oettle & reichler
datentechnik

Schießgrabenstr. 28a
8900 Augsburg 1

Tel.: (0821) 15 46 32

F D C

Floppy & Memory Extension Controller

H A N D B U C H

Inhaltsverzeichnis

Schaltungs-Beschreibung

Die FDC-Baugruppe erlaubt den gemischten Betrieb von maximal vier softsektorierten 3, 5.25 und 8 Zoll Laufwerken. Der Datenaustausch mit dem Systemspeicher erfolgt per DMA. Die DMA kann im 1M-Byte Adreßbereich jede beliebige 64k-Seite ansprechen. Memory to Memory Transfers zwischen verschiedenen Pages mit einer Übertragungsrate von 1 M-Byte/sec sind ebenso möglich. Die FDC-Baugruppe eignet sich damit hervorragend zum Aufbau einer Halbleiter- RAM-Floppy. Dies erbringt eine bis zu 20-fache Geschwindigkeitssteigerung gegenüber herkömmlichen Laufwerken.

Besonderen Wert wurde auf einen sicheren Datentransport vom und zum Diskettenlaufwerk gelegt. Beim Lesen sorgt ein integrierter "Phase Locked Loop" Datenseperator für die einwandfreie Aufarbeitung des seriellen Datenstroms von der Diskette. Beim Schreiben garantiert eine "Dynamic write precompensation"-Schaltung die sichere Aufzeichnung.

Adress-Extension A16 - A19

Der B-Port der Z-80 PIO definiert in den aktiven Phasen der DMA eine 64 K-Byte Page, aus der die DMA liest und eine weitere, in die sie schreibt. Schreib- und Lese-Bank können selbstverständlich auch identisch sein. Bit 0-3 PIO-Port B bestimmen dabei die Page aus der gelesen werden soll, Bit 4-7 die Page, in die geschrieben werden soll. In der aktiven Phase der DMA werden die Daten dieser beiden 4-Bit Ports abwechselnd zum Lesen oder Schreiben auf den Adreßbus A16 - A19 geschaltet.

```
PIO Port B:     Bit  7   6   5   4   3   2   1   0
                    A19 A18 A17 A16 A19 A18 A17 A16
                    !_____________! !_____________!
                       DMA Write       DMA Read
```

Der Hardwareaufbau dieser Schaltung erfordert es, daß die DMA sowohl bei Speicher- als auch bei I/O-Operationen auf "early end" programmiert wird. Bei identischer Read- und Write-Page ist dies nicht unbedingt erforderlich.

Bussteuerung:

Die BUS-Steuerung erfolgt über einen 256 x 4 PROM (24 S 10, IC 09). In Abhängigkeit der 8 Eingangssignale und der internen Programmierung, werden 3 Ausgangssignale erzeugt.

Pinbelegung Bus-Prom IC 09:

A7	Pin 15	BAI	D3	Pin 9	DMA inactive
A6	Pin 1	M1	D2	Pin 10	DIR DATA
A5	Pin 2	BAO	D1	Pin 11	EN DATA
A4	Pin 3	RD	D0	Pin 12	NC
A3	Pin 4	IORQ			
A2	Pin 7	IO-SEL			
A1	Pin 6	IEI			
A0	Pin 5	IEO			

Funktion der Bussteuerung:

1. Während der CPU-aktiven Zeit:

   Die Adreß- und Steuersignale treiben vom Systembus ins Innere der Karte. Der Datenbus treibt normalerweise auch nach innen. Ausnahmen sind I/O-Lese- und Interupt-Acknowledge-Zyklen, die die DMA, PIO oder den FDC ansprechen. In diesen Fällen treibt die Karte auf den Systembus.

2. Während der DMA-aktiven Zeit:

   Die Adreß- und Steuersignale treiben nach außen auf den Systembus. Der Datenbus treibt normalerweise auch nach außen. Ausnahmen sind I/O-Lesezyklen, die nicht die PIO oder den FDC ansprechen sowie Memory-Lesezyklen.

I/O Adreßbelegung

Die FDC Karte belegt insgesamt 8 I/O-Adressen. Diese 8 Adressen können in zwei 4-Byte Blöcken beliebig in dem 256-Byte I/O Adreßraum angeordnet werden. Ein 4-Byte Block wird von der PIO belegt.

Der zweite 4-Byte Block teilt sich folgendermaßen auf:

XXXX XX00 = FDC upD 756
XXXX XX01 = FDC upd 756
XXXX XX10 = DMA
XXXX XX11 = Motor on

Die oberen 6-Bit's (A7 - A2) werden durch Programmieren des I/O Prom's IC 02 (TBP 24 SA 10) bestimmt.

Standard-Adreßbelegung:

Standardmäßig belegt das FDC-Modul folgende 8 I/O-Adressen:

20h - 23h	**Z-80 PIO**
	20h Port A Data
	21h Port B Data
	22h Port A Control
	23h Port B Control
24h - 25h	**uPD 765 FDC**
	24h FDC Command Port
	25h FDC Data Port
26h	**Z-80 DMA**
27h	**Motor on I und II**

Pinbelegung IC 02 (24 SA 10):

A7	Pin 15	A5	D3	Pin 9	CE-PIO
A6	Pin 1	A7	D2	Pin 10	IO-SEL
A5	Pin 2	A6	D1	Pin 11	CE-FDC+765
A4	Pin 3	A0	D0	Pin 12	MOT-ON
A3	Pin 4	A4			
A2	Pin 7	A2			
A1	Pin 6	A3			
A0	Pin 5	A1			

INTACK-DMA

Wird die DMA auf halber Systemfrequenz betrieben, so ist es nicht mehr möglich den DMA-Baustein in die Interrupt-Kette einzubinden. Der M1-Eingang der DMA wird deshalb auf High-Potential geführt um die DMA daran zu hindern, den Interrupt-Vektor während des INTACK-Zyklus auf den BUS zu legen. Das INT-Signal der DMA wird nicht mit dem BUS verbunden, sondern auf einen Eingang der Z-80 PIO gelegt, die stellvertretend für die DMA einen Vektor-Interrupt auslöst. Gleichzeitig stellt dieses INT-Signal den Terminal-Count-Puls (TC) für den FDC-Baustein dar. Das Rücksetzen des DMA-Interrupts muß softwaremäßig geschehen (0C3h Reset DMA), da die DMA einen RETI-Opcode (ED 4D) nicht mehr erkennen kann (halbe Systemfrequenz).

Head-Load-Schaltung

Um ein gleichzeitiges Absenken der Schreib/Lese Köpfe aller vier Laufwerke zu verhindern, ist es möglich, jedem Laufwerk ein getrenntes Head-Load-Signal zur Verfügung zu stellen. Welches der vier Laufwerke ein aktives Head-Load-Signal bekommt, bestimmen Bits 3 und 4 des A-Ports der PIO. Ein RS-Flip-Flop gewährleistet, daß die Köpfe aller angesprochenen Laufwerke nach einem Zugriff eine bestimmte "Head Unload Time" (über FDC oder Monoflop II programmierbar) abgesenkt bleiben und bei einem erneuten Zugriff nicht wieder neu geladen werden müssen. Insbesondere bei Transfers zwischen zwei Laufwerken ist dies durch schnelleren Zugriff und geringes "Kopfklappern" von Vorteil.

PIO Parallel Input Output Controller

Die Beschaltung der 2 x 8 I/O-Leitungen des Z-80 PIO-Bausteins ist folgendermaßen:

Channel A

PA 0	Pin 15	Input	INT DMA	High active
PA 1	Pin 14	Input	Motor on 1	High active
PA 2	Pin 13	Output	Enable Mot on 1	Low active
PA 3	Pin 12	Output	Select Drive 0	High active
PA 4	Pin 10	Output	Select Drive 1	High active
PA 5	Pin 9	Output	MINI Drive	Low active
PA 6	Pin 8	Input	INT 765	High active
PA 7	Pin 7	Output	Terminal Count 765	High active

Channel B

PB 0	Pin 27	Output	A16 DMA-Read Adress
PB 1	Pin 28	Output	A17 DMA-Read Adress
PB 2	Pin 29	Output	A18 DMA-Read Adress
PB 3	Pin 30	Output	A19 DMA-Read Adress
PB 4	Pin 31	Output	A16 DMA-Write Adress
PB 5	Pin 32	Output	A17 DMA-Write Adress
PB 6	Pin 33	Output	A18 DMA-Write Adress
PB 7	Pin 34	Output	A19 DMA-Write Adress

Steckbrücken

1. DMA Clock

Die Z-80 DMA läßt sich wahlweise auf halber Systemfrequenz betreiben. Da Z-80 DMA Bausteine derzeit nur in der A-Version (4 MHz) gefertigt werden, ist dies immer dann erforderlich, wenn die Systemfrequenz 4 MHz überschreitet.

2. Wait DMA

Ist diese Brücke gesteckt, werden bei jedem I/O-Zugriff auf die DMA Wait's ausgegeben. Die Anzahl der Wait's bewegt sich, in Abhängigkeit von der Steckbrücke DMA-Clock, zwischen 2 und 4 Wait-Stait's. Wird die DMA auf halber Systemfrequenz betrieben, so muß diese Brücke gesteckt sein.

3. BAI from BUSAK

Die BUS-Aviable-In-Leitung (BAI) der DMA wird normalerweise von einer BUS-Aviable-Out (BAO) Leitung einer höherpriorisierten DMA beschaltet. Soll die DMA auf der FDC-Karte höchste Priorität erhalten oder ist es der einzige DMA-Baustein im System, so ist das BUS-Aviable-In-Signal (BAI) vom BUS-Acknowledge (BUSAK) Signal der CPU abzuleiten (Brücke gesteckt).

4. Pin 6 MINI

Pin Nr. 6 des 34-poligen MINI Steckers kann wahlweise mit Ready bzw. Drive-Select 3 beschaltet werden. Bitte beachten Sie die Angaben des angeschlossenen Laufwerks.

5. Head Load

Die Head Load Leitung (Pin 2 bzw. 18) kann wahlweise mit dem Head-Load des uPD 765 oder mit einem nur bei Laufwerk A aktiven Head-Load 0 beschaltet werden.

6. Dynamic Write Precompensation

Auf diesem 18-poligen Pfostensteckerfeld kann der Precompensation-Wert beim Schreiben auf Diskette eingestellt werden. Je nach Steckweise ergeben sich folgende Werte:

Standard				MINI			
P2	**P1**	**P0**	**Precomp.**	**P2**	**P1**	**P0**	**Precomp.**
0	0	0	0 ns	0	0	0	0 ns
0	0	1	62,5 ns	0	0	1	125 ns
0	1	0	125 ns	0	1	0	250 ns
0	1	1	187,5 ns	0	1	1	375 ns
1	0	0	250 ns	1	0	0	500 ns
1	0	1	250 ns	1	0	1	500 ns
1	1	0	312,5 ns	1	1	0	625 ns
1	1	1	312,5 ns	1	1	1	625 ns

Es muß je Reihe eine Brücke gesteckt sein, also insgesamt 3 Steckbrücken. Für jede der drei Reihen ergeben sich vier Steck-Möglichkeiten:

1, H = 1
2, L = 0
3, OUT = 0 bei inneren Spuren, 1 bei äußeren Spuren
4, IN = 1 bei inneren Spuren, 0 bei äußeren Spuren

Bei Steckweise 3 und 4 läßt sich der Precompensation-Wert für die inneren und äußeren Spuren getrennt einstellen.

Der Precompensation Wert ist abhängig vom verwendeten Floppy-Laufwerk. Bitte beachten Sie die dort angegebenen Werte. Sind keine genauen Angaben vorhanden empfiehlt sich folgende Steckweise: P2 = 0, P1 = 1, P2 = 0 (125/250 ns).

Steckerbelegung zu den Laufwerken

Standard

2	Low Current
4	Fault Reset
6	Fault
8	
10	Two Sided
12	Motor on 1
14	Side Select
16	Motor on 0
18	Head Load / Head Load 0
20	Index
22	Ready
24	Head Load 1
26	Drive Select 0
28	Drive Select 1
30	Drive Select 2
32	Drive Select 3
34	Direction
36	Step
38	Write Data
40	Write Enable
42	Track 00
44	Write Protect
46	Read Data
48	Head Load 2
50	Head Load 3

MINI

2	Head Load / Head Load 0
4	Head Load 1
6	Drive Select 3 / Ready
8	Index
10	Drive Select 0
12	Drive Select 1
14	Drive Select 2
16	Motor on
18	Direction
20	Step
22	Write Data
24	Write Enable
26	Track 00
28	Write Protect
30	Read Data
32	Side Select
34	Ready

µPD765A

INSTRUCTION SET ① ②

Phase	R/W	Data Bus D7 D6 D5 D4 D3 D2 D1 D0	Remarks
READ DATA			
Command	W	MT MF SK 0 0 1 1 0	Command Codes
	W	X X X X X HD US1 US0	
	W	C	Sector ID information prior to Command execution. The 4 bytes are commanded against header on Floppy Disk.
	W	H	
	W	R	
	W	N	
	W	EOT	
	W	GPL	
	W	DTL	
Execution			Data-transfer between the FDD and main-system
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	Sector ID information after Command execution
	R	H	
	R	R	
	R	N	
READ DELETED DATA			
Command	W	MT MF SK 0 1 1 0 0	Command Codes
	W	X X X X X HD US1 US0	
	W	C	Sector ID information prior to Command execution. The 4 bytes are commanded against header on Floppy Disk.
	W	H	
	W	R	
	W	N	
	W	EOT	
	W	GPL	
	W	DTL	
Execution			Data-transfer between the FDD and main-system
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	Sector ID information after Command execution
	R	H	
	R	R	
	R	N	
WRITE DATA			
Command	W	MT MF 0 0 0 1 0 1	Command Codes
	W	X X X X X HD US1 US0	
	W	C	Sector ID information prior to Command execution. The 4 bytes are commanded against header on Floppy Disk.
	W	H	
	W	R	
	W	N	
	W	EOT	
	W	GPL	
	W	DTL	
Execution			Data-transfer between the main-system and FDD
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	Sector ID information after Command execution
	R	H	
	R	R	
	R	N	
WRITE DELETED DATA			
Command	W	MT MF 0 0 1 0 0 1	Command Codes
	W	X X X X X HD US1 US0	
	W	C	Sector ID information prior to Command execution. The 4 bytes are commanded against header on Floppy Disk.
	W	H	
	W	R	
	W	N	
	W	EOT	
	W	GPL	
	W	DTL	
Execution			Data-transfer between the FDD and main-system
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	Sector ID information after Command execution
	R	H	
	R	R	
	R	N	

Phase	R/W	Data Bus D7 D6 D5 D4 D3 D2 D1 D0	Remarks
READ A TRACK			
Command	W	0 MF SK 0 0 0 1 0	Command Codes
	W	X X X X X HD US1 US0	
	W	C	Sector ID information prior to Command execution
	W	H	
	W	R	
	W	N	
	W	EOT	
	W	GPL	
	W	DTL	
Execution			Data-transfer between the FDD and main-system. FDC reads all data fields from index hole to EOT.
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	Sector ID information after Command execution
	R	H	
	R	R	
	R	N	
READ ID			
Command	W	0 MF 0 0 1 0 1 0	Commands
	W	X X X X X HD US1 US0	
Execution			The first correct ID information on the Cylinder is stored in Data Register
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	Sector ID information read during Execution Phase from Floppy Disk
	R	H	
	R	R	
	R	N	
FORMAT A TRACK			
Command	W	0 MF 0 0 1 1 0 1	Command Codes
	W	X X X X X HD US1 US0	
	W	N	Bytes/Sector
	W	SC	Sectors/Track
	W	GPL	Gap 3
	W	D	Filler Byte
Execution			FDC formats an entire track
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	In this case, the ID information has no meaning
	R	H	
	R	R	
	R	N	
SCAN EQUAL			
Command	W	MT MF SK 1 0 0 0 1	Command Codes
	W	X X X X X HD US1 US0	
	W	C	Sector ID information prior to Command execution
	W	H	
	W	R	
	W	N	
	W	EOT	
	W	GPL	
	W	STP	
Execution			Data-compared between the FDD and main-system
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	Sector ID information after Command execution
	R	H	
	R	R	
	R	N	

Note: ① Symbols used in this table are described at the end of this section.
② A0 should equal binary 1 for all operations.
③ X = Don't care, usually made to equal binary 0.

INSTRUCTION SET (CONT.)

µPD765A

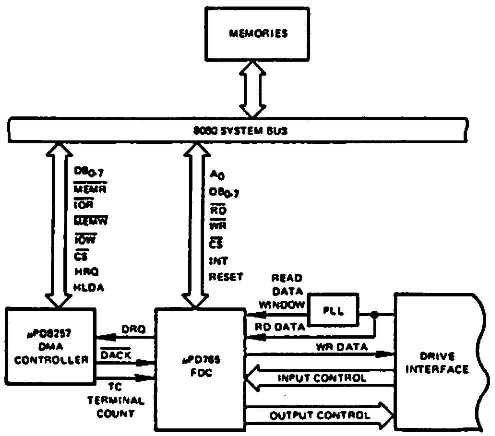
Phase	R/W	Data Bus D7 D6 D5 D4 D3 D2 D1 D0	Remarks
SCAN LOW OR EQUAL			
Command	W	MT MF SK 1 1 0 0 1	Command Codes
	W	X X X X X HD US1 US0	
	W	C	Sector ID information prior Command execution
	W	H	
	W	R	
	W	N	
	W	EOT	
	W	GPL	
	W	STP	
Execution			Data-compared between the FDD and main-system
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	Sector ID information after Command execution
	R	H	
	R	R	
	R	N	
SCAN HIGH OR EQUAL			
Command	W	MT MF SK 1 1 1 0 1	Command Codes
	W	X X X X X HD US1 US0	
	W	C	Sector ID information prior Command execution
	W	H	
	W	R	
	W	N	
	W	EOT	
	W	GPL	
	W	STP	
Execution			Data-compared between the FDD and main-system
Result	R	ST 0	Status information after Command execution
	R	ST 1	
	R	ST 2	
	R	C	Sector ID information after Command execution
	R	H	
	R	R	
	R	N	

Phase	R/W	Data Bus D7 D6 D5 D4 D3 D2 D1 D0	Remarks
RECALIBRATE			
Command	W	0 0 0 0 0 1 1 1	Command Codes
	W	X X X X X 0 US1 US0	
Execution			Head retracted to Track 0
SENSE INTERRUPT STATUS			
Command	W	0 0 0 0 1 0 0 0	Command Codes
Result	R	ST0	Status information at the end of seek-operation about the FDC
	R	PCN	
SPECIFY			
Command	W	0 0 0 0 0 0 1 1	Command Codes
	W	SRT — HUT	
	W	HLT — ND	
SENSE DRIVE STATUS			
Command	W	0 0 0 0 0 1 0 0	Command Codes
	W	X X X X X HD US1 US0	
Result	R	ST 3	Status information about FDD
SEEK			
Command	W	0 0 0 0 1 1 1 1	Command Codes
	W	X X X X X HD US1 US0	
	W	NCN	
Execution			Head is positioned over proper Cylinder on Diskette
INVALID			
Command	W	Invalid Codes	Invalid Command Codes (NoOp — FDC goes into Standby State)
Result	R	ST 0	ST 0 = 80 (16)

COMMAND SYMBOL DESCRIPTION

SYMBOL	NAME	DESCRIPTION
A_0	Address Line 0	A_0 controls selection of Main Status Register (A_0 = 0) or Data Register (A_0 = 1)
C	Cylinder Number	C stands for the current/selected Cylinder (track) number 0 through 76 of the medium.
D	Data	D stands for the data pattern which is going to be written into a Sector.
D_7-D_0	Data Bus	8-bit Data Bus, where D_7 stands for a most significant bit, and D_0 stands for a least significant bit.
DTL	Data Length	When N is defined as 00, DTL stands for the data length which users are going to read out or write into the Sector.
EOT	End of Track	EOT stands for the final Sector number on a Cylinder. During Read or Write operation FDC will stop date transfer after a sector # equal to EOT.
GPL	Gap Length	GPL stands for the length of Gap 3. During Read/Write commands this value determines the number of bytes that VCOs will stay low after two CRC bytes. During Format command it determines the size of Gap 3.
H	Head Address	H stands for head number 0 or 1, as specified in ID field.
HD	Head	HD stands for a selected head number 0 or 1 and controls the polarity of pin 27. (H = HD in all command words.)
HLT	Head Load Time	HLT stands for the head load time in the FDD (2 to 254 ms in 2 ms increments).
HUT	Head Unload Time	HUT stands for the head unload time after a read or write operation has occurred (16 to 240 ms in 16 ms increments).
MF	FM or MFM Mode	If MF is low, FM mode is selected, and if it is high, MFM mode is selected.
MT	Multi-Track	If MT is high, a multi-track operation is to be performed. If MT = 1 after finishing Read/Write operation on side 0 FDC will automatically start searching for sector 1 on side 1.

