

MPR-II/FDC-DOC Ausgabe Vers 1.1 20. Dez. 1982

MPR-II/FDC-DOC

L A K O S A Microcomputer GmbH

## Inhaltsverzeichnis

Kapitel	Überschrift	Seite
1	Eigenschaften und Kompatibilität.....	1
2	Arbeitsweise.....	3
2.1	Businterface.....	3
2.1.1	Anschlußbelegung für MPR-II BUS (ECB kompatibel)..	4
2.2	Laufwerkanschluß.....	5
2.2.1	Steckerbelegung Laufwerksanschluß.....	6
2.3	Kernbereich.....	7
2.4	Schnelle Synchron Logik.....	8
2.4.1	Der Ablauf der schnellen Synchron Logik.....	8
3	FDC Programmierung.....	9
3.1.1	Double/Single Density Switch.....	9
3.1.2	Subsystem Status Byte (78H).....	9
3.1.3	Result Type RT (79H).....	9
3.1.4	Result Byte (7BH).....	10
3.2	Der I/O Parameter Block.....	10
3.2.1	Aufbau des IOPB.....	11
3.2.1.1	Das erste Byte.....	11
3.2.1.2	Das zweite Byte (Steuerbefehl).....	11
3.2.1.3	Das dritte Byte (Sektoranzahl).....	12
3.2.1.4	Das vierte Byte (Track Nummer).....	12
3.2.1.5	Das fünfte Byte (Sektor Start Nummer).....	12
3.2.1.6	Das sechste Byte (Puffer Adresse LOW).....	12
3.2.1.7	Das siebte Byte (Puffer Adresse HIGH).....	12
3.3	Die Diskettenbefehle.....	13
3.3.1	RECALIBRATE.....	13
3.3.2	SEEK.....	13
3.3.3	FORMAT TRACK.....	13
3.3.4	WRITE DATA.....	15
3.3.5	WRITE DELETED DATA.....	15
3.3.6	READ DATA.....	15
3.3.7	VERIFY CRC.....	15
3.4	Fehler Bits (Inhalt Result Byte).....	15
4	Amerkung.....	16

## 1 Eigenschaften und Kompatibilität

---

Mit der Baugruppe MPR-II/FDC können bis zu vier Diskettenlaufwerke gleichen Typs an das Mikrocomputer-Baugruppensystem MPR-II angeschlossen werden. Aufgabe der Baugruppe ist es, bitparallele Daten vom MPR-II Systembus (ECB) als bitserielle Daten zum Laufwerk und umgekehrt zu übertragen. Da auf dieser Baugruppe auch alle Steuersignale für das Laufwerk erzeugt und die vom Laufwerk ausgehenden verarbeitet werden, kann der Anwender seine Laufwerke direkt anschließen.

Die Baugruppe MPR-II/FDC ist vorbereitet für den Anschluß von 8" Standard Laufwerken und 5 1/4" Mini Laufwerken. Es können Laufwerke für einseitiges (SINGLE SIDED;SS) Beschreiben oder solche für doppelseitiges (DOUBLE SIDED;DS) Beschreiben der Diskette angeschlossen werden.

Die Aufzeichnung erfolgt mit Frequenzmodulation (FM) und einfacher Schreibdichte (SINGLE DENSITY; SD) oder modifizierter Frequenzmodulation (MFM) und doppelter Schreibdichte (DOUBLE DENSITY; DD). Das FM-Format in Verbindung mit SS ist kompatibel zu Systemen der kommerziellen Datenverarbeitung (IBM 3740). Die Daten können durch einen hardwaremäßigen Schreibschutz gesichert werden.

Zu der Baugruppe MPR-II/FDC wird ein Programm mit der Bezeichnung MPR-II/FIRM angeboten, welches die Baugruppe steuert und die wesentlichen Eigenschaften der Baugruppe erzeugt. Dies Programm wird im Eprom geliefert und wird zum Betrieb der Baugruppe benötigt, wenn der Anwender keine eigene Firmware bereitstellt. So ist diese Baugruppe mit entsprechender Firmware in vielfältigen Anwendungen, bei denen intelligente Floppy-Disk-Steuerungen benötigt werden, einsetzbar.

