

1. Der PCI-Bus

1.1. Grundlagen

1.1.1. Der PCI-Bus im PC

Der PCI-Bus wurde ursprünglich (Anfang der 90er Jahre) als reiner Lokalkbus für Motherboards "gewöhnlicher" PCs entwickelt. Er ist vorzugsweise zum Anschließen peripherer Einrichtungen vorgesehen (PCI = Peripheral Components Interconnect). Solche Einrichtungen können sowohl als Schaltkreise auf dem Motherboard als auch in Form von Steckkarten ausgeführt sein. Die ursprünglichen Entwicklungsziele lassen sich folgendermaßen beschreiben:

- der Bus sollte leistungsfähiger sein (höhere Datenraten, kürzere Latenzzeiten) als die seinerzeit gängigen Bussysteme (ISA, EISA, MCA),
- der Bus sollte moderne Systemkonzepte unterstützen (32-Bit-Adressierung, Caches, Multi-Master-Betrieb, Burst-Zugriffe, automatische, durch Software steuerbare Konfiguration (Stichwort: Plug and Play)),
- es sollte ein echter, sowohl in funktioneller als auch in elektrischer Hinsicht wohldefinierter Standard geschaffen werden,
- Abwärtskompatibilität zu den herkömmlichen PC-Systemen,
- Kostenoptimierung auf den Einsatzzweck hin (Lokalkbus in üblichen PCs - also Verzicht auf eine "total universelle" Auslegung (in bewußter Abgrenzung beispielsweise zu VME-Bus, Multibus und Futurebus)).

Die Abbildungen 1.1 bis 1.4 veranschaulichen die typische Nutzung des PCI-Bus in Personalcomputern.

Abbildung 1.1 Der PCI-Bus in einem PC (rechts oben: das PCI-Logo; Quelle: PCI SIG)

Grundlage ist die herkömmliche PC-Struktur mit einem einzigen zentralen Prozessor, der direkt mit dem Arbeitsspeicher zusammengeschaltet ist. Der Peripheriebus ist dem Prozessor-Arbeitsspeicher-Verbund über besondere Koppel- und Steuerschaltungen nachgeordnet (Abbildung 1.2). In der PCI-Terminologie heißen derartige Buskoppelschaltungen "Brücken" (Bridges).

Abbildung 1.2 Die Anordnung des Peripheriebus in einem PC-System (stark vereinfacht). 1 - Steuerung des Prozessorbus (Host Bus Interface State Machine), 2 - Steuerung des Peripheriebus (Peripheral Bus Interface State Machine)

Weshalb verwendet man nicht einen einzigen, wirklich universellen Bus?

Der Prozessor-Arbeitsspeicher-Verbund muß in sich auf Leistung optimiert sein. Die Auslegung der Signalwege und Steuerschaltungen wird somit stets vom Prozessortyp und von der jeweiligen Arbeitsspeichertechnologie (z. B. FPM-DRAM, EDO-DRAM, SDRAM, Rambus usw.) abhängen. Des Weiteren ist das Cache-Subsystem (der L2-Cache) zu berücksichtigen. Der Peripheriebus hingegen sollte ein allgemeingültiger, gleichsam "überzeitlicher" Standard sein, um sicherzustellen, daß man die periphere Ausstattung der PCs unabhängig von Prozessortyp, Arbeitsspeichertechnologie usw. jederzeit erweitern bzw. verändern kann.

Die Lösung: beide Bussysteme (Prozessor- bzw. Host-Bus) und Peripheriebus werden über eine *Brücken-Hardware* (Bus Bridge) zusammengeschaltet.

Die Probleme:

1. beide Bussysteme unterscheiden sich üblicherweise stark voneinander (hinsichtlich Zugriffsbreite, Fehlerkontrolle, Steuerprinzipien, Taktfrequenzen bzw. Datenraten usw.),
2. im allgemeinen Fall sind Zugriffe in beiden Richtungen zu unterstützen: (1) der Prozessor muß - wie es sich von selbst versteht - auf den Peripheriebus zugreifen können. Darüber hinaus ist es (2) von Vorteil, wenn periphere Einrichtungen ihrerseits direkten Zugang zum Arbeitsspeicher haben.

Wir wollen den einen Bus als primären Bus bezeichnen, den anderen als sekundären. Ist eine Einrichtung am primären Bus als Master wirksam und möchte diese auf eine Einrichtung (als Slave) am sekundären Bus zugreifen, so wirkt die Brücke am primären Bus als Slave und am sekundären als Master. Befindet sich hingegen der Master am sekundären und der Slave am primären Bus, verhält es sich umgekehrt. Tabelle 1.1 gibt einen Überblick über die Datenflüsse bei Schreib- und Lesezugriffen.