SYMBOL	NAME	DESCRIPTION
N	Number	N stands for the number of data bytes written in a Sector.
NCN	New Cylinder Number	NCN stands for a new Cylinder number, which is going to be reached as a result of the Seek operation. Desired position of Head.
ND	Non-DMA Mode	ND stands for operation in the Non-DMA Mode.
PCN	Present Cylinder Number	PCN stands for the Cylinder number at the completion of SENSE INTERRUPT STATUS Command. Position of Head at present time.
R	Record	R stands for the Sector number, which will be read or written.
R/W	Read/Write	R/W stands for either Read (R) or Write (W) signal.
SC	Sector	SC indicates the number of Sectors per Cylinder.
SK	Skip	SK stands for Skip Deleted Data Address Mark.
SRT	Step Rate Time	SRT stands for the Stepping Rate for the FDD. (1 to 16 ms in 1 ms increments.) Stepping Rate applies to all drives, (F = 1 ms, E = 2 ms, etc.).
ST 0 ST 1 ST 2 ST 3	Status 0 Status 1 Status 2 Status 3	ST 0-3 stand for one of four registers which store the status information after a command has been executed. This information is available during the result phase after command execution. These registers should not be confused with the main status register (selected by A_0 = 0). ST 0-3 may be read only after a command has been executed and contain information relevant to that particular command.
STP		During a Scan operation, if STP = 1, the data in contiguous sectors is compared byte by byte with data sent from the processor (or DMA); and if STP = 2, then alternate sectors are read and compared.
US0, US1	Unit Select	US stands for a selected drive number 0 or 1.



PROCESSOR INTERFACE

During Command or Result Phases the Main Status Register (described earlier) must be read by the processor before each byte of information is written into or read from the Data Register. After each byte of data read or written to Data Register, CPU should wait for 12 μs before reading MSR. Bits D6 and D7 in the Main Status Register must be in a 0 and 1 state, respectively, before each byte of the command word may be written into the μPD765. Many of the commands require multiple bytes, and as a result the Main Status Register must be read prior to each byte transfer to the μPD765. On the other hand, during the Result Phase, D6 and D7 in the Main Status Register must both be 1's (D6 = 1 and D7 = 1) before reading each byte from the Data Register. Note, this reading of the Main Status Register before each byte transfer to the μPD765 is required in only the Command and Result Phases, and NOT during the Execution Phase.

During the Execution Phase, the Main Status Register need not be read. If the μPD765 is in the NON-DMA Mode, then the receipt of each data byte (if μPD765 is reading data from FDD) is indicated by an Interrupt signal on pin 18 (INT = 1). The generation of a Read signal ($\overline{RD}$ = 0) or Write signal ($\overline{WR}$ = 0) will reset the Interrupt as well as output the Data onto the Data Bus. If the processor cannot handle Interrupts fast enough (every 13 μs) for MFM and 27 μs for FM mode, then it may poll the Main Status Register and then bit D7 (RQM) functions just like the Interrupt signal. If a Write Command is in process then the WR signal performs the reset to the Interrupt signal.

If the μPD765 is in the DMA Mode, no Interrupts are generated during the Execution Phase. The μPD765 generates DRQ's (DMA Requests) when each byte of data is available. The DMA Controller responds to this request with both a $\overline{DACK}$ = 0 (DMA Acknowledge) and a $\overline{RD}$ = 0 (Read signal). When the DMA Acknowledge signal goes low ($\overline{DACK}$ = 0) then the DMA Request is reset (DRQ = 0). If a Write Command has been programmed then a $\overline{WR}$ signal will appear instead of $\overline{RD}$. After the Execution Phase has been completed (Terminal Count has occurred) or EOT sector was read/written, then an Interrupt will occur (INT = 1). This signifies the beginning of the Result Phase. When the first byte of data is read during the Result Phase, the Interrupt is automatically reset (INT = 0).

It is important to note that during the Result Phase all bytes shown in the Command Table must be read. The Read Data Command, for example has seven bytes of data in the Result Phase. All seven bytes must be read in order to successfully complete the Read Data Command. The μPD765 will not accept a new command until all seven bytes have been read. Other commands may require fewer bytes to be read during the Result Phase.

The μPD765 contains five Status Registers. The Main Status Register mentioned above may be read by the processor at any time. The other four Status Registers (ST0, ST1, ST2, and ST3) are only available during the Result Phase, and may be read only after completing a command. The particular command which has been executed determines how many of the Status Registers will be read.

The bytes of data which are sent to the μPD765 to form the Command Phase, and are read out of the μPD765 in the Result Phase, must occur in the order shown in the Command Table. That is, the Command Code must be sent first and the other bytes sent in the prescribed sequence. No foreshortening of the Command or Result Phases are allowed. After the last byte of data in the Command Phase is sent to the μPD765, the Execution Phase automatically starts. In a similar fashion, when the last byte of data is read out in the Result Phase, the command is automatically ended and the μPD765 is ready for a new command.

POLLING FEATURE OF THE μPD765

After the Specify command has been sent to the μPD765, the Unit Select line US0 and US1 will automatically go into a polling mode. In between commands (and between step pulses in the SEEK command) the μPD765 polls all four FDD's looking for a change in the Ready line from any of the drives. If the Ready line changes state (usually due to a door opening or closing) then the μPD765 will generate an interrupt. When Status Register 0 (ST0) is read (after Sense Interrupt Status is issued), Not Ready (NR) will be indicated. The polling of the Ready line by the μPD765 occurs continuously between commands, thus notifying the processor which drives are on or off line. Each drive is polled every 1.024 ms except during the Read/Write commands.

FUNCTIONAL DESCRIPTION OF COMMANDS

READ DATA

A set of nine (9) byte words are required to place the FDC into the Read Data Mode. After the Read Data command has been issued the FDC loads the head (if it is in the unloaded state), waits the specified head settling time (defined in the Specify Command), and begins reading ID Address Marks and ID fields. When the current sector number ("R") stored in the ID Register (IDR) compares with the sector number read off the diskette, then the FDC outputs data (from the data field) byte-to-byte to the main system via the data bus.

After completion of the read operation from the current sector, the Sector Number is incremented by one, and the data from the next sector is read and output on the data bus. This continuous read function is called a "Multi-Sector Read Operation." The Read Data Command may be terminated by the receipt of a Terminal Count signal. TC should be issued at the same time that the $\overline{DACK}$ for the last byte of data is sent. Upon receipt of this signal, the FDC stops outputting data to the processor, but will continue to read data from the current sector, check CRC (Cyclic Redundancy Count) bytes, and then at the end of the sector terminate the Read Data Command.

The amount of data which can be handled with a single command to the FDC depends upon MT (multi-track), MF (MFM/FM), and N (Number of Bytes/Sector). Table 1 below shows the Transfer Capacity.

Multi-Track MT	MFM/FM MF	Bytes/Sector N	Maximum Transfer Capacity (Bytes/Sector) (Number of Sectors)	Final Sector Read from Diskette
0 0	0 1	00 01	(128) (26) = 3,328 (256) (26) = 6,656	26 at Side 0 or 26 at Side 1
1 1	0 1	00 01	(128) (52) = 6,656 (256) (52) = 13,312	26 at Side 1
0 0	0 1	01 02	(256) (15) = 3,840 (512) (15) = 7,680	15 at Side 0 or 15 at Side 1
1 1	0 1	01 02	(256) (30) = 7,680 (512) (30) = 15,360	15 at Side 1
0 0	0 1	02 03	(512) (8) = 4,096 (1024) (8) = 8,192	8 at Side 0 or 8 at Side 1
1 1	0 1	02 03	(512) (16) = 8,192 (1024) (16) = 16,384	8 at Side 1

Table 1. Transfer Capacity

The "multi-track" function (MT) allows the FDC to read data from both sides of the diskette. For a particular cylinder, data will be transferred starting at Sector 1, Side 0 and completing at Sector L, Side 1 (Sector L = last sector on the side). Note, this function pertains to only one cylinder (the same track) on each side of the diskette.

When N = 0, then DTL defines the data length which the FDC must treat as a sector. If DTL is smaller than the actual data length in a Sector, the data beyond DTL in the Sector, is not sent to the Data Bus. The FDC reads (internally) the complete Sector performing the CRC check, and depending upon the manner of command termination, may perform a Multi-Sector Read Operation. When N is non-zero, then DTL has no meaning and should be set to FF Hexidecimal.

At the completion of the Read Data Command, the head is not unloaded until after Head Unload Time Interval (specified in the Specify Command) has elapsed. If the processor issues another command before the head unloads then the head settling time may be saved between subsequent reads. This time out is particularly valuable when a diskette is copied from one drive to another.

If the FDC detects the Index Hole twice without finding the right sector, (indicated in "R"), then the FDC sets the ND (No Data) flag in Status Register 1 to a 1 (high), and terminates the Read Data Command. (Status Register 0 also has bits 7 and 6 set to 0 and 1 respectively.)

After reading the ID and Data Fields in each sector, the FDC checks the CRC bytes. If a read error is detected (incorrect CRC in ID field), the FDC sets the DE (Data Error) flag in Status Register 1 to a 1 (high), and if a CRC error occurs in the Data Field the FDC also sets the DD (Data Error in Data Field) flag in Status Register 2 to a 1 (high), and terminates the Read Data Command. (Status Register 0 also has bits 7 and 6 set to 0 and 1 respectively.)

If the FDC reads a Deleted Data Address Mark off the diskette, and the SK bit (bit D5 in the first Command Word) is not set (SK = 0), then the FDC sets the CM (Control Mark) flag in Status Register 2 to a 1 (high), and terminates the Read Data Command, after reading all the data in the Sector. If SK = 1, the FDC skips the sector with the Deleted Data Address Mark and reads the next sector. The CRC bits in the deleted data field are not checked when SK = 1.

During disk data transfers between the FDC and the processor, via the data bus, the FDC must be serviced by the processor every 27 μs in the FM Mode, and every 13 μs in the MFM Mode, or the FDC sets the OR (Over Run) flag in Status Register 1 to a 1 (high), and terminates the Read Data Command.

If the processor terminates a read (or write) operation in the FDC, then the ID Information in the Result Phase is dependent upon the state of the MT bit and EOT byte. Table 2 shows the values for C, H, R, and N, when the processor terminates the Command.

μPD765A

MT	HD	Final Sector Transferred to Processor	ID Information at Result Phase C	H	R	N
0	0	Less than EOT	NC	NC	R + 1	NC
	0	Equal to EOT	C + 1	NC	R = 01	NC
	1	Less than EOT	NC	NC	R + 1	NC
	1	Equal to EOT	C + 1	NC	R = 01	NC
1	0	Less than EOT	NC	NC	R + 1	NC
	0	Equal to EOT	NC	LSB	R = 01	NC
	1	Less than EOT	NC	NC	R + 1	NC
	1	Equal to EOT	C + 1	LSB	R = 01	NC

Notes: 1 NC (No Change): The same value as the one at the beginning of command execution.
2 LSB (Least Significant Bit): The least significant bit of H is complemented.

WRITE DATA

A set of nine (9) bytes are required to set the FDC into the Write Data mode. After the Write Data command has been issued the FDC loads the head (if it is in the unloaded state), waits the specified Head Settling Time (defined in the Specify Command), and begins reading ID Fields. When all four bytes loaded during the command (C, H, R, N) match the four bytes of the ID field from the diskette, the FDC takes data from the processor byte-by-byte via the data bus, and outputs it to the FDD.

After writing data into the current sector, the Sector Number stored in "R" is incremented by one, and the next data field is written into. The FDC continues this "Multi-Sector Write Operation" until the issuance of a Terminal Count signal. If a Terminal Count signal is sent to the FDC it continues writing into the current sector to complete the data field. If the Terminal Count signal is received while a data field is being written then the remainder of the data field is filled with 00 (zeros).

The FDC reads the ID field of each sector and checks the CRC bytes. If the FDC detects a read error (incorrect CRC) in one of the ID Fields, it sets the DE (Data Error) flag of Status Register 1 to a 1 (high), and terminates the Write Data Command. (Status Register 0 also has bits 7 and 6 set to 0 and 1 respectively.)

The Write Command operates in much the same manner as the Read Command. The following items are the same, and one should refer to the Read Data Command for details:

- Transfer Capacity
- EN (End of Cylinder) Flag
- ND (No Data) Flag
- Head Unload Time Interval
- ID Information when the processor terminates command (see Table 2)
- Definition of DTL when N = 0 and when N ≠ 0

In the Write Data mode, data transfers between the processor and FDC, via the Data Bus, must occur every 27 μs in the FM mode, and every 13 μs in the MFM mode. If the time interval between data transfers is longer than this then the FDC sets the OR (Over Run) flag in Status Register 1 to a 1 (high), and terminates the Write Data Command. (Status Register 0 also has bit 7 and 6 set to 0 and 1 respectively.)

WRITE DELETED DATA

This command is the same as the Write Data Command except a Deleted Data Address Mark is written at the beginning of the Data Field instead of the normal Data Address Mark.

READ DELETED DATA

This command is the same as the Read Data Command except that when the FDC detects a Data Address Mark at the beginning of a Data Field (and SK = 0 (low), it will read all the data in the sector and set the CM flag in Status Register 2 to a 1 (high), and then terminate the command. If SK = 1, then the FDC skips the sector with the Data Address Mark and reads the next sector.

READ A TRACK

This command is similar to READ DATA Command except that this is a continuous READ operation where the entire data field from each of the sectors are read. Immediately after encountering the INDEX HOLE, the FDC starts reading all data fields on the track, as continuous blocks of data. If the FDC finds an error in the ID or DATA CRC check bytes, it continues to read data from the track. The FDC compares the ID information read from each sector with the value stored in the IDR, and sets the ND flag of Status Register 1 to a 1 (high) if there is no comparison. Multi-track or skip operations are not allowed with this command.

This command terminates when number of sectors read is equal to EOT. If the FDC does not find an ID Address Mark on the diskette after it encounters the INDEX HOLE for the second time, then it sets the MA (missing address mark) flag in Status Register 1 to a 1 (high), and terminates the command. (Status Register 0 has bits 7 and 6 set to 0 and 1 respectively.)

FUNCTIONAL DESCRIPTION OF COMMANDS (CONT.)

READ ID

The READ ID Command is used to give the present position of the recording head. The FDC stores the values from the first ID Field it is able to read. If no proper ID Address Mark is found on the diskette, before the INDEX HOLE is encountered for the second time then the MA (Missing Address Mark) flag in Status Register 1 is set to a 1 (high), and if no data is found then the ND (No Data) flag is also set in Status Register 1 to a 1 (high). The command is then terminated with Bits 7 and 6 in Status Register 0 set to 0 and 1 respectively. During this command there is no data transfer between FDC and the CPU except during the result phase.

FORMAT A TRACK

The Format Command allows an entire track to be formatted. After the INDEX HOLE is detected, Data is written on the Diskette; Gaps, Address Marks, ID Fields and Data Fields, all per the IBM System 34 (Double Density) or System 3740 (Single Density) Format are recorded. The particular format which will be written is controlled by the values programmed into N (number of bytes/sector), SC (sectors/cylinder), GPL (Gap Length), and D (Data Pattern) which are supplied by the processor during the Command Phase. The Data Field is filled with the Byte of data stored in D. The ID Field for each sector is supplied by the processor; that is, four data requests per sector are made by the FDC for C (Cylinder Number), H (Head Number), R (Sector Number) and N (Number of Bytes/Sector). This allows the diskette to be formatted with non-sequential sector numbers, if desired.

The processor must send new values for C, H, R, and N to the µPD765 for each sector on the track. If FDC is set for DMA mode, it will issue 4 DMA requests per sector. If it is set for interrupt mode, it will issue four interrupts per sector and the processor must supply C, H, R and N load for each sector. The contents of the R register is incremented by one after each sector is formatted, thus, the R register contains a value of R when it is read during the Result Phase. This incrementing and formatting continues for the whole track until the FDC encounters the INDEX HOLE for the second time, whereupon it terminates the command.

If a FAULT signal is received from the FDD at the end of a write operation, then the FDC sets the EC flag of Status Register 0 to a 1 (high), and terminates the command after setting bits 7 and 6 of Status Register 0 to 0 and 1 respectively. Also the loss of a READY signal at the beginning of a command execution phase causes bits 7 and 6 of Status Register 0 to be set to 0 and 1 respectively.

Table 3 shows the relationship between N, SC, and GPL for various sector sizes:

	8" STANDARD FLOPPY					5¼" MINI FLOPPY				
FORMAT	SECTOR SIZE	N	SC	GPL ①	GPL ②	SECTOR SIZE	N	SC	GPL ①	GPL ②
FM Mode	128 bytes/Sector	00	1A	07	1B	128 bytes/Sector	00	12	07	09
	256	01	0F	0E	2A	128	00	10	10	19
	512	02	08	1B	3A	256	01	08	18	30
	1024 bytes/Sector	03	04	47	8A	512	02	04	46	87
	2048	04	02	C8	FF	1024	03	02	C8	FF
	4096	05	01	C8	FF	2048	04	01	C8	FF
MFM Mode	256	01	1A	0E	36	256	01	12	0A	0C
	512	02	0F	1B	54	256	01	10	20	32
	1024	03	08	35	74	512	02	08	2A	50
	2048	04	04	99	FF	1024	03	04	80	F0
	4096	05	02	C8	FF	2048	04	02	C8	FF
	8192	06	01	C8	FF	4096	05	01	C8	FF

Table 3

Note: ① Suggested values of GPL in Read or Write Commands to avoid splice point between data field and ID field of contiguous sections.