Mögliche Anwendungen wären:

- LOW-COST Stand-alone Diskettenkopiergeräte für softsektorierte Disketten.
- Emulation von Übertragungsprotokollen verschiedener Diskettencontrollern.
- Integration von Dateiverwaltungen in die Diskettensteuerung zum Anschluß der Diskettensteuerung an einfache Anwendersysteme

So wird durch dies Programm nicht nur die Umsetzung und Aufbereitung der Steuerbefehle für den FDC-Baustein 765 o. 8272 vorgenommen, sondern auch der Datentransfer durch direkten Speicherzugriff (DMA) vom und zum Diskettenlaufwerk. Es wird daher kein spezieller DMA Baustein benötigt.

Die Baugruppe ist mit einer Basis Messerleiste zum Anschluß an das Bus-System und mit einer 50 pol.Front-Messerleiste zum Anschluß der (des) Disketten-Laufwerk(s) ausgerüstet.

Die Baugruppe ist kompatibel zu allen lieferbaren Baugruppen des MPR-II Systems und zu den ECB-Baugruppen die DMA-Fähig sind.

2 Arbeitsweise

Das Blockschaltbild der Baugruppe MPR-II/FDC ist in Bild und der ausführliche Schaltplan im Anhang dargestellt. Die Arbeitsweise wird im folgenden anhand des Blockschaltbildes erläutert.

2.1 Businterface

In dem Funktionsblock "MPR-II Busanschluß" werden der Adreß-, Daten- und Steuerbus gepuffert sowie die Adressen der I/O Ports festgelegt (durch das Adreßselektions PROM PR 0 FDC IC 18).

PROM PR0 (IC18) legt den Adressbereich der Ports folgendermaßen fest.

Adresse | Bedeutung

78H u 88H	!	FDC Status	
79H u.89H	!	FDC Result Type	(lesen)/10PB LOW BYTE (schreiben)
7AH u.8AH	!	IOPB HIGH BYTE	(schreiben)
7BH u.8BH	!	FDC Result BYTE	(lesen)
7EH <del>u.8EH</del>	!	Density Control	(schreiben)/(lesen)

Die anderen Adressen im Bereich 78H - 7FH u. 88H u. 8FH sind für spätere Erweiterungen der Firmware reserviert.

SD:	Port 7XH	10PB (Byte 2)	Port 8XH	10PB (Byte 2)	
	Drive 0	0000xxxxB	Drive 0	00010xxxB	
	Drive 1	00110xxxB	Drive 1	00100xxxB	
	Drive 2	00010xxxB	Drive 2	00000xxxB	
	Drive 3	00100xxxB	Drive 3	00110xxxB	
DD:	Port 7XH	10PB (Byte 2)	Port 8XH	MODE 1 10PB (Byte 2)	MODE 1 10PB (Byte 2)
	Drive 0	00000xxxB	Drive 0	SD { 00000xxxB	DD { 00000xxxB
	Drive 1	00010xxxB	Drive 1	SD { 00110xxxB	DD { 00010xxxB
	Drive 2	00100xxxB	Drive 2	SD { 00010xxxB	DD { 00110xxxB
	Drive 3	00110xxxB	Drive 3	SD { 00100xxxB	DD { 00100xxxB
	Port 8XH	MODE 2 10PB (Byte 2)	MODE 2 10PB (Byte 2)	MODE 3 10PB (Byte 2)	
	Drive 0	SD { 00010xxxB	DD { 00000xxxB		
	Drive 1	SD { 00000xxxB	DD { 00010xxxB		
	Drive 2	SD { 00100xxxB	DD { 00100xxxB		
	Drive 3	SD { 00110xxxB	DD { 00110xxxB		

