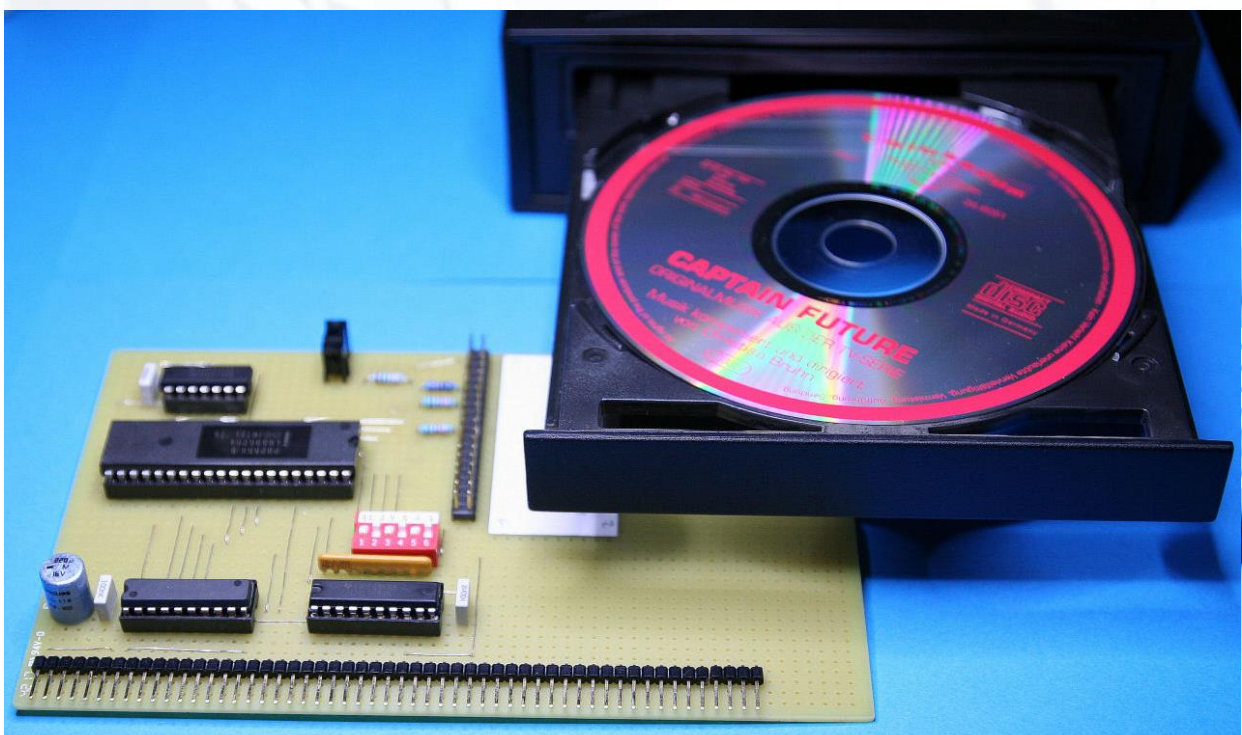


Spezifikation

CD-Player mit NKC-MC-IDE-Karte und
68008-Programm für den NDR-Klein-Computer



Version 1.0

Idee:

Sascha Neuschl
 Pirolweg 21
 48167 Münster
 Email: scn69@gmx.de

Schaltplan und Layout der NKC-MC-IDE-Karte:

Jens Mewes

Dokumentenhistorie

Version	Autor(en)	Änderung	Datum
0.9	Neuschl, Sascha	Entwurf	26.11.2018
1.0	Neuschl, Sascha	Erste Version	19.12.2018

Inhaltsverzeichnis

1	Vorwort.....	4
1.1	Idee.....	4
1.2	Ansatz.....	4
1.3	Aktueller Stand.....	4
2	Beschreibung des Konzepts	4
3	Schaltungsprinzip.....	5
3.1	Adresslogik	5
3.2	Datenbuspuffer	5
3.3	PIO 8255A	5
4	IDE-Schnittstelle	7
4.1	Belegung des 40-poligen IDE-Steckers	7
4.2	PIO-Betriebsarten	8
4.3	Schnittstellenkommunikation.....	9
4.4	ATA- und ATAPI-Geräte in der IDE-Schnittstelle	12
4.5	ATAPI-Command Packets	14
4.6	Ablauf ATA-Packet Command und Command Packet.....	15
5	Schaltplan, Layout, Bestückungsplan und Stückliste NKC-MC-IDE-Karte	19
5.1	Schaltplan:	19
5.2	Layout:	20
5.3	Bestückungsplan:.....	21
5.4	Stückliste:.....	22
6	Anmerkungen.....	23
6.1	Eingänge und Ausgänge:.....	23
6.2	Allgemeines Verhalten:	23
7	Aufbau und Test des NKC-CD-Players.....	24
7.1	Programmierung mit Schichtenkonzept:	24
7.2	Schematischer Ablauf des Programms:.....	25
8	Bedienprogramm	26
8.1	Parameter und Systemkonfiguration.....	26
8.2	Funktionen	26
8.3	Masken	28
9	Anhang.....	39
9.1	Datenblätter TTL-Bausteine:	39
9.1.1	74LS688.....	39
9.1.2	74LS245.....	39
9.1.3	74S04.....	40
9.2	Datenblatt PIO:	40
9.2.1	8255A.....	40
9.3	Verweis auf Datenblätter komplexer Bausteine und Spezifikationen / Quellennachweis	41

1 Vorwort

1.1 Idee

Ich habe ja bereits eine Tunerkarte und ein 68008 - Bedienprogramm für den NDR-Klein-Computer (NKC) entwickelt, die auch schon nachgebaut und in Betrieb genommen wurden.

Da dachte ich mir, jetzt fehlt noch eine Möglichkeit, um Musik aus der Konserve zu hören. Und das sollte mit einem **CD-Player** möglich werden ...

1.2 Ansatz

Kein CD-Player ohne CD-ROM-Laufwerk! Also wurde eine IDE-Schnittstellen-Karte benötigt. Die existierende GIDE-Karte mit dem FPGA war mir zu kompliziert, weshalb ich mich für die MC-IDE-Karte entschieden habe, auf der ein PIO 8255A seinen Dienst tut.

Das Ton-Signal des CD-ROM-Laufwerks sollte über den 2. Eingang des I2C-Sound-ICs TDA 8425 auf der NKC-Tunerkarte verarbeitet werden. Damit konnten wieder Lautstärke, Höhen und Bässe eingestellt werden.

1.3 Aktueller Stand

In Kürze liegt eine industrielle Platine für den Nachbau vor.

Das CD-Player-Programm liegt in Version 1.0 vor.

2 Beschreibung des Konzepts

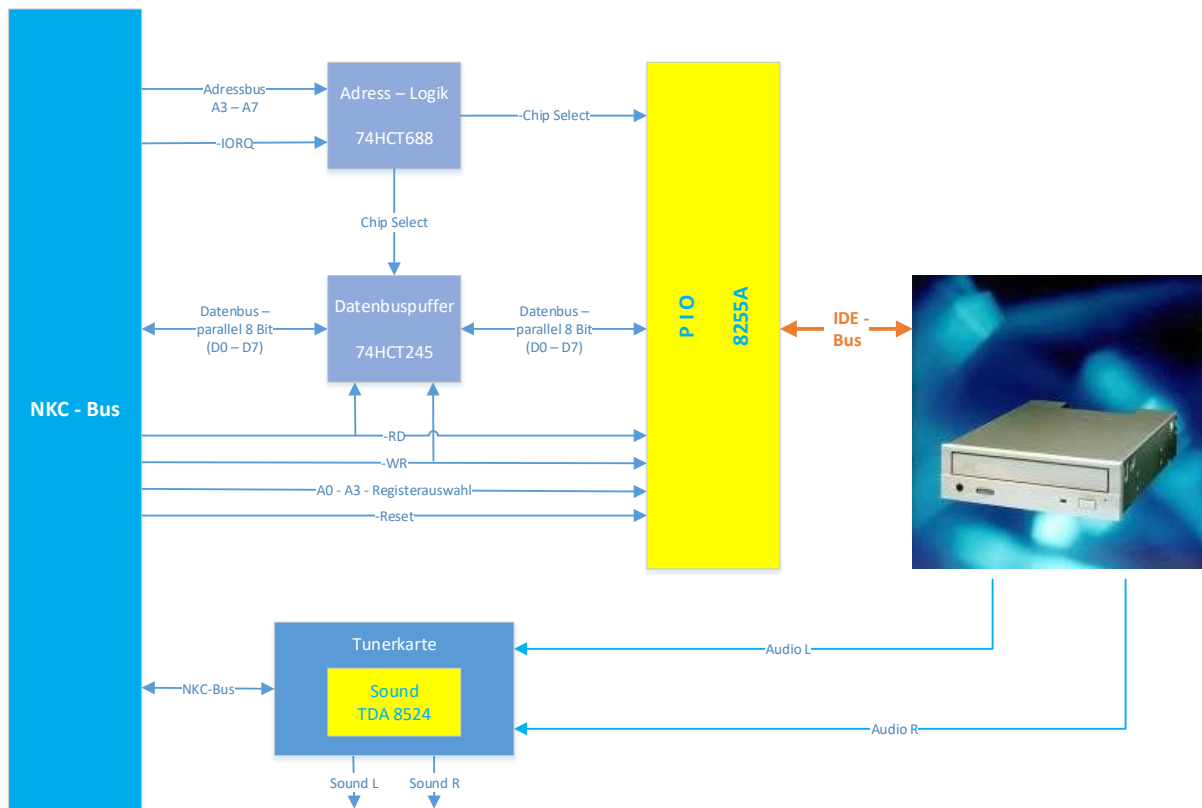
Das Konzept des **NKC-CD-Players** besteht darin, ein **CD-ROM-Laufwerk** über eine **IDE-Schnittstelle** an den NKC anzubinden und dieses als **ATAPI-Gerät** mit einem Programm anzusteuern.

Der **Audioausgang des CD-ROM-Laufwerks** wird an den **2. Eingang des I2C-Soundbausteins** auf der Tunerkarte geführt.