② Suggested values of GPL in format command.

③ In MFM mode FDC can not perform a read/write/format operation with 128 bytes/sector. (N = 00)

④ All the values are hexidecimal.

SCAN COMMANDS

The SCAN Commands allow data which is being read from the diskette to be compared against data which is being supplied from the main system. The FDC compares the data on a byte-by-byte basis, and looks for a sector of data which meets the conditions of $D_{FDD} = D_{Processor}$, $D_{FDD} \leq D_{Processor}$, or $D_{FDD} \geq D_{Processor}$. The hexidecemial byte of FF either from memory or from FDD can be used as a mask byte because it always meet the condition of the compare. Ones complement arithmetic is used for comparison (FF = largest number, 00 = smallest number). After a whole sector of data is compared, if the conditions are not met, the sector number is incremental (R + STP → R), and the scan operation is continued. The scan operation continues until one of the following conditions occur; the conditions for scan are met (equal, low, or high), the last sector on the track is reached (EOT), or the terminal count signal is received.

µPD765A

If the conditions for scan are met then the FDC sets the SH (Scan Hit) flag of Status Register 2 to a 1 (high), and terminates the Scan Command. If the conditions for scan are not met between the starting sector (as specified by R) and the last sector on the cylinder (EOT), then the FDC sets the SN (Scan Not Satisfied) flag of Status Register 2 to a 1 (high), and terminates the Scan Command. The receipt of a TERMINAL COUNT signal from the Processor or DMA Controller during the scan operation will cause the FDC to complete the comparison of the particular byte which is in process, and then to terminate the command. Table 4 shows the status of bits SH and SN under various conditions of SCAN.

COMMAND	STATUS REGISTER 2		COMMENTS
	BIT 2 = SN	BIT 3 = SH	
Scan Equal	0	1	$D_{FDD} = D_{Processor}$
	1	0	$D_{FDD} \neq D_{Processor}$
Scan Low or Equal	0	1	$D_{FDD} = D_{Processor}$
	0	0	$D_{FDD} < D_{Processor}$
	1	0	$D_{FDD} > D_{Processor}$
Scan High or Equal	0	1	$D_{FDD} = D_{Processor}$
	0	0	$D_{FDD} > D_{Processor}$
	1	0	$D_{FDD} < D_{Processor}$

Table 4

If the FDC encounters a Deleted Data Address Mark on one of the sectors (and SK = 0), then it regards the sector as the last sector on the cylinder, sets CM (Control Mark) flag of Status Register 2 to a 1 (high) and terminates the command. If SK = 1, the FDC skips the sector with the Deleted Address Mark, and reads the next sector. In the second case (SK = 1), the FDC sets the CM (Control Mark) flag of Status Register 2 to a 1 (high) in order to show that a Deleted Sector had been encountered.

When either the STP (contiguous sectors = 01, or alternate sectors = 02 sectors are read) or the MT (Multi-Track) are programmed, it is necessary to remember that the last sector on the track must be read. For example, if STP = 02, MT = 0, the sectors are numbered sequentially 1 through 26, and we start the Scan Command at sector 21; the following will happen. Sectors 21, 23, and 25 will be read, then the next sector (26) will be skipped and the Index Hole will be encountered before the EOT value of 26 can be read. This will result in an abnormal termination of the command. If the EOT had been set at 25 or the scanning started at sector 20, then the Scan Command would be completed in a normal manner.

During the Scan Command data is supplied by either the processor or DMA Controller for comparison against the data read from the diskette. In order to avoid having the OR (Over Run) flag set in Status Register 1, it is necessary to have the data available in less than 27 µs (FM Mode) or 13 µs (MFM Mode). If an Overrun occurs the FDC ends the command with bits 7 and 6 of Status Register 0 set to 0 and 1, respectively.

SEEK

The read/write head within the FDD is moved from cylinder to cylinder under control of the Seek Command. FDC has four independent Present Cylinder Registers for each drive. They are clear only after Recalibrate command. The FDC compares the PCN (Present Cylinder Number) which is the current head position with the NCN (New Cylinder Number), and if there is a difference performs the following operation:

PCN < NCN: Direction signal to FDD set to a 1 (high), and Step Pulses are issued. (Step In.)
PCN > NCN: Direction signal to FDD set to a 0 (low), and Step Pulses are issued. (Step Out.)

The rate at which Step Pulses are issued is controlled by SRT (Stepping Rate Time) in the SPECIFY Command. After each Step Pulse is issued NCN is compared against PCN, and when NCN = PCN, then the SE (Seek End) flag is set in Status Register 0 to a 1 (high), and the command is terminated. At this point FDC interrupt goes high. Bits DB_0-DB_3 in Main Status Register are set during seek operation and are cleared by Sense Interrupt Status command.

During the Command Phase of the Seek operation the FDC is in the FDC BUSY state, but during the Execution Phase it is in the NON BUSY state. While the FDC is in the NON BUSY state, another Seek Command may be issued, and in this manner parallel seek operations may be done on up to 4 Drives at once. No other command could be issued for as long as FDC is in process of sending Step Pulses to any drive.

If an FDD is in a NOT READY state at the beginning of the command execution phase or during the seek operation, then the NR (NOT READY) flag is set in Status Register 0 to a 1 (high), and the command is terminated after bits 7 and 6 of Status Register 0 are set to 0 and 1 respectively.

If the time to write 3 bytes of seek command exceeds 150 µs, the timing between first two Step Pulses may be shorter than set in the Specify command by as much as 1 ms.

RECALIBRATE

The function of this command is to retract the read/write head within the FDD to the Track 0 position. The FDC clears the contents of the PCN counter, and checks the status of the Track 0 signal from the FDD. As long as the Track 0 signal is low, the Direction signal remains 0 (low) and Step Pulses are issued. When the Track 0 signal goes high, the SE (SEEK END) flag in Status Register 0 is set to a 1 (high) and the command is terminated. If the Track 0 signal is still low after 77 Step Pulse have been issued, the FDC sets the SE (SEEK END) and EC (EQUIPMENT CHECK) flags of Status Register 0 to both 1s (highs), and terminates the command after bits 7 and 6 of Status Register 0 is set to 0 and 1 respectively.

The ability to do overlap RECALIBRATE Commands to multiple FDDs and the loss of the READY signal, as described in the SEEK Command, also applies to the RECALIBRATE Command.

SENSE INTERRUPT STATUS

An Interrupt signal is generated by the FDC for one of the following reasons:

1. Upon entering the Result Phase of:
   a. Read Data Command
   b. Read a Track Command
   c. Read ID Command
   d. Read Deleted Data Command
   e. Write Data Command
   f. Format a Cylinder Command
   g. Write Deleted Data Command
   h. Scan Commands
2. Ready Line of FDD changes state
3. End of Seek or Recalibrate Command
4. During Execution Phase in the NON-DMA Mode

Interrupts caused by reasons 1 and 4 above occur during normal command operations and are easily discernible by the processor. During an execution phase in NON-DMA Mode, DBS in Main Status Register is high. Upon entering Result Phase this bit gets clear. Reason 1 and 4 does not require Sense Interrupt Status command. The interrupt is cleared by reading/writing data to FDC. Interrupts caused by reasons 2 and 3 above may be uniquely identified with the aid of the Sense Interrupt Status Command. This command when issued resets the interrupt signal and via bits 5, 6, and 7 of Status Register 0 identifies the cause of the interrupt.

SEEK END BIT 5	INTERRUPT CODE BIT 6	INTERRUPT CODE BIT 7	CAUSE
0	1	1	Ready Line changed state, either polarity
1	0	0	Normal Termination of Seek or Recalibrate Command
1	1	0	Abnormal Termination of Seek or Recalibrate Command

Table 5

Neither the Seek or Recalibrate Command have a Result Phase. Therefore, it is mandatory to use the Sense Interrupt Status Command after these commands to effectively terminate them and to provide verification of where the head is positioned (PCN).

Issuing Sense Interrupt Status Command without interrupt pending is treated as an invalid command.

SPECIFY

The Specify Command sets the initial values for each of the three internal timers. The HUT (Head Unload Time) defines the time from the end of the Execution Phase of one of the Read/Write Commands to the head unload state. This timer is programmable from 16 to 240 ms in increments of 16 ms (01 = 16 ms, 02 = 32 ms . . . . OF = 240 ms). The SRT (Step Rate Time) defines the time interval between adjacent step pulses. This timer is programmable from 1 to 16 ms in increments of 1 ms (F = 1 ms, E = 2 ms, D = 3 ms, etc.). The HLT (Head Load Time) defines the time between when the Head Load signal goes high and when the Read/Write operation starts. This timer is programmable from 2 to 254 ms in increments of 2 ms (01 = 2 ms, 02 = 4 ms, 03 = 6 ms . . . 7F = 254 ms).

The time intervals mentioned above are a direct function of the clock (CLK on pin 19). Times indicated above are for an 8 MHz clock, if the clock was reduced to 4 MHz (mini-floppy application) then all time intervals are increased by a factor of 2.

The choice of DMA or NON-DMA operation is made by the ND (NON-DMA) bit. When this bit is high (ND = 1) the NON-DMA mode is selected, and when ND = 0 the DMA mode is selected.

SENSE DRIVE STATUS

This command may be used by the processor whenever it wishes to obtain the status of the FDDs. Status Register 3 contains the Drive Status information stored internally in FDC registers.

INVALID

If an invalid command is sent to the FDC (a command not defined above), then the FDC will terminate the command after bits 7 and 6 of Status Register 0 are set to 1 and 0 respectively. No interrupt is generated by the μPD765 during this condition. Bit 6 and bit 7 (DIO and RQM) in the Main Status Register are both high ("1") indicating to the processor that the μPD765 is in the Result Phase and the contents of Status Register 0 (STO) must be read. When the processor reads Status Register 0 it will find a 80 hex indicating an invalid command was received.

A Sense Interrupt Status Command must be sent after a Seek or Recalibrate Interrupt, otherwise the FDC will consider the next command to be an Invalid Command.

In some applications the user may wish to use this command as a No-Op command, to place the FDC in a standby or no operation state.

BIT NO.	BIT NAME	BIT SYMBOL	DESCRIPTION
			STATUS REGISTER 0
D_7	Interrupt Code	IC	$D_7 = 0$ and $D_6 = 0$ Normal Termination of Command, (NT). Command was completed and properly executed.
D_6			$D_7 = 0$ and $D_6 = 1$ Abnormal Termination of Command, (AT). Execution of Command was started, but was not successfully completed.
			$D_7 = 1$ and $D_6 = 0$ Invalid Command issue, (IC). Command which was issued was never started.
			$D_7 = 1$ and $D_6 = 1$ Abnormal Termination because during command execution the ready signal from FDD changed state.
D_5	Seek End	SE	When the FDC completes the SEEK Command, this flag is set to 1 (high).
D_4	Equipment Check	EC	If a fault Signal is received from the FDD, or if the Track 0 Signal fails to occur after 77 Step Pulses (Recalibrate Command) then this flag is set.
D_3	Not Ready	NR	When the FDD is in the not-ready state and a read or write command is issued, this flag is set. If a read or write command is issued to Side 1 of a single sided drive, then this flag is set.
D_2	Head Address	HD	This flag is used to indicate the state of the head at Interrupt.
D_1	Unit Select 1	US 1	These flags are used to indicate a Drive Unit.
D_0	Unit Select 0	US 0	Number at Interrupt.
			STATUS REGISTER 1
D_7	End of Cylinder	EN	When the FDC tries to access a Sector beyond the final Sector of a Cylinder, this flag is set.
D_6			Not used. This bit is always 0 (low).
D_5	Data Error	DE	When the FDC detects a CRC error in either the ID field or the data field, this flag is set.
D_4	Over Run	OR	If the FDC is not serviced by the main-systems during data transfers, within a certain time interval, this flag is set.
D_3			Not used. This bit always 0 (low).
D_2	No Data	ND	During execution of READ DATA, WRITE DELETED DATA or SCAN Command, if the FDC cannot find the Sector specified in the IDR Register, this flag is set.
			During executing the READ ID Command, if the FDC cannot read the ID field without an error, then this flag is set.
			During the execution of the READ A Cylinder Command, if the starting sector cannot be found, then this flag is set.

µPD765A

BIT NO.	BIT NAME	BIT SYMBOL	DESCRIPTION
STATUS REGISTER 1 (CONT.)			
D_1	Not Writable	NW	During execution of WRITE DATA, WRITE DELETED DATA or Format A Cylinder Command, if the FDC detects a write protect signal from the FDD, then this flag is set.
D_0	Missing Address Mark	MA	If the FDC cannot detect the ID Address Mark after encountering the index hole twice, then this flag is set.
			If the FDC cannot detect the Data Address Mark or Deleted Data Address Mark, this flag is set. Also at the same time, the MD (Missing Address Mark in Data Field) of Status Register 2 is set.
STATUS REGISTER 2			
D_7			Not used. This bit is always 0 (low).
D_6	Control Mark	CM	During executing the READ DATA or SCAN Command, if the FDC encounters a Sector which contains a Deleted Data Address Mark, this flag is set.
D_5	Data Error in Data Field	DD	If the FDC detects a CRC error in the data field then this flag is set.
D_4	Wrong Cylinder	WC	This bit is related with the ND bit, and when the contents of C on the medium is different from that stored in the IDR, this flag is set.
D_3	Scan Equal Hit	SH	During execution, the SCAN Command, if the condition of "equal" is satisfied, this flag is set.
D_2	Scan Not Satisfied	SN	During executing the SCAN Command, if the FDC cannot find a Sector on the cylinder which meets the condition, then this flag is set.
D_1	Bad Cylinder	BC	This bit is related with the ND bit, and when the content of C on the medium is different from that stored in the IDR and the content of C is FF, then this flag is set.
D_0	Missing Address Mark in Data Field	MD	When data is read from the medium, if the FDC cannot find a Data Address Mark or Deleted Data Address Mark, then this flag is set.
STATUS REGISTER 3			
D_7	Fault	FT	This bit is used to indicate the status of the Fault signal from the FDD.
D_6	Write Protected	WP	This bit is used to indicate the status of the Write Protected signal from the FDD.
D_5	Ready	RY	This bit is used to indicate the status of the Ready signal from the FDD.
D_4	Track 0	T0	This bit is used to indicate the status of the Track 0 signal from the FDD.
D_3	Two Side	TS	This bit is used to indicate the status of the Two Side signal from the FDD.
D_2	Head Address	HD	This bit is used to indicate the status of Side Select signal to the FDD.
D_1	Unit Select 1	US 1	This bit is used to indicate the status of the Unit Select 1 signal to the FDD.
D_0	Unit Select 0	US 0	This bit is used to indicate the status of the Unit Select 0 signal to the FDD.

µPD765A

PIN IDENTIFICATION

PIN NO.	PIN SYMBOL	PIN NAME	INPUT/ OUTPUT	CONNECTION TO	FUNCTION
1	RST	Reset	Input	Processor	Places FDC in idle state. Resets output lines to FDD to "0" (low). Does not effect SRT, HUT or HLT in Specify command. If RDY pin is held high during Reset, FDC will generate Interrupt 1-25 ms later. To clear this interrupt use Sense Interrupt Status command.
2	$\overline{RD}$	Read	Input (1)	Processor	Control signal for transfer of data from FDC to Data Bus, when "0" (low).
3	$\overline{WR}$	Write	Input (1)	Processor	Control signal for transfer of data to FDC via Data Bus, when "0" (low).
4	$\overline{CS}$	Chip Select	Input	Processor	IC selected when "0" (low), allowing $\overline{RD}$ and $\overline{WR}$ to be enabled.
5	A_0	Data/Status Reg Select	Input (1)	Processor	Selects Data Reg (A_0=1) or Status Reg (A_0=0) contents of the FDC to be sent to Data Bus.
6-13	DB_0-DB_7	Data Bus	Input/ Output (1)	Processor	Bi-Directional 8-Bit Data Bus.
14	DRQ	Data DMA Request	Output	DMA	DMA Request is being made by FDC when DRQ="1".
15	$\overline{DACK}$	DMA Acknowledge	Input	DMA	DMA cycle is active when "0" (low) and Controller is performing DMA transfer.
16	TC	Terminal Count	Input	DMA	Indicates the termination of a DMA transfer when "1" (high). It terminates data transfer during Read/Write/Scan command in DMA or interrupt mode.
17	IDX	Index	Input	FDD	Indicates the beginning of a disk track.
18	INT	Interrupt	Output	Processor	Interrupt Request Generated by FDC.
19	CLK	Clock	Input		Single Phase 8 MHz Squarewave Clock.
20	GND	Ground			D.C. Power Return.
21	WCK	Write Clock	Input		Write data rate to FDD. FM = 500 kHz, MFM = 1 MHz, with a pulse width of 250 ns for both FM and MFM.
22	RDW	Read Data Window	Input	Phase Lock Loop	Generated by PLL, and used to sample data from FDD.
23	RDD	Read Data	Input	FDD	Read data from FDD, containing clock and data bits.
24	VCO	VCO Sync	Output	Phase Lock Loop	Inhibits VCO in PLL when "0" (low), enables VCO when "1."
25	WE	Write Enable	Output	FDD	Enables write data into FDD.
26	MFM	MFM Mode	Output	Phase Lock Loop	MFM mode when "1," FM mode when "0."
27	HD	Head Select	Output	FDD	Head 1 selected when "1" (high). Head 0 selected when "0" (low).
28,29	US_1,US_0	Unit Select	Output	FDD	FDD Unit Selected.
30	WDA	Write Data	Output	FDD	Serial clock and data bits to FDD.
31,32	PS_1,PS_0	Precompensation (pre-shift)	Output	FDD	Write precompensation status during MFM mode. Determines early, late, and normal times.
33	FLT/TR_0	Fault/Track 0	Input	FDD	Senses FDD fault condition, in Read/Write mode; and Track 0 condition in Seek mode.
34	WP/TS	Write Protect/ Two-Side	Input	FDD	Senses Write Protect status in Read/Write mode; and Two Side Media in Seek mode.
35	RDY	Ready	Input	FDD	Indicates FDD is ready to send or receive data.
36	HDL	Head Load	Output	FDD	Command which causes read/write head in FDD to contact diskette.
37	FR/STP	Flt Reset/Step	Output	FDD	Resets fault F.F. in FDD in Read/Write mode, contains step pulses to move head to another cylinder in Seek mode.
38	LCT/DIR	Low Current/ Direction	Output	FDD	Lowers Write current on inner tracks in Read/Write mode, determines direction head will step in Seek mode. A fault reset pulse is issued at the beginning of each Read or Write command prior to the occurrence of the Head Load signal.
39	$\overline{RW}$/SEEK	Read Write/SEEK	Output	FDD	When "1" (high) Seek mode selected and when "0" (low) Read/Write mode selected.
40	V_{CC}	+5V			DC Power.

Note: (1) Disabled when CS = 1.