2.1.1 Anschlußbelegung für MPR-II BUS (ECB kompatibel)

A		C	!							
+5V	----	* 1	*----	+5V	!	A0	5C	!	+5V	1A,C
D5	----	* 2	*----	D0	!	A1	7C	!	GND	32A,C
D6	----	* 3	*----	D7	!	A2	6A	!	VCMOS	24A
D3	----	* 4	*----	D2	!	A3	6C	!	+12V	13A
D4	----	* 5	*----	A0	!	A4	7A	!	-12V	14A
A2	----	* 6	*----	A3	!	A5	8A	!	-5V	15A
A4	----	* 7	*----	A1	!	A6	9A	!	+15V	19A
A5	----	* 8	*----	A8	!	A7	9C	!		
A6	----	* 9	*----	A7	!	A8	8C	!	-RD	24C
-WAIT	----	* 10	*----	-IACK	!	A9	30A	!	-WR	22C
-BUSRQ	----	* 11	*----	(IEI)!	!	A10	18C	!	-M1	20A
A18	----	* 12	*----	A19	!	A11	17C	!	-MREQ	30C
+12V	----	* 13	*----		!	A12	27C	!	-IORQ	27A
-12V	----	* 14	*----	D1	!	A13	29A	!		
-5V	----	* 15	*----	-15V	!	A14	18A	!	-INT	21C
2PHI	----	* 16	*----	(IEO)!	!	A15	28C	!	-NMI	20C
A17	----	* 17	*----	A11	!	A16	19C	!	(IEI)!	11C
A14	----	* 18	*----	A10	!	A17	17A	!	(IEO)!	16C
+15V	----	* 19	*----	A16	!	A18	19A	!	-IACK	10C
-M1	----	* 20	*----	-NMI	!	A19	12C	!	-WAIT	10A
	----	* 21	*----	-INT	!			!	-HALT	25C
	----	* 22	*----	-WR	!	D0	2C	!	-RESET	31C
	----	* 23	*----		!	D1	14C	!	-PWRCL	26C
VCMOS	----	* 24	*----	-RD	!	D2	4C	!	-RFRSH	28A
NPHI	----	* 25	*----	-HALT	!	D3	4A	!		
	----	* 26	*----	-PWCLR	!	D4	5A	!	PHI	29C
-IORQ	----	* 27	*----	A12	!	D5	2A	!	2PHI	16A
-RFRSH	----	* 28	*----	A15	!	D6	3A	!	NPHI	25A
A13	----	* 29	*----	PHI	!	D7	3C	!	-BUSRQ	11A
A9	----	* 30	*----	-MREQ	!			!	-BUSAK	31A
-BUSAK	----	* 31	*----	-RESET	!			!		
GND	----	* 32	*----	GND	!			!		

(!) Anschlüsse werden durch die Platinen geschleift !

## 2.2 Laufwerkanschluß

Die Laufwerke werden über eine 50 pol. Messerleiste für Flachkabelsteckverbindungen an den Controller angeschlossen. Dabei wurde darauf geachtet, daß die Steckerbelegung so erfolgt ist, daß standard 8 Zoll Laufwerke durch eins zu eins verdrahtete Kabel verwendet werden können.

Dieser Block übernimmt ebenfalls das Senden und Empfangen von Signalen für das (die) Laufwerk(e). Wobei Sendeleitungen über Open Kollektor Treiber und Empfangsleitungen über 150 Ohm Pull-up Widerstände angeschlossen sind. Eine Ausnahme ist die RAWDATA/ Leitung. Diese Leitung wird zur besseren Anpassung mit Reihenschaltung aus 220 Ohm u. 330 Ohm abgeschlossen. Im folgendem ist die Belegung und die Bedeutung der einzelnen Laufwerkssignale wiedergegeben.

## 2.2.1 Steckerbelegung Laufwerksanschluß

## Belegung 50 pol. Stecker J2

Pin Nr.	Bezeichn.	Beschreibung
2	LOW CURR/	Reduziert den Schreibstrom ! bei inneren Spuren Ausgang
4	(MOT1OFF)	Hilfsausgang
6	(MOT2OFF)	Hilfsausgang
8	(MOT3OFF)	Hilfsausgang (NC)
10	TWOSIDE	Zweiseitiges Laufwerk ! Eingang nicht benutzt
12		! unbenutzt
14	HDSEL/	Kopfauswahl Ausgang
16		! unbenutzt
18	HDLOAD/	Lädt den Kopf Ausgang
20	INDEX/	Index Impuls Eingang
22	READY/	Ready Eingang
24	(MOT4OFF/)	Hilfseingang
26	SEL0/	Laufwerksauswahl 0 ! Ausgang
28	SEL1/	Laufwerksauswahl 1
30	SEL2/	Laufwerksauswahl 2
32	SEL3/	Laufwerksauswahl 3
34	DIR/	! Legt die Richtung der Kopf- ! bewegung fest Ausgang
36	STEP/	! Stepimpulse zur Kopf- ! bewegung Ausgang
38	WRD/	Schreibdaten Ausgang
40	WRENA/	Schreibgatter Freigabe ! Ausgang
42	TR0/	! Spur Null Sensor ! Eingang
44	WRPRT/	! Schreibschutz Sensor ! Eingang
46	RAWDAT/	! Lesedaten Clock-u. Daten- ! Impulse ungetrennt Eingang
48		! unbenutzt
50		! unbenutzt

Die Anschlüsse mit den Pin Nr. 1 - 49 stellen die Rückleitungen der Signale da und sind mit der Signal Masse verbunden. Von der Benutzung der Leitungen mit den Nummern

4, 6, 8, 10, und 24 wird abgeraten. Sie sind für spätere Erweiterungen vorgesehen.