Folgende aktive Komponenten werden eingesetzt:

Funktion	Baustein	Beschreibung	Datenblatt
PIO	8255A	Der Baustein stellt das IDE-Interface zwischen dem parallelen NKC-Bus und dem CD-ROM-Laufwerk dar	8255A.pdf
CD-ROM-Laufwerk	CD-ROM-Laufwerk	IDE-CD-ROM-Laufwerk mit Audioausgang	generisch
I2C-Soundbautein	TDA 8425	I2C-Soundbaustein auf der Tunerkarte zum Einstellen von Lautstärke, Höhen und Bässen	TDA 8425.pdf

3 Schaltungsprinzip



3.1 Adresslogik

Die Adresslogik wird mit dem Baustein **74HCT688** abgebildet. Er benötigt zur Freigabe das **IO-Request-Signal (-IORQ)** des NKC-Busses. Des Weiteren werden die **Datenleitungen A3 bis A7** zugeführt und mit den Werten, die am **JP1** (siehe Schaltplan) eingestellt werden, verglichen. Am **JP1** wird somit die **IO-Adresse des PIO 8255A** eingestellt. Bemerkung: Da der 74LS688 8 Zustände vergleicht, wir aber nur 6 benötigen, werden die Eingänge P0 / Q0 und P1 / Q1 (siehe Schaltbild 74HCT688) auf 0V gelegt, damit diese immer gleich sind. Die **Datenleitungen A0 – A3** sind direkt an **den PIO 8255A** geführt, um dort interne Register anzusprechen. Sind die angelegten Adressdaten des **JP1** und der **Datenleitungen A2 bis A7** identisch, dann wird das Signal **-Chip Select** ausgelöst, was einerseits den **Datenpuffer 74HCT245** und den **PIO 8255A** frei gibt.

3.2 Datenbuspuffer

Als **Datenpuffer** wird ein **74HCT245** eingesetzt. Seine Freigabe erfolgt durch das Signal **-Chip Select**. Die Richtung, also ob Daten zum **PIO 8255A** gesendet oder von ihm geholt werden, bestimmt das Signal **-RD** vom NKC-Bus. Die zu sendenden oder zu holenden Daten werden über die Datenleitungen **D0 bis D7** vom bzw. zum **NKC-Bus** übertragen.

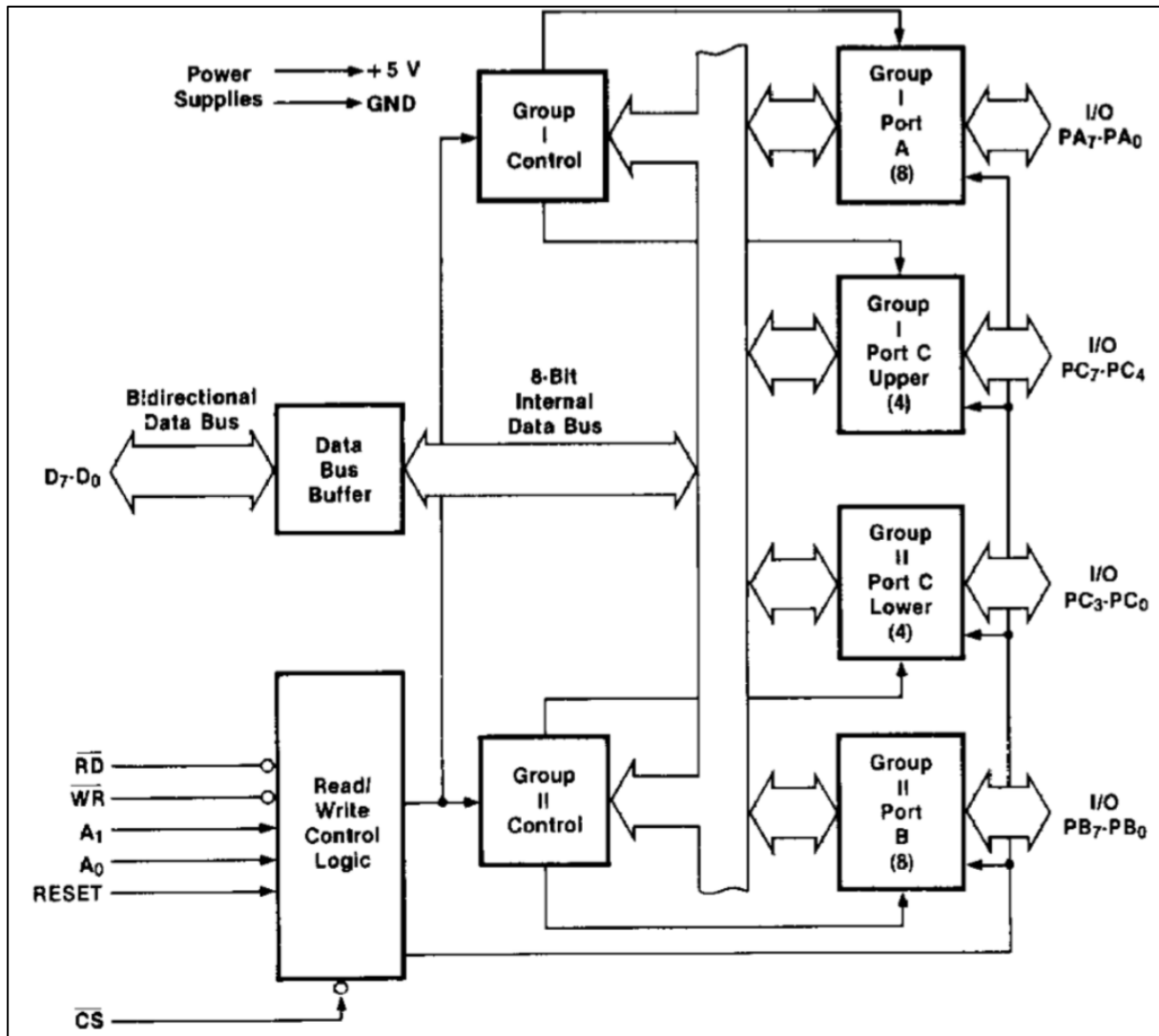
3.3 PIO 8255A

Der **PIO 8255A** wird durch das **Signal -Chip Select** freigegeben. Mit den Adressleitungen **A0 / A1** vom **NKC-Bus** wird entschieden, **welches Register** des Bausteins angesprochen wird. Daten, um aus dem **PIO 8255A** zu lesen oder auf ihn zu schreiben, gelangen über die durch den **74HCT245** gepufferten Datenleitungen **D0 bis D7** zu ihm. Der Zugriff des Lesens wird über das Signal **-RD** und

CD-Player für den NDR-Klein-Computer

der des Schreibens über das Signal \overline{WR} des NKC-Busses ausgelöst. Zudem ist das \overline{Reset} -Signal des NKC-Busses an den **PIO 8255A** gelegt, um ihn bei einem System-Reset zurückzusetzen.

Blockschaltbild:



Zur Realisierung der IDE-Schnittstelle besitzt der Baustein 3 bidirektionale I/O-Ports:

Port	Zuordnung / Funktion	Modus	Beschreibung
A	Erste 8 Bit des IDE-Datenbusses	0 - input / output	Bidirektionaler 16-Bit-Datenbus der IDE-Schnittstelle
B	Zweite 8 Bit des IDE-Datenbusses	0 - input / output	
C	8 Bit des IDE-Steuerbusses	0 - output	Unidirektionaler 8-Bit-Steuerbus der IDE-Schnittstelle

CD-Player für den NDR-Klein-Computer

Für die hier benötigten Zwecke reicht es, den „**Mode 0 - basic operation**“ des Bausteins einzusetzen. Wie dieser für **Lesen und Schreiben der I/O-Ports** angewendet wird, ist der untenstehenden Abbildung zu entnehmen. Außerdem wird darauf bei der Beschreibung des 68008-Programms in Kapitel 6 eingegangen.

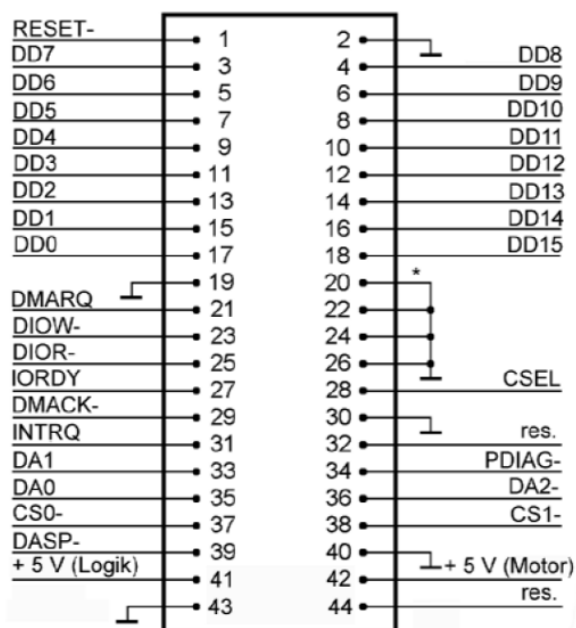
<p>Modes</p> <p>The μPD8255A can be operated in modes 0, 1 or 2 which are selected by appropriate control words and are detailed below.</p> <p>Mode 0</p> <p>Mode 0 provides basic input and output operations through each of the ports A, B, and C. Output data is latched and input data follows the peripheral. No “handshaking” strobes are needed.</p>	<p>Basic Operation</p> <p><i>Input Operation (Read)</i></p> <table border="1"> <thead> <tr> <th>A₁</th> <th>A₀</th> <th>RD</th> <th>WR</th> <th>CS</th> <th></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>0</td> <td>PORT A → DATA BUS</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>1</td> <td>0</td> <td>PORT B → DATA BUS</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>1</td> <td>0</td> <td>PORT C → DATA BUS</td> </tr> </tbody> </table> <p><i>Output Operation (Write)</i></p> <table border="1"> <thead> <tr> <th>A₁</th> <th>A₀</th> <th>RD</th> <th>WR</th> <th>CS</th> <th></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>DATA BUS → PORT A</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>DATA BUS → PORT B</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>DATA BUS → PORT C</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>DATA BUS → CONTROL</td> </tr> </tbody> </table>	A ₁	A ₀	RD	WR	CS		0	0	0	1	0	PORT A → DATA BUS	0	1	0	1	0	PORT B → DATA BUS	1	0	0	1	0	PORT C → DATA BUS	A ₁	A ₀	RD	WR	CS		0	0	1	0	0	DATA BUS → PORT A	0	1	1	0	0	DATA BUS → PORT B	1	0	1	0	0	DATA BUS → PORT C	1	1	1	0	0	DATA BUS → CONTROL
A ₁	A ₀	RD	WR	CS																																																			
0	0	0	1	0	PORT A → DATA BUS																																																		
0	1	0	1	0	PORT B → DATA BUS																																																		
1	0	0	1	0	PORT C → DATA BUS																																																		
A ₁	A ₀	RD	WR	CS																																																			
0	0	1	0	0	DATA BUS → PORT A																																																		
0	1	1	0	0	DATA BUS → PORT B																																																		
1	0	1	0	0	DATA BUS → PORT C																																																		
1	1	1	0	0	DATA BUS → CONTROL																																																		