CAPACITANCE

$T_a = 25°C$; $f_c = 1$ MHz; $V_{CC} = 0V$

PARAMETER	SYMBOL	LIMITS MIN	LIMITS TYP	LIMITS MAX	UNIT	TEST CONDITIONS
Clock Input Capacitance	$C_{IN(\Phi)}$			20	pF	All Pins Except Pin Under Test Tied to AC Ground
Input Capacitance	C_{IN}			10	pF	
Output Capacitance	C_{OUT}			20	pF	

General Description (Continued)

The Z-80 DMA contains direct interfacing to and independent control of system buses, as well as sophisticated bus and interrupt controls. Many programmable features, including variable cycle timing and auto-restart, minimize CPU software overhead. They are especially useful in adapting this special-purpose transfer processor to a broad variety of memory, I/O and CPU environments.

The Z-80 DMA is an n-channel silicon-gate depletion-load device packaged in a 40-pin plastic or ceramic DIP. It uses a single +5 V power supply and the standard Z-80 Family single-phase clock.

Functional Description

Classes of Operation. The Z-80 DMA has three basic classes of operation:

- Transfers of data between two ports (memory or I/O peripheral)
- Searches for a particular 8-bit maskable byte at a single port in memory or an I/O peripheral
- Combined transfers with simultaneous search between two ports

Figure 4 illustrates the basic functions served by these classes of operation.

During a transfer, the DMA assumes control of the system address and data buses. Data is read from one addressable port and written to the other addressable port, byte by byte. The ports may be programmed to be either system main memory or peripheral I/O devices. Thus, a block of data may be written from one peripheral to another, from one area of main memory to another, or from a peripheral to main memory and vice versa.

During a search-only operation, data is read from the source port and compared byte by byte with a DMA-internal register containing a programmable match byte. This match byte may optionally be masked so that only certain bits within the match byte are compared. Search rates up to 1.25M bytes per second can be obtained with the 2.5 MHz Z-80 DMA or 2M bytes per second with the 4 MHz Z-80A DMA.

In combined searches and transfers, data is transferred between two ports while simultaneously searching for a bit-maskable byte match.

Data transfers or searches can be programmed to stop or interrupt under various conditions. In addition, CPU-readable status bits can be programmed to reflect the condition.

Modes of Operation. The Z-80 DMA can be programmed to operate in one of three transfer and/or search modes:

- *Byte-at-a-Time:* data operations are performed one byte at a time. Between each byte operation the system buses are released to the CPU. The buses are requested again for each succeeding byte operation.
- *Burst:* data operations continue until a port's Ready line to the DMA goes inactive. The DMA then stops and releases the system buses after completing its current byte operation.
- *Continuous:* data operations continue until the end of the programmed block of data is reached before the system buses are released. If a port's Ready line goes inactive before this occurs, the DMA simply pauses until the Ready line comes active again.

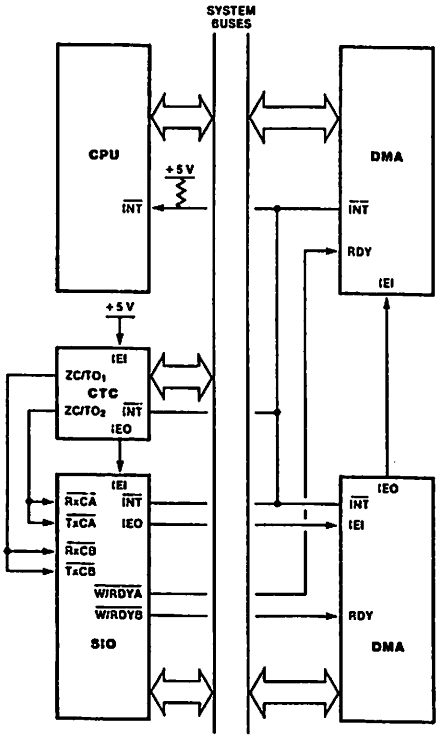


Figure 3. Typical Z-80 Environment

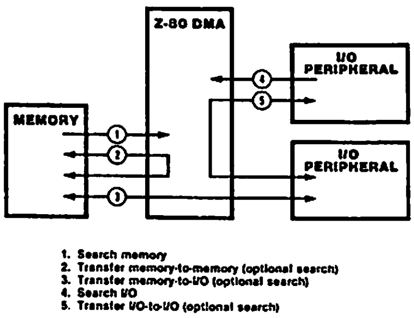


Figure 4. Basic Functions of the Z-80 DMA

Functional Description (Continued)

In all modes, once a byte of data is read into the DMA, the operation on the byte will be completed in an orderly fashion, regardless of the state of other signals (including a port's Ready line).

Due to the DMA's high-speed buffered method of reading data, operations on one byte are not completed until the next byte is read in. This means that total transfer or search block lengths must be two or more bytes, and that block lengths programmed into the DMA must be one byte less than the desired block length (count is N-1 where N is the block length).

Commands and Status. The Z-80 DMA has several writable control registers and readable status registers available to the CPU. Control bytes can be written to the DMA whenever the DMA is not controlling the system buses, but the act of writing a control byte to the DMA disables the DMA until it is again enabled by a specific command. Status bytes can also be read at any such time, but writing the Read Status Byte command or the Initiate Read Sequence command disables the DMA.

Control bytes to the DMA include those which effect immediate command actions such as enable, disable, reset, load starting-address buffers, continue, clear counters, clear status bits and the like. In addition, many mode-setting control bytes can be written, including mode and class of operation, port configuration, starting addresses, block length, address counting rule, match and match-mask byte, interrupt conditions, interrupt vector, status-affects-vector condition, pulse counting, auto restart, Ready-line and Wait-line rules, and read mask.

Readable status registers include a general status byte reflecting Ready-line, end-of-block, byte-match and interrupt conditions, as well as 2-byte registers for the current byte count, Port A address and Port B address.

Variable Cycle. The Z-80 DMA has the unique feature of programmable operation-cycle length. This is valuable in tailoring the DMA to the particular requirements of other system components (fast or slow) and maximizes the data-transfer rate. It also eliminates external logic for signal conditioning.

There are two aspects to the variable cycle feature. First, the entire read and write cycles (periods) associated with the source and destination ports can be independently programmed as 2, 3 or 4 T-cycles long (more if Wait cycles are used), thereby increasing or decreasing the speed with which all DMA signals change (Figure 5).

Second, the four signals in each port specifically associated with transfers of data (I/O Request, Memory Request, Read, and Write) can each have its active trailing edge terminated one-half T-cycle early. This adds a further dimension of flexibility and speed, allowing such things as shorter-than-normal Read or Write signals that go inactive before data starts to change.

Address Generation. Two 16-bit addresses are generated by the Z-80 DMA for every transfer operation, one address for the source port and another for the destination port. Each address can be either variable or fixed. Variable addresses can increment or decrement from the programmed starting address. The fixed-address capability eliminates the need for separate enabling wires to I/O ports.

Port addresses are multiplexed onto the system address bus, depending on whether the DMA is reading the source port or writing to the destination port. Two readable address counters (2 bytes each) keep the current address of each port.

Auto Restart. The starting addresses of either port can be reloaded automatically at the end of a block. This option is selected by the Auto Restart control bit. The byte counter is cleared when the addresses are reloaded.

The Auto Restart feature relieves the CPU of software overhead for repetitive operations such as CRT refresh and many others. Moreover, when the CPU has access to the buses during byte-at-a-time or burst transfers, different starting addresses can be written into buffer registers during transfers, causing the Auto Restart to begin at a new location.

Interrupts. The Z-80 DMA can be programmed to interrupt the CPU on three conditions:

- Interrupt on Ready (before requesting bus)
- Interrupt on Match
- Interrupt on End of Block

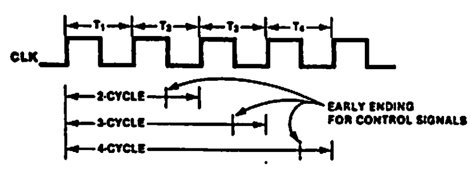


Figure 5. Variable Cycle Length

Functional Description (Continued)

Any of these interrupts cause an interrupt-pending status bit to be set, and each of them can optionally alter the DMA's interrupt vector. Due to the buffered constraint mentioned under "Modes of Operation," interrupts on Match at End of Block are caused by matches to the byte just prior to the last byte in the block.

The DMA shares the Z-80 Family's elaborate interrupt scheme, which provides fast interrupt service in real-time applications. In a Z-80 CPU environment, the DMA passes its internally modifiable 8-bit interrupt vector to the CPU, which adds an additional eight bits to form the memory address of the interrupt-routine table. This table contains the address of the beginning of the interrupt routine itself. In this process, CPU control is transferred directly to the interrupt routine, so that the next instruction executed after an interrupt acknowledge is the first instruction of the interrupt routine itself.

Pulse Generation. External devices can keep track of how many bytes have been transferred by using the DMA's pulse output, which provides a signal at 256-byte intervals. The interval sequence may be offset at the beginning by 1 to 255 bytes.

The Interrupt line outputs the pulse signal in a manner that prevents misinterpretation by the CPU as an interrupt request, since it only appears when the Bus Request and Bus Acknowledge lines are both active.

Pin Description

A_0-A_{15}. *System Address Bus* (output, 3-state). Addresses generated by the DMA are sent to both source and destination ports (main memory or I/O peripherals) on these lines.

$\overline{\text{BAI}}$. *Bus Acknowledge In* (input, active Low). Signals that the system buses have been released for DMA control. In multiple-DMA configurations, the $\overline{\text{BAI}}$ pin of the highest priority DMA is normally connected to the Bus Acknowledge pin of the CPU. Lower-priority DMAs have their $\overline{\text{BAI}}$ connected to the $\overline{\text{BAO}}$ of a higher-priority DMA.

$\overline{\text{BAO}}$. *Bus Acknowledge Out* (output, active Low). In a multiple-DMA configuration, this pin signals that no other higher-priority DMA has requested the system buses. $\overline{\text{BAI}}$ and $\overline{\text{BAO}}$ form a daisy chain for multiple-DMA priority resolution over bus control.

$\overline{\text{BUSREQ}}$. *Bus Request* (bidirectional, active Low, open drain). As an output, it sends requests for control of the system address bus, data bus and control bus to the CPU. As an input, when multiple DMAs are strung together in a priority daisy chain via $\overline{\text{BAI}}$ and $\overline{\text{BAO}}$, it senses when another DMA has requested the buses and causes this DMA to refrain from bus requesting until the other DMA is finished. Because it is a bidirectional pin, there cannot be any buffers between this DMA and any other DMA. It can, however, have a buffer between it and the CPU because it is unidirectional into the CPU. A pull-up resistor is connected to this pin.

$\overline{\text{CE}}/\overline{\text{WAIT}}$. *Chip Enable and Wait* (input, active Low). Normally this functions only as a $\overline{\text{CE}}$ line, but it can also be programmed to serve a $\overline{\text{WAIT}}$ function. As a $\overline{\text{CE}}$ line from the CPU, it becomes active when $\overline{\text{WR}}$ and $\overline{\text{IORQ}}$ are active and the I/O port address on the system address bus is the DMA's address, thereby allowing a transfer of control or command bytes from the CPU to the DMA. As a $\overline{\text{WAIT}}$ line from memory or I/O devices, after the DMA has received a bus-request acknowledge from the CPU, it causes wait states to be inserted in the DMA's operation cycles thereby slowing the DMA to a speed that matches the memory or I/O device.

CLK. *System Clock* (input). Standard Z-80 single-phase clock at 2.5 MHz (Z-80 DMA) or 4.0 MHz (Z-80A DMA). For slower system clocks, a TTL gate with a pullup resistor may be adequate to meet the timing and voltage level specification. For higher-speed systems, use a clock driver with an active pullup to meet the V_{IH} specification and risetime requirements. In all cases there should be a resistive pullup to the power supply of 10K ohms (max) to ensure proper power when the DMA is reset.

D_0-D_7. *System Data Bus* (bidirectional, 3-state). Commands from the CPU, DMA status, and data from memory or I/O peripherals are transferred on these lines.

IEI. *Interrupt Enable In* (input, active High). This is used with IEO to form a priority daisy chain when there is more than one interrupt-driven device. A High on this line indicates that no other device of higher priority is being serviced by a CPU interrupt service routine.

IEO. *Interrupt Enable Out* (output, active High). IEO is High only if IEI is High and the CPU is not servicing an interrupt from this DMA. Thus, this signal blocks lower-priority devices from interrupting while a higher-priority device is being serviced by its CPU interrupt service routine.

Pin Description (Continued)

$\overline{\text{INT}}$/PULSE. *Interrupt Request* (output, active Low, open drain). This requests a CPU interrupt. The CPU acknowledges the interrupt by pulling its $\overline{\text{IORQ}}$ output Low during an $\overline{\text{M1}}$ cycle. It is typically connected to the $\overline{\text{INT}}$ pin of the CPU with a pullup resistor and tied to all other $\overline{\text{INT}}$ pins in the system. This pin can also be used to generate periodic pulses to an external device. It can be used this way only when the DMA is bus master (i.e., the CPU's $\overline{\text{BUSREQ}}$ and $\overline{\text{BUSACK}}$ lines are both Low and the CPU cannot see interrupts).

$\overline{\text{IORQ}}$. *Input/Output Request* (bidirectional, active Low, 3-state). As an input, this indicates that the lower half of the address bus holds a valid I/O port address for transfer of control or status bytes from or to the CPU, respectively; this DMA is the addressed port if its $\overline{\text{CE}}$ pin and its $\overline{\text{WR}}$ or $\overline{\text{RD}}$ pins are simultaneously active. As an output, after the DMA has taken control of the system buses, it indicates that the 8-bit or 16-bit address bus holds a valid port address for another I/O device involved in a DMA transfer of data. When $\overline{\text{IORQ}}$ and $\overline{\text{M1}}$ are both active simultaneously, an interrupt acknowledge is indicated.

$\overline{\text{M1}}$. *Machine Cycle One* (input, active Low). Indicates that the current CPU machine cycle is an instruction fetch. It is used by the DMA to decode the return-from-interrupt instruction (RETI) (ED-4D) sent by the CPU. During two-byte instruction fetches, $\overline{\text{M1}}$ is active as each opcode byte is fetched. An interrupt acknowledge is indicated when both $\overline{\text{M1}}$ and $\overline{\text{IORQ}}$ are active.

$\overline{\text{MREQ}}$. *Memory Request* (output, active Low, 3-state). This indicates that the address bus holds a valid address for a memory read or write operation. After the DMA has taken control of the system buses, it indicates a DMA transfer request from or to memory.

$\overline{\text{RD}}$. *Read* (bidirectional, active Low, 3-state). As an input, this indicates that the CPU wants to read status bytes from the DMA's read registers. As an output, after the DMA has taken control of the system buses, it indicates a DMA-controlled read from a memory or I/O port address.

RDY. *Ready* (input, programmable active Low or High). This is monitored by the DMA to determine when a peripheral device associated with a DMA port is ready for a read or write operation. Depending on the mode of DMA operation (Byte, Burst or Continuous), the RDY line indirectly controls DMA activity by causing the $\overline{\text{BUSREQ}}$ line to go Low or High.

$\overline{\text{WR}}$. *Write* (bidirectional, active Low, 3-state). As an input, this indicates that the CPU wants to write control or command bytes to the DMA write registers. As an output, after the DMA has taken control of the system buses, it indicates a DMA-controlled write to a memory or I/O port address.

Internal Structure

The internal structure of the Z-80 DMA includes driver and receiver circuitry for interfacing with an 8-bit system data bus, a 16-bit system address bus, and system control lines (Figure 6). In a Z-80 CPU environment, the DMA can be tied directly to the analogous pins on the CPU (Figure 7) with no additional buffering, except for the $\overline{\text{CE}}/\overline{\text{WAIT}}$ line.

The DMA's internal data bus interfaces with the system data bus and services all internal logic and registers. Addresses generated from this logic for Ports A and B (source and destination) of the DMA's single transfer channel are multiplexed onto the system address bus.

Specialized logic circuits in the DMA are dedicated to the various functions of external bus interfacing, internal bus control, byte matching, byte counting, periodic pulse generation, CPU interrupts, bus requests, and address generation. A set of twenty-one writable control registers and seven readable status registers provides the means by which the CPU governs and monitors the activities of these logic circuits. All registers are eight bits wide, with double-byte information stored in adjacent registers. The two address counters (two bytes each) for Ports A and B are buffered by the two starting addresses.

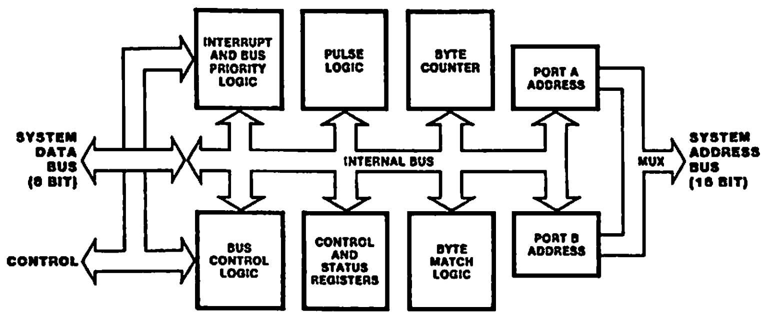


Figure 6. Block Diagram

Internal Structure (Continued)

The 21 writable control registers are organized into seven base-register groups, most of which have multiple registers. The base registers in each writable group contain both control/command bits and pointer bits that can be set to address other registers within the group. The seven readable status registers have no analogous second-level registers.

The registers are designated as follows, according to their base-register groups:

WR0-WR6 — Write Register groups 0 through 6 (7 base registers plus 14 associated registers)
RR0-RR6 — Read Registers 0 through 6

Writing to a register within a write-register group involves first writing to the base register, with the appropriate pointer bits set, then writing to one or more of the other registers within the group. All seven of the readable status registers are accessed sequentially according to a programmable mask contained in one of the writable registers. The section entitled "Programming" explains this in more detail.

A pipelining scheme is used for reading data in. The programmed block length is the number of bytes compared to the byte counter, which increments at the end of each cycle. In searches, data byte comparisons with the match byte are made during the read cycle of the next byte. Matches are, therefore, discovered only after the next byte is read in.

In multiple-DMA configurations, interrupt-request daisy chains are prioritized by the order in which their IEI and IEO lines are connected (Zilog Application Note 03-0041-01, *The Z-80 Family Program Interrupt Structure*). The system bus, however, may not be pre-empted. Any DMA that gains access to the system bus keeps the bus until it is finished.