Hinweis:

In der PCI-Terminologie heißt der Slave "Target" (Abschnitt 1.1.3.). Wir verwenden diesen Begriff bereits in Tabelle 1.1.

Zugriff	Datenfluß	Wirkung der Brücke	
		am primären Bus	am sekundären Bus
<i>Master am primären, Target am sekundären Bus</i>			
Schreiben	Master → primärer Bus → Brücke → sekundärer Bus → Target (der sekundäre Bus empfängt die Daten vom primären)	<ul style="list-style-type: none"> ■ ist Target, ■ nimmt Daten entgegen <p style="text-align: right;">⇒</p>	<ul style="list-style-type: none"> ■ ist Master, ■ schreibt Daten <p style="text-align: right;">⇒*)</p>
Lesen	Target → sekundärer Bus → Brücke → primärer Bus → Master (der primäre Bus empfängt die Daten vom sekundären)	<ul style="list-style-type: none"> ■ ist Target, ■ liefert Daten <p style="text-align: right;">←</p>	<ul style="list-style-type: none"> ■ ist Master, ■ liest Daten <p style="text-align: right;">←*)</p>
<i>Master am sekundären, Target am primären Bus</i>			
Schreiben	Master → sekundärer Bus → Brücke → primärer Bus → Target (der primäre Bus empfängt die Daten vom sekundären)	<ul style="list-style-type: none"> ■ ist Master, ■ schreibt Daten <p style="text-align: right;">←</p>	<ul style="list-style-type: none"> ■ ist Target, ■ nimmt Daten entgegen <p style="text-align: right;">←*)</p>
Lesen	Target → primärer Bus → Brücke → sekundärer Bus → Master (der sekundäre Bus empfängt die Daten vom primären)	<ul style="list-style-type: none"> ■ ist Master, ■ liest Daten <p style="text-align: right;">⇒</p>	<ul style="list-style-type: none"> ■ ist Target, ■ liefert Daten <p style="text-align: right;">⇒*)</p>

*) : Pfeile kennzeichnen Richtung des Informationsflusses

Tabelle 1.1 Datenflüsse über eine Bus-Brücke bei Schreib- und Lesezugriffen

Eine solche Bus-Brücke ist eine aktive Schaltung (mit eigenen State Machines usw.), die folgende Funktionen ausführen kann:

- Adressierung: wandelt die Adresse, wie sie vom primären Bus geliefert wird, in die Darstellung um, wie sie der sekundäre Bus erwartet,
- Datenwegkopplung: wandelt die Daten in die Darstellung um, wie sie der jeweilige Bus erwartet. Dies kann folgende Sonderfunktionen einschließen:
 - Ausführung mehrerer Buszugriffe (wenn sich beide Bussysteme in der Zugriffsbreite unterscheiden; Beispiel: 32-Bit-Zugriffe auf einen 16-Bit-Bus),
 - Ausführung von Read-Modify-Write-Zyklen (wenn die Zugriffsbreite des empfangenden Bus größer und wenn byteweises Schreiben nicht möglich ist; Beispiel: 16-Bit-Zugriffe auf einen Arbeitsspeicher mit 8 Bytes Zugriffsbreite und ECC),

- byte- bzw. bitweises Umordnen (wenn das eine Bussystem für Rechts- und das andere für Linksadressierung ausgelegt ist) bzw. wenn bestimmte Bytes auf bestimmten Datenleitungen erwartet werden (Byte Swapping; als Beispiel vgl. die Beschreibung des ISA-Bus im Fehlersuchhandbuch, insbesondere Abbildung 5.103).
- Steuerung: diese Schaltungen steuern sowohl die Informationswandlungen innerhalb der Bus-Brücke als auch die Bussysteme selbst. Hierzu sind auf jeder Seite der Brücke (und auch in den einzelnen Einrichtungen) sequentielle Steuerschaltungen (Bus Interface State Machines) erforderlich.

Hinweis: Dieses Prinzip ist allen im vorliegenden Heft beschriebenen Systemen gemeinsam: alle Einrichtungen sind State Machines, die die Buszyklen beobachten und die Zustandsübergänge verfolgen. Im Gegensatz zu den "einfachen" Bussystemen, ist es nicht so, daß die jeweils aktuelle Busbelegung allein den jeweiligen Zustand repräsentiert und somit im wesentlichen nur kombinatorisch entschlüsselt werden muß.