4 IDE-Schnittstelle

Der **PIO 8255A** übersetzt die **parallelen 8-Bit-IO-Daten** des NKC in die **parallelen Daten der IDE-Schnittstelle** um und umgekehrt, um das CD-ROM-Laufwerk anzusprechen. Die **IDE-Schnittstelle** ist folgendermaßen aufgebaut:

4.1 Belegung des 40-poligen IDE-Steckers

Layout des Steckers:



CD-Player für den NDR-Klein-Computer

Beschreibung der PINs / Signale:

Pin	ATA-Signal	Bedeutung
1	RESET	Setzt alle Laufwerke an diesem Anschluss zurück
2, 19, 22, 24, 26, 30, 40	GND	Masse
3-5-7- 9-11-13-15-17	DD7..DD0	Datenbus, Bits 7..0, Low-Byte bei 16-Bit-Datenübertragung
18-16-14-12-10-8-6-4	DD15..DD8	Datenbus, Bits 15..8, High-Byte bei 16-Bit-Datenübertragung
20	KEYPIN	Pin fehlt, damit der Stecker nicht verkehrt eingesetzt werden kann. Wird aber, abweichend vom Standard, teilweise auch zur Stromversorgung eingesetzt (z. B. für Disk-on-Module-Flashspeicher)
21, 29	DMARQ, DMACK	DMA-Request und -Acknowledgment, DMA-Steuersignale (optional)
23	DIOW	Signal zum Daten schreiben
25	DIOR	Signal zum Daten lesen
27	IORDY	I/O-Ready, low-Pegel: Benötigt zusätzliche Taktzyklen für den gegenwärtigen I/O-Zyklus (häufig nicht verwendet)
28	CABLE SELECT, an sehr alten Festplatten SPINDLE SYNC	Zuordnung des Laufwerks als DRIVE0=Low oder DRIVE1=high, an sehr alten Festplatten Spindelsynchronisation zwischen Device 0 und 1 (damals <i>Master</i> und <i>Slave</i>), z. B. für Drive-Arrays (häufig nicht implementiert)
31	INTRQ	Interrupt-Anforderung
32	IOCS16	Auswahl des 16-Bit-Transfer
34, 39	PDIAG, DASP	Passed Diagnostic vom Device 1, Drive Active/Device 1 Present, Rückmeldungen von Device 1 an Device 0 bei der Initialisierung
36-33-35	DA2..DA0	Adressbus Bits 2..0, Adressierung der internen Register innerhalb des <i>Command-</i> bzw. <i>Control-Register-Satzes</i> des Laufwerks
37, 38	CS1Fx, CS3Fx (auch CS0, CS1)	Chip-Auswahlsignale des Hosts, um die Registersätze <i>Command Register</i> bzw. <i>Control Register</i> auszuwählen. Normalerweise liegen diese bei der PC-Architektur im I/O-Adressraum ab 1F0h bzw. ab 3F0h.

4.2 PIO-Betriebsarten

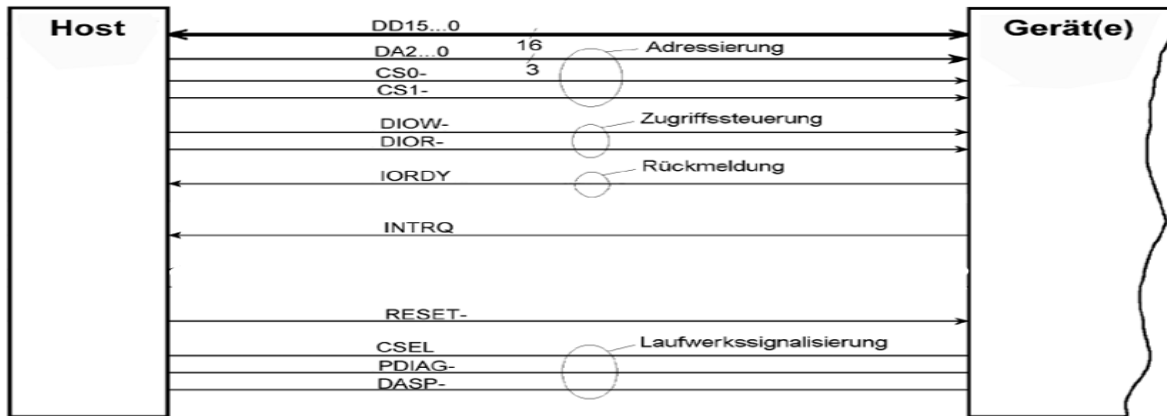
Es wird hier ausschließlich der **PIO-Mode** mit nur **e i n e m Laufwerk** verwendet (kein DMA)!

Betriebsart	minimale Zykluszeit t_c	maximale Datenrate ^{*)}
PIO Mode 0	600 ns	3,33 MBytes/s
PIO Mode 1	383 ns	5,22 MBytes/s
PIO Mode 2	240 ns	8,33 MBytes/s
PIO Mode 3	180 ns	11,11 MBytes/s
PIO Mode 4	120 ns	16,67 MBytes/s

*) in jedem Zyklus werden 2 Bytes übertragen; Datenrate = 2 : t_c

4.3 Schnittstellenkommunikation

Kommunikation zwischen PIO 8255A (Host) und CD-ROM-Laufwerk (Gerät):



Rücksetzen:

RESET

Hardwareseitiges Rücksetzen (z. B. nach dem Einschalten). Erregung: vom Host. Minimale Impulsbreite: 25 µs.

Grundsätzliche Datenkonfiguration:

Zugriffsbreite	beteiligte Datenleitungen	Anwendung
8 Bits	DD7...0	Registerzugriffe,
16 Bits	DD15...0	Datenzugriffe

Adressierung:

DA2...0

Die drei Adreßleitungen dienen zur Registerauswahl (vgl. Abschnitt 9.3.8.). Erregung: vom Host.

CS1-, CS0-

Auswahl des Registerblocks (vgl. Abschnitt 9.3.8.):

- CS0-: Kommandoregisterblock,
- CS1-: Steuerregisterblock.

In DMA-Zugriffen müssen beide Signale inaktiv sein. Erregung: vom Host.

Bei allen unzulässigen Belegungen der DA- und CS-Signale (ungültige Registeradressen, CS1- und CS0- beide aktiv) ignoriert das Gerät den Zugriff und hält den Datenbus hochohmig.

CD-Player für den NDR-Klein-Computer

Gerätezuweisung:

CSEL

Eine Festbelegung, die dann wirksam wird, wenn in den Geräten Kabelauswahl (Cable Select) aktiviert ist:

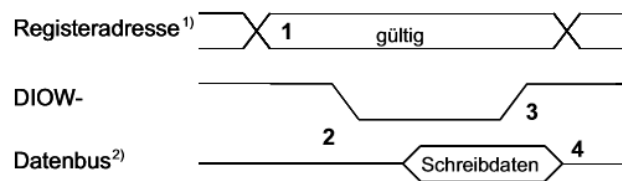
- das Gerät, das CSEL = 0 empfängt, wird zu Gerät 0,
- das Gerät, das CSEL = 1 empfängt, wird zu Gerät 1.

Die Nutzung von CSEL erfordert ein entsprechendes Kabel. An einem Kabel, in dem die CSEL-Anschlüsse 1:1 durchverbunden sind, funktioniert diese Form der Zuweisung nicht.

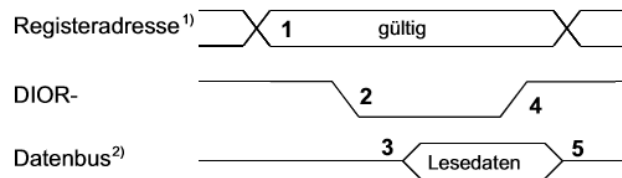
Zugriffssteuerung:

Registerzugriffe sind herkömmliche Zugriffe mit einer Zugriffsbreite von 8 Bits (Abbildungen 9.36 bis 9.38, Tabellen 9.15, 9.16).

a) Schreibzugriff



b) Lesezugriff



- 1): CS0-, CS1-, DA2...0
 2): Datenbits DD7...0 (8-Bit-Zugriffe)

Abbildung 9.36 Registerzugriffe (1). Schreiben und Lesen

Erklärung:

- a) Schreibzugriff: 1 - Host schaltet Registeradresse auf; 2 - Host aktiviert das Strobosignal und belegt den Datenbus mit dem zu schreibenden Byte; 3 - Host deaktiviert das Strobosignal. Infolgedessen werden die Schreibdaten vom Gerät übernommen; 4 - Host gibt Datenbus frei.
- b) Lesezugriff: Der Datenbus ist anfänglich hochohmig. 1 - Host schaltet Registeradresse auf; 2 - Host aktiviert das Strobosignal; 3 - Gerät belegt den Datenbus mit dem gelesenen Byte; 4 - Host übernimmt die Lesedaten und deaktiviert das Strobosignal; 5 - Gerät gibt Datenbus frei.

DIOW- (Ultra DMA: STOP)

Erregung: vom Host. Die Nutzung hängt von der Betriebsart ab:

- herkömmliche Zugriffe: Strobe-Signal für Schreibzugriffe. Datenübernahme (ins Gerät) mit der Low-High-Flanke.
- Ultra-DMA-Zugriffe: Endesignalisierung vom Host. Vor Beginn der Datenübertragung muß der Host STOP deaktivieren. Aktivierung während der laufenden Übertragung zeigt das Ende des Übertragungsablaufs an.

DIOR- (Ultra DMA: HDMARDY- (Lesen) oder HSTROBE (Schreiben))

Erregung: vom Host. Die Nutzung hängt von der Betriebsart ab:

- herkömmliche Zugriffe: Strobe-Signal für Lesezugriffe. Datenübernahme (in den Host) mit der Low-High-Flanke.
- Ultra DMA (Lesen): Übertragungssteuersignal. HDMARDY- = 0 zeigt dem Gerät an, daß der Host zur Datenübernahme bereit ist. HMARDY- = 1 bewirkt, daß die laufende Datenübertragung zeitweilig angehalten wird (eine Art Wartezustand).
- Ultra DMA (Schreiben): Strobosignal für Schreibzugriffe, das die Daten begleitet (gleiche Signalflußrichtung). Datenübertragung mit beiden Signalflanken.