Write Registers

WR0	Base register byte Port A starting address (low byte) Port A starting address (high byte) Block length (low byte) Block length (high byte)
WR1	Base register byte Port A variable-timing byte
WR2	Base register byte Port B variable-timing byte
WR3	Base register byte Mask byte Match byte
WR4	Base register byte Port B starting address (low byte) Port B starting address (high byte) Interrupt control byte Pulse control byte Interrupt vector
WR5	Base register byte
WR6	Base register byte Read mask

Read Registers

RR0	Status byte
RR1	Byte counter (low byte)
RR2	Byte counter (high byte)
RR3	Port A address counter (low byte)
RR4	Port A address counter (high byte)
RR5	Port B address counter (low byte)
RR6	Port B address counter (high byte)

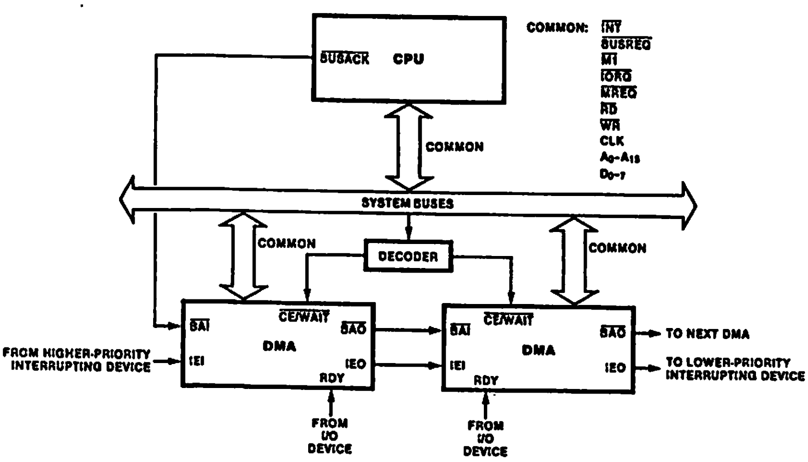


Figure 7. Multiple-DMA Interconnection to the Z-80 CPU

Programming The Z-80 DMA has two programmable fundamental states: (1) an enabled state, in which it can gain control of the system buses and direct the transfer of data between ports, and (2) a disabled state, in which it can initiate neither bus requests nor data transfers. When the DMA is powered up or reset by any means, it is automatically placed into the disabled state. Program commands can be written to it by the CPU in either state, but this automatically puts the DMA in the disabled state, which is maintained until an enable command is issued by the CPU. The CPU must program the DMA in advance of any data search or transfer by addressing it as an I/O port and sending a sequence of control bytes using an Output instruction (such as OTIR for the Z-80 CPU).

Writing. Control or command bytes are written into one or more of the Write Register groups (WR0-WR6) by first writing to the base register byte in that group. All groups have base registers and most groups have additional associated registers. The associated registers in a group are sequentially accessed by first writing a byte to the base register containing register-group identification and pointer bits (1's) to one or more of that base register's associated registers.

This is illustrated in Figure 8b. In this figure, the sequence in which associated registers within a group can be written to is shown by the vertical position of the associated registers. For example, if a byte written to the DMA contains the bits that identify WR0 (bits D0, D1 and D7), and also contains 1's in the bit positions that point to the associated "Port A Starting Address (low byte)" and "Port A Starting Address (high byte)," then the next two bytes written to the DMA will be stored in these two registers, in that order.

Reading. The Read Registers (RR0-RR6) are read by the CPU by addressing the DMA as an I/O port using an Input instruction (such as INIR for the Z-80 CPU). The readable bytes contain DMA status, byte-counter values, and port addresses since the last DMA reset. The registers are always read in a fixed sequence beginning with RR0 and ending with RR6. However, the register read in this sequence is determined by programming the Read Mask in WR6. The sequence of reading is initialized by writing an Initiate Read Sequence or Set Read Status command to WR6. After a Reset DMA, the sequence must be initialized with the Initiate Read Sequence command or a Read Status command. The sequence of reading all registers that are not excluded by the Read Mask register must be completed before a new Initiate Read Sequence or Read Status command.

Fixed-Address Programming. A special circumstance arises when programming a destination port to have a fixed address. The load command in WR6 only loads a fixed address to a port selected as the source, not to a port selected as the destination. Therefore, a fixed destination address must be loaded by temporarily declaring it a fixed-source address and subsequently declaring the true source as such, thereby implicitly making the other a destination.

The following example illustrates the steps in this procedure, assuming that transfers are to occur from a variable-address source (Port A) to a fixed-address destination (Port B):

1. Temporarily declare Port B as source in WR0.
2. Load Port B address in WR6.
3. Declare Port A as source in WR0.
4. Load Port A address in WR6.
5. Enable DMA in WR6.

Figure 9 illustrates a program to transfer data from memory (Port A) to a peripheral device (Port B). In this example, the Port A memory starting address is 1050_H and the Port B peripheral fixed address is 05_H. Note that the data flow is 1001_H bytes—one more than specified by the block length. The table of DMA commands may be stored in consecutive memory locations and transferred to the DMA with an output instruction such as the Z-80 CPU's OTIR instruction.

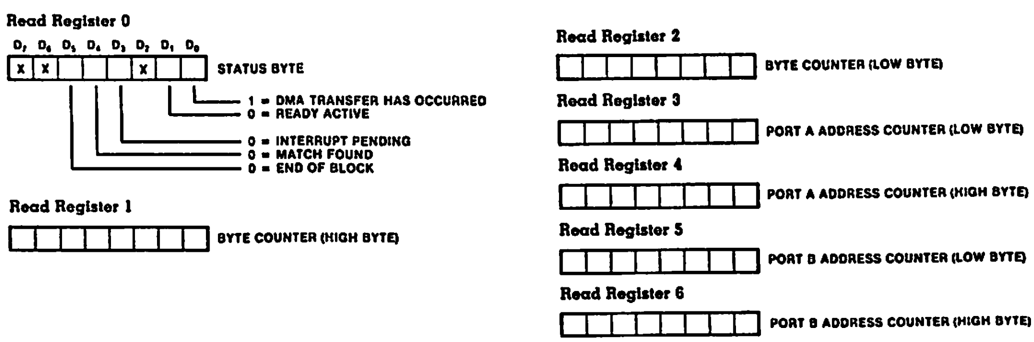


Figure 8a. Read Registers

Programming
(Continued)

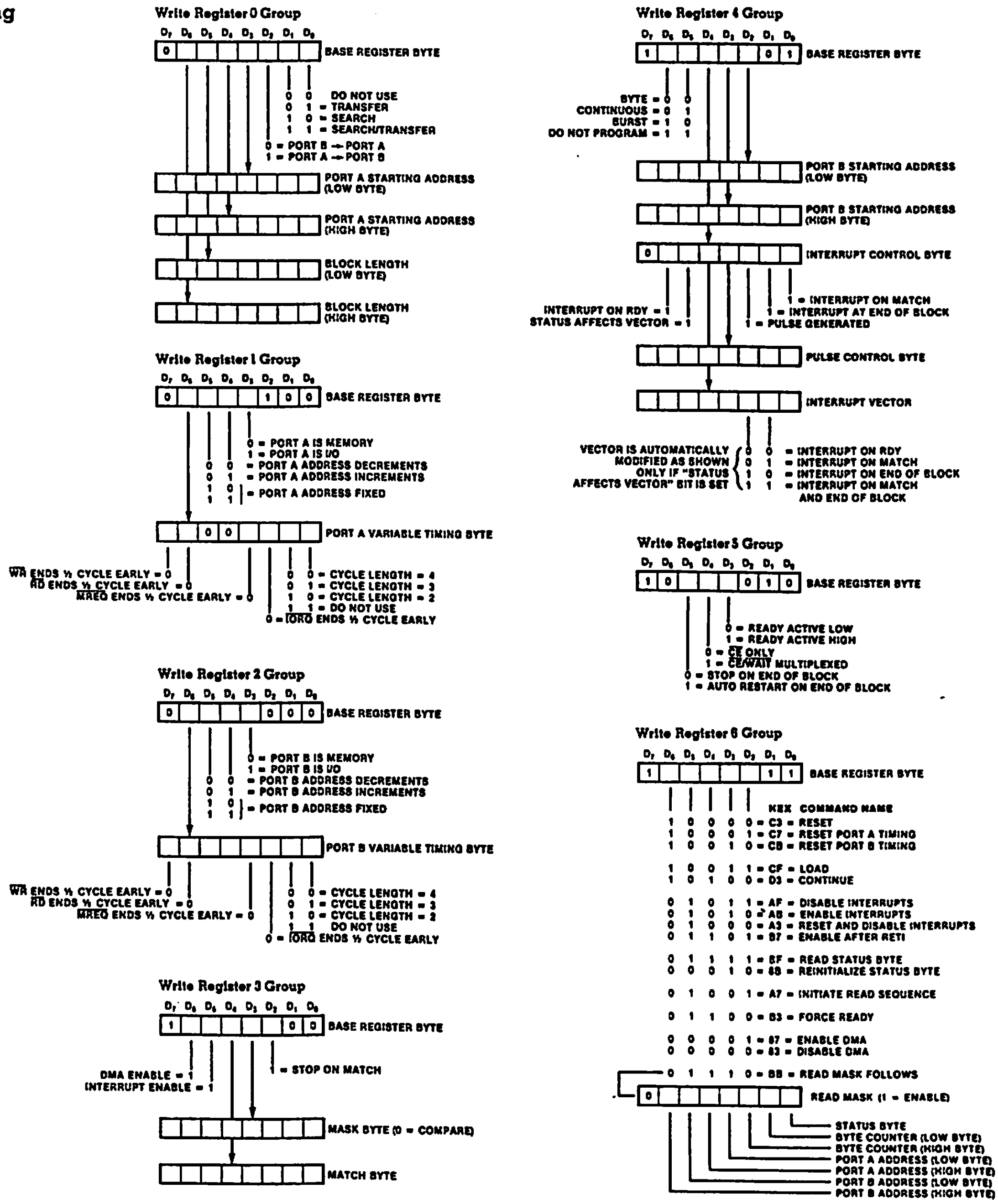


Figure 8b. Write Registers

Programm Beispiel:

Die im Folgendem aufgeführten Programmbeispiele sollen die prinzipielle Programmierung des FDC-Moduls verdeutlichen. Das Beispiel zeigt die Ansteuerung eines Mini-Laufwerks vom Typ Teac FD-55B mit der Laufwerksnummer A. Der Interrupt der DMA wird zur Erzeugung des Terminal-Count-Pulses beim Lese- oder Schreib Transfer genutzt.

```
0100                ORG 100H
                    MACLIB Z80

                    ;         CHIP BASE
                    ;         =========

0020 =              P$FDC$PIO         EQU      20H
0024 =              P$FDC$765         EQU      24H
0026 =              P$FDC$DMA         EQU      26H
0027 =              P$FDC$MOT         EQU      27H

                    ;         PORTADRESS EQUATES
                    ;         ==================

0024 =              P$FDC$765$CMD     EQU P$FDC$765+0H ; 765 COMMAND PORT
0025 =              P$FDC$765$DAT     EQU P$FDC$765+1H ; 765 DATA PORT

0020 =              P$FDC$CTR$DAT     EQU P$FDC$PIO+0H ; PIO PORTA-DATA
0022 =              P$FDC$CTR$MODE    EQU P$FDC$PIO+2H ; PIO PORTA-CTR
0021 =              P$FDC$BANK$DAT    EQU P$FDC$PIO+1H ; BANK PIO PORTB-DATA
0023 =              P$FDC$BANK$MODE   EQU P$FDC$PIO+3H ; BANK PIO PORTB-CTR

                    ;BANKPORT(DMA):    PB0-PB3 = BANKADRESS FOR RAED
                    ;                  PB4-PB7 = BANKADRESS FOR WRITE (A16-A19)


                     ;**************** INIT FDC *********************
                     ;**********************************************

                    INITFDC:

                     ; PORT A Z-80 PIO

0100 3EFFD322        MVI A,0FFH ! OUT P$FDC$CTR$MODE ; MODE CTR. WORD, MODE 3
0104 3E43D322        MVI A,43H ! OUT P$FDC$CTR$MODE  ; I/O CTR. WORD
0108 3E03D322        MVI A,03H ! OUT P$FDC$CTR$MODE  ; INTERRUPT DISABLE WORD

                     ; PORT B Z-80 PIO

010C 3E0FD323        MVI A,0FH ! OUT P$FDC$BANK$MODE
0110 3E00D320        MVI A,0H ! OUT P$FDC$CTR$DAT    ; SELECT DRIVE A, MOTOR ON,
                                                     ; MINIDRIVE 250/125 kHz
```



```
                ;INIT UPD 765

                INI765:

0114 0600        MVI B,0
                 DJNZ INI765+2                     ; DELAY
0116+10FE        DB      10H,INI765+2-$-1
0118 DB24FE80    IN P$FDC$765$CMD ! CPI  80H        ; REQUEST FOR MASTER ?
                 JRZ SPECIFY                        ; NO -> SKIP
011C+2804        DB      28H,SPECIFY-$-1
011E DB25        IN P$FDC$765$DAT                   ; ELSE READ DATA
                 JR INI765                          ; AND TRY IT AGAIN
0120+18F2        DB      18H,INI765-$-1

                SPECIFY:

0122 214401      LXI H,SPEC5-1                      ; SPECIFY PARAMETER FD55-B DRIVE
0125 010303      LXI B,0303H                        ; SPECIFY CMD. 3 BYTES
0128 CD7C01      CALL CMDFD1                        ; SEND COMMAND

                ; RECALIBRATE DRIVE TO TRACK ZERO

                RECAL:

012B 010702      LXI B,0207H                        ; RECAL.CMD =07 - 2 BYTES
012E CD6E01      CALL MOTO

                ; SENSE DRIVE INTERUPT STATUS

0131 010801     SENSE:  LXI B,0108H                 ; COMMAND SENSED.= 08 - 1 BYTE
0134 CD7901      CALL   CMDFD
0137 CD8C01      CALL   NEXT                        ; FETCH RESULT
013A 47          MOV    B,A                         ; SAVE IT
013B FE80        CPI    80H                         ; INVALID CMD.?
013D C48C01      CNZ    NEXT                        ; NO -> NEXT RESULT
                 BIT    5,B                         ; SEEK FINISHED ?
0140+CB68        DB     0CBH,5*8+B+40H
                 JRZ    SENSE                       ; NO -> LOOP BACK
0142+28ED        DB     28H,SENSE-$-1
0144 C9          RET                                ; YES -> ALL DONE


                ;*********** SPECIFY PARAMETER FOR TEAC FD-55B ***************
                ;*************************************************************

0145 DF         SPEC5    DB      1101$1111B
0146 12                  DB      0001$0010B         ;SPECIFY PARAM. 5 1/4 TEAC
                                                    ;HLT = 36 MS
                                                    ;HUT = 480 MS
                                                    ;SRT = 6 MS

                ;********** COMMAND TABLE FOR UPD 765 ***************
                ;****************************************************

0147 46         CMDTAB:  DB      046H     ; MFM READ COMMAND
0148 00         UNIT:    DB      0        ; HEAD0/UNIT A
0149 00         TRACK:   DB      0        ; TRACK 0
014A 00         HEAD:    DB      0        ; HEAD 0
014B 01         SECTOR:  DB      1        ; SECTOR 1
014C 02         N:       DB      2        ; N=2 512BYTES
014D 0A         EOT:     DB      10       ; EOT
014E 10         GPL:     DB      10H      ; GPL
014F FF         DTL:     DB      0FFH     ; DTL
```



```
                 ;********** COMMAND TABLE FOR DMA ****************
                 ;************************************************

                 DMACMDT:

0150 C3           DB      0C3H     ; WR6   RESET DMA
0151 79           DB      79H      ; WR0   TRANSFER B->A

                 ZDMAMDMA:

0152 0030         DW      3000H    ; DESTINATION / SOURCE

                 DMALEN:

0154 FF01         DW      512-1    ; LENGTH
0156 14           DB      14H      ; WR1   A=MEM ADR. INCR
0157 C7           DB      0C7H     ; WR6   RESET PORT A TIMING
0158 28           DB      28H      ; WR2   B=I/O ADR. FIXED
0159 CB           DB      0CBH     ; WR6   RESET PORT B TIMING
015A 95           DB      95H      ; WR4   BYTE, START & INT FOLLOWS
015B 25           DB      P$FDC$765$DAT    ;PORTADR
015C 12           DB      12H      ; INTERRUPT AT END OF BLOCK
015D 00           DB      00       ; LOW INT.-VECTOR
015E 8A           DB      8AH      ; WR5   RDY ACTIV H ,CE ONLY, STOP
                                   ;       AT END OF BLOCK
015F CF           DB      0CFH     ; WR6   LOAD
0160 AB           DB      0ABH     ; ENABLE INT
0161 AB           DB      0ABH
0162 AB           DB      0ABH
0163 87           DB      87H      ; ENABLE DMA


                 ;*********** SENSE DRIVE STATUS ***************
                 ;**********************************************

                 SDRS:

0164 010402       LXI     B,0204H          ; SDRS.CMD = 04 - 2 BYTES
0167 CD7901       CALL    CMDFD
016A CD8C01       CALL    NEXT
016D C9           RET


                 ;******************* MOTOR ON *************************
                 ;WAIT UNTIL DISK READY, THEN TRANSMIT CMD. IN BC TO FDD *
                 ;********************************************************

                 MOTO:

016E C5           PUSH B                   ; SAVE CMD.

                 MOTO1:
016F DB27         IN P$FDC$MOT             ; FORCE MOTOR ON
0171 CD6401       CALL SDRS
                  BIT 5,A                  ; DRIVE READY ?
0174+CB6F         DB      0CBH,5*8+A+40H
                  JRZ MOTO1                ; NO->WAIT
0176+28F7         DB      28H,MOTO1-$-1
0178 C1           POP B
```

```
                ;*************** COMMAND TO UPD 765 ********************
                ;TRANSMITT COMAND IN BC TO FLOPPY DISK CONTROLER *
                ;B=NUMBER OF BYTES TO TRANSMITT, C = COMMAND
                ;*******************************************************

                CMDFD:
0179 214701      LXI H,CMDTAB

                CMDFD1:
017C DB24        IN P$FDC$765$CMD
017E E6C0FE80    ANI 0C0H ! CPI 80H             ; REQUEST FOR MASTER ?
                 JRNZ CMDFD1                    ; NO -> WAIT
0182+20F8        DB      20H,CMDFD1-$-1
0184 79D325      MOV A,C ! OUT P$FDC$765$DAT    ; SEND COMMAND
0187 234E        INX H ! MOV C,M ! DJNZ CMDFD1  ; NEXT COMMAND
0189+10F1        DB      10H,CMDFD1-$-1
018B C9          RET

                ;***************** NEXT RESULT ***********************
                ;          READ NEXT RESULT BYTE FROM FDC
                ;*****************************************************

                NEXT:

018C DB24        IN P$FDC$765$CMD
018E E6C0FEC0    ANI 0C0H ! CPI 0C0H
                 JRNZ NEXT                      ; REQUEST FOR MASTER ?
0192+20F8        DB      20H,NEXT-$-1
0194 DB25        IN P$FDC$765$DAT               ; GET RESULT IN A
0196 C9          RET

                ;********* INTERRUPT SERVICE ROUTINE FOR DMA *********
                ;*****************************************************

                TERMINI:

                 EXAF
0197+08          DB      08H
0198 3EC3D326    MVI A,0C3H! OUT P$FDC$DMA      ; RESET DMA
                 EXAF
019C+08          DB      08H
019D FBC9        EI ! RET

                ;************* FETCH RESULTS FROM FDC ****************
                ;          CHECK FOR R/W ERRORS, IF ANY Z=1
                ;*****************************************************

                RESULT:

019F 0606        MVI     B,6            ; 7 RESULTS
01A1 CD8C01      CALL    NEXT           ; FETCH FIRST
01A4 213302      LXI     H,RESLTB
01A7 77          MOV     M,A            ; SAVE IT
01A8 E6C0        ANI     0C0H           ; ANY ERROR IN STATUS 0
01AA CD9701      CALL    TERMINI        ; ABORT OPERATION
01AD 4F          MOV     C,A            ; SAVE STATUS 0
01AE CD8C01     RESLOP:  CALL    NEXT
01B1 2377        INX H ! MOV M,A        ; SAVE RESULT
                 DJNZ    RESLOP
01B3+10F9        DB      10H,RESLOP-$-1
01B5 79B7C9      MOV A,C ! ORA A ! RET  ; RET WITH ERROR IN A
```



```
                  ;************* SEEK ****************
                  ; SEEK TO TRACK IN A, DRIVE IN UNIT
                  ;**********************************

                  SEEK:
01B8 324901        STA TRACK
01BB 010F03        LXI B,030FH                ; COMMAND SEEK = 0F - 2 BYTES
01BE CD6E01        CALL MOTO
01C1 C33101        JMP SENSE


                  ; ************ READ & WRITE ENTRY POINTS ***************
                  ; R/W ONE SECTOR IN <SECTOR>, LENGTH IN <N> AND <DMALEN>
                  ; R/W TRACK IN <TRACK>, SEEK-COMMAND GIVEN
                  ; R/W HEAD IN <HEAD> AND <UNIT>
                  ; R/W DRIVE IN <UNIT>, SELECT IT TO PIO-PORT
                  ; DESTINATION/SOURCE IN <ZDMADMA>
                  ; ****************************************************

                  ;********** WRITE SELECTED SECTOR TO FDD *************
                  ;****************************************************

                  WRITEIT:
01C4 216001        LXI H,DMACMDT+10H
01C7 360523        MVI M,5 ! INX H            ; CODE: A -> B
01CA 36CF          MVI M,0CFH                 ; CODE: LOAD
01CC 3A4701F605 LDA CMDTAB ! ORI 5
                   JR RWIT                    ; COMMON CODE FOLLOWS
01D1+180D          DB      18H,RWIT-$-1


                  ;********** READ SELECTED RECORD FROM FDD ************
                  ;****************************************************

                  READIT:
01D3 216001        LXI H,DMACMDT+10H
01D6 36AB23        MVI M,0ABH ! INX H
01D9 36AB          MVI M,0ABH
01DB 3A4701F606 LDA CMDTAB ! ORI 6

                  RWIT:
01E0 323A02        STA     CMDCODE            ; COMAND CODE FOR READ/WRITE
01E3 060A          MVI     B,10               ; RETRY-COUNTER
                  RWOP:
01E5 C5            PUSH    B
01E6 AF            XRA A
01E7 D321          OUT P$FDC$BANK$DAT         ; READ/WRITE BANK 0
01E9 012614        LXI B,14H*256+P$FDC$DMA
01EC 215001        LXI     H,DMACMDT
                   OUTIR                      ; PUT CMD TO DMA
01EF+EDB3          DB      0EDH,0B3H
01F1 0609          MVI     B,9                ; 9 COMMANDS
01F3 3A3A02        LDA     CMDCODE
01F6 4F            MOV     C,A                ; COMMAND IN C
01F7 CD6E01        CALL    MOTO
01FA CD9F01        CALL    RESULT
01FD C1            POP     B
01FE C8            RZ                         ; NO ERRORS -> RET
                   DJNZ    RWOP               ; ERROR-> TRY IT AGAIN
01FF+10E4          DB      10H,RWOP-$-1
0201 C9            RET                        ; RET WITH Z=0
```



```
                ;******************* INTERBANK MOVE *******************
                ; PERFORMED BY Z80-DMA
                ; SOURCEBANK IN LOWBYTE, DESTINATIONBANK IN HIGHBYTE OF <A>
                ; DESTINATION ADRESS IN <HL>
                ; SOURCE ADRESS IN <DE>
                ; LENGTH IN <BC>
                ;*****************************************************

                INTER$MOVE:
0202 D321        OUT P$FDC$BANK$DAT              ; SELECT BANKS
0204 222102      SHLD    MOVE$DEST
                 SDED    MOVE$SRC
0207+ED53        DB      0EDH,53H
0209+2A02        DW      MOVE$SRC
020B 0B          DCX     B
                 SBCD    MOVE$LEN
020C+ED43        DB      0EDH,43H
020E+2302        DW      MOVE$LEN

                 ; PERFORM MOVE
0210 012614      LXI     B,1400H+P$FDC$DMA       ; C = DMA PORT , B = LENGTH
0213 211F02      LXI     H,DMAT2
                 OUTIR                           ; MOVE COMMANDS TO DMA
0216+EDB3        DB      0EDH,0B3H

                 ; WAIT TILL TRANSFER FINISHED
0218 DB20       DMARDY:  IN P$FDC$CTR$DAT
                 BIT 0,A                         ; INTERRUPT FROM DMA ?
021A+CB47        DB      0CBH,0*8+A+40H
                 JRZ DMARDY                      ; NO -> WAIT
021C+28FA        DB      28H,DMARDY-$-1
021E C9          RET


                ;************* DMA - INTERMOVE  COMMAND TABLE *************
                ;TRANSFER CHANNEL B (SOURCE) TO A (DEST)
                ;*********************************************************

021F C3         DMAT2:   DB      0C3H     ; RESET
0220 79          DB      79H      ; WR0 TRANSFER B->A
                MOVEDEST:
0221             DS      2        ; A STARTING ADRESS = DESTINATION
                MOVELEN:
0223             DS      2        ; BLOCK LENGTH -1
0225 54          DB      54H      ; WR1 A IS MEM  ADR INC , TIMING FOLLOWS
0226 02          DB      2        ; EARLY END ON IORQ,MREQ,RD,WR, 2T
0227 50          DB      50H      ; WR2 B IS MEM ADR INC , TIMING FOLLOWS
0228 02          DB      2
0229 BD          DB      0BDH     ; WR4 CONTINUOUS
                MOVESRC:
022A             DS      2        ; B STARTING ADDRESS = SOURCE
022C 12          DB      12H      ; INT AT END OF BLOCK
022D 00          DB      0        ; INTERRUPT VECTOR L
022E 82          DB      82H      ; STOP AT END OF BLOCK,CE-ONLY,RDY ACT LOW
022F B3          DB      0B3H     ; FORCE READY
0230 CF          DB      0CFH     ; LOAD
0231 AB          DB      0ABH     ; ENABLE INT
0232 87          DB      87H      ; ENABLE DMA


0233            RESLTB:  DS      7        ; RESULT TABLE FOR FDC
023A            CMDCODE: DS      1
023B             END
```