Mehrere Bussysteme

Über weitere Brückenschaltungen (Bus Bridges) können verschiedenartige Bussysteme miteinander verkoppelt werden. Typische PCs haben neben einem PCI-Bus noch einen herkömmlichen Erweiterungsbus (Abbildungen 1.3 und 1.4). Dieser "althergebrachte" Erweiterungsbus (Legacy Bus; Legacy = "Erblast") ist meistens ein ISA-Bus; es gibt aber auch solche PCs mit EISA- oder MCA-Bussystem.

Hinweis: Die mechanische Auslegung des PCI-Bus ist derart spezifiziert, daß PCI-Steckkarten in PCs mit allen genannten herkömmlichen Bussystemen eingesetzt werden können (zu Spitzfindigkeiten siehe Abschnitt 1.1.4.4.).

Abbildung 1.3 PC mit mehreren Peripheriebussystemen (Prinzip)

Erklärung:

Die Abbildung zeigt eine Struktur, wie sie für die PCs des Massen-Marktes typisch ist. Es sind insgesamt 5 Bussysteme vorhanden:

1. der eigentliche Prozessor-Bus (Host-Bus). Er verbindet den Prozessor mit dem L2-Cache sowie mit einem ersten Steuerschaltkreis (1).
2. das DRAM- bzw.- Arbeitsspeicher-Interface,
3. der PCI-Bus. Dieser ist über die Host-to-PCI Bridge im ersten Steuerschaltkreis mit dem Verbund aus Prozessor, L2-Cache und Arbeitsspeicher gekoppelt.
4. der herkömmliche Erweiterungsbus (typischerweise: ISA). Er ist dem PCI-Bus über einen zweiten Steuerschaltkreis (PCI-Legacy Bridge) nachgeschaltet.
5. ein Motherboard-Peripheriebus, an den die "Elementarperipherie" des PC angeschlossen ist (Floppy-Disk-Controller, Tastaturcontroller, BIOS-ROM usw.). Dies ist typischerweise der vom ISA-Bus gleichsam abgezweigte X-Bus bzw. der LPC-Bus (Kapitel 3).

Abbildung 1.4 PC mit mehreren Bussystemen (Ausführungsbeispiel; Quelle: Intel)

Erklärung:

Dieses Pentium-II-System kann mit zwei Prozessoren ausgerüstet werden. (Hinweis zum L2-Cache: der Pentium II hat einen in der Steckkassette (SEC Cartridge) eingebauten L2-Cache.) Des weiteren sind

ein PCI-Bus, ein AGP-Bus und ein ISA-Bus vorgesehen. Die gesamte Brücken- und Steuerhardware ist in nur 2 Schaltkreisen (82443LX, 82371SB) untergebracht. Diese Schaltkreise enthalten zudem mehrere zum "Industriestandard" gehörende periphere Anschlußsteuerungen (z. B. für 2 IDE- und 2 USB-Ports).

Übersicht über die Merkmale des PCI-Bus

In den folgenden Abschnitten wird der PCI-Bus in Struktur und Funktionsweise näher beschrieben. Wir beginnen mit einer kurzen Aufzählung:

- Steuerprinzipien: Multimaster-Bussystem mit zentraler Steuerung. Vollsynchrone Arbeitsweise (Bustakt).
- Masterauswahl (Arbitrierung): Prinzip der unabhängigen Anforderungen, zentrale Vermittlung,
- Übertragung von Daten und Adressen: zeitmultiplex über gemeinsame Signalwege,
- Übertragungssteuerung: synchronisiertes Handshaking,
- Übertragungsbreite: 32 Bits, auf 64 Bits erweiterbar,
- Adreßräume: 3 (Speicher, Ein- und Ausgabe, Konfiguration),
- geographische Adressierung: ist vorgesehen (Konfigurationsadreßraum),
- Fehlerkontrolle: durch Paritätsprüfung.

