DA Mark	DA Field	CRC	GAP					
6x00	1xFE	1	00	1	00	2	FF	6x00	Z	128	Byte	Byte	FF
	1x	1x	1x	1x	1x	1x	2	27x					
	Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte

TN = Tracknumber Z = FB Data valid

SN = Sektornummer = F8 Data deleted

Obiges Format entspricht dem IBM-Typ elektronisch sektoriert.

Bei Double Density Laufwerken erhöht sich die Länge des Datenfeldes auf 256 Bytes.

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: 3	Page: 2
Level: 2	Author: Stotz		Date: 05.03.80	

3.3 Besonderheiten des Blockweisen Betriebs

Ein Block beginnt immer an einer Sektorkante, d.h. mit dem 1. Byte des Datenfeldes. Jedoch gehört dieses Byte noch nicht zur eigentlichen Blockinformation, sondern bildet den sogenannten Längenschlüssel LS. In ihm wird dargestellt, wieviele Bytes des Sektors noch zum Block gehören. Ist der LS größer als 127 (254 bei DD) bedeutet das, daß der Block länger als ein Sektor ist und im Folgesektor fortzusetzen ist.

Ein Blockende ist demnach an einem Längenschlüssel ≤ 127 (≤ 254 bei DD) zu erkennen.

Vereinbarungsgemäß ist eine Filemarke ein Block mit der Länge 0 Datenbytes, d.h. der LS (1.Byte im Sektor) ist 0.

Die Kapazität eines Sektors im blockweisen Betrieb beträgt also 127 Byte (254 Byte bei DD). Dies ist bei Kapazitätsberechnungen zu berücksichtigen.

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter:	Page:
Level: 2	Author: Stotz		A	1
			Date:	05.03.80

A n h a n g

Beschreibung des Floppy - Zusatzprogrammes
für Formatieren.

Programmier: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter: A	Page: 2
Level: 2	Author: Stotz		Date: 05.03.80	

Das Zusatzmodul ist nicht Bestandteil des Treibers im PROM, sondern wird in den RAM - Bereich dazugeladen.
Das Modul ist verschieblich.
Die Schnittstelle ist identisch mit der des Treibers.
Dieses Programm wird für die Formatier Routinen verwendet.
Folgende Routinen sind möglich:

A 1.0 Funktion Hardware Identifizieren (0E)

Es wird die vorhandene Hardware identifiziert und das Programm dafür eingestellt.

Aufruf : siehe allgemeine Aufrufparameter (1.4)

Rückgabe: siehe allgemeine Rückgabeparameter (1.5)

Dieser Aufruf muß vor einem Formatieraufruf erfolgen.

A 2.0 Funktion Spur-Formatieren (0B)

Es wird die angegebene Spur neu formatiert. Dabei ist die Sektorfolge auf der Diskette fortlaufend (Sequenz 01), also 1, 2, 3, ... 15, 16...

Aufruf : Spurnummer RE

Rückgabe: siehe allgemeine Rückgabeparameter (1.5).

Programmier: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter:	Page:
Level: 2	Author: Stotz		A	3
			Date:	05.03.80

A 3.0 Funktion Formatieren mit wählbarer Sektorsequenz (09)

Es wird die ganze Floppy mit der angegebenen Sektorsequenz formatiert. Mit der Sequenz kann die physikalische Folge der Sektoren auf der Floppy geändert werden.

Es sind insgesamt 8 (13 bei 8" Floppy) verschiedene Sequenzen möglich:

Beispiel:

Sequenz = 1	Sektorfolge 1, 2, 3 ...
Sequenz = 2	Sektorfolge 1, 3, 5, 7 ...
·	
·	
·	
Sequenz = 8	Sektorfolge 1, 9, 2, 10, 3, ...
Aufruf :	Sequenz RD
Rückgabe :	siehe allgemeine Rückgabeparameter (1.5).

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter:	Page:
Level: 2	Author: Stotz		Date: 05.03.80	

Fehlerprotokoll

Fehlerbeschreibung:

Fehlerbehebung:

Neuer Programmname:

Neue Programmbibliothek:

Eintragung in das Programmfreigabeprotokoll:

Unterschrift

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter:	Page:
Level: 2	Author: Stotz		Date: 05.03.80	

Lfd. Nr.	Art der Änderung	Veranlassung	durchgeführt am , von

Programmer: Stotz		FLOPPY-TREIBER SCHNITTSTELLE	Chapter:	Page:
Level: 2	Author: Stotz		Date: 05.03.80	

Lfd. Nr.	Programmierung			Progr. Sprache	Programm oder Programmssystemname Name-Bildschirm-Datenträger Schreibdichte/Version z.B. AAAAAA-Z-AZZ/ZZ-ZZ
	Unterschrift	Datum			
1	Progr.:				
	Test:				
	Dokum.:	<i>Stotz</i>	21.08.79		
2	Progr.:				
	Test:				
	Dokum.:	<i>Stotz</i>	05.03.80		
	Progr.:				
	Test:				
	Dokum.:				
	Progr.:				
	Test:				
	Dokum.:				
					Anzahl Bandmarken

A = Assembler
B = Basic