FDC1 - Bestückungsliste

IC's

IC 00,01	74 LS 245
IC 03	TBP 24SA10
IC 03,04	74 LS 245
IC 05	74 241
IC 06,07	74 LS 74
IC 08	74 LS 08
IC 09	TBP 24S10
IC 10	74 LS 74
IC 11	74 LS 04
IC 12	74 38
IC 13	Z-80 A DMA
IC 14	74 LS 257
IC 15	74 LS 123
IC 16	Z-80 B PIO
IC 17	upd 765
IC 18	74 LS 02
IC 19	74 LS 139
IC 20	74 S 163
IC 21	74 LS 240
IC 22	74 07
IC 23	74 LS 139
IC 24	74 07
IC 25	74 LS 14
IC 26	74 06
IC 27	FDC 9229 B
IC 28	74 S 04
IC 29	74 LS 279

Widerstände

R 00	10 KOhm
R 01	47 KOhm
R 02	330 Ohm
R 03,04	4,7 KOhm
R 09-15	150 Ohm
R 16	entfällt
R 17-20	4,7 KOhm
R 21-24	680 Ohm
R 25	330 Ohm
R 26,27	150 Ohm

Kondensatoren

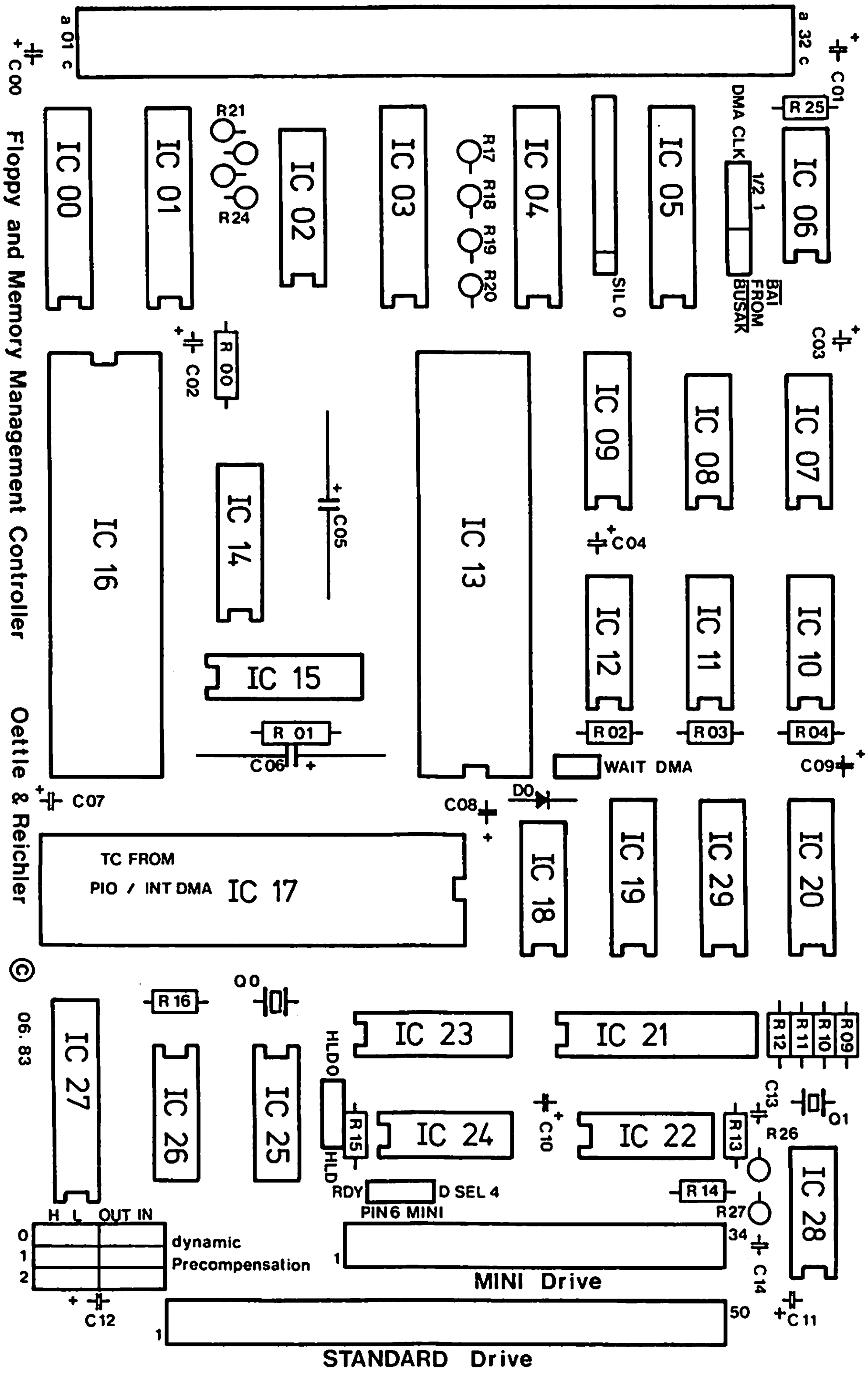
C 00-04	4,7 uF
C 05,06	470 uF
C 07,08	4,7 uF
C 09	100 nF
C 10	4,7 uF
C 11	100 nF
C 12	4,7 uF
C 13	120 pF
C 14	entfällt