Die Anschlüsse mit den Nummern 4, 6, 8 u. 24 sind Hilfsausgänge. Sie sind auf der Controller Baugruppe nicht angeschlossen, können aber für allgemeine Ausgabezwecke benutzt werden. So können diese Pins z.B. zur Steuerung der Antriebsmotore von Diskettenlaufwerken benutzt werden, wie es auch beim MPR-II/SYS geschieht. Um dies zu bewirken, müssen auf der Baugruppe die Ausgänge BIT 4 u. BIT 5 (Pin Nr. 15 u. 6) von IC 16 mit Pin 4 u. 6 des Steckers verbunden werden. Somit können Motore zweier Laufwerke im eins aus zwei Code geschaltet werden.

### 2.3 Kernbereich

Der Kernbereich der Baugruppe umfasst ein spezielles Mikrocomputersystem bestehend aus Z80-CPU, max. 4 KByte EPROM, 2 KByte RAM, integrierten Floppy-Disk-Contoller Baustein (FDC 765 o. 8272) und der dazugehörigen Dekodierlogik. Diese Logik definiert den lokalen Adressraum folgendermaßen:

- Der Speicheradressraum wird in zwei Hälften zu je 32 KByte aufgeteilt. Die untere Hälfte, das sind die Adressen 0H - 7FFFH, belegt das EPROM, das RAM belegt die obere Hälfte, das sind die Adressen 8000H - 0FFFFH.
- Der I/O Adressraum ist ebenfalls in zwei Hälften geteilt, da das Adressbit A7 nicht zur Dekodierung benutzt wird. Die Aufteilung ergibt sich nun zu:

0F0H	-----	I/O Kanal zur Kommunikation mit Hauptrechner
0F2H	-----	FDC Baugruppe Steuerport
0F4H	-----	I/O Adressport
0F6H	-----	FDC Baugruppe Statusport
0FCH	-----	DMA Port 8272
0FEH	-----	Status 8272
0FFH	-----	Daten 8272

Der I/O Adressbereich wird durch PROM PR2 (IC11) und der Speicheradressbereich durch PROM PR1 (IC7) festgelegt.

## 2.4 Schnelle Synchron Logik

Ein wesentlicher Teil dieser Steuerung ist die Ausführung der Datenübertragung durch die lokale Z80 CPU. Um den dabei auftretenden zeitlichen Anforderungen gerecht zu werden, bei MFM DD muß alle 15  $\mu$ sec ein Byte übertragen werden, und daher eine Interrupt gesteuerte Datenübertragung nicht anwendbar ist, wird die CPU durch zusätzliche Hardware bei dieser Aufgabe unterstützt. Durch diese synchron Hardware ist es nun möglich, daß die CPU durch Ausführung von BLOCK-TRANSFER-Befehlen (INIR, OTIR) den zeitlichen Anforderungen genügen kann.

### 2.4.1 Der Ablauf der schnellen Synchron Logik

Im folgenden wird der Ablauf am Beispiel eines Disketten Lesebefehls dargestellt.

Nachdem die Steuerung von der System-CPU den Befehl zum Lesen von Daten erhalten, interpretiert und die entsprechenden Parameter dem FDC 8272 übergeben hat, werden im Steuerregister (IC 16) Bit 0 u. Bit 3 gesetzt. Dadurch wird die schnelle Synchron Logik bereitgemacht. Bit 0 fordert durch Flipflop 2IC28 mit den Datentransfer synchronisiert den Systembus an und läßt die lokale CPU bis zum Auftreten von Daten WAIT-STATES ausführen. Das Vorhandensein von Daten gibt der FDC 8272 durch das Signal DRQ (DMAREQUEST) bekannt. Die lokale CPU beendet nun die WAIT-STATES, liest das Datenbyte aus dem Datenregister des FDC 8272 aus und überträgt es durch die Bustreiber in den System-Speicher. Desweiteren benötigt der FDC 8272 zur Übertragung von Daten im DMA-Modus eine entsprechende "Handshake Sequence" der Signale DRQ u. DACK/ dies wird durch die Schaltung aus IC 14, IC 30, IC 19 und Teilen von IC 26 u. IC 15 erzeugt. BIT 3 (IC 16) gibt über eine logische Verknüpfung in IC 19 den Eingang NMI/ der lokalen CPU frei. Dieser Interrupt Eingang wird zur bearbeitung von Interrupts des FDC 8272 bei der Datenübertragung benutzt. Dies zeigt im allg. Fehler beim Lesen oder Schreiben der Diskette an, welche dann die Übertragung beenden und angezeigt werden.