Stichworte zur Kostenoptimierung:

- zentrale Steuerung: wenn zweckmäßig, werden zentrale Steuermittel vorgesehen (das betrifft z. B. die Masterauswahl),
- Zeitmultiplex-Auslegung: durch zeitmultiplexe Übertragung von Adressen und Daten über dieselben Signalwege verringert sich die Anzahl der Busleitungen und Anschlüsse,
- elektrische Auslegung: Optimierung auf CMOS-Technologie; PCI-Einrichtungen lassen sich komplett in einzelnen CMOS-Schaltkreisen realisieren, besondere Bustreiber und -empfänger sind nicht erforderlich,
- im einzelnen werden u. a. folgende Tatsachen ausgenutzt, um den PCI-Bus kostengünstig zu gestalten:
 - Motherboards haben vergleichsweise geringe Abmessungen, so daß typischerweise Bus-Leitungslängen von höchstens 10...30 cm zu erwarten sind,
 - es ist nicht schwierig, auf Motherboards Einzelleitungen zwischen den angeschlossenen Einrichtungen und den zentralen Steuerschaltungen zu verlegen (im Gegensatz beispielsweise zu einem Bussystem, das über Kabel geführt ist),
 - es sind ohnehin zentrale Steuermittel vorhanden (die Motherboard-Steuerschaltkreise) - und darin kann man auch noch die PCI-Steuerung unterbringen.

Erweiterung des PCI-Bus

Der PCI-Bus ist sowohl in seiner Ausdehnung (Leitungslänge) als auch hinsichtlich der Zahl der anschließbaren Einrichtungen ziemlich strengen Beschränkungen unterworfen (Näheres in Abschnitt 1.1.8.). Man kann also einen PCI-Bus nicht einfach "verlängern", um größere Systeme aufzubauen. Der Ausweg: mehrere PCI-Bussysteme vorsehen und über besondere Brücken (PCI-to-PCI Bridges) untereinander verbinden (Näheres in Abschnitt 1.5.).

1.1.2. Entwicklungsgeschichte und Standardisierung

Die anfängliche Initiative ging im besonderen von der Fa. Intel aus. Eine erste Spezifikation wurde 1992 vorgestellt. Erste Produkte erschienen 1993/94. PCI ist aber kein herstellerspezifischer Standard. Vielmehr gibt es ein hersteller-unabhängiges Standardisierungsgremium: die Peripheral Components Interconnect Special Interest Group (PCI SIG), in der jeder Mitglied werden kann, der sich an der Weiterentwicklung des PCI-Bus beteiligen möchte.

PCI-Standards sind zwar nicht kostenlos, aber ohne weiteres zugänglich (und durchaus bezahlbar). Zu den einzelnen Spezifikationen, weiteren Informationsquellen und Bezugsmöglichkeiten sei auf den Anhang verwiesen.

Der Standard, der den PCI-Bus an sich beschreibt, heißt genau "PCI Local Bus Specification". Hiervon gibt es verschiedene Ausgaben (Revisions; Tabelle 1.2).

Ausgabe (Revision)	Stichworte zum Inhalt	Ausgabejahr
1.0	1. Entwurf	1992
2.0	Spezifikation von Steckverbinder und Steckkarten	1993
2.1	Klärung von Problemstellen (Clarifications)*), Erweiterung auf 66 MHz Bustakt	1995
2.2	u. a. 3,3 V Hilfsspannung, maximale Wiederanlaufzeit, verbesserte Interrupt-Signalisierung, Tauschen von Funktionseinheiten bei anliegender Betriebsspannung (Hot Plugging)	1998/1999

*) : Problemstellen = Festlegungen des Standards, die zu Schwierigkeiten oder zu unterschiedlichen Auffassungen Anlaß gegeben haben

Tabelle 1.2 Die (bisherigen) Ausgaben (Revisions) des PCI-Standards

Hinweise:

1. Anwendungspraktische Bedeutung hat der Standard ab Ausgabe (Revision) 2.0.
2. Jeder von der PCI SIG herausgegebene Standard enthält eine Liste der vorausgegangenen Ausgaben sowie ein Verzeichnis der Änderungen zur jeweils unmittelbar vorhergehenden Ausgabe.
3. Die Änderungen heißen ECRs (Engineering Change Requests).
4. Eine bestimmte Ausgabe (z. B. Revision 2.1) wird erst dann durch eine neue ersetzt, wenn sie sozusagen "genug zusammengekommen ist". "Kleinigkeiten" werden durch eingeschobene Änderungsmitteilungen (ECRs) erledigt.
5. PCI-Standards kann jedermann beziehen. Die Pressemitteilungen, Neuigkeitsmeldungen usw. der PCI SIG geben einen gewissen Einblick in das Änderungsgeschehen. Um allerdings in den offiziellen Änderungsdienst einbezogen zu werden, muß man Mitglied der PCI SIG sein (was sich aber (Kosten) wohl nur für Hersteller einschlägiger Hardware lohnt).

Abwandlungen

Der PCI-Bus wurde im Laufe der Zeit überaus erfolgreich; er hat die Bussysteme EISA und MC A sowie den VESA-Lokalbus praktisch vollkommen verdrängt. Dieser Erfolg hat die Entwickler und Hersteller zu weitergehenden Bemühungen angeregt:

- dem PCI-Bus wurden - über die ursprüngliche Absicht hinaus - weitere Einsatzgebiete erschlossen (Stichworte: SmallPCI, PC/104-Plus, IndustrialPCI, CompactPCI),
- Interfaces für spezielle Einsatzgebiete bauen auf den PCI-Wirkprinzipien auf (Stichworte: AGP, LPC, CardBus). PCI-Wirkprinzipien kommen sogar *auf* Schaltkreisen zur Anwendung, nämlich um Standardzellen auf größeren ASICs untereinander zu verbinden.

Zu den Zukunfts-Aussichten des PCI-Bus siehe weiterhin Abschnitt 1.7.

1.1.3. PCI-typische Begriffe

Master/Initiator - Target

In der allgemeinen Fachsprache heißt die Einrichtung, die den jeweiligen Buszugriff anfordert und einleitet (in anderer Redeweise: die die "Busherrschaft" (Bus Ownership) ausübt), "Master" und die jeweils ausgewählte (adressierte) Einrichtung "Slave". In der PCI-Literatur verwendet man neben dem Begriff "Master" (bzw. "Busmaster") auch den Begriff "Initiator" und statt "Slave" den Begriff "Target".

Ressource, Einrichtung (Device), Funktion

Diese Begriffe sind nicht gleichbedeutend:

Ressource ist der Allgemeinbegriff für Hardware, die an den PCI-Bus angeschlossen ist. Man unterscheidet folgende Arten von Ressourcen: Master, Targets und Brücken (Bridges). Die zentralen Schaltmittel (Takterzeugung, Rücksetzen, Mastervermittlung, Fehlerauswertung usw.) heißen zusammengefaßt "zentrale Ressource" (Central Resource).

Eine *Einrichtung (Device)* ist physisch an den Bus angeschlossen (wie z. B. weiter unten in Abbildung 1.5 gezeigt).

Eine *Funktion* ist gekennzeichnet durch bestimmte Ausschnitte aus den Adreßräumen des PCI-Bus. In anderer Redeweise: eine Funktion ist ein "logisches Gerät" innerhalb einer "physischen" Einrichtung (daß eine Einrichtung mehrere Gerätefunktionen steuern kann, kennen wir von anderen Systemen her (SCSI, E-A-Interface der IBM-Mainframes)). Am PCI-Bus gilt:

- es gibt Einrichtungen, die nur eine Funktion enthalten (*Single Function Devices*) und solche mit mehreren Funktionen (*Multi-Function Devices*),
- eine Einrichtung kann bis zu 8 Funktionen enthalten,
- jeder Funktion ist ein bestimmter Ausschnitt aus dem Konfigurationsadreßraum zugeordnet.

Hinweis:

Im Interesse der Kürze sprechen wir im folgenden stets von "Einrichtungen" - und meinen damit sowohl *Devices* als auch *Functions*. Sind die Unterschiede zu beachten, so weisen wir jeweils an Ort und Stelle darauf hin.

PCI-Konfigurationen: 32/64 Bits, 33/66 MHz, 5/3,3 V

PCI-Systeme werden grundsätzlich nach folgenden Kennwerten unterschieden:

- Zugriffsbreite: 32 oder 64 Bits,
- Taktfrequenz: 33 oder 66 MHz,
- Signalpegel: 5 V oder 3,3 V.

Aus Tabelle 1.3 geht hervor, welche Kombinationen zulässig sind und welche nicht.

Signalpegel: 5 V	Signalpegel: 3,3 V	
<ul style="list-style-type: none"> ■ 32 Bits, 33 MHz, ■ 64 Bits, 33 MHz 	<ul style="list-style-type: none"> ■ 32 Bits, 33 MHz, ■ 64 Bits, 33 MHz 	<ul style="list-style-type: none"> ■ 32 Bits, 66 MHz, ■ 64 Bits, 66 MHz

Tabelle 1.3 Zulässige PCI-Konfigurationen

1.1.4. Mechanische Auslegung

1.1.4.1. PCI-Einrichtungen

PCI-Einrichtungen können grundsätzlich auf zweierlei Weise realisiert sein:

1. als (fest eingelötete) Schaltkreise auf dem Motherboard (“Plattform”- bzw. “planare” Implementierung),
2. als Steckkarten (Slot-Implementierung). Hierbei gibt es wiederum 2 Varianten:
 - Slots auf dem Motherboard (“Plattform”- bzw. “planare” Anordnung),
 - Slots auf Erweiterungs-Leiterplatten (Riser Cards), wie sie z. B. für LPX- und NLX - Motherboards vorgesehen sind.

Allen Ausführungsformen ist gemeinsam, daß (meistens) die eigentliche PCI-Einrichtung aus einem einzigen Schaltkreis besteht oder über einen einzigen Schaltkreis mit dem PCI-Bus gekoppelt ist.

Abbildung 1.5 PCI-Einrichtung als Schaltkreis (Quelle: PCI SIG)

Erklärung:

- 1) Adapterschaltkreis,
- 2) Steckverbinder der Steckkarte,
- 3) alle “echten” (d. h. von allen Einrichtungen gemeinsam genutzten) Busleitungen der “Grundausrüstung (32-Bit-Organisation) müssen unterhalb dieser Linie angeschlossen werden,
- 4) Signale der 64-Bit-Erweiterung,
- 5) wahlweise Anschlüsse für Testinterface (Boundary Scan),
- 6) jede Busmaster-Einrichtung hat je einen Anschluß für Busanforderung (REQ) und Busbestätigung (GNT),
- 7) jede Einrichtung hat einen IDSEL-Auswahleingang für die geographische Adressierung. Auf dem Motherboard ist dieser Anschluß mit jeweils einem der Signale AD31...11 verbunden.

a, b, c: die Maßpfeile bezeichnen zulässige Leitungslängen. Einzelheiten weiter unten in Abschnitt 1.1.8.2.

Die Abbildung zeigt den Schaltkreis auf einer Steckkarte. Auf dem Motherboard fest eingelötete PCI-Schaltkreise sehen ähnlich aus. Wichtig ist, daß es keine besonderen Treiber, Buskoppelschaltkreise o. dergl. gibt, sondern daß Treiber, Empfänger und Bussteuerschaltungen in einem einzigen Schaltkreis vereinigt sind.

Hinweis:

Die elektrischen Anforderungen der PCI-Spezifikation sind auch nur auf diese Weise zu erfüllen; eine PCI-Einrichtung läßt sich praktisch nicht “diskret” (d. h. mit Schaltkreisen geringeren Integrationsgrades (Bustreiber, Gattern, Registern usw.)) aufbauen.

Infolge der vielen Anschlüsse kommen nur bestimmte Gehäuseformen für PCI-Schaltkreise in Frage. Neben dem in Abbildung 1.5 gezeigten QFP-Gehäuse (QFP = Quad Flatpack) sind dies vor allem BGA-Gehäuse (BGA = Ball Grid Array; solche Gehäuse haben eine Vielzahl von halbkugelförmigen Lötkontakten auf ihrer Unterseite). PGA-Gehäuse (PGA = Pin Grid Array (rasterförmige Anordnung von Lötstiften; Durchsteckmontage)) werden typischerweise als zu teuer angesehen (auch sind die parasitären Kapazitäten der PGA-Gehäuse vergleichsweise hoch).

Hinweise:

1. Das Datenmaterial der einschlägigen Hersteller (Internet) enthält zahlreiche Ausführungsbeispiele.
2. Die praktische Konsequenz der bevorzugten Gehäusebauformen: PCI-Schaltkreise sind nur schwer austauschbar (ohne Sonderausrüstung: QFP notfalls (mit viel Zeit und Geschick), BGA praktisch gar nicht).

1.1.4.2. PCI-Signalpegel: 5 V und 3,3 V

Es gibt 2 Arten von PCI-Systemen: 5 V und 3,3 V. Diese Bezeichnungen erinnern an typische Logik-Speisespannungen, stehen aber tatsächlich für bestimmte Spezifikationen der Signalpegel (Einzelheiten in Abschnitt 1.1.5.3.). Tatsächlich liegen den Bezeichnungen gewisse “Vorzugslösungen” zugrunde:

- “5 V” = 5 V Speisespannung, Signalpegel gemäß TTL-Spezifikation,
- “3,3 V” = 3,3 V Speisespannung, Signalpegel gemäß CMOS-Spezifikation.

Grundsätzlich können aber 3,3-V-Einrichtungen mit 5 V Speisespannung betrieben werden (und umgekehrt). Zu Einzelheiten der Spannungsversