

Rolf-Dieter Klein

Winchester-Anschluß

Für mc-CP/M-Computer und NDR-Klein-Computer, Teil 1

In mc 1985, Heft 2 wurde die Baugruppe FLO2 vorgestellt, die für den NDR-Klein-Computer und seinen engen Verwandten, den mc-CP/M-Computer, das Floppy-Problem endgültig löste. Offen blieb immer noch der Festplatten-Anschluß. Lesen Sie heute, wie man eine Festplatte an Z80-ECB-Systeme und damit an die Computer von Rolf-Dieter Klein anschließt.

In diesem Artikel ist die Rede von drei verschiedenen, aber zu einem System gehörenden Interface-Karten. Die FLO2-Karte ist das Floppy-Interface für den NDR-Klein-Computer, das auch an den mc-CP/M-Computer angeschlossen werden kann, wenn man einen EC-Bus-

Adapter verwendet. Die FLOSASI-Karte ist die kombinierte Interface-Karte für den mc-CP/M-Computer, die sowohl die Elektronik von FLO2 enthält als auch die Elektronik zur Steuerung eines Festplatten-Controllers über den SASI-Bus. Die dritte Platine ist eine Baugruppe mit

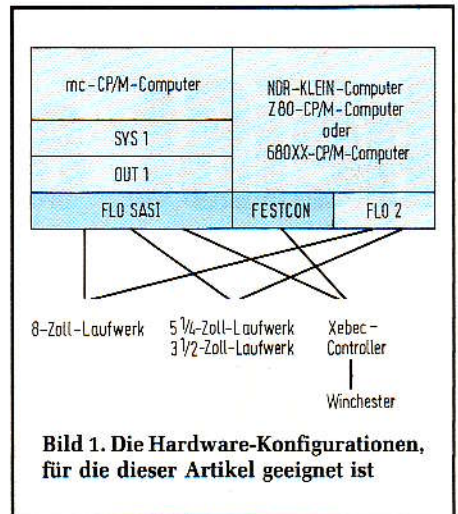


Bild 1. Die Hardware-Konfigurationen, für die dieser Artikel geeignet ist

dem Namen FESTCON. Sie enthält die „SASI-Elektronik“ aus der FLOSASI-Baugruppe einzeln und ist als Festplatten-Steuer-Interface für den NDR-Klein-Computer gedacht. Bild 1 zeigt den Zusammenhang schematisch. Bild 2 zeigt in einer Übersicht, welche Software-Pakete für welche Zusammenstellung an Baugruppen geeignet oder besser notwendig sind.

Wer schon den mc-CP/M-Computer mit FLO1 besitzt und den Monitor 3.4, der benötigt nur den SASI-Teil allein, wenn er die Winchester-Unterprogramme in seinen alten Monitor zusätzlich einbaut („Patchen“ nennt man das). Im neuen Monitor 1.1 für den mc-CP/M-Computer sind bereits die Winchester-Routinen enthalten. Es werden die Laufwerke BASF 6188 oder RO-352 unterstützt. Bei anderen Laufwerk-Typen muß man eine Tabelle im EPROM ändern, die die Laufwerksparameter enthält. Der NDR-Klein-Computer benutzt in der Z80-Version den Monitor FLOMON 1.5. Darin sind keine Winchester-Routinen enthalten, man muß sie mit einem Patch einfügen. Dabei können die Routinen automatisch nach Start des CP/M nachgeladen werden.

Der NDR-Klein-Computer und der mc-CP/M-Computer verwenden jetzt ein- und dasselbe BIOS. Beim CP/M68k, im Betrieb zusammen mit dem Grundprogramm 4.3, muß nur ein neues BIOS ins System integriert werden. In diesem BIOS befinden sich alle Winchester-Routinen.

Der SASI-Bus

Bild 3 zeigt das Verbindungsschema zum Winchester-Laufwerk. Die FLOSASI/FESTCON-Baugruppe ist über

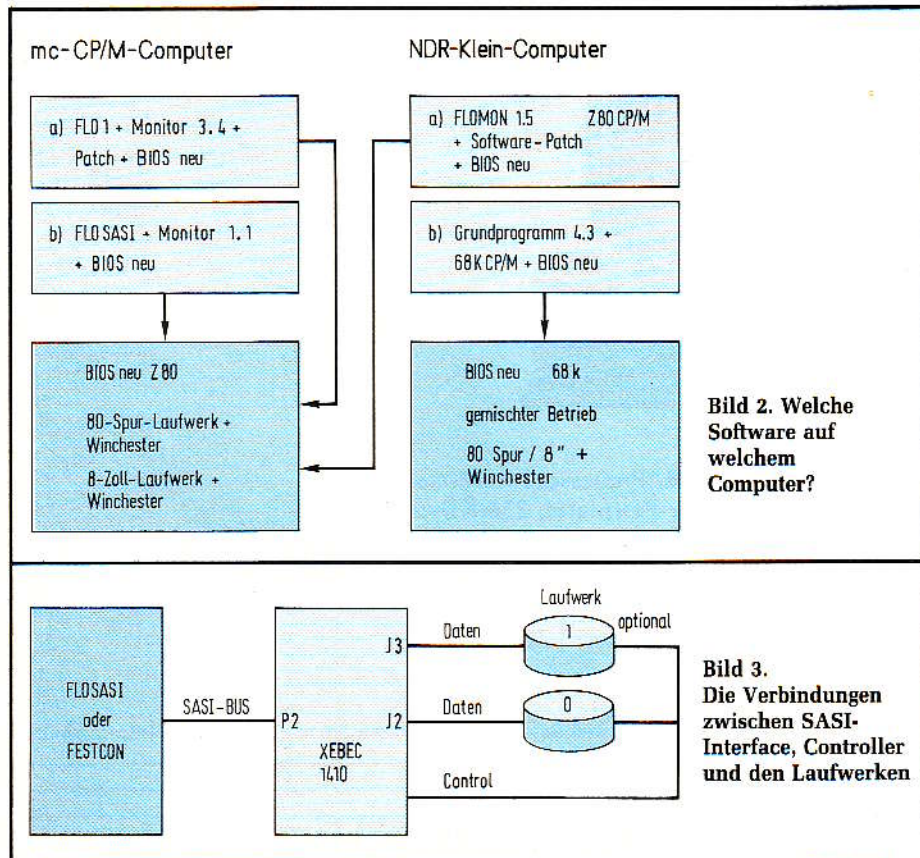


Bild 2. Welche Software auf welchem Computer?

Bild 3. Die Verbindungen zwischen SASI-Interface, Controller und den Laufwerken

ein 50poliges Kabel mit dem Winchester-Controller verbunden. Wir verwenden hier den Xebec-Controller 1410, der sogar einen eigenen Mikroprozessor verwendet. Dieser kann maximal zwei Winchester-Laufwerke betreiben, die über einen gemeinsamen, 34poligen Steuerbus und zusätzlich über je ein 20poliges Kabel, das die Schreib- und Leseleitungen beinhaltet, verbunden sind. Bild 4 zeigt die Schaltung des SASI-Interface. Die Baugruppe belegt vier Adressen, deren Bereich über eine Reihe von Brücken (ST1) eingestellt werden kann. Einen Bausatz gibt es bei der Firma Graf, Kempten. Der SASI-Bus selbst ist nichts weiter Aufregendes, er besteht aus einem Datenbus mit den Signalen $\overline{DB0}$ bis $\overline{DB7}$, die allerdings negiert sind, und ein paar Steuersignalen, die zur Kontrolle des Ablaufs dienen. Dazu gehören das Signal $\overline{C/D}$, das bestimmt, ob gerade Daten oder Befehle übertragen werden, das Si-

gnal $\overline{T/O}$, das die Übertragungsrichtung festlegt, das Signal \overline{BSY} , das angibt, wann der Controller aktiv ist, das Signal \overline{MSG} , das zur Erkennung des letzten Übertragungsbytes dient und das Signal \overline{REQ} , das bei jeder Byteübertragung aktiviert wird. Das Signal \overline{ACK} ist für die Rückmeldung an den Controller gedacht, und das Signal \overline{SEL} dient zur Adressierung eines einzelnen Controllers, denn es können bis zu 8 Controller gleichzeitig am SASI-Bus angeschlossen sein. Das Signal \overline{RST} dient als Rücksetzsignal für den Controller. Bild 5 zeigt die Belegung des SASI-Stekkers. Bild 6 zeigt die Belegung der Steuerleitungen vom Controller zur Winchester und Bild 7 die Belegung der Datenleitungen vom Controller zur Winchester. Bild 8 und Bild 9 zeigen das Layout und Bild 10 den Bestückungsplan der Baugruppe FLOSASI.

Das Timing

Ein Puls auf der \overline{RST} -Leitung setzt den Controller (bzw. alle Controller, die auf dem Bus liegen) in die Grundstellung. Bild 11 zeigt den Vorgang beim Selektieren des Controllers. Dazu wird die Controller-Adresse auf die Datenleitungen gelegt. Ein einzelnes Bit steht dabei für einen Controller. Bit 0 also für den ersten Controller. Ist es gesetzt, liegen also auf dem SASI-Bus 0 V bei Bit 0, so wird der Controller mit der ersten Adresse selektiert. Dazu muß aber ein weiteres Signal, \overline{SEL} , hinzugefügt werden. Auch dort genügt ein kurzer Impuls. Daraufhin wird die \overline{BUSY} -Leitung auf 0 V gehen und der Controller bleibt nun so lange aktiv, bis er einen Befehl ausgeführt hat. Bild 12 zeigt den Vorgang beim Lesen von Daten oder Statusinformationen, die vom Controller geliefert werden. Dabei werden zuerst die Signale $\overline{T/O}$ und $\overline{C/D}$

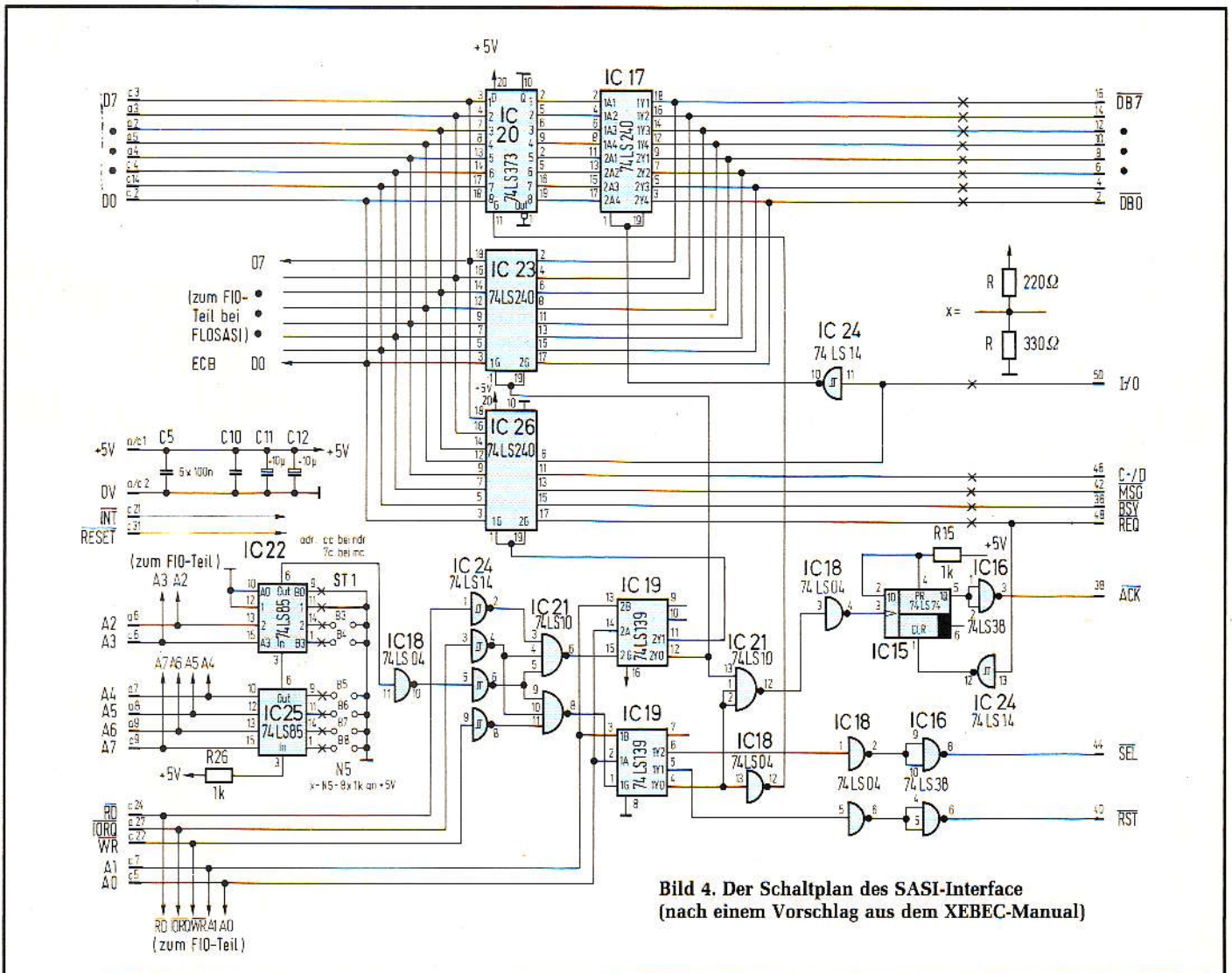


Bild 4. Der Schaltplan des SASI-Interface (nach einem Vorschlag aus dem XEBEC-Manual)

Signal Pin	Ground Return	Signal Name
2	1	DATA0-
4	3	DATA1-
6	5	DATA2-
8	7	DATA3-
10	9	DATA4-
12	11	DATA5-
14	13	DATA6-
16	15	DATA7-
18	17	Spare
20	19	Spare
22	21	Spare
24	23	Spare
26	25	Reserved
28	27	Spare
30	29	Spare
32	31	Spare
34	33	Spare
36	35	BUSY-
38	37	ACK-
40	39	RST-
42	41	MSG-
44	43	SEL-
46	45	C-/D
48	47	REQ-
50	49	I-/O

Bild 5.
Der SASI-Bus

CONNECTOR J1, CONTROL SIGNALS, PIN ASSIGNMENTS

Signal Pin	Ground Return	Signal Name
2	1	Reduced Write Current/ Head Select 2 ³ /-Change Cartridge
4	3	Head Select 2 ²
6	5	Write Select Gate
8	7	Seek Complete
10	9	Track 00
12	11	Write Fault
14	13	Head Select 2 ⁰
16	15	Spare / -Sector
18	17	Head Select 2 ¹
20	19	Index
22	21	Ready
24	23	Step
26	25	Drive Select 1
28	27	Drive Select 2
30	29	Reserved
32	31	Reserved
34	33	Direction In

Bild 6. Die Steuerleitungen der ST-506

gültig. Dann wird der Datenbus stabil und erst jetzt wird das Signal $\overline{\text{REQ}}$ auf 0 V gelegt. Jetzt kann man die Daten lesen. Danach wird dem Controller mit Anlegen von 0 V an das Signal $\overline{\text{ACK}}$ mitgeteilt, daß die Daten übernommen wurden. Daraufhin wird das Signal $\overline{\text{REQ}}$ weggenommen und dann nimmt man selbst (also der Computer) das Signal $\overline{\text{ACK}}$ wieder weg. Die Erzeugung des $\overline{\text{ACK}}$ -Signales geschieht bei unserer Schaltung automatisch. Mit der steigenden Flanke eines Lese- oder Schreibsignales ($\overline{\text{RD}}$ oder $\overline{\text{WR}}$) wird $\overline{\text{ACK}}$ aktiviert und durch ein $\overline{\text{REQ}}$ -Signal, das auf 5 V liegt, wieder weggenommen. Das $\overline{\text{REQ}}$ -Signal kann man über einen Datenport abfragen (wie auch die Signale $\overline{\text{C/D}}$ usw.).

Bild 13 zeigt den Vorgang beim Schreiben. Hier zeigt das Signal $\overline{\text{REQ}}$ an, daß Daten vom Controller erwartet werden. Dann muß der Computer die Daten auf den Datenbus legen und danach durch $\overline{\text{ACK}}$ dem Controller mitteilen, daß die Daten gültig sind. Die Übertragung von Befehlen läuft mit demselben Timing, die Unterscheidung, ob Befehle oder Daten übertragen werden, liefert das Signal $\overline{\text{C/D}}$, das vom Controller erzeugt wird. Der verwendete Xebec-Controller ist übrigens in der Lage, die Daten sehr

schnell zu übertragen, so daß man bei Z80 den $\overline{\text{OTIR}}$ - oder $\overline{\text{INIR}}$ -Befehl verwenden kann, und das $\overline{\text{REQ}}$ bei der Übertragung nicht abwarten muß. Umgekehrt kann die CPU aber auch beliebig langsam sein, sie muß nicht mit der Datenrate der Winchester mithalten können, denn ein gelesener Sektor wird immer im Controller zwischengespeichert und dann asynchron übertragen.

Aufbau und Test der Schaltung

Die Schaltung selbst ist einfach, es werden nur normale TTL-Bausteine verwendet. Beim Anschluß des Controllers und der Winchester muß man sorgfältig alle Kabelanschlüsse prüfen. Achtung: Man sollte dazu unbedingt die Handbücher von Winchester und Controller zu Rate ziehen. Der Controller benötigt übrigens

CONNECTORS J2 and J3, SIGNALS, PIN ASSIGNMENTS

Signal Pin	Ground Return	Signal Name
1	2	Drive Selected
3	4	Spare / -Recalibrate
5	6	Spare / -Write protected
7	8	Reserved
9	10	Spare / -Cartridge Changed
11	12	Ground (GND)
13		MFM Write Data+
14		MFM Write Data-
15	16	Ground (GND)
17		MFM Read Data+
18		MFM Read Data-
20	19	Ground (GND)

Bild 7. Die Datenleitungen der ST-506

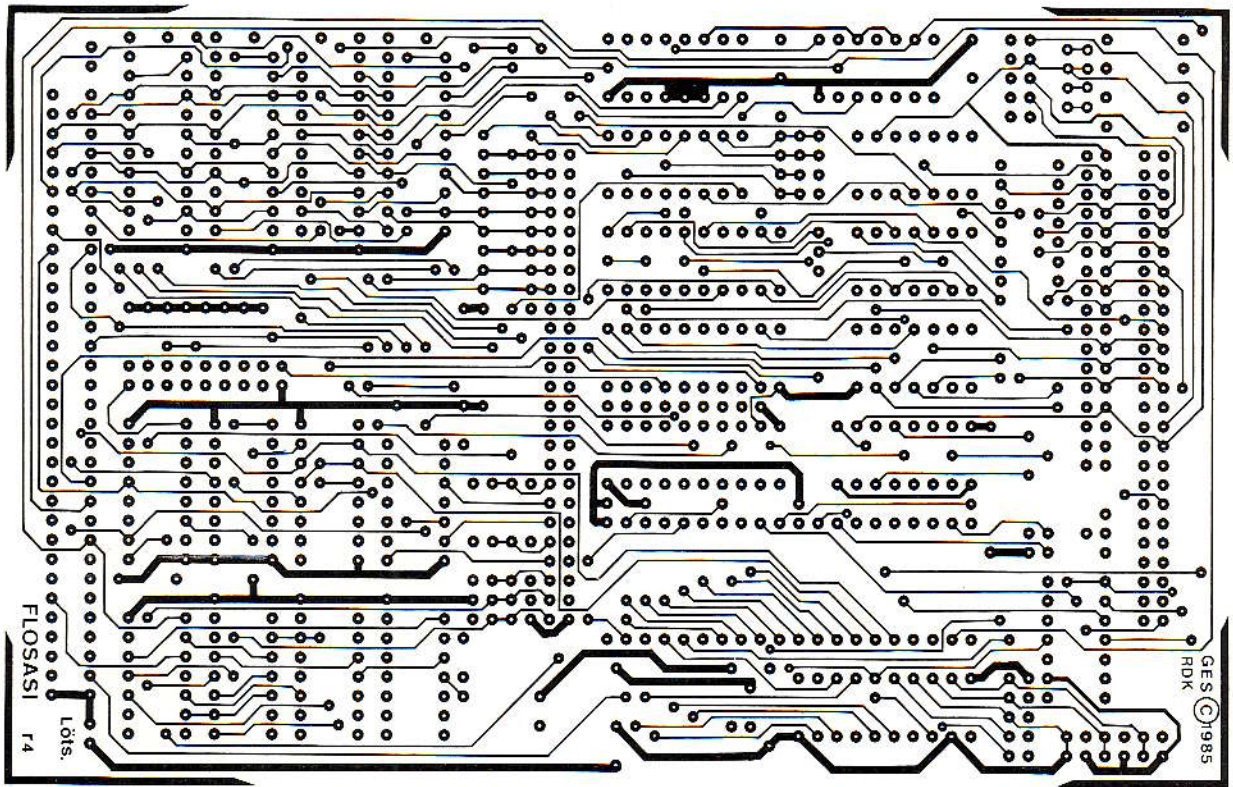


Bild 8. Layout FLOSASI, Lötseite

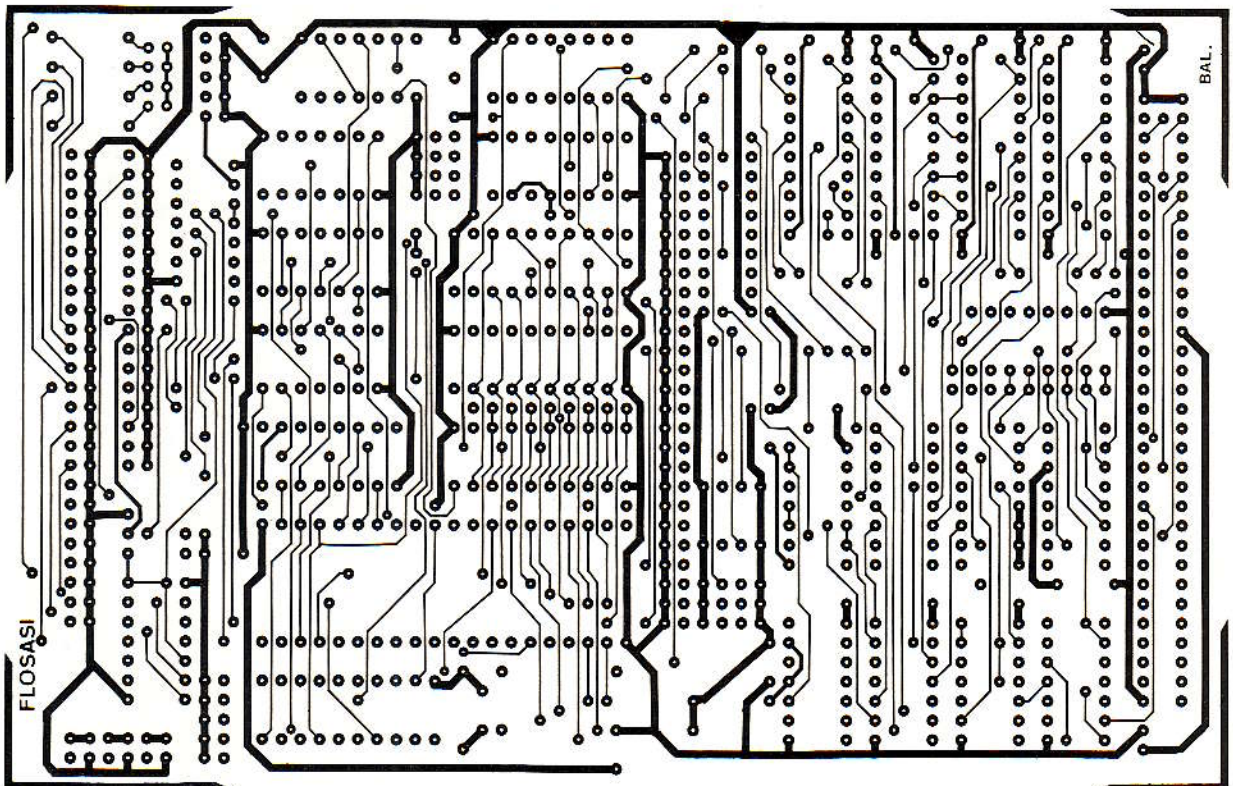


Bild 9. Layout FLOSASI, Bestückungsseite

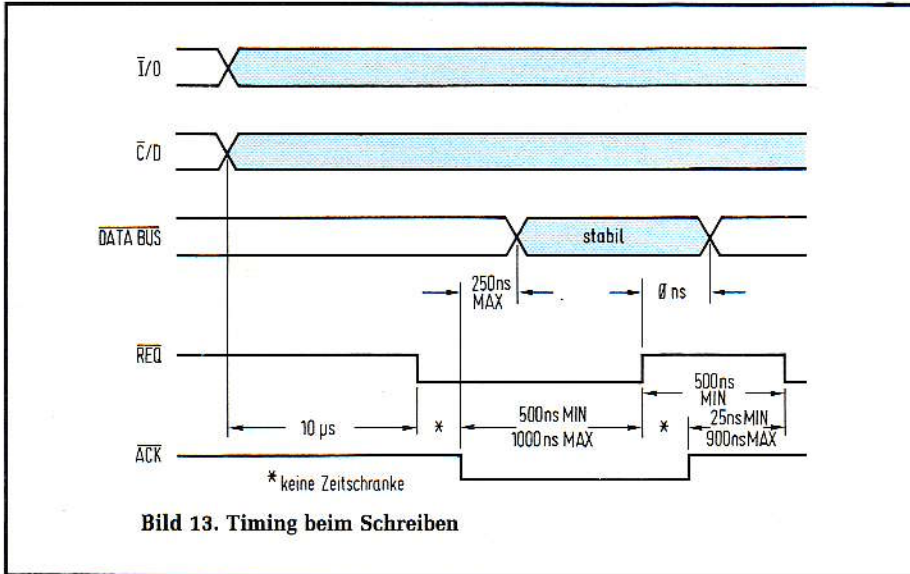


Bild 13. Timing beim Schreiben

Form einer Tabelle gleich am Anfang des Programms eingetragen. Die Tabelle findet man unter der Marke mit dem Namen DRVTBL.: Die ersten beiden Bytes bestimmen die Anzahl der Zylinder des Laufwerks. Dabei wird das höherwertige Byte zuerst abgelegt. In unserem Beispiel hat das Laufwerk 306 Zylinder. Danach gibt ein Byte die Anzahl der Schreib-Leseköpfe an. Hier sind es vier Köpfe. Maximal kann man 8 Köpfe angeben. Die Anzahl der Köpfe gibt, multipliziert mit der Anzahl der Zylinder, die Zahl der verfügba-

ren Spuren. Das Laufwerk soll mit 256 Byte pro Sektor formatiert werden (Achtung, Brücke am XEBEC-Controller nach Handbuch überprüfen), dann sind genau 32 Sektoren pro Spur möglich. Somit ergibt sich eine nutzbare Speicherkapazität von $256 \cdot 32 \cdot 4 \cdot 306 \text{ Byte} = 10,027008 \text{ MByte}$. Jetzt folgen zwei Bytes, die den Zylinder für den reduzierten Schreibstrom angeben. Hier wurde der Wert 128 gewählt. Diese Angaben erfährt man aus den technischen Handbüchern zum jeweiligen

Tabelle 1: Befehle des Xebec-1410A-Controller

Aufbau eines Befehls:							
7	6	5	4	3	2	1	0
--- Kommando Klasse ---		Gerät		----- Befehl -----			
x	x	----- Adresse MSBLSB -----		----- Adresse MSB -----			
----- Adresse MSBLSB -----		----- Adresse LSB -----		----- Interleave-Faktor, oder Blockzähler -----			
----- Adresse MSBLSB -----		----- Adresse LSB -----		----- Steuerfeld -----			

Steuerfeld:
 Bit 7:
 Wenn das Bit gesetzt ist, werden auftretende Fehler ohne Neuversuche direkt gemeldet.
 Bit 6:
 Wenn das Bit gesetzt ist, wird vor erneutem Leseversuch im Fehlerfall zuerst eine Korrektur versucht.
 Bit 5:
 nicht verwendet.
 Bit 4:
 Wenn gesetzt, so besitzt das Laufwerk eine Servo-Information auf jeder Spur.
 Bit 3-0:
 Dadurch wird die Steprate des Laufwerks bestimmt:
 0000 3 ms (default)
 0100 200 µs gepuffertes Schreiten
 0101 70 µs gepuffertes Schreiten
 0110 30 µs gepuffertes Schreiten
 0111 15 µs gepuffertes Schreiten
 1000 12 µs gepuffertes Schreiten

Tabelle 2: Stückliste (SASI-Teil)

IC 15 74LS74
IC 16 74LS38
IC 17 74LS240
IC 18 74LS04
IC 19 74LS139
IC 20 74LS373
IC 21 74LS10
IC 22 74LS85
IC 23 74LS240
IC 24 74LS14
IC 25 74LS85
IC 26 74LS240
5 × 14polige IC-Fassung
3 × 16polige IC-Fassung
4 × 20polige IC-Fassung
1 × 8poliger DIL-Schalter
1 × 220 Ω Widerstandsnetz × 8
1 × 330 Ω Widerstandsnetz × 8
1 × 1 kΩ Widerstandsnetz × 8
5 × 330 Ω
5 × 220 Ω
R15, R26 2 × 1kΩ
C5-C10 6 × 100 nF
C11, C12 2 × 10 mF Tantal
1 × 50polige doppelreihige Stiftleiste
1 × ECB (VG-Leiste) oder NDR-Stecker

Winchester-Laufwerk. Ebenfalls den Wert für den Zylinder, bei dem die Schreibprekompensation beginnt. Auch dieser Wert wird mit zwei Bytes abgelegt. Dabei muß immer der höherwertige Teil zuerst abgelegt werden. Die Anzahl der verfügbaren Sektoren wird mit einer EQU-Anweisung an den Namen SEKANZ zugewiesen. Damit kann das Programm später die Anzahl der Sektoren zur Lesekontrolle bestimmen. Der Wert wurde hier auf 65 535 begrenzt. Bei Laufwerken mit größerer Kapazität muß man das Programm leicht modifizieren. Ein Beispiel für ein größeres Laufwerk ist gleich hinter der Tabelle zu sehen. Der XEBEC-Controller verfügt über eine Vielzahl von Kommandos, die hier in der Tabelle 1 angegeben sind. Zunächst wird der Controller mit dem Unterprogramm WIINI zurückgesetzt. Dann werden die Werte der Laufwerkstabelle DRVTAB mit dem Kommando INIT (initlc) an den Controller übertragen. Danach wird mit dem Befehl RECALC das Laufwerk auf Spur 0 eingestellt. Im Hauptprogramm werden danach zwei Diagnose-Befehle ausgegeben, die den Controller testen. Danach erst wird formatiert. Das Formatieren geht eigentlich ganz einfach, denn es gibt einen Befehl dafür, der das ganze Laufwerk formatiert (das kann einige Minuten dauern). Danach wird mit einem besonderen Befehl ein Schnell-Prüfverfahren

Die Befehlsliste des Controllers

Klasse 0, Code 0:
Test Drive Ready. Damit kann man prüfen, ob das Laufwerk bereit ist.

Klasse 0, Code 1:
Recalibrate. Der Laufwerkskopf wird auf Spur 00 gefahren.

Klasse 0, Code 3:
Request Sense Status. Nach Auftreten eines Fehlercodes muß man diesen Befehl schicken, um die genauere Fehlerursache zu erfahren.

Klasse 0, Code 4:
Format Drive. Das Laufwerk wird formatiert. Dabei werden alle Daten gelöscht.

Klasse 0, Code 5:
Check Track Format. Damit kann die Formatierung geprüft werden. Daten werden dabei nicht gelesen.

Klasse 0, Code 6:
Format Track. Damit kann eine einzelne Spur formatiert werden.

Klasse 0, Code 7:
Format Bad Track. Eine Marke wird beim Formatieren gesetzt, die angibt, daß es sich um eine defekte Spur handelt.

Klasse 0, Code 8:
Read. Ein oder mehrere Sektoren werden gelesen.

Klasse 0, Code 9:
Read Verify. Arbeitet wie Read, nur daß keine Daten zum Hauptrechner übertragen werden. Damit kann das Datenfeld nach einem Formatier- oder Schreibvorgang geprüft werden.

Klasse 0, Code 0Ah:
Write. Ein oder mehrere Sektoren werden geschrieben. Die geschriebenen Daten werden nicht prüfgelesen.

Klasse 0, Code 0Bh:
Seek. Damit kann man eine Spur anfahren. Normalerweise wird das durch das Schreib- oder Lesekommando automatisch durchgeführt, hier hat man jedoch die Möglichkeit, z. B. bei mehreren Laufwerken, schon im voraus bei einem Laufwerk die Spur anzufahren, und bei anderen Daten zu lesen, um so Zeit zu sparen.

Klasse 0, Code 0Ch:
Initialize Drive Characteristics. In nachfolgenden 8 Parameter-Byte werden die Laufwerkdaten übertragen.
Byte 0: MSB maximale Zylinderzahl
Byte 1: LSB maximale Zylinderzahl
Byte 2: Anzahl der Schreib-Leseköpfe
Byte 3: MSB Zylinder für reduzierten Schreibstrom
Byte 4: LSB Zylinder für reduzierten Schreibstrom
Byte 5: MSB Zylinder für Präcompensation
Byte 6: LSB Zylinder für Präcompensation
Byte 7: ECC (error correcting code) Länge.

Klasse 0, Code 0Dh:
Read ECC Burst Error Length. Dieser Befehl ist nur nach einem korrigierbaren Datenfehler gültig. Ein Byte wird übertragen das die Länge des korrigierten Fehlers angibt.

Klasse 0, Code 0Eh:
Format Alternate Track. Damit ist es möglich, eine Ersatzspur zu formatieren, um eine mögliche defekte Spur zu ersetzen.

Klasse 0, Code 0Fh:
Write Sector Buffer. Damit kann man in den internen Sektor-Zwischenspeicher direkt schreiben.

Klasse 0, Code 10h:
Read Sector Buffer. Nach einem Schreibvorgang kann man z. B. damit das interne RAM zu Kontrollzwecken auslesen.

Klasse 7, Code 0:
Damit wird das interne RAM getestet.

Klasse 7, Code 1...2 reserviert.

Klasse 7, Code 3:
Drive Diagnostic. Damit wird das Lauf-

werk, wie auch das Interface getestet. Daten werden dabei nicht zerstört. Auf allen Spuren wird der Kopf von Sektor 0 geprüft. Das Laufwerk muß vorher formatiert worden sein.

Klasse 7, Code 4:
Controller Internal Diagnostics. Der Inhalt des EPROMs wird mit einer Prüfsumme verglichen, wie auch die ECC-Schaltung. Das Laufwerk wird nicht angesprochen.

Klasse 7, Code 5:
Read Long. Damit kann man die ECC-Schaltung prüfen. Anstelle von 256 Bytes werden 260 Bytes übertragen, in denen die ECC-Bytes enthalten sind.

Klasse 7, Code 6:
Write Long. Zusätzlich werden die ECC-Bytes mitgeschickt und direkt aufgezeichnet.

Klasse 7, Code 7:
Retry Statistics. Damit kann man eine Statistik über aufgetretene Fehler erhalten. Dazu werden 8 Bytes zurückübertragen.

Erweiterungen beim Controller 1410A:

Klasse 0, Code 11h:
Initialize Format. Wenn man diesen Befehl vor einem Formatier-Befehl gibt, so wird nach dem Formatieren die Laufwerkinformation auf Zylinder 0 gespeichert.

Klasse 0, Code 12h:
Read Initialize Data. Da die Laufwerkinformationen bei der neueren Ausführung des Controllers mit auf die Platte geschrieben werden und von dort automatisch gelesen werden, kann man mit diesem Befehl die aktuellen Daten abfragen.

Klasse 0, Code 13h:
Stop Drive. Der Befehl ist für Winchester-Wechselplatten gedacht, um den Motor bei Plattenwechsel abzuschalten.

durchlaufen, und wenn dabei kein Fehler auftrat, so wird jeder Sektor einzeln gelesen. Dies dauert ebenfalls einige Minuten. Alle 256 Sektoren wird daher zur Beruhigung des Benutzers ein Punkt auf den Bildschirm ausgegeben. Im Programm ist ebenfalls noch ein Schreib-Befehl implementiert, der für eigene erste Versuche dienen kann.

(Fortsetzung folgt)

Literatur

[1] Handbücher: Xebec Controller 1410; BASF-6188; RO-352; WD-1002. Diese Handbücher erhält man in der Regel erst beim Kauf des Gerätes.

```

*****
; kleines Formatierprogramm, fuer den
; ersten Laufwerkstest
; Achtung, Adresse des Controllers und
; Parameter des Laufwerks eintragen.
; (C) 1985 Rolf-Dieter Klein V 1.0
*****
0005      bdos equ 5

0000'      start:
0000'      C3 00E4'      jp main
0003'      C3 00DE'      jp lesnur      ; test nur pruefen
;
; Diese Tabelle muss man aendern, wenn
; man andere Laufwerke verwenden will.

9880      sekanz equ 39308      ; cylinder * koepfe * 32
; bei 256 Bytes/sek, R0352/BASF6188

```

Bild 14. Formatierprogramm


```

0153' CD 000E' ; Recalibratre und Init ausfuehren
0154' 3E E4 ; Controller testen
0155' CD 006F' call taskout
0156' CD 00A3' call getstat
0157' CD 00A3' JP nz,main
0158' C2 00E4' call type
0159' CD 00CA' defb 0dh,0ah,'Controller ok',0dh,0ah,'$'
0160' 00 0A 43 6F
0161' 6E 74 72 6F
0162' 01 6C 6C 65 72
0163' 20 6F 6B 00
0164' 0A 24
0165' 3E E0
0166' CD 006F' call taskout
0167' CD 00A3' call getstat
0168' JP nz,main
0169' CD 00CA' call type
0170' defb 0dh,0ah,'RAM ok, Formattieren beginnt'
0171' 00 0A 52 41
0172' 40 20 6F 68
0173' 2C 20 46 6F
0174' 72 60 61 74
0175' 69 65 72 65
0176' 6E 20 62 65
0177' 67 69 6E 6E
0178' 74
0179' 00 0A 62 69
0180' 74 74 65 20
0181' 77 61 72 74
0182' 65 6E 20 2E
0183' 2E 2E 00 0A
0184' 24
0185' 21 0270' ld hl,taskform ; recal bei init durchgefuehrt.
0186' CD 0083' call taskhl ; Formattieren beginnt.
0187' CD 00A3' .call getstat
0188' DA 01E7' err:
0189'
0190' CD 00CA' call type
0191' defb 0dh,0ah,'*** Formattier Fehler ***',0dh,0ah,'$'
0192' 00 0A 2A 2A
0193' 2A 2A 20 46
0194' 6F 72 60 61
0195' 74 69 65 72
0196' 20 46 65 68
0197' 6C 65 72 20
0198' 2A 2A 2A 2A
0199' 00 0A 24
0200' C3 0000
0201'
0202' CD 00CA' call type
0203' defb 0dh,0ah,'Sektor 0 Test',0dh,0ah,'$'
0204' 00 0A 53 65
0205' 6B 74 6F 72
0206' 20 30 20 54
0207' 65 73 74 00
0208' 0A 24
0209' 3E C3
0210' CD 006F' call taskout
0211' CD 00A3' call getstat
0212' JP nz,err ; Fehler sonst.
0213' Testnur: ; nur Lese-Test Einsprung.
0214' call type
0215' defb 0dh,0ah,'Nun folgt Prueflesen',0dh,0ah,'$'
0216' 00 0A 4C 75
0217' 6E 20 66 6F
0218' 6C 67 74 20
0219'
0220' 50 72 75 65
0221' 66 6C 65 73
0222' 65 6E 0D 0A
0223' 24
0224' 11 0000
0225' 06 00
0226'
0227' C5
0228' D5
0229' CD 004C'
0230' D1
0231' C1
0232' 20 92
0233' 13
0234' 04
0235' 20 09
0236' C5
0237' D5
0238' CD 00CA'
0239' 2E 24
0240' 01
0241' C1
0242'
0243' 21 9880
0244' AF
0245' ED 52
0246' 7C
0247' B5
0248' C2 0237'
0249' CD 00CA'
0250' 00 0A
0251' 4F 4B 2E 20
0252' 4C 61 75 66
0253' 77 65 72 68
0254' 20 66 6F 72
0255' 6D 61 74 69
0256' 65 72 74 2E
0257' 20 45 6E 64
0258' 65
0259' 00 0A 24
0260' C3 0000
0261'
0262' 0270' taskform:
0263' defb format,0,0,0,0,0 ; hier ggf. aendern.
0264'
0265' task: defs 6 ;task block
0266'
0267' oldsec: defw 0 ;frueherer sector a 120
0268'
0269' .mvr: defb 0ffh ;wichtig init auf ff,wiini etc
0270'
0271' .mwf: defb 0 ;befehl =1 read =2 write
0272' sector: defw 0 ;sectoradresse a 128
0273' .mwna: defw 0 ;adresse ziel,quelle
0274'
0275' .mstbuf: defs 256 ; Sektor-Buffer fuer Prueflesen
0276'
0277' end start

```