IORDY (Ultra DMA: DSTROBE (Lesen) oder DDMARDY- (Schreiben))

Erregung: vom Gerät. Die Nutzung hängt von der Betriebsart ab:

- herkömmliche Zugriffe: Verlängerung des aktuellen Zugriffs (Einfügen von Wartezuständen). Folgende Zugriffe können verlängert werden: (1) Registerzugriffe, (2) PIO-Zugriffe (typischerweise ab Modus 3).
- Ultra DMA (Lesen): Strobosignal für Lesezugriffe, das die Daten begleitet (gleiche Signalflußrichtung). Datenübertragung mit beiden Signalflanken.
- Ultra DMA (Schreiben): Übertragungssteuersignal. DDMARDY- = 0 zeigt dem Host an, daß das Gerät zur Datenübernahme bereit ist. DDMARDY- = 1 bewirkt, daß die laufende Datenübertragung zeitweilig angehalten wird (eine Art Wartezustand).

4.4 ATA- und ATAPI-Geräte in der IDE-Schnittstelle

ATA-Geräte (Festplatten) werden mit bestimmten **Commands** angesteuert. Für **ATAPI-Geräte (CD-ROM)** ist dieser Satz an Kommandos zum Teil gegenüber Standard-ATA-Geräten eingeschränkt aber an anderer Stelle auch erweitert. (Für genaue Beschreibung siehe „sff8020i.pdf“.):

Ein **Command** wird durch **Beschreiben der Register** des Geräts abgesetzt. Der Registersatz wird auch „**Task File**“ genannt.

Registersatz bei ATAPI-Geräten – Task File (Für genaue Beschreibung siehe „sff8020i.pdf“.):

Registerblock	CS		Registeradresse DA				Register	
	1-	0-	2	1	0	Hex	Lesezugriff	Schreibzugriff
Kommando- registerblock	1	0	0	0	0	0	Datenregister	
	1	0	0	0	1	1	Fehlerregister	Funktionsmerkmale
	1	0	0	1	0	2	Unterbrechungsursache	
	1	0	0	1	1	3		
	1	0	1	0	0	4	Byteanzahl niedrig	
	1	0	1	0	1	5	Byteanzahl hoch	
	1	0	1	1	0	6	Geräteauswahl	
	1	0	1	1	1	7	Zustandsregister	Kommandoregister
Steuer- registerblock	0	1	1	1	0	6	Zustandsregister 2	Gerätesteuerregister

ATAPI = AT Attachment Packet Interface. Diese Spezifikation wurde entwickelt, um an das ATA-Interface weitere Geräte (außer Festplatten) auf reguläre Weise anschließen zu können. Die Grenzen der herkömmlichen ATA-Softwareschnittstelle liegen in dem beschränkten Registersatz, der nur 12 Bytpositionen enthält, die für Steuer- und Zustandsangaben verwendet werden können.

Der Grundgedanke: die Kommandos (und - in umgekehrter Richtung - Zustandsangaben usw.) werden in Form von Datenpaketen (d. h. über das Datenregister) übertragen. Hierzu hat man u. a. ein spezielles ATA-Kommando „Paketübertragung“ (PACKET) definiert. Dieses Kommando wird wie ein gewöhnliches ATA-Kommando in das Kommandoregister geladen. Die weiteren Register haben dann eine jeweils spezifische Bedeutung. Das Kommando bewirkt, daß nachfolgend das eigentliche Zugriffskommando über das Datenregister als Paket zum Gerät übertragen wird. Die Formate dieser Pakete hat man weitgehend dem SCSI-Standard angeglichen. ATAPI verwendet Kommandobeschreibungsblöcke (CDBs) von 12 oder 16 Bytes Länge. Diese Pakete werden stets in einer PIO-Betriebsart übertragen.

CD-Player für den NDR-Klein-Computer

Liste der ATA-Kommandos für ATAPI-Geräte (Für genaue Beschreibung siehe „sff8020i.pdf“.):

Command	used	Code	Error Register						Status Register					
			BBK	UNC	IDNF	ABRT	TKONF	AMNF	DRDY	DWF	DSC	CORR	ERR	
Acknowledge Media Change	N	DB				V							V	
ATAPI Soft Reset	M	08												
Boot - Post-boot	N	DC				V							V	
Boot - Pre-boot	N	DD				V							V	
Check Power Mode	M	E5				V			V	V	V		V	
Door Lock	N	DE				V							V	
Door Unlock	N	DF				V							V	
Download Microcode	N	92				V			V	V			V	
Media eject	N	ED				V			V		V		V	
Execute Drive Diagnostics	M	90	Special Drive Diagnostic Errors						V					V
Format Track	N ¹	50				V							V	
Identify Drive	N	EC				V							V	
Idle	O	E3				V			V	V	V		V	
Idle Immediate	M	E1				V			V	V	V		V	
Initialize Drive Parameters	N ¹	91				V							V	
NOP	M	00				V			V				V	
ATAPI Pkt. Command	M	A0	Contains Packet Command Status						V			V	V	
ATAPI Identify Device	M	A1				V			V		V		V	
Service	O	A2				V			V				V	
Read Buffer	N	E4				V							V	
Read DMA (w/retry)	N	C8				V							V	
Read DMA (wo/retry)	N	C9				V							V	
Read Long (w/retry)	N	22				V							V	
Read Long (wo/retry)	N ¹	23				V							V	
Read Multiple	N	C4				V							V	
Read Sector(s) (w/retry)	N ¹	20				V							V	
Read Sector(s) (wo/retry)	N ¹	21				V							V	
Read Verify Sector(s) (w/retry)	N ¹	40				V							V	
Read Verify Sector(S) (wo/retry)	N ¹	41				V							V	
Recalibrate	N	1x				V							V	

Command	used	Code	Error Register						Status Register				
			BBK	UNC	IDNF	ABRT	TKONF	AMNF	DRDY	DWF	DSC	CORR	ERR
Seek	N	7x				V							V
Set Features	M	EF				V			V	V	V		V
Set Multiple Mode	N	C6				V							V
Sleep	M	E6				V			V	V	V		V
Standby	O	E2				V			V	V	V		V
Standby Immediate	M	E0				V			V	V	V		V
Write Buffer	N	E8				V							V
Write DMA (w/retry)	N	CA				V							V
Write DMA (wo/retry)	N	CB				V							V
Write Long (w/retry)	N ¹	32				V							V
Write Long (wo/retry)	N ¹	33				V							V
Write Multiple	N	C5				V							V
Write Same	N	E9				V							V
Write Sector(s) (w/retry)	N ¹	30				V							V
Write Sector(s) (wo/retry)	N ¹	31				V							V
Write Verify	N	3C				V							V
Invalid Command Code						V			V	V	V		V

V = valid on this command
 Y = Used by the CD-ROM ATAPI Drive, as specified by the ATA Standard
 N = Not supported by ATAPI CD-ROM Drives
 M = Mandatory for ATA
 Shaded = Commands utilized by the ATAPI CD-ROM Drive

1. This command is specified as Mandatory for ATA, but *shall* NOT be supported by ATAPI Devices

CD-Player für den NDR-Klein-Computer

Ein „**Command**“ besitzt eine **Parameterliste**, mit der die **Register des Task Files** geschrieben werden. Hier ein Beispiel für das Command „IDENTIFY DEVICE“:

Register	7	6	5	4	3	2	1	0
Features	na							
Sector Count	na							
Sector Number	na							
Cylinder Low	na							
Cylinder High	na							
Device/Head	obs	na	obs	DEV	na	na	na	na
Command	ECh							

na = not available, obs = obsolete, DEV = device 0 /1

4.5 ATAPI-Command Packets

Mit dem **ATA-Kommando „ATAPI-Packet-Command“** werden ein „**Command Packet**“ und dessen **Parameter** an das **ATAPI-Gerät** übertragen. Ein „**Command Packet**“ wird - im Gegensatz zu einem ATA Command - **über das Datenregister übertragen** und nicht den „Task File“!

Liste der „**Command Packets**“ (Für genaue Beschreibung siehe „sff8020i.pdf“.):

Command	Opcode	May Return Not Ready Error
INQUIRY	12h	No
LOAD / UNLOAD CD	A6h	Yes
MECHANISM STATUS	BDh	No
MODE SELECT(10)	55h	No
MODE SENSE(10)	5Ah	No
PAUSE/RESUME	4Bh	Yes
PLAY AUDIO	45h	Yes
PLAY AUDIO MSF	47h	Yes
PLAY CD	BCh	Yes
PREVENT/ALLOW MEDIUM REMOVAL	1Eh	
READ(10)	28h	Yes
READ(12)	A8h	Yes
READ CD-ROM CAPACITY	25h	Yes
READ CD	BEh	Yes
READ CD MSF	B9h	Yes
READ HEADER	44h	Yes
READ SUB-CHANNEL	42h	Yes
READ TOC	43h	Yes
REQUEST SENSE	03h	No
SCAN	BAh	Yes
SEEK	2Bh	Yes
SET CD SPEED	BBh	No
STOP PLAY / SCAN	4Eh	Yes
START STOP UNIT	1Bh	Yes
TEST UNIT READY	00h	Yes

CD-Player für den NDR-Klein-Computer

Ein „**Command Packet**“ besitzt eine **Parameterliste**, die über das Datenregister übertragen wird und im Standard so aussieht. Der „**Operation Code**“ entspricht z.B. \$BC für das **Packet** „PLAY CD“:

Byte	Bit	7	6	5	4	3	2	1	0
0		Operation Code							
1		Reserved				Reserved			
2	(MSB)	Logical Block Address (if required)							
3									
4									
5									
6									
6		Reserved							
7 - 8	(MSB)	Transfer Length (if required) or Parameter List Length (if required) or Allocation Length (if required)(LSB)							
9		Reserved							
10		Reserved							
11		Reserved							

4.6 Ablauf ATA-Packet Command und Command Packet

Die Befehlssequenz für das **Auslösen des „ATAPI-Packet-Commands“** und die anschließende **Übertragung des „Command Packets“** sieht – wie folgt – aus.