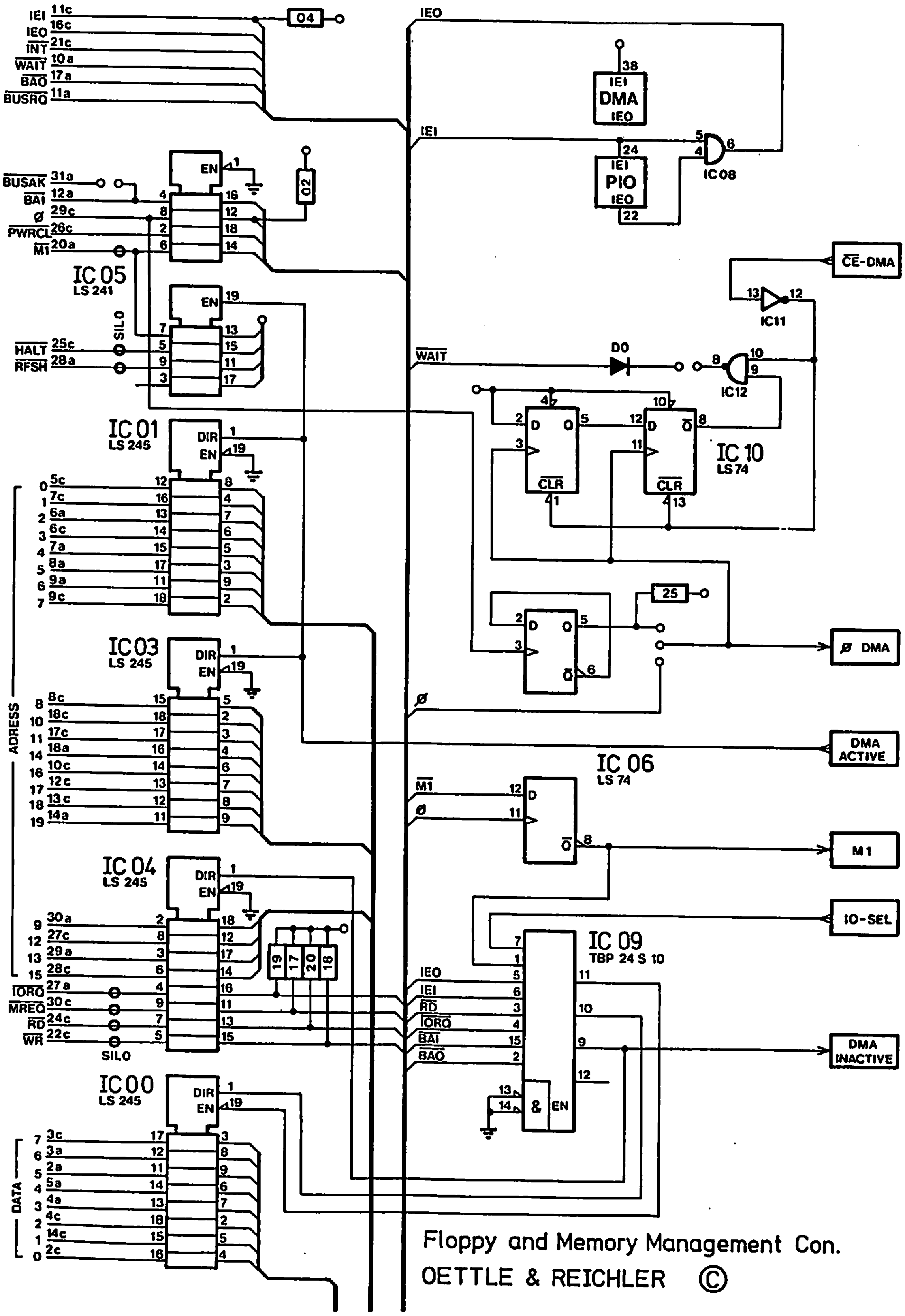
Quarze

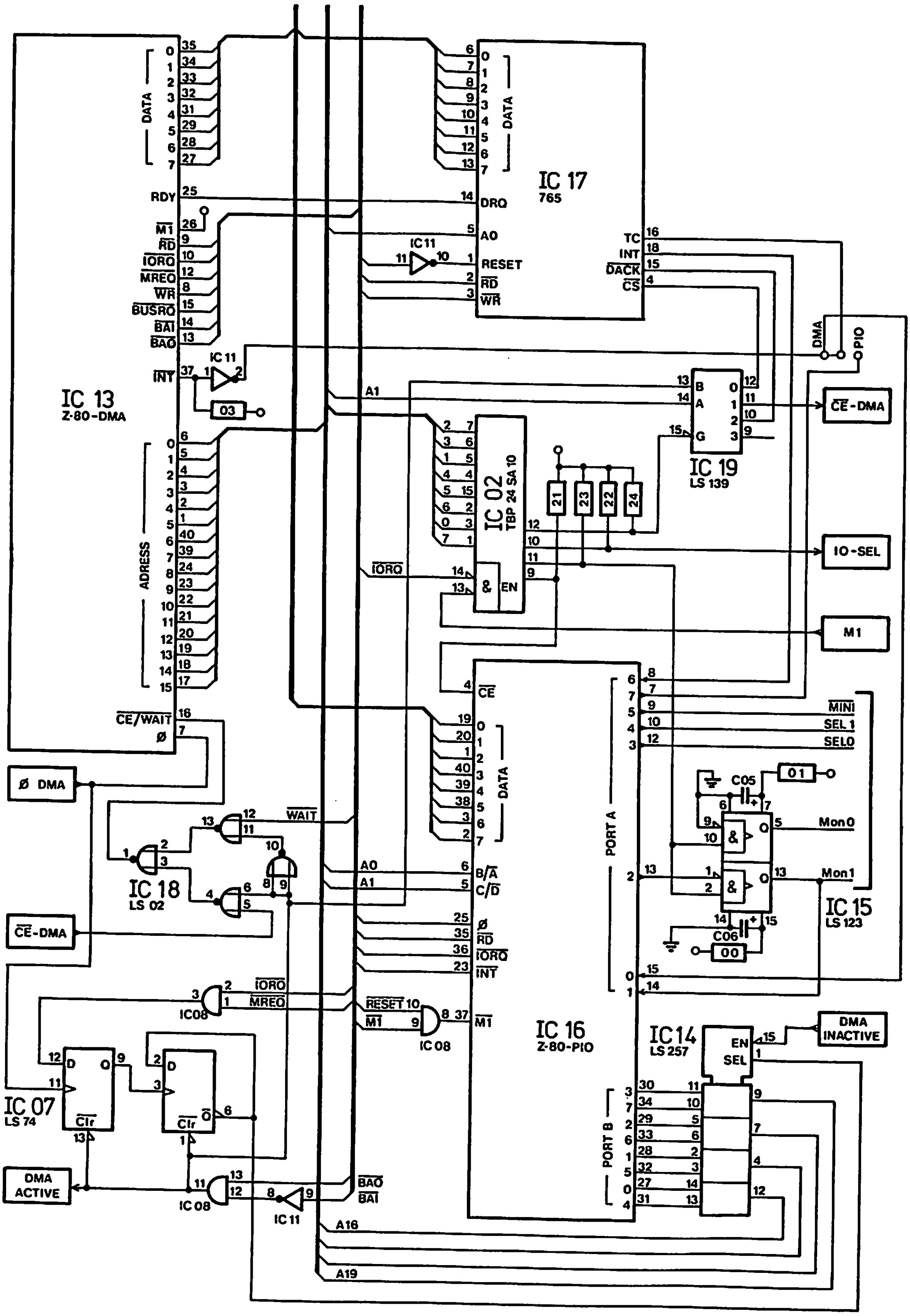
Q 1	32 MHz

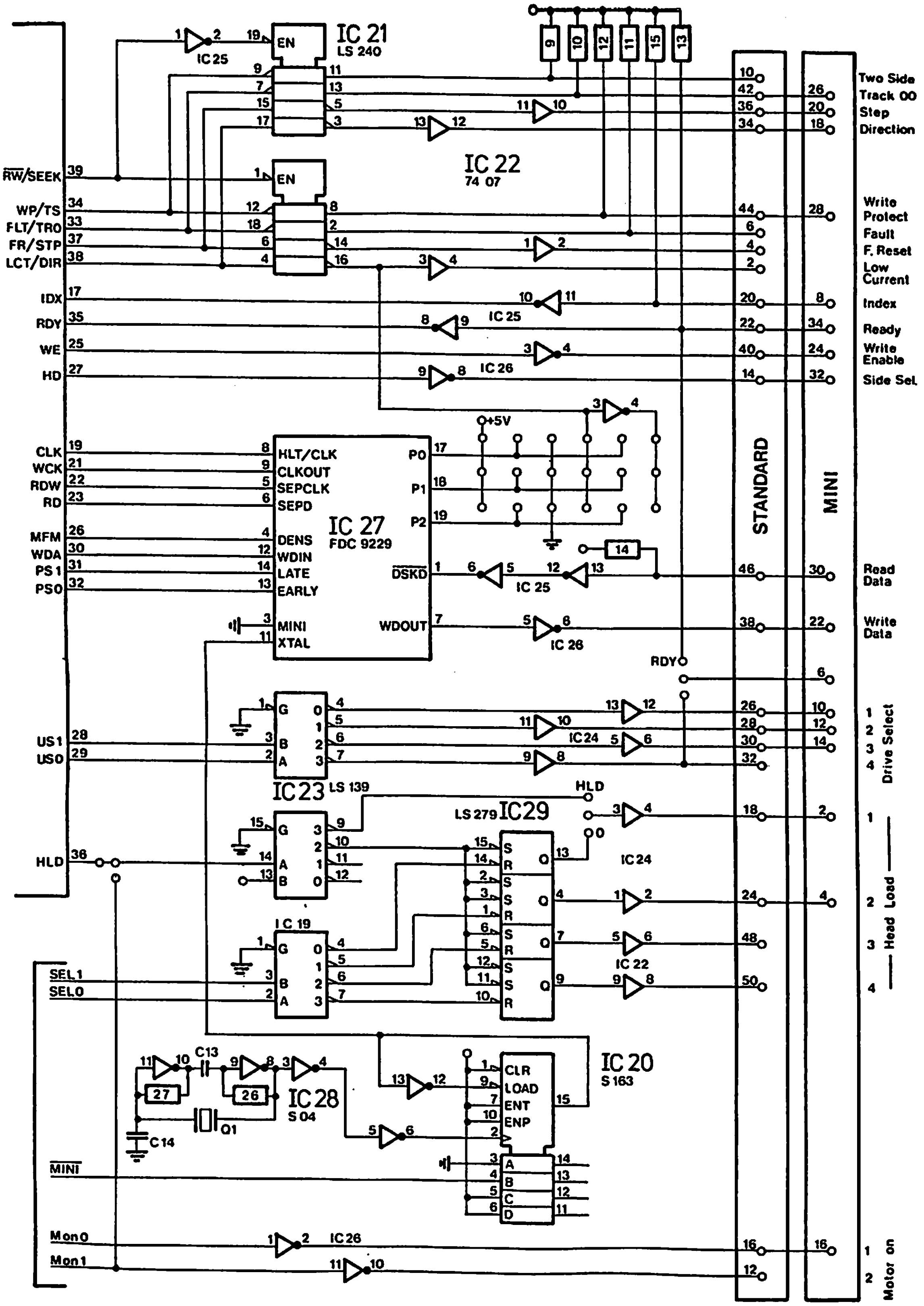
Netzwerke

SIL 0	7 x 4,7 K


a 01 c
a 32 c
C00
C01
R25
DMA CLK
1/2 1
BAI FROM BUSAK
R21
R24
R17 R18 R19 R20
SIL 0
IC 00
IC 01
IC 02
IC 03
IC 04
IC 05
IC 06
Floppy and Memory Management Controller
Oettle & Reichler
© 06.83
R 00
C02
C03
IC 16
IC 14
C05
IC 13
IC 09
IC 08
IC 07
C04
IC 12
IC 11
IC 10
IC 15
R 01
C06
R02
R03
R04
WAIT DMA
C09
C07
C08
D0
TC FROM
PIO / INT DMA
IC 17
IC 18
IC 19
IC 29
IC 20
R16
00
IC 23
IC 21
R12 R11 R10 R09
IC 27
IC 26
IC 25
HLD0
HLD
R15
IC 24
C10
IC 22
R13
C13
Q1
R26
RDY
D SEL 4
PIN6 MINI
R14
R27
IC 28
H L OUT IN
0
1
2
dynamic
Precompensation
1
34
C14
MINI Drive
C12
1
50
C11
STANDARD Drive


IEI
IEO
INT
WAIT
BAO
BUSRQ
04
IEO
DMA
PIO
IC08
BUSAK
BAI
Ø
PWRCL
M1
EN
IC 05
LS 241
02
CE-DMA
IC11
SILO
HALT
RFSH
WAIT
DO
IC12
IC 01
LS 245
DIR
EN
IC 10
LS 74
ADRESS
IC 03
LS 245
25
Ø DMA
Ø
DMA ACTIVE
IC 06
LS 74
M1
IC 04
LS 245
M1
IO-SEL
IC 09
TBP 24 S 10
IORQ
MREQ
RD
WR
SILO
DMA INACTIVE
IC 00
LS 245
DATA
Floppy and Memory Management Con.
OETTLE & REICHLER ©


IC 13
Z-80-DMA
DATA
RDY
M1
RD
IORQ
MREQ
WR
BUSRQ
BAI
BAO
INT
ADRESS
CE/WAIT
Ø
IC 17
765
DRQ
A0
RESET
TC
INT
DACK
CS
IC 11
IC 02
TBP 24 SA 10
EN
IC 19
LS 139
DMA
PIO
CE-DMA
IO-SEL
M1
Ø DMA
WAIT
IC 18
LS 02
A0
A1
B/A
C/D
IC 16
Z-80-PIO
PORT A
PORT B
MINI
SEL 1
SEL0
C05
C06
Mon0
Mon1
IC 15
LS 123
RESET
IC 08
IC 07
LS 74
Clr
DMA ACTIVE
BAO
BAI
A16
A19
IC14
LS 257
SEL
DMA INACTIVE


IC 21
LS 240
IC 25
IC 22
74 07
RW/SEEK
WP/TS
FLT/TRO
FR/STP
LCT/DIR
IDX
RDY
WE
HD
CLK
WCK
RDW
RD
MFM
WDA
PS 1
PS0
HLT/CLK
CLKOUT
SEPCLK
SEPD
DENS
WDIN
LATE
EARLY
MINI
XTAL
P0
P1
P2
DSKD
WDOUT
IC 27
FDC 9229
IC 26
+5V
STANDARD
MINI
Two Side
Track 00
Step
Direction
Write Protect
Fault
F. Reset
Low Current
Index
Ready
Write Enable
Side Sel.
Read Data
Write Data
RDY
US1
US0
IC23 LS 139
IC24
Drive Select
HLD
LS 279 IC29
IC 19
SEL 1
SEL0
Head Load
IC 20
S 163
CLR
LOAD
ENT
ENP
C13
C14
Q1
IC28
S 04
MINI
Mon 0
Mon 1
Motor on

oettle & reichler
datentechnik

Schießgrabenstr. 28
8900 Augsburg 1

Tel.: (0821) 15 46 32

R A M

256 K-Byte Speicherbaugruppe

Inhaltsverzeichnis

Übersicht:

Die dynamische RAM1 Baugruppe läßt sich in vier Stufen zu 64 K-Byte auf maximal 256 K-Byte ausbauen. Als Speicher werden 4164 Bausteine verwendet. Es können dabei wahlweise Chip's mit und ohne automatischem Refresh eingesetzt werden.
Zum Refresh des dynamischen Speichermediums ist ein RFSH Signal vom Systembus erforderlich. Ebenfalls muß bei Verwendung von Speichern ohne automatischen Refesh auf den Adressen A0-A6 eine Refresh-adresse bereit gestellt werden.
Auf einem von einem Z-80 gesteuerten Systembus sind beide Voraussetzungen erfüllt.

Adreßbereich

Der auf der RAM Karte bestückte Speicherbereich kann beliebig in einem 1 M-Byte großen Adreßbereich verschoben werden. Die Adreßlage wird dabei ausschließlich über Programmierung des PROM's IC 06 festgelegt. Dieser ist in der Lage 4 K-Byte große Speicherblöcke zu dekodieren und in Abhängigkeit seiner Programmierung einem der vier Speicherreihen ein RAS Signal zur Verfügung zu stellen.

Precharge Schaltung

Das Timingverhalten des Z80 macht es erforderlich während des M1 Zyklus den Precharge zwischen Speicherzugriff und Refresh aufzuweiten um die minimale Prechage Zeit der Speicher nicht zu unterschreiten. Mit IC 01 (LS 74) wird dazu der Opcode Fetch vorzeitig beendet, wodurch den Speichern eine größere Precharge Zeit bis zum Beginn des Refresh Zugriffs verbleibt.

Dynamische RAM Driver

Um eine optimale Ansteuerung der Speicher zu erzielen können alternativ für IC 09 und IC 10 spezielle dynamische RAM Driver eingesetzt werden.

Steckbrücken

Speicher Timing RAS - MUX

Die Zeitspanne zwischen fallender RAS Flanke und Umschalten der Adressen kann über die Brücken E,F,G,H und J in Schritten zu 6 ns exakt eingestellt werden. Diese Zeitspanne ist in Abhängigkeit der verwendeten Speicher einzustellen. In Datenblättern ist diese Zeit mit der Bezeichnung tRAH (Row Adress Hold Time) zu finden. Diese minimal angegebene Zeit ist mit einem Sicherheitsfaktor 1,75 zu multiplizieren und auf den Steckbrücken einzustellen. J entspricht dabei einer typischen Zeit von 26 ns, H - 32ns, G - 38ns, F - 44ns, E - 50ns.

Speicher Timing MUX - CAS

Ebenfalls kann die Zeitspanne zwischen Umschalten der Adressen und fallender Flanke von CAS über die Brücken A,B,C und D eingestellt werden. Diese mit tASC (Column Adress Setup Time) bezeichnete Zeit wird in den meisten Datenblättern mit minimal 0 ns angegeben. Dieser Zeit ist der gleiche Sicherheitsfaktor wie oben (0,75 x tRAH) zuzuschlagen und auf den entsprechenden Steckbrücken einzustellen. Steckweise D entspricht 12 ns, C - 18ns, B - 24ns, A - 30ns.

Hardware Beschreibung

Datentreiber

Die Daten vom und zum BUS werden von zwei uPD 8226 (IC 13 und IC 14) getrieben. Diese treiben grundsätzlich zum Karteninneren. Ist RD (24c) und CAS aktiv wird die Richtung gewechselt und die Speicherkarte treibt auf den BUS. Führt der Eingang DESLCT (26a) Low Pegel wird ein Treiben auf den BUS verhindert. Dies ist zum Beispiel sinnvoll während des BOOT Vorgangs bei dem der RAM Speicher zeitweise ausgeblendet werden muß.

Adreßmultiplexing

Die 16 Adreßleitungen A0 bis A15 werden über die beiden Muliplexer 74 S 158 (IC 11 und IC 12) auf 8 Adreßleitungen reduziert und an die Speicher geführt. Standardmäßig werden zwischen Speicher und Multiplexer acht 33 Ohm Widerstände geschaltet. Diese haben zur Aufgabe gefährliches Über- und Unterschwingen der Adreßleitungen zu unterdrücken. Optional können auch spezielle RAM Driver wie z. B. AmZ 8165 eingesetzt werden.

Adreßdekodierung

Die Adreßdekodierung übernimmt ein 256x4 PROM (IC 06). Die 8 Eingänge sind mit A12 - A19 beschaltet. In Abhängigkeit dieser Signale und der internen Programmierung wird einer der vier Ausgänge auf High Potential gelegt.
Ein nachgeschaltetes Latch 74 LS 75 (IC 07) 'friert' diese vier Signale während der MREQ aktiven Zeit ein.
Aus der UND Verknüpfung (74 S 00, IC 08) der vier Signale mit MREQ werden die RAS Signale abgeleitet.
Grundsätzlich ist immer nur ein RAS Signal aktiv. Eine Ausnahme bildet der Refresh Zyklus bei dem alle RAS Leitungen aktiviert werden. Das RFSH Signal an den CE Eingängen des PROM's (IC 06) deaktiviert den PROM während der Refresh Phase. Demzufolge führen alle Ausgänge High Pegel (Pull-up SIL 0) und die vier RAS Signale werden während MREQ aktiviert.

Precharge Extension

Bei höheren Systemfrequenzen wird beim M1 Opcode Fetch die Zeit zwischen Speicherzugriff und Refesh zu knapp. Die minimale Prechagezeit wird unterschritten und der Speicher verliert Daten. Abhilfe schafft eine Precharge Extension Schaltung. Diese, bestehend aus IC 01 (74 LS 74), beendet den Opcode Speicherzugriff vorzeitig mit der steigenden Flanke von T3.

Multiplex Timing

Das Multipex Timing kann über Steckbrücken in Schritten zu 6ns exakt auf den verwendeten Speicher abgestimmt werden.
Zeit zwischen fallender RAS Flanke und Adreßumschaltung:
Die vier RAS Leitungen werden über eine ODER Verknüpfung (IC 02, 74 LS 21) und eine UND Verknüpfung (IC 03, 74 S 02) mit RFSH auf eine Verzögerungskette geführt (IC 00, 74 LS 241). Auf einer der Brücken J,H,G,F oder E wird das Signal zur Umschaltung der Adressen abgegriffen.
Zeit zwischen Adreßumschaltung und fallender CAS Flanke:
Über eine zweite Verzögerungskette (D,C,B oder A) wird das CAS Signal abgeleitet. IC 04 (74 LS 08) bewirkt eine UND Verknüpfung mit dem MREQ Signal.

Auto Refresh Speicher

Mit den beiden Steckbrücken MN und LK kann das Signaltiming genau auf die verwendeten Speicher abgestimmt werden.

Steckbrücke LK:
Steckweise L bewirkt, daß während des Refresh Vorgangs RAS nicht aktiviert wird.
Steckweise K bewirkt ein aktiv werden des RAS Signals, wenn MREQ aktiv ist.

Steckbrücke MN:
Mit dieser Brücke wird das Timingverhalten des RFSH Signals (Pin 1) an den Speichern bestimmt.
Steckweise M deaktiviert das RFSH Signal (Pin 1) vollständig unabhängig von anderen Signalen.
Steckweise N bewirkt eine UND Verknüpfung der beiden BUS Signale MREQ (30c) und RFSH (28a).
Ist weder M noch N gesteckt wird das BUS Signal RFSH direkt an die Speicher geführt.

ECB-BUS Belegung

	A		C	
*	+5V	01	+5V	*
*	D5	02	D0	*
*	D6	03	D7	*
*	D3	04	D2	*
*	D4	05	A0	*
*	A2	06	A3	*
*	A4	07	A1	*
*	A5	08	A8	*
*	A6	09	A7	*
*	$\overline{\text{WAIT}}$	10	A16	*
	$\overline{\text{BUSRQ}}$	11	IEI	*
*	$\overline{\text{BAI}}$	12	A17	*
	+12V	13	A18	*
*	A19	14	D1	*
	-5V	15	-15V	
	2xØ	16	IEO	*
*	$\overline{\text{BAO}}$	17	A11	*
*	A14	18	A10	*
	+15V	19		
*	$\overline{\text{M1}}$	20	$\overline{\text{NMI}}$	
		21	$\overline{\text{INT}}$	
		22	$\overline{\text{WR}}$	*
	$\overline{\text{PFL}}$	23		
	VCMOS	24	$\overline{\text{RD}}$	*
		25	$\overline{\text{HALT}}$	
*	$\overline{\text{DESLCT}}$	26	$\overline{\text{PWRCL}}$	
	$\overline{\text{IORQ}}$	27	A12	*
*	$\overline{\text{RFSH}}$	28	A15	*
*	A13	29	Ø	*
*	A9	30	$\overline{\text{MREQ}}$	*
	$\overline{\text{BUSAK}}$	31	$\overline{\text{RESET}}$	
*	GND	32	GND	*

* = Signale sind auf der Speicherbaugruppe verdrahtet

RAM1 - Bestückung

Speicher

M 00-37	4164

IC's

IC 00	74 LS 241
IC 01	74 LS 74
IC 02	74 LS 21
IC 03	74 S 02
IC 04	74 LS 08
IC 05	74 LS 14
IC 06	TBP 24 SA 10
IC 07	74 LS 75
IC 08	74 S 00
IC 09	74 LS 240 (AmZ8165)
IC 10	8 x 33 Ohm (AmZ8165)
IC 11,12	74 S 158
IC 13,14	8226

Widerstände

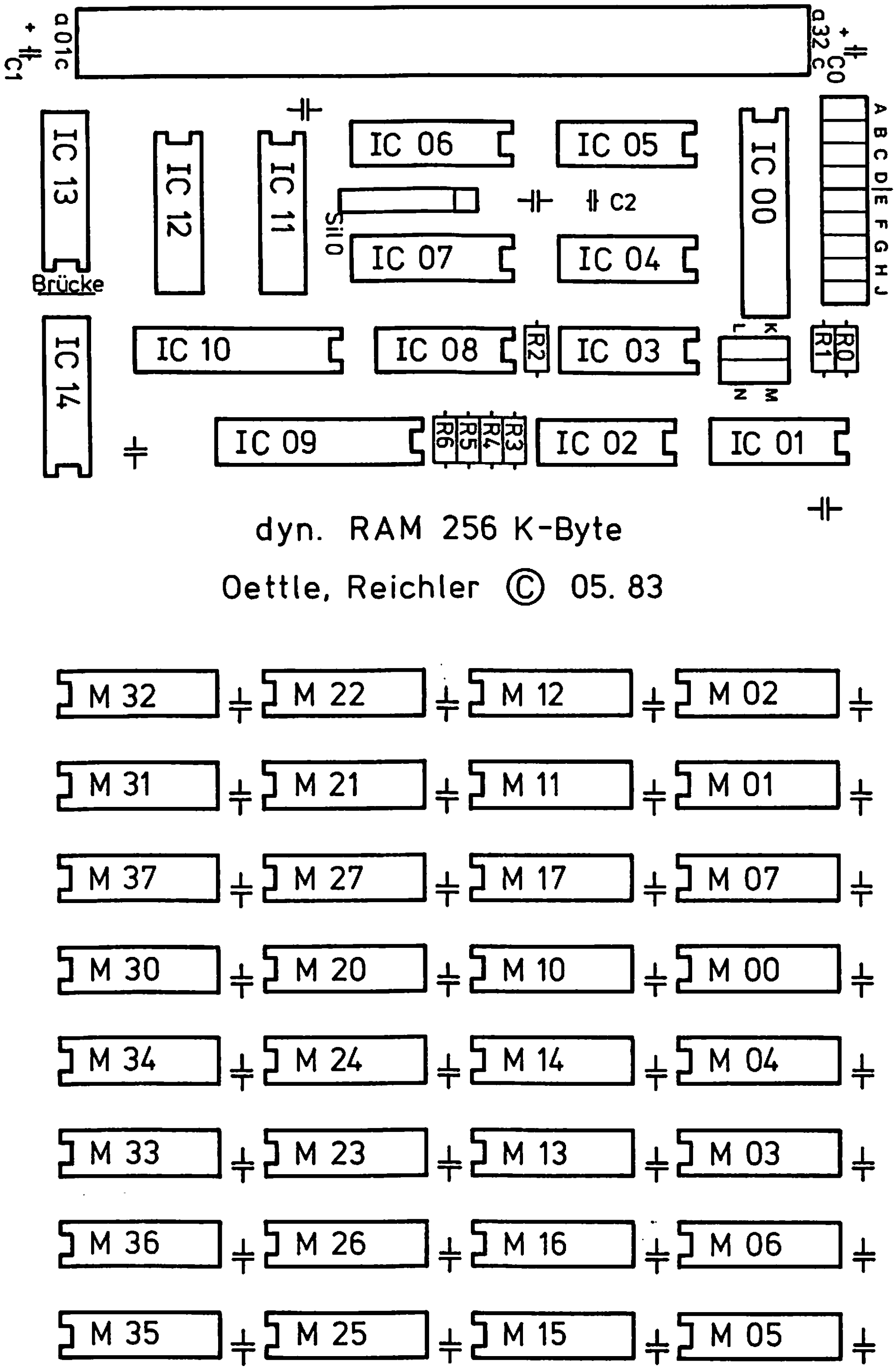
R 0,1	4,7 K Ohm
R 2-6	33 Ohm

Netzwerke

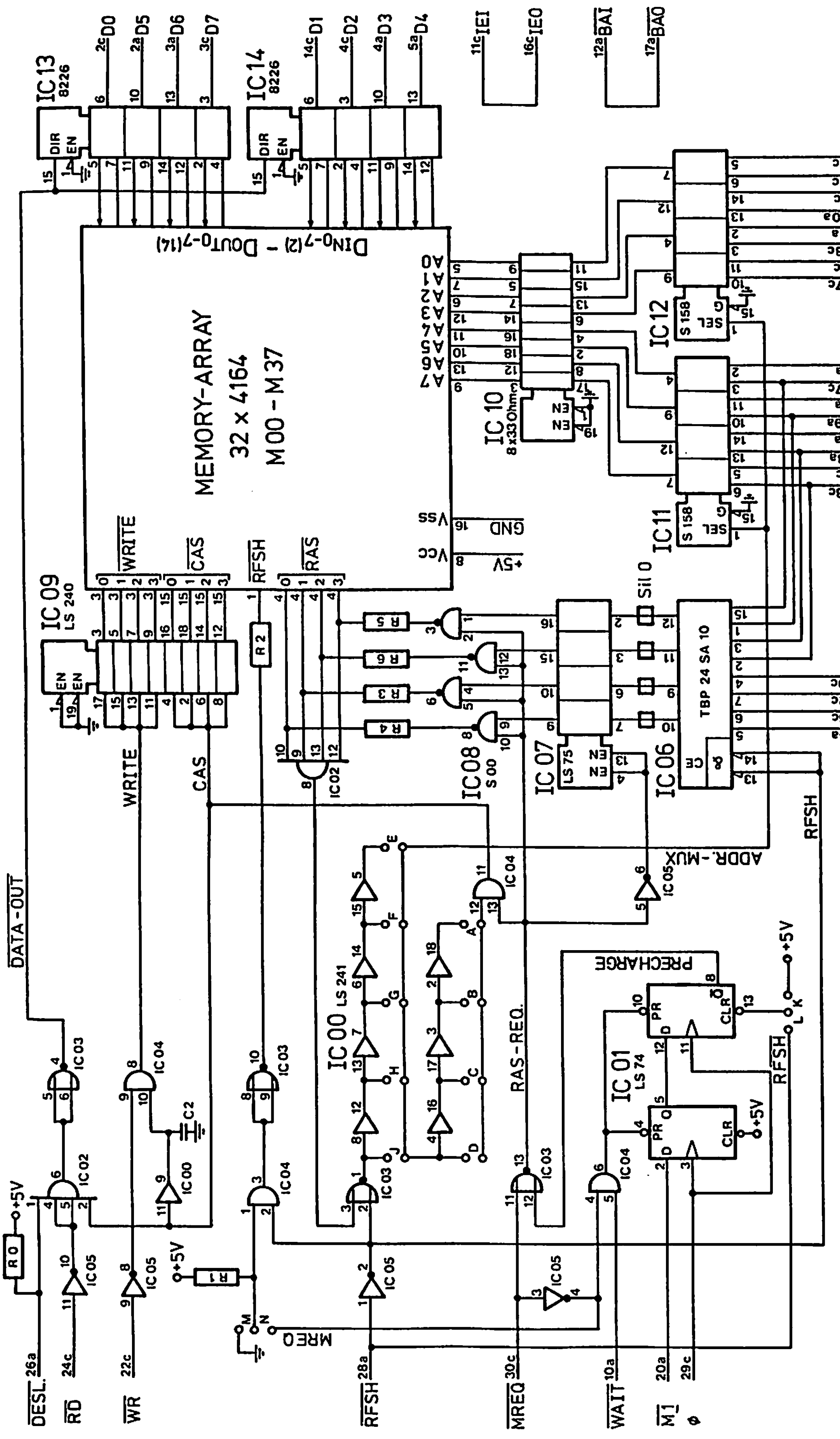
SIL 0	5 x 680 Ohm

Kondensatoren

C 0,1	4,7 uF, 16 V
C 2	entfällt
C sonst.	100 nF



a01c
C1
a32c
C0
IC 13
Brücke
IC 12
IC 11
Sil10
IC 06
IC 05
IC 07
IC 04
C2
IC 00
A B C D E F G H J
K L M N
R0 R1
IC 14
IC 10
IC 08
R2
IC 03
IC 09
R6 R5 R4 R3
IC 02
IC 01
dyn. RAM 256 K-Byte
Oettle, Reichler © 05. 83
M 32
M 22
M 12
M 02
M 31
M 21
M 11
M 01
M 37
M 27
M 17
M 07
M 30
M 20
M 10
M 00
M 34
M 24
M 14
M 04
M 33
M 23
M 13
M 03
M 36
M 26
M 16
M 06
M 35
M 25
M 15
M 05


MEMORY-ARRAY
32 x 4164
M00 - M37
IC 13 8226
IC 14 8226
IC 12 S 158
IC 11 S 158
IC 10 8x33Ohm
IC 09 LS 240
IC 08 S 00
IC 07 LS 75
IC 06 TBP 24 SA 10
IC 00 LS 241
IC 01 LS 74
DATA-OUT
WRITE
CAS
RAS
RFSH
RAS-REQ.
ADDR.-MUX
PRECHARGE
DESL.
RD
WR
MREQ
WAIT
M1
GND
+5V

oettle & reichler
datentechnik

Schießgrabenstr. 28 a
8900 Augsburg 1

Tel.: (0821) 15 46 32

B U S

ECB BUS Rückwandverdrahtung

Inhaltsverzeichnis

Signalbelegung

	A		C	
	+5V	01	+5V	
*	D5	02	D0	*
*	D6	03	D7	*
*	D3	04	D2	*
*	D4	05	A0	*
*	A2	06	A3	*
*	A4	07	A1	*
*	A5	08	A8	*
*	A6	09	A7	*
	$\overline{WAIT}$	10	A16	*
	$\overline{BUSRQ}$	11	IEI	
	$\overline{BAI}$	12	A17	*
	+12V	13	A18	*
*	A19	14	D1	*
	-5V	15	-15V	
	2xØ	16	IEO	
	$\overline{BAO}$	17	A11	*
*	A14	18	A10	*
	+15V	19		
*	$\overline{M1}$	20	$\overline{NMI}$	
		21	$\overline{INT}$	
		22	$\overline{WR}$	*
	$\overline{PFL}$	23		
	VCMOS	24	$\overline{RD}$	*
		25	$\overline{HALT}$	*
	$\overline{DESLCT}$	26	$\overline{PWRCL}$	
*	$\overline{IORQ}$	27	A12	*
*	$\overline{RFSH}$	28	A15	*
*	A13	29	Ø	
*	A9	30	$\overline{MREQ}$	*
	$\overline{BUSAK}$	31	$\overline{RESET}$	
	GND	32	GND	

* = über 390 Ohm an 2,9 Volt abgeschlossen

PFL = Power Fail
DESLCT = DESELECT

IEI-IEO / BAI-BAO Kette

Diese vier Signale der Interrupt- und DMA-Kette sind im Gegensatz zu den anderen Signalleitungen nicht parallel durchverdrahtet. Der IEO-Ausgang ist mit dem IEI-Eingang des nächsten Steckplatzes und BAO mit BAI verbunden. Der Ausgang des Steckplatzes mit der höheren Priorität geht jeweils auf den Eingang des nächst niedrigeren.

ECB-BUS-Europakarten, die diese Signale nicht benützen, müssen auf der Platine IEI mit IEO und BAI mit BAO verbinden. Sowohl die Interrupt-, als auch die DMA-Kette wird somit unterbrechungsfrei an die jeweils nächst niedrigere Einheit weitergegeben.

Montage Hinweise

VG Steckverbinder

Die insgesamt acht 64-poligen VG-Messerleisten sind auf der Seite mit der Massefläche einzulöten. Auf Seitenrichtigkeit (Verpolungsschutz-Aussparungen) ist unbedingt zu achten.

Terminierung

Die Terminierung hat die Aufgabe durch Erniedrigung der Impedanz Reflexionen und Übersprech-Störungen zu unterdrücken. Die terminierten Leitungen sind auf Seite 3 (Signalbelegung) durch einen Stern gekennzeichnet.
Werden Baugruppen anderer Hersteller auf dem System-Bus betrieben, ist zu überprüfen, ob die terminierten Bus-Signale ausreichend gepuffert sind.

Die Bauteile der Terminierung sind ebenfalls auf der Masseseite einzulöten.
Bei dem Kondensator C 1 sowie den vier Widerstandsnetzwerken ist auf die Polarisation zu achten.
Der Spannungsregler LM 317 ist liegend auf der Seite mit dem Bestückungsdruck einzulöten.

Power-Stecker

Zehn der insgesamt 14 Kontakte dienen zur Heranführung der Spannungen -5V, +5V und +12V. Die beiden Spannungen +5V und +12V sind mit einem Sense Anschluß (+S und -S) versehen. Die verbleibenden vier Kontakte dienen zur Versorgung eines Laufwerks oder ähnlicher Einheiten mit den Spannungen +5V und +12V.

Signalbelegung Power-Stecker

-5V	-5V
	GND
+5V	-S
	GND
	+5S
	+S
+12V	-S
	GND
	+12V
	+S
DISK	+12V
	GND
	GND
	+5V

Der 14-polige Power-Stecker ist auf der Positionsdruckseite einzubauen. Die Enden des 90 Grad Winkelsteckers zeigen dabei zur Kartenmitte. Um ein einwandfreies Stecken zu ermöglichen, ist der 90 Grad Winkel durch leicht schrägen Einbau zu vergrößern.

Akku Schaltung

Die drei Akkus sind auf der Positionsdruck-seite stehend einzubauen.
Die vier Bauteile D0 - D2 und R0 sind auf der Seite mit der Massefläche kurzbeinig einzulöten.

BUS1 - Bestückung

1.Terminierung

Widerstände

R 1	150
R 2	120

Widerstandsnetzwerke

SIL 0-3	9 x 390 Ohm

Kondensatoren

C 1	100 uF
C 2,3	100 nF

Spannungsregler

317	LM 317

2. Akku-Schaltung

Widerstände

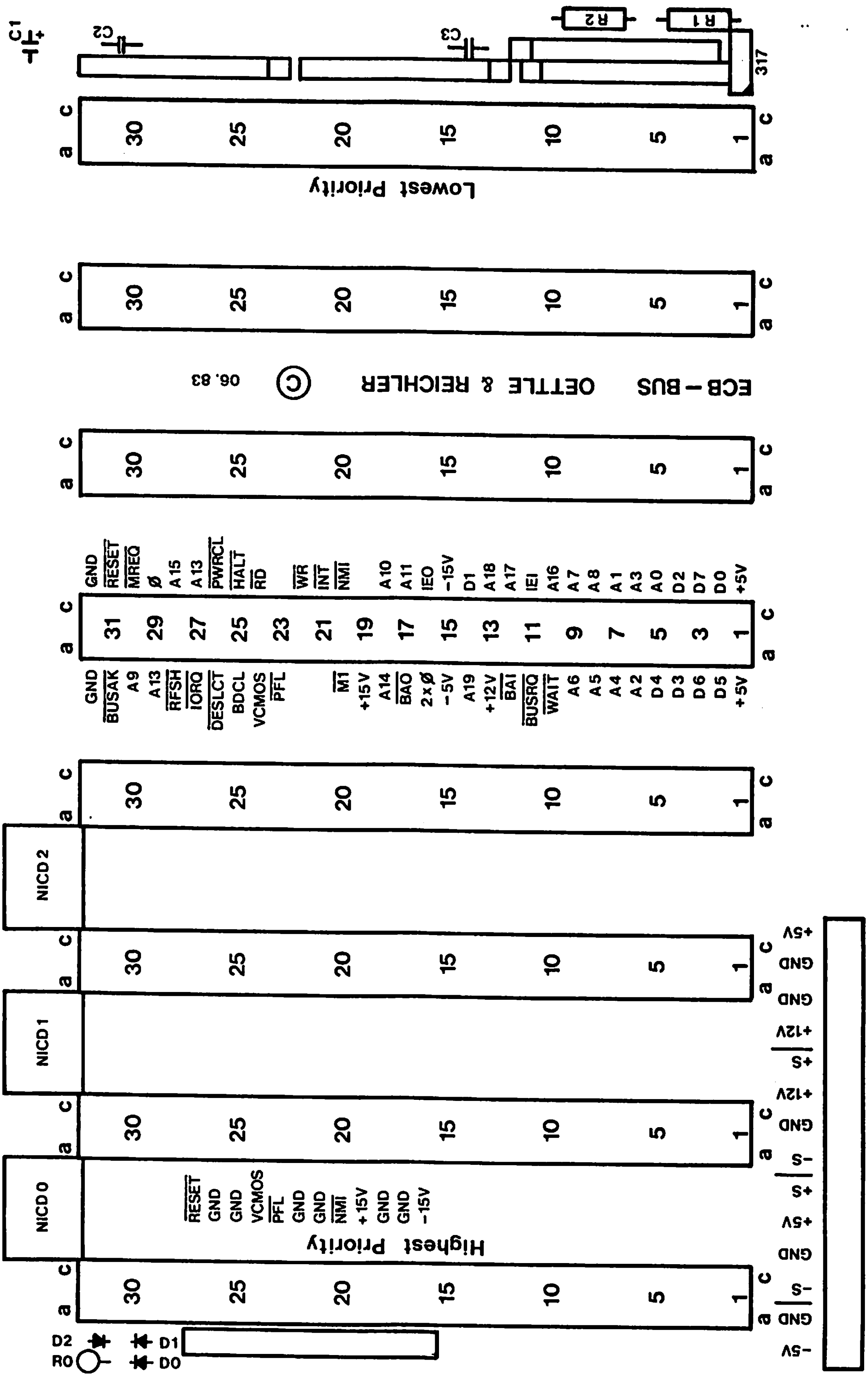
R 0	33

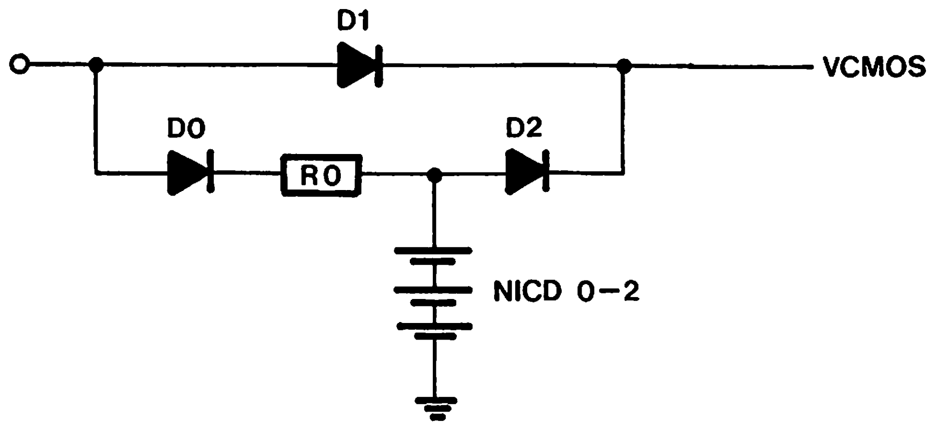
Dioden

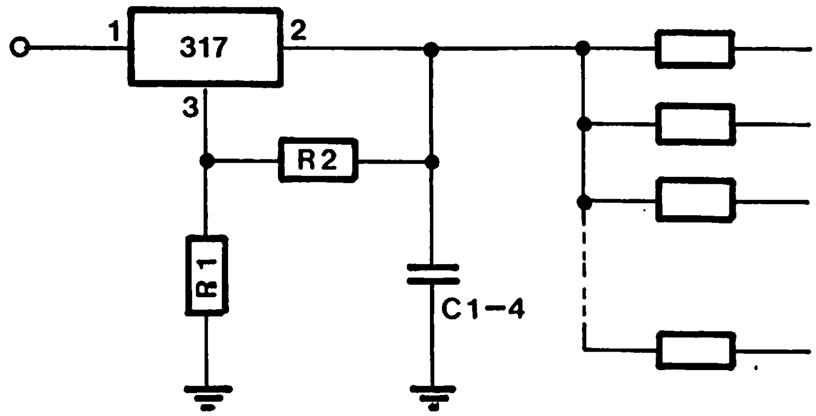
D 0	1 N 4148
D 1	1 N 4001
D 2	AA 143

Akkus

NICD 0-2	Varta 60 DK-F


Lowest Priority
ECB-BUS OETTLE & REICHLER © 06.83
Highest Priority
NICD 0
NICD 1
NICD 2
R1
R2
C1
C2
C3
317
D0
D1
D2
R0


D1
VCMOS
D0
R0
D2
NICD 0-2


1
317
2
3
R2
R1
C1-4