### 3 FDC Programmierung

-----

Die FDC-Baugruppe wird direkt auf den ECB-Bus gesteckt. Nach einem RESET Signal an Pin 31C läuft im FDC zunächst eine interne Initialisierung ab. Die Kommunikation zwischen FDC u. Host-CPU erfolgt über mehrere I/O Ports diese sind:

#### 3.1.1 Double/Single Density Switch

Der Controller läßt sich über das Port 7EH softwaremäßig auf DD o. SD einstellen. Hierzu ist ein Schreibzyklus mit:

Bit 6 = 1 (Double Density)  
 Bit 6 = 0 (Single Density)

*für DD Port 7E*

*Drive Mode 0: 40H auf Port 7EH auszuführen. Das Double Density Format entspricht DD 34 mit 26 Sektoren a' 256 Byte u. 77 Spuren. Bit 0 und 1 definieren dabei den Drive-Mode für DD. Siehe hierzu MPR-II/MONI-DOC u.a. x Kapitel 4 Abs. 6 und MPR-II/SYS-DOC Kapitel 3.4.1.1.1. DDMODE 3 definiert auch Port 884 auf DD.*

*4x DD*

#### 3.1.2 Subsystem Status Byte (78H) (88H)

*00: unteres LWVS :F4: In diesem Byte übergibt der FDC der Host CPU folgendes:*  
*01: oberes LWVS :E5:*  
*02: unteres LWVS :F4: Bit 0 - Ready Status des Laufwerkes 0.*  
*unteres LWRS :F5: Bit 1 - Ready Status des Laufwerkes 1.*  
*oberes LWVS :F4: Bit 2 - Result Present Bit (FDC hat Operation beendet).*  
*" LWRS :F5: Bit 3 - FDC Present Bit.*  
*Bit 4 - Double Density Bit.*  
*Bit 5 - Ready Status des Laufwerkes 2.*  
*Bit 6 - Ready Status des Laufwerkes 3.*

#### 3.1.3 Result Type RT (79H) (89H)

Dies Byte gibt an ob das Result Byte den Drive Status enthält (RT=2H) oder die Disk I/O Fehler (RT=0)

### 3.1.4 Result Byte (7BH) (8BH)

Dies Byte enthält bei RT=0 die Disk I/O Fehler Bits:

Bit 0 = Deleted Record  
Bit 1 = Checksum Error  
Bit 2 = Seek Error  
Bit 3 = Address Error  
Bit 4 = Data Over/Underrun  
Bit 5 = Disk is Write Protected  
Bit 6 = Don't care  
Bit 7 = Drive not ready

Bei RT = 2 hat der FDC einen Wechsel im Ready Status eines Laufwerkes erkannt. Das Result Byte enthält dann folgende Bit:

Bit 0-3 = 0  
Bit 4 = Ready Status Drive 2  
Bit 5 = Ready Status Drive 3  
Bit 6 = Ready Status Drive 0  
Bit 7 = Ready Status Drive 1

mit 0 ==> not ready  
1 ==> ready

Der Controller führt sieben Disk Befehle aus, diese sind:

-RECALIBRATE  
-SEEK  
-FORMAT TRACK  
-READ DATA  
-WRITE DATA  
-WRITE DELETED DATA  
-VERIFY CRC

### 3.2 Der I/O Parameter Block

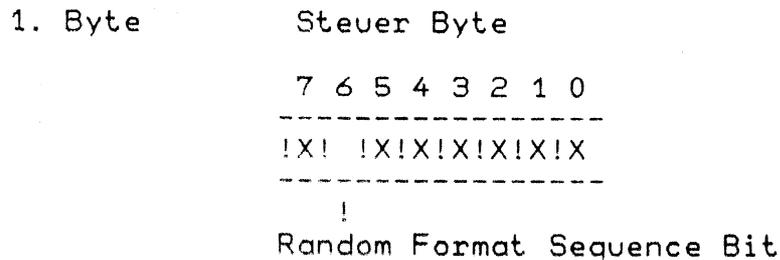
Die Haupt CPU veranlaßt den FDC zur Ausführung einer dieser Operationen durch Ausgabe einer 16 bit Adresse, die auf das erste Bytes eines sieben Bytes umfassenden I/O Parameter Block (IOPB) weist. Das niederwertige Byte wird dem FDC über die I/O Adresse 79H (8BH) übergeben, das höherwertige auf Adresse 7AH. Der FDC liest (8AH)

dann den IOPB durch DMA Zyklen. Der IOPB legt nun die Disk-Operationen fest und enthält alle dazu benötigten Parameter.

### 3.2.1 Aufbau des IOPB

Im folgendem werden die einzelnen Bytes des IOPB eingehend besprochen.

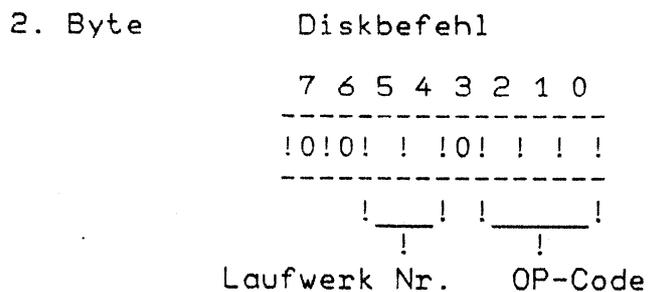
#### 3.2.1.1 Das erste Byte



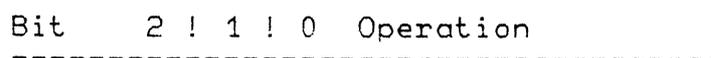
Das Random Format Bit (6) legt die Methode der Anordnung der Sektoren auf der Diskette fest. Wenn dies Bit logisch null ist, werden die Sektoren sequentiell angeordnet. Wenn es logisch eins ist werden die Sektoren in der Reihenfolge angeordnet wie die Sektoradressen in dem durch Byte 5 u. 6 des IOPB adressierten Array angeordnet sind.

#### 3.2.1.2 Das zweite Byte (Steuerbefehl)

Im zweiten Byte IOPB werden Laufwerks Nr. und der auszuführende Befehl festgelegt.



Bits 0-2 haben folgende Bedeutung:



0 ! 0 ! 0	NOP
0 ! 0 ! 1	Seek
0 ! 1 ! 0	Format Track
0 ! 1 ! 1	Recalibrate
1 ! 0 ! 0	Read Data
1 ! 0 ! 1	Verify CRC
1 ! 1 ! 0	Write Data
1 ! 1 ! 1	Write Deleted Data

### 3.2.1.3 Das dritte Byte (Sektoranzahl)

Diese binäre Zahl legt die Anzahl der zu Übertragenden Sektoren fest. Es ist möglich mehrere Sektoren zu Übertragen, die maximale Anzahl darf jedoch nicht größer als 26 sein. In diesem Fall wird ein Adressfehler erzeugt.

### 3.2.1.4 Das vierte Byte (Track Nummer)

Die maximale Track Nummer ist 76 (4CH). Bei binären Zahlen außerhalb des Bereichs 0 - 76 wird ein Adressfehler erzeugt.

### 3.2.1.5 Das fünfte Byte (Sektor Start Nummer)

Bit 4 - 0 dieses Bytes enthält als binären Wert die Nummer des ersten zu Übertragenden Sektors. Gültige Werte sind 1 - 26 (1AH) einschließlich. Es ist zu beachten, daß die Summe aus Byte 3 u. Byte 5 kleiner gleich 26 sein muß.

### 3.2.1.6 Das sechste Byte (Puffer Adresse LOW)

Dies Byte enthält die niederwertigen 8 Bits der 16 Bit Puffer-Speicher-Adresse.

### 3.2.1.7 Das siebte Byte (Puffer Adresse HIGH)

Dies Byte enthält die höherwertigen 8 Bits der 16 Bit Puffer-Speicher-Adresse. Die Bytes sechs u. sieben bilden zusammen die 16 Bit Puffer-Speicher-Adresse im Systemspeicher. Bei Leseoperationen (READ DATA) werden Daten vom Diskettenlaufwerk zu diesem Puffer übertragen. Bei Schreiboperationen werden Daten von diesem Puffer zum Diskettenlaufwerk übertragen. Bei FORMAT TRACK Operationen enthält dieser Puffer das Muster der Sektoranordnung und

0 ! 0 ! 0	NOP
0 ! 0 ! 1	Seek
0 ! 1 ! 0	Format Track
0 ! 1 ! 1	Recalibrate
1 ! 0 ! 0	Read Data
1 ! 0 ! 1	Verify CRC
1 ! 1 ! 0	Write Data
1 ! 1 ! 1	Write Deleted Data

### 3.2.1.3 Das dritte Byte (Sektoranzahl)

Diese binäre Zahl legt die Anzahl der zu Übertragenden Sektoren fest. Es ist möglich mehrere Sektoren zu Übertragen, die maximale Anzahl darf jedoch nicht größer als 26 sein. In diesem Fall wird ein Adressfehler erzeugt.

### 3.2.1.4 Das vierte Byte (Track Nummer)

Die maximale Track Nummer ist 76 (4CH). Bei binären Zahlen außerhalb des Bereichs 0 - 76 wird ein Adressfehler erzeugt.

### 3.2.1.5 Das fünfte Byte (Sektor Start Nummer)

Bit 4 - 0 dieses Bytes enthält als binären Wert die Nummer des ersten zu Übertragenden Sektors. Gültige Werte sind 1 - 26 (1AH) einschließlich. Es ist zu beachten, daß die Summe aus Byte 3 u. Byte 5 kleiner gleich 26 sein muß.

### 3.2.1.6 Das sechste Byte (Puffer Adresse LOW)

Dies Byte enthält die niederwertigen 8 Bits der 16 Bit Puffer-Speicher-Adresse.

### 3.2.1.7 Das siebte Byte (Puffer Adresse HIGH)

Dies Byte enthält die höherwertigen 8 Bits der 16 Bit Puffer-Speicher-Adresse. Die Bytes sechs u. sieben bilden zusammen die 16 Bit Puffer-Speicher-Adresse im Systemspeicher. Bei Leseoperationen (READ DATA) werden Daten vom Diskettenlaufwerk zu diesem Puffer Übertragen. Bei Schreiboperationen werden Daten von diesem Puffer zum Diskettenlaufwerk Übertragen. Bei FORMAT TRACK Operationen enthält dieser Puffer das Muster der Sektoranordnung und

das Fäll Byte der Spur. Der FDC führt diesen Transfer über DMA durch.

Anmerkung: Die oben beschriebenen Befehle gelten bei Einsatz der MPR-II/FDC Standard Firmware.

### 3.3 Die Diskettenbefehle

Die Disketten Steuerung ist in der Lage sieben verschiedene Befehle auszuführen: RECALIBRATE, SEEK, FORMAT TRACK, WRITE DATA (mit Daten Marken), WRITE DATA (mit gelöschten Daten Marken), READ DATA u. VERIFY CRC. Um einen dieser Befehle auszuführen muß die Haupt-CPU der Disketten Steuerung die Adresse des IOPB mitteilen. Das zweite Byte im IOPB spezifiziert den Disketten Befehl. Nachdem die Disketten Steuerung das höherwertige Byte des IOPB erhalten hat, greift die Steuerung auf den IOPB zu, bestimmt den Befehl, verarbeitet die dazugehörenden Parameter und setzt nach dessen Ausführung das RESULT PRESENT Bit im FDC Status Byte (X8H).

#### 3.3.1 RECALIBRATE

Dieser Befehl bringt den Schreib-Lesekopf des Diskettenlaufwerks zur Spur-Null. Dazu werden vom FDC solange Step Impulse ausgesendet, bis das der Kopf den Spur-Null-Sensor des Laufwerkes betätigt. Dies ist oft der erste Befehl nachdem eine Diskette eingelegt wurde oder wenn ein SEEK Fehler aufgetreten ist.

#### 3.3.2 SEEK

Dieser Befehl bringt den Schreib-Lesekopf des Laufwerkes zu der Spur, die im Byte 4 des IOPB festgelegt ist. Bevor nun dieser Befehl beendet wird liest die Steuerung ein Adressfeld der Spur. Wenn die Steuerung dabei feststellt, daß der Kopf nicht über der gewünschten Spur ist, wird ein "SEEK Fehler" erzeugt.

#### 3.3.3 FORMAT TRACK

Dieser Befehl veranlaßt die Steuerung, die durch Byte 4 festgelgte Spur neu zu initialisieren, indem sie alle benötigten Adressmarken, Lücken, Adressfelder und Datenfelder auf der Diskette erzeugt.

Hierbei gibt es zwei Möglichkeiten die Sektoren auf der Diskette anzuordnen. Diese beiden Methoden werden durch Bit 6 (RANDOM FORMAT BIT) im 1. Byte des IOPB festgelegt. Wenn dies Bit logisch null ist, dann stimmt die logische Reihenfolge der Sektoren mit der physikalischen überein d.h. Sektor Adresse 1 wird in den ersten Sektor, Sektor Adresse 2 wird in den zweiten Sektor geschrieben usw. Die Datenfelder werden bei dieser Methode mit dem Byte aufgefüllt, welches durch den Inhalt von Byte 7 u. Byte 6 des IOPB adressiert wird. Es brauchen keine weiteren Daten Bytes in diesem Puffer abgelegt werden.

Wenn nun das Random Format Bit logisch eins ist, dann braucht die logische Reihenfolge der Sektoren nicht mehr mit der physikalischen Reihenfolge übereinzusteimmen. Vielmehr wird die logische Reihenfolge nun durch das 52 Byte Array bestimmt welches durch Byte 6 u. 7 des IOPB adressiert wird. Dies Array hat folgenden Aufbau:

Byte	!	Inhalt (HEX)	!	Bemerkung
1	!	01	!	erste Sektor Adresse
2	!	E5	!	Füll Byte der Spur
3	!	07	!	zweite Sektor Adresse
4	!	XX	!	Dummy Byte
5	!	0D	!	dritte Sektor Adresse
6	!	XX	!	Dummy Byte
7	!	13	!	vierte Sektor Adresse
8	!	XX	!	Dummy Byte
9	!	19	!	fünfte Sektor Adresse
10	!	XX	!	Dummy Byte
.	!	.	!	!
.	!	.	!	usw.

In diesem Fall wird zum Beispiel in den ersten physikalischen Sektor die Adresse 01 eingetragen, in den nächsten Sektor wird die Adresse 07 eingetragen und so weiter bis das in allen Adressfeldern der Spur eine Sektor Adresse eingetragen ist. Die Datenfelder dieser Spur werden (in diesem Beispiel) mit 0E5H aufgefüllt.

Wenn der Kopf zu Beginn des FORMAT TRACK Kommandos nicht über der durch Byte 4 des IOPB adressierten Spur ist, bringt die Steuerung den Schreib- Lesekopf dorthin.

### 3.3.4 WRITE DATA

Dieser Befehl überträgt  $N * 128$  Bytes kontinuierliche Daten vom Datenpuffer im Systemspeicher (adressiert durch Byte 6 u. 7 des IOPB) auf die Diskette. Hierbei wird  $N$  durch den Inhalt des Byte 3 des IOPB definiert. Beim Schreiben werden die Datenfelder mit Daten Marken versehen. Eine Multi-Sektor Operation (z.B.  $n \geq 2$ ) kann bei jedem Sektor beginnen, sie darf aber nicht hinter den letzten logischen Sektor einer Spur führen. Wenn der Kopf zu Beginn des WRITE DATA Kommandos nicht über der durch Byte 4 des IOPB adressierten Spur ist, bringt die Steuerung den Schreib- Lesekopf dorthin.

### 3.3.5 WRITE DELETED DATA

Dieser Befehl ist identisch zum vorherigen außer das Deleted Data Marken geschrieben werden.

### 3.3.6 READ DATA

Dieser Befehl hat die entsprechende umgekehrte Wirkung wie der Befehl WRITE DATA. Darüberhinaus werden 2 Bytes von CRC Bit erzeugt und mit den vorher geschriebenen verglichen. Ergibt sich hierbei ein Unterschied, wird ein "CRC Fehler" erzeugt. Eine Multi-Sektor Operation (z.B.  $n \geq 2$ ) kann bei jedem Sektor beginnen, sie darf aber nicht hinter den letzten logischen Sektor einer Spur führen. Wenn der Kopf zu Beginn des READ DATA Kommandos nicht über der durch Byte 4 des IOPB adressierten Spur ist, bringt die Steuerung den Schreib- Lesekopf dorthin.

### 3.3.7 VERIFY CRC

Dieser Befehl hat die gleiche Wirkung wie der vorherige Befehl, außer das keine Daten übertragen werden.

## 3.4 Fehler Bits (Inhalt Result Byte)

Zusätzlich zu den weiter oben beschriebenen Bedeutung der Fehler Bits ergeben sich noch einige weitere Fehleranzeigen durch Kombinationen der Fehlerbits. Diese sind:

Inhalt RB ! Bedeutung

---

1BH	!	Es gibt keinen Sektor mit der Adresse
1FH	!	Falscher Zylinder
0EH	!	Es gibt keine Adressmarke

4 Anmerkung

Beim Einsatz von MPR-II/FIRM ist zu beachten in welchem PROM die Firmware geliefert wird. Beim 2716 ist auf der Leiterplatte nichts zu ändern. Beim 2732 ist auf der Leiterplatte die Brücke d einzusetzen u. die Verbindung Pin 21 mit +5V zu trennen.

*Standard*