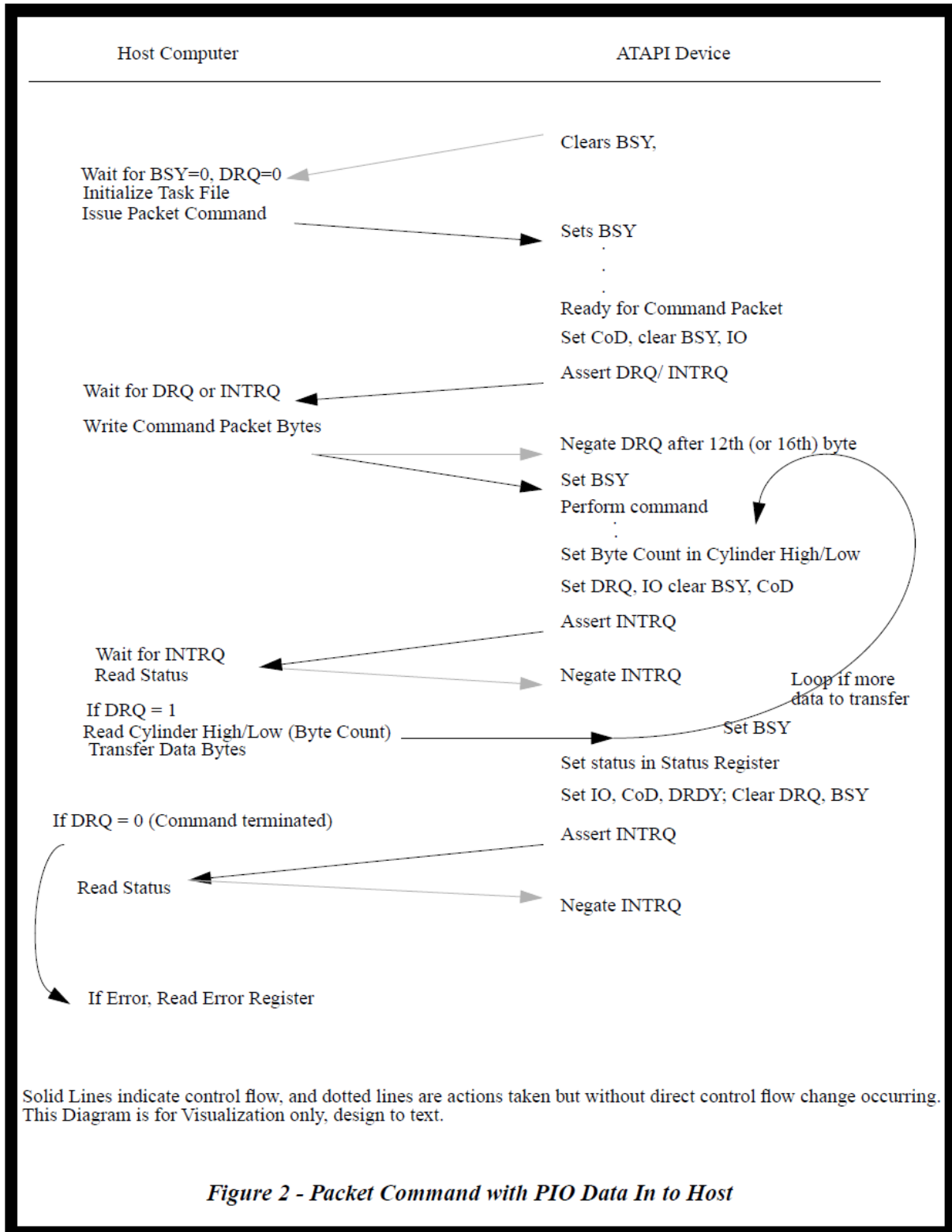
Fall A: Das Command Packet löst eine Datenübertragung an den Host aus:

Beispiele sind die Packets „inquiry“ und „read“, deren Ausführung eine Datenübertragung einer unbekanntenen Anzahl von Bytes vom Gerät in den Computer zur Folge hat:

1. Der Computer wartet auf BSY = 0, DRQ = 0 und initialisiert den Task File.
2. Der Computer schreibt nun das Packet Command (\$0A) in das Command Register.
3. Das Gerät setzt BSY = 1 vor dem nächsten Lesen des Status Registers und bereitet sich auf den Empfang des Command Packets vor.
4. Wenn das Gerät bereit zum Empfang des Command Packets ist, dann setzt es COD = 1, IO = 0 und DRQ = 1. Manche Geräte setzen auch INTRQ.
5. Nachdem der Computer DRQ = 1 festgestellt hat, überträgt er die 12 Bytes (6 Datenworte) des Packet Commands in das Datenregister des Geräts.
6. Das Gerät setzt DRQ = 0, wenn die 12 Bytes des Command Packets vollständig übertragen sind und danach BSY = 1. Es liest das Feature und Byte Count Register – wie vom Computer bestimmt – aus und bereitet sich auf den Datentransfer an den Computer vor.
7. Wenn Daten zur Übertragung vorhanden sind, schreibt das Gerät die Register Cylinder High und LOW, setzt COD = 0, IO = 1, DRQ = 1, BSY = 0 und INTRQ = 1.
8. Nachdem der Computer INTRQ = 1 festgestellt hat, liest er DRQ:
 - a. Wenn DRQ = 0, hat das Gerät das Command Packet abgebrochen.
 - b. Wenn DRQ = 1, dann soll der Computer die Anzahl Daten aus dem Cylinder High / LOW Register via Datenregister lesen.
 Nach dem Lesen von DRQ wird INTRQ = 0 vom Gerät gesetzt.
9. Wenn mehr Daten zu übertragen sind, setzt das Gerät DRQ = 0 und BSY = 1, und die Prozedur beginnt erneut ab Schritt 7.
10. Wenn das Gerät mit der gesamten Datenübertragung fertig ist und den Status präsentiert, setzt es COD = 1, IO = 1, DRDY = 1, BSY = 0, DRQ = 0, INTRQ = 1
11. Der Computer wartet auf INTRQ = 0 und DRQ = 0 und liest dann das Statusregister und ggfs. Das Error Register aus.

CD-Player für den NDR-Klein-Computer

Das Gerät verwendet das DRQ-Signal um anzuzeigen, dass es zum Datentransfer bereit ist. Es wird während des Transfers des letzten Bytes auf „Null“ gesetzt. Das gilt sowohl für das Command Packet als auch für normale Schreib-// Lesedaten!



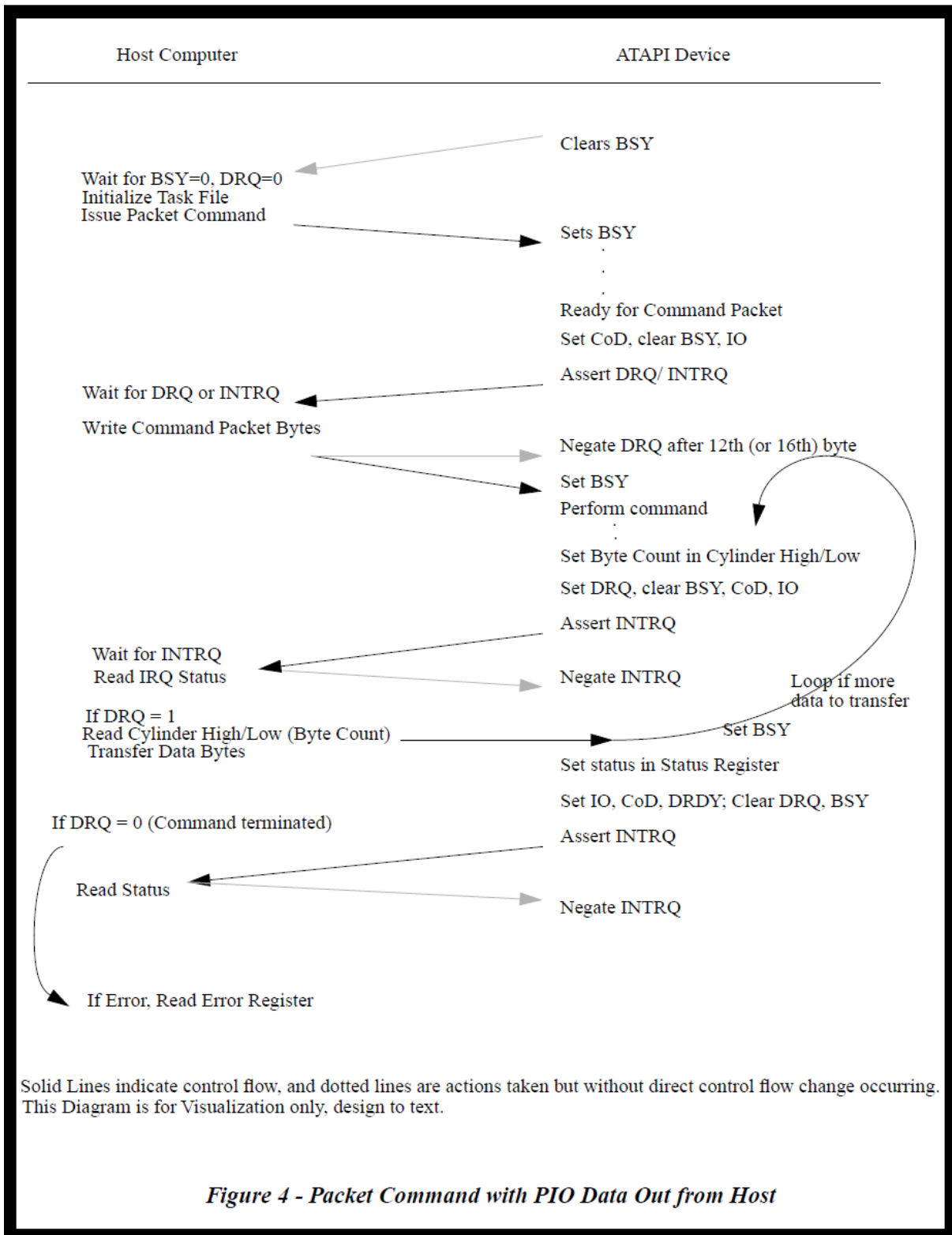
Fall B: Das Command Packet löst eine Datenübertragung an das ATAPI-Gerät aus:

Beispiel ist das Packet „modeselect“, dessen Ausführung eine Datenübertragung einer bekannten Anzahl von Bytes vom Computer zum Gerät zur Folge hat:

1. Der Computer wartet auf BSY = 0, DRQ = 0 und initialisiert den Task File mit dem Schreiben der notwendigen Parameter in Features, Byte Count und Drive / Head Register.
2. Der Computer schreibt nun das Packet Command (\$0A) in das Command Register.
3. Das Gerät setzt BSY = 1 vor dem nächsten Lesen des Status Registers und bereitet sich auf den Empfang des Command Packets vor.
4. Wenn das Gerät bereit zum Empfang des Command Packets ist, dann setzt es COD = 1, IO = 0 und DRQ = 1. Manche Geräte setzen auch INTRQ.
5. Nachdem der Computer DRQ = 1 festgestellt hat, überträgt er die 12 Bytes (6 Datenworte) des Packet Commands in das Datenregister des Geräts.
6. Das Gerät setzt DRQ = 0, wenn die 12 Bytes des Command Packets vollständig übertragen sind und danach BSY = 1. Es liest das Feature und Byte Count Register – wie vom Computer bestimmt – aus und bereitet sich auf den Datentransfer vom Computer vor.
7. Wenn das Gerät für die Übertragung bereit ist, setzt es den Byte Count (Cylinder High / LOW Register) auf den Wert, wie viele Daten das Gerät übertragen möchte, setzt COD = 0, IO = 0, DRQ = 1, BSY = 0 und INTRQ = 1.
8. Nachdem der Computer INTRQ = 1 festgestellt hat, liest er DRQ:
 - a. Wenn DRQ = 0, hat das Gerät das Command Packet abgebrochen.
 - b. Wenn DRQ = 1, dann soll der Computer die Anzahl Daten aus dem Cylinder High / LOW Register via Datenregister schreiben.
 Nach dem Lesen von DRQ wird INTRQ = 0 vom Gerät gesetzt.
9. Wenn mehr Daten zu übertragen sind, setzt das Gerät DRQ = 0 und BSY = 1, und die Prozedur beginnt erneut ab Schritt 7.
10. Wenn das Gerät mit der gesamten Datenübertragung fertig ist und den Status präsentiert, setzt es COD = 1, IO = 1, DRDY = 1, BSY = 0, DRQ = 0, INTRQ = 1
11. Der Computer wartet auf INTRQ = 0 und DRQ = 0 und liest dann das Statusregister und ggfs. Das Error Register aus.

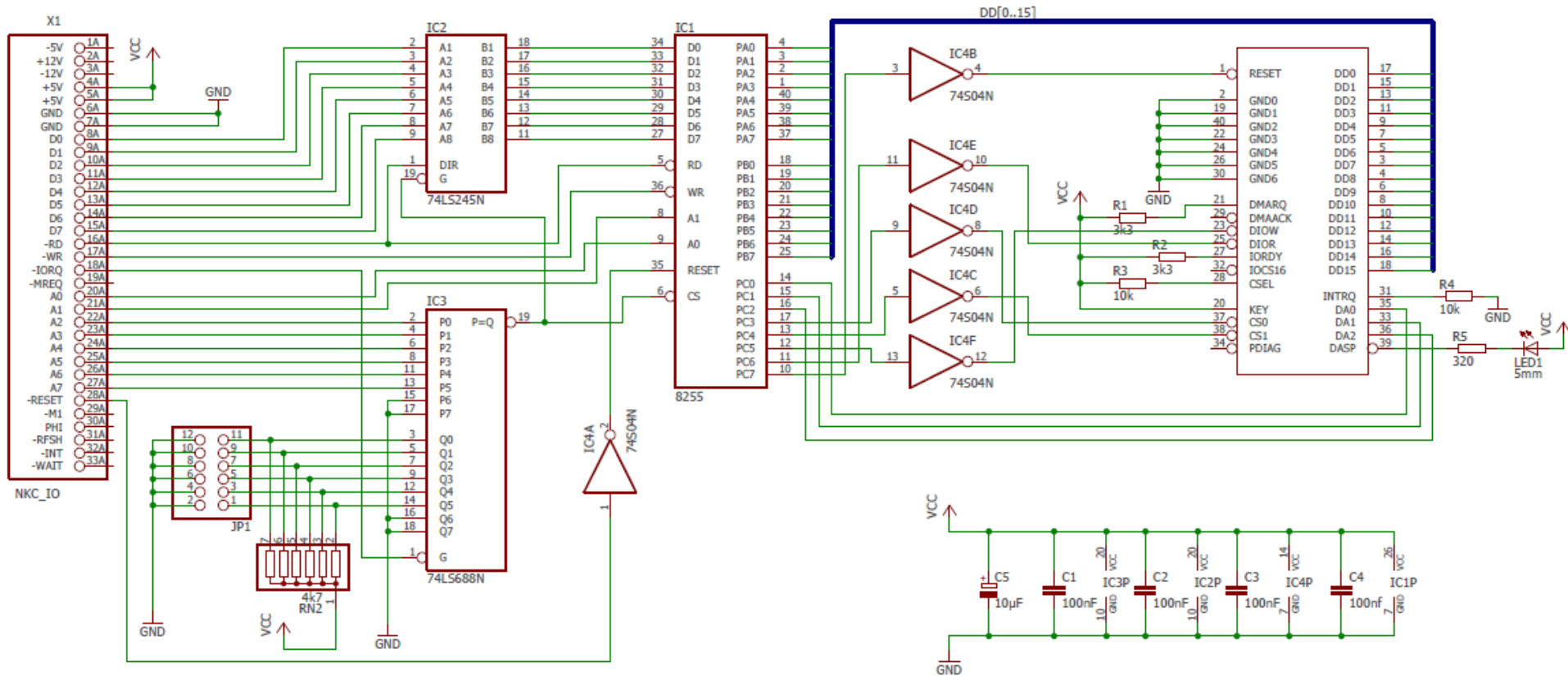
Das Gerät verwendet das DRQ-Signal um anzuzeigen, dass es zum Datentransfer bereit ist. Es wird während des Transfers des letzten Bytes auf „Null“ gesetzt. Das gilt sowohl für das Command Packet als auch für normale Schreib- / Lesedaten!

CD-Player für den NDR-Klein-Computer



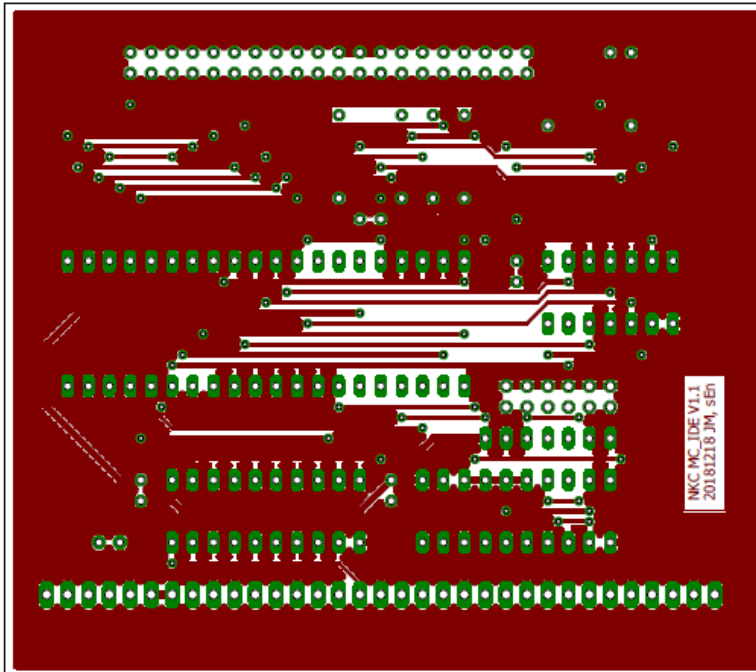
5 Schaltplan, Layout, Bestückungsplan und Stückliste NKC-MC-IDE-Karte

5.1 Schaltplan:

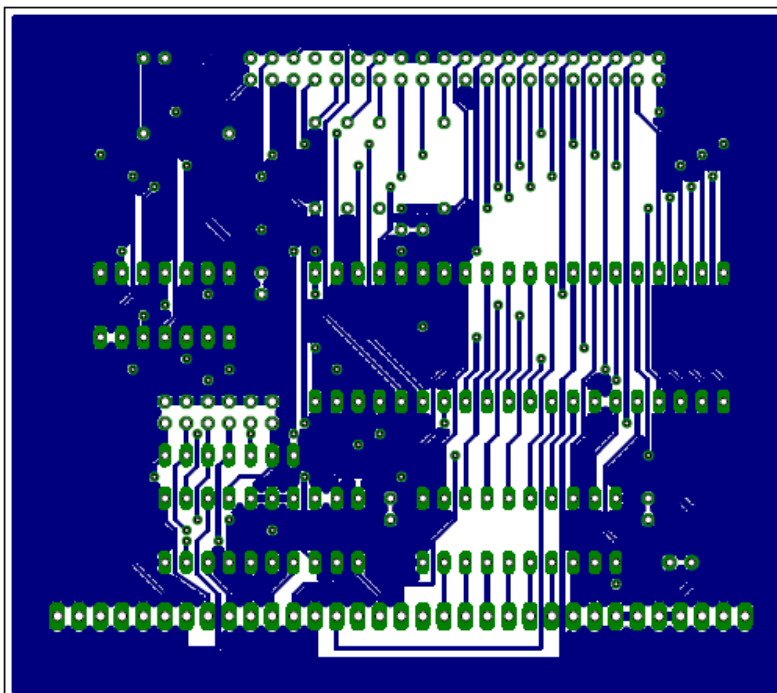


5.2 Layout:

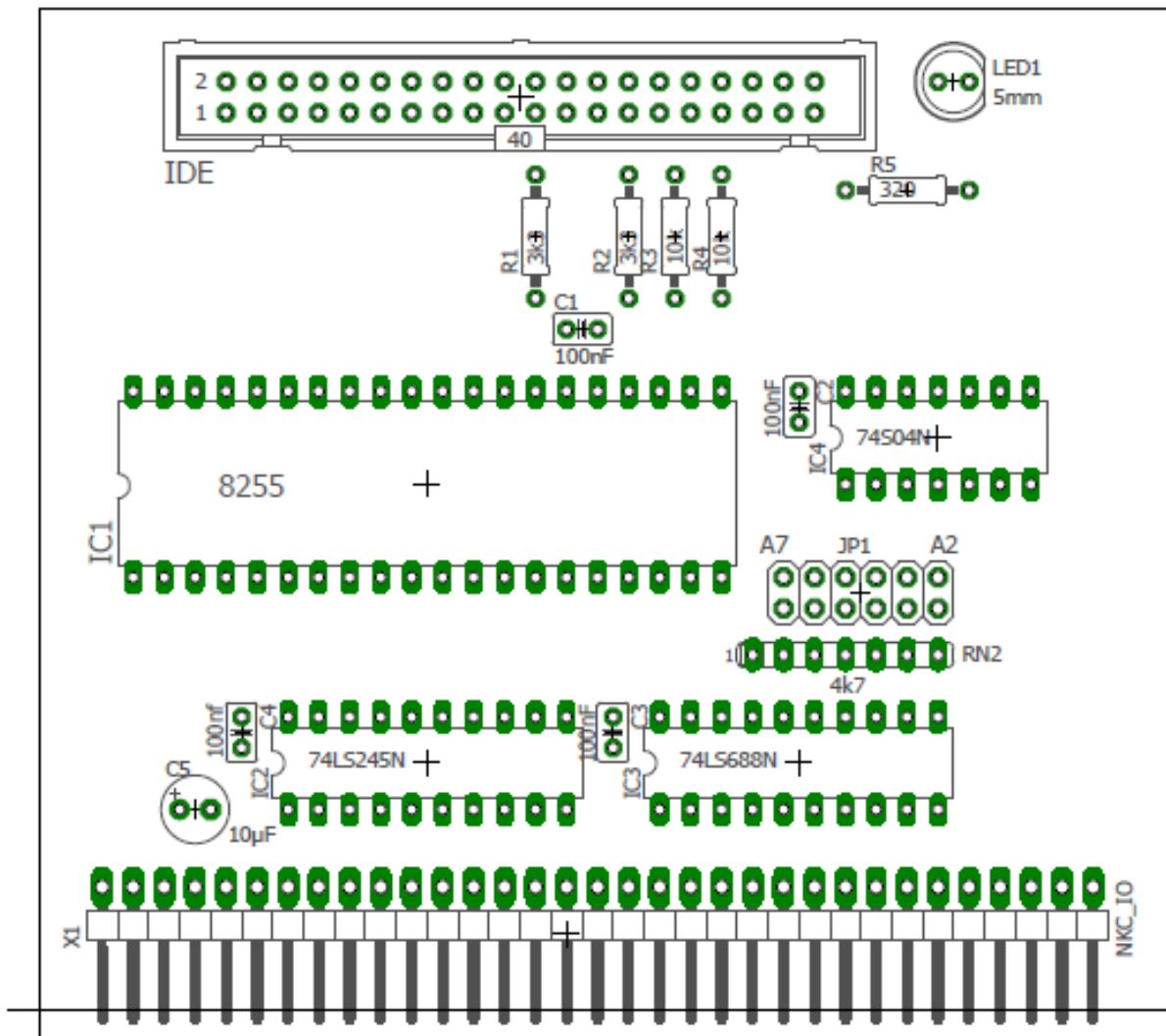
Bestückungsseite:



Rückseite:



5.3 Bestückungsplan:



5.4 Stückliste:

NKC-IDE-Karte

Mechanisch:

Industrielle Platine	1 Stück
Steckerleiste - einreihig, abgewinkelt, 33 polig (X1)	1 Stück
Steckerleiste - zweireihig, gerade, 6 polig (JP1)	1 Stück
Steckerleiste - zweireihig, gerade, 20 polig / IDE-Stecker	1 Stück
IC-Fassung - 14 polig (IC 4)	1 Stück
IC-Fassung - 20 polig (IC 2, 3)	2 Stück
IC-Fassung - 40 polig (IC 1)	1 Stück
DIE-Flachbandleitung für Verbindung Karte-Laufwerk	1 Stück

Widerstände (1/4 Watt):

320 Ω (R5)	1 Stück
3,3 K Ω (R1, R2)	2 Stück
10 K Ω (R3, R4)	2 Stück
4,7 K X 6 – Netzwerk (RN2)	1 Stück

Kondensatoren:

100 nF (C1 – C4)	4 Stück
47 μ F (C5)	1 Stück

LED:

5mm. Rot	1 Stück
----------	---------

TTL:

74 S / HCT 04 (IC 4)	1 Stück
74 LS / HCT 245 (IC 2)	1 Stück
74 LS / HCT 688 (IC 3)	1 Stück

PIO:

8255A (IC 1)	1 Stück
--------------	---------

Laufwerk:

IDE-CD-ROM-Laufwerk mit Audio-Anschluss	1 Stück
---	---------

6 Anmerkungen

6.1 Eingänge und Ausgänge:

Hier werden alle Ein- und Ausgänge sowie Steckverbinder und Jumper der NKC-MC-IDE-Karte und des CD-ROM-Laufwerks beschrieben:

Steckverbinder	Beschreibung	Standard	Bemerkung
JP1 (Karte)	Jumper zur Einstellung der NKC-IO-Adresse des PIO 8255A	ja	Im Bedienprogramm ist die IO-Adresse \$FFFFFF54 verwendet.
IDE (Karte)	IDE 40 PIN - Anschluss	ja	Mit IDE (Laufwerk) verbinden.
IDE (Laufwerk)	IDE 40 PIN - Anschluss	ja	Mit IDE (Karte) verbinden.
SV1 (Laufwerk)	Audio-Ausgang	ja	Entweder direkt mit einem (Vor-) Verstärker oder mit SV4 der NKC-Tunerkarte verbinden.
SV2 (Laufwerk)	Versorgungsspannung	ja	Direkt an das Netzteil des NKC anschließen.

6.2 Allgemeines Verhalten:

Beim Betrieb der NKC-MC-IDE-Karte wurden traten keine instabilen Situationen auf.

7 Aufbau und Test des NKC-CD-Players

Die Hardware des **NKC-CD-Players** besteht mindestens aus der **NKC-MC-IDE-Karte** und einem **CD-ROM-Laufwerk mit Audio-Ausgang**.

Ich verwende hier auch noch die **NKC-Tunerkarte** mit ihrem über den I2C-Bus programmierbaren **Vorverstärker / Klangsteller**.

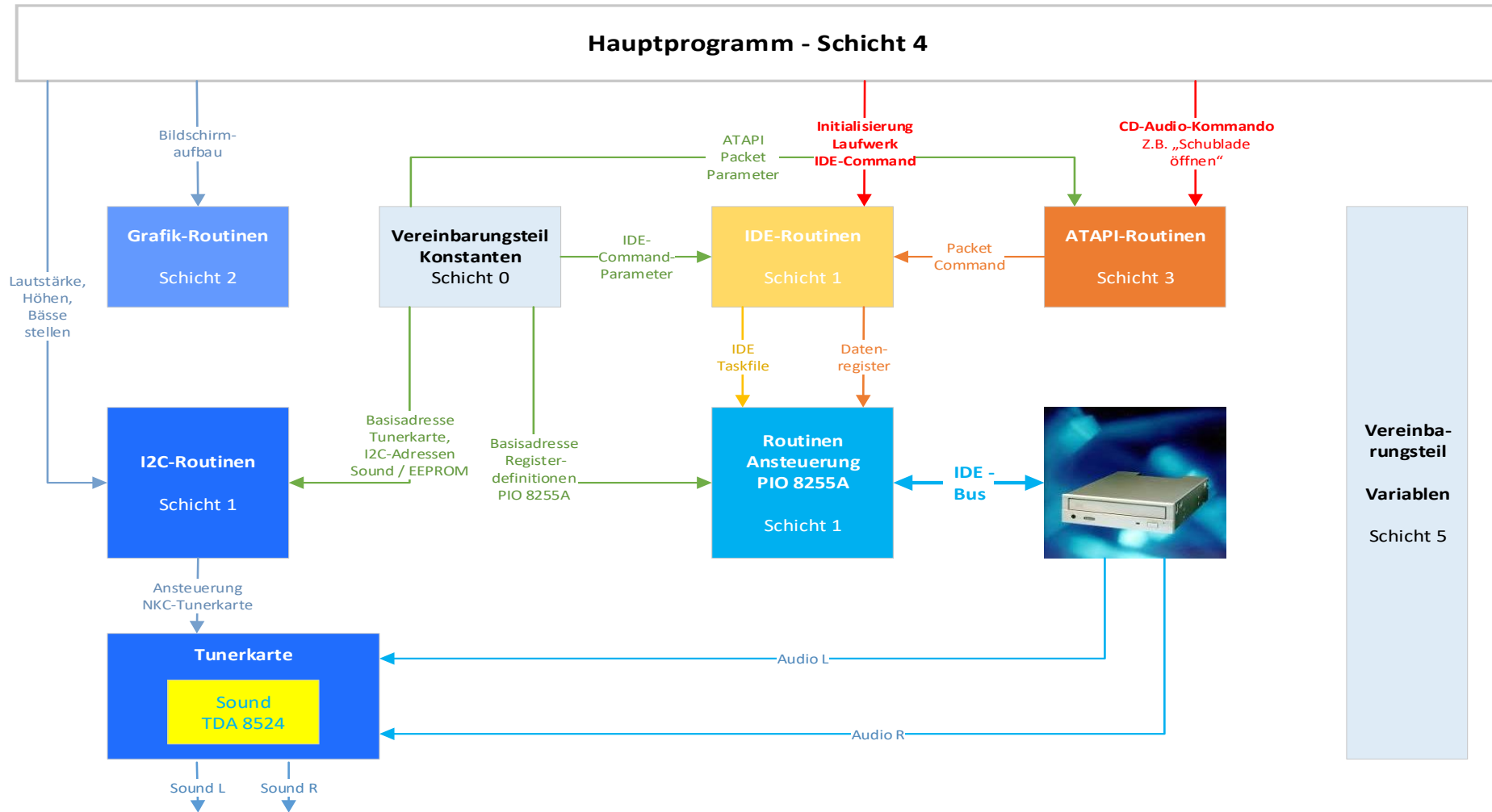
7.1 Programmierung mit Schichtenkonzept:

Die Programmierung des NKC-Players erfolgt in einem **Schichtenkonzept**:

Schicht	Objekt	Beschreibung / Bemerkung
0	Vereinbarungsteil-Konstanten	Konstantendeklaration
1	Variableninitialisierung, CI-Tastatur-eingaberoutine - intern, I2C-Bussteuerung für die NKC-Tunerkarte, Basisroutinen zur Ansteuerung des PIO 8255A, IDE-Routinen, ATAPI-Routinen, Uhrabfrage	Für I2C-Steuerung siehe Projekt „NKC-Tunerkarte“. Als IDE-Routinen z.B. „identify drive“ oder „packet command“. Als ATAPI-Routinen z.B. „test unit ready“ oder „table of contents“.
2	Bildschirmmaske, Grafik-Subroutinen	
3	CD-Player - Routinen - Aktionen, die aus der Eingabe im Hauptprogramm resultieren ...	U.a. Toneinstellung, Playlist, Single- und Listplay, Wiederholung, Shuffleplay ...
4	Hauptprogramm mit Hauptprogrammsschleife	Initialisierung von Variablen, I2C-Bus und Ton auf der Tunerkarte sowie des Laufwerks. Ständige Abfrage der Urzeit, der Tastatur und von Trackpositionsdaten während Play
5	Vereinbarungsteil-Variablen	Variablendeklaration

CD-Player für den NDR-Klein-Computer

7.2 Schematischer Ablauf des Programms:



Aus dem Hauptprogramm heraus geschehen folgende, prinzipielle Aktionen:

- Aufbau der Bildschirmmaske
- Stellen von Lautstärke, Höhen und Bässe via I2C-Bus auf der NKC-Tunerkarte
- Initialisierung des Laufwerks direkt mit IDE-Commands (Taskfile) über IDE-Routinen
- CD-Audio-Commands via IDE-Packet-Command und entsprechendem Command Packet über ATAPI-Routinen
- Eine IDE-Routine steuert wiederum die Routinen zur Ansteuerung des PIO 8255A an.
- Der PIO 8255A stellt die IDE-Schnittstelle zum Laufwerk bereit.

8 Bedienprogramm

Es gibt ein fertiges Bedienprogramm in 680XX-Assembler für den NKC-CD-Player. Die aktuelle Version ist V2.0 und als Dateien „nkccdpv20.asm“ und „nkccdpv20.68k“ verfügbar.

8.1 Parameter und Systemkonfiguration

- **Basisadresse der NKC-MC-IDE-Karte (PIO 8255A)**
- **Basisadresse NKC-Tunerkarte**
- **I2C-Adressen von Soundbaustein und EEPROM auf der NKC-Tunerkarte**
- **Konstanten für IDE-Kommandos**
- **Konstanten für ATA Commands**
- **Konstanten für ATAPI Command Packets**
- **Uhrenkarte**

8.2 Funktionen

- **CD:**
 - o Öffnen / Schließen des Laufwerks
 - o Spielen eines einzelnen, ausgewählten Tracks
 - o Festlegen einer Playliste durch Markieren / Entmarkieren einzelner Tracks
 - o Spielen der Playliste bzw. der gesamten CD, da am Anfang alle Tracks markiert sind.
 - o Pause
 - o Stop
 - o Abspielmodus:
 - Normal
 - Wiederholung
 - Shuffle
 - o Programm beenden
- **Audioeinstellungen:**
 - o Lautstärke
 - o Höhen
 - o Bässe
- **Anzeigen:**
 - o Datum und Uhrzeit, wenn eine Uhrenkarte im NKC vorhanden ist
 - o Totale Spielzeit einer CD, wenn diese geladen ist
 - o Abspielmodus:
 - Normal = keine Anzeige
 - Wiederholung = Ellipse
 - Shuffle = Kreuz
 - o Inhaltsverzeichnis mit Track-Nummern, wenn CD geladen ist
 - o Markierung am Track, ob dieser zur Playlist gehört oder nicht:
 - Playlist = *

CD-Player für den NDR-Klein-Computer

- Keine Playlist = SPACE
 - Totale Spielzeit des ausgewählten Tracks, wenn eine CD geladen ist
 - Abgelaufene Spielzeit eines Tracks, wenn dieser gespielt wird:
 - als Anzeigebalken
 - als Zeitangabe unter dem Track-Total
 - Audio-Einstellung – Schieber für:
 - Lautstärke
 - Höhen
 - Bässe
 - Info / Status:
 - Fehler- oder Statusmeldung
 - Play
 - Pause
 - Stop
 - Beschreibung der Funktionen
- **Speicherung der Audio-Einstellungen** auf dem EEPROM der NKC-Tunerkarte:

8.3 Masken

Für dieses Programm existiert nur 1 Bildschirmmaske, die je nach Funktion / Zustand mit unterschiedlichen Anzeigen gefüllt ist:

- Keine CD geladen:



CD-Player für den NDR-Klein-Computer

- Schublade offen:



CD-Player für den NDR-Klein-Computer

- CD geladen:



CD-Player für den NDR-Klein-Computer

- Play:



CD-Player für den NDR-Klein-Computer

- Pause:



CD-Player für den NDR-Klein-Computer

- Stop:



CD-Player für den NDR-Klein-Computer

- Modus „Wiederholung“:



CD-Player für den NDR-Klein-Computer

- Modus „Shuffle“:



CD-Player für den NDR-Klein-Computer

- Modus „Wiederholung / Shuffle“ und Playliste durch Markierung bestimmter Tracks:

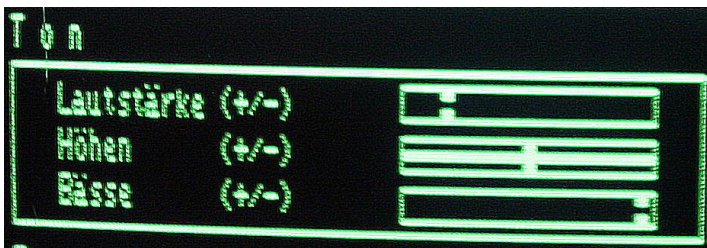


CD-Player für den NDR-Klein-Computer

- Lautstärke einstellen:



- Höhen einstellen:



- Bässe einstellen:



CD-Player für den NDR-Klein-Computer

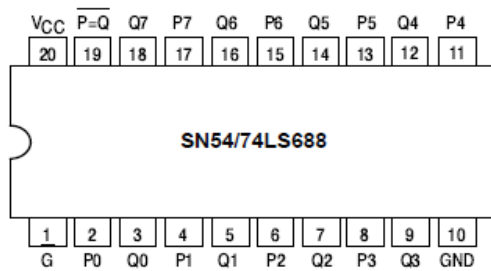
- Programmende:



9 Anhang

9.1 Datenblätter TTL-Bausteine:

9.1.1 74LS688



TYPE	P = Q	P > Q	OUTPUT ENABLE	OUTPUT CONFIGURATION	PULLUP
LS682	yes	yes	no	totem-pole	yes
LS684	yes	yes	no	totem-pole	no
LS688	yes	no	yes	totem-pole	no

FUNCTION TABLE

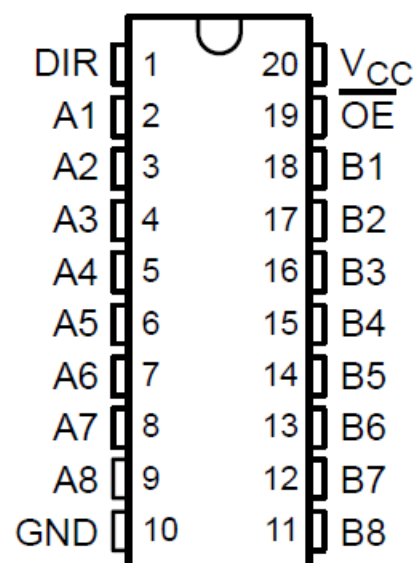
INPUTS			OUTPUTS	
DATA	ENABLES		$\overline{P = Q}$	$\overline{P > Q}$
P, Q	$\overline{G}, \overline{GT}$	$\overline{G2}$	$\overline{P = Q}$	$\overline{P > Q}$
P = Q	L	L	L	H
P > Q	L	L	H	L
P < Q	L	L	H	H
X	H	H	H	H

H = HIGH Level, L = LOW Level, X = Irrelevant

9.1.2 74LS245

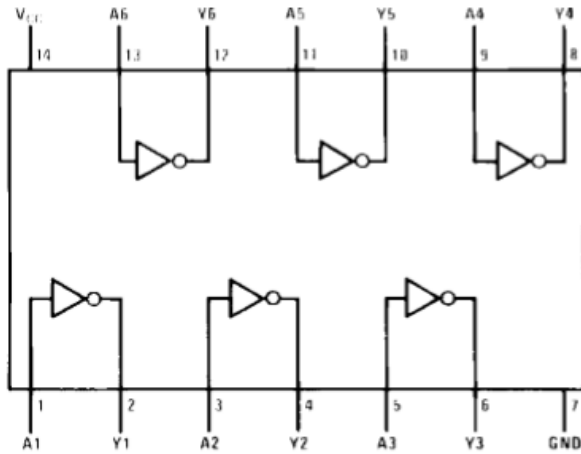
FUNCTION TABLE

INPUTS		OPERATION
\overline{OE}	DIR	
L	L	B data to A bus
L	H	A data to B bus
H	X	Isolation



9.1.3 74S04

Connection Diagram



Function Table

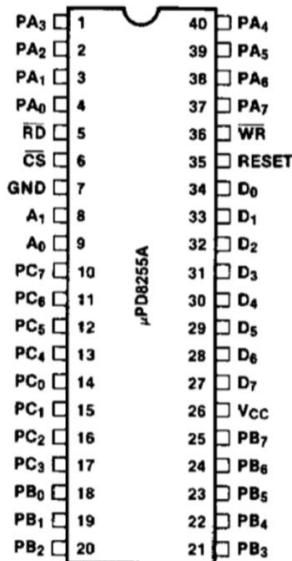
$$Y = \bar{A}$$

Input	Output
A	Y
L	H
H	L

H = HIGH Logic Level
 L = LOW Logic Level

9.2 Datenblatt PIO:

9.2.1 8255A



Pin Identification

No.	Symbol	Function
1-4, 37-40	PA ₇ -PA ₀	Port A (I/O)
5	\overline{RD}	Read input
6	\overline{CS}	Chip select input
7	GND	Ground
8,9	A ₁ , A ₀	Port address inputs
10-17	PC ₇ -PC ₀	Port C (I/O)
18-25	PB ₇ -PB ₀	Port B (I/O)
26	V _{CC}	+5 V power supply
27-34	D ₇ -D ₀	Bidirectional data bus
35	RESET	Reset input
36	\overline{WR}	Write input

9.3 Verweis auf Datenblätter komplexer Bausteine und Spezifikationen / Quellennachweis

Baustein/Objekt	Funktion	Datenblatt / Spezifikation
8255A	PIO	8255A.pdf
IDE / ATA / ATAPI-Schnittstelle	Schnittstelle zu IDE / ATA / ATAPI-Geräten	d1321r3-ATA-ATAPI-5.pdf, sff8020i.pdf, Technik_der_Personalcomputer_TL09.pdf
TDA 8425	I2C-Sound	TDA8425.pdf
Spezifikation NKC-Tunerkarte	I2C-Sound (2. Eingang)	SP_NKC_Tunerkarte_V15_20180429.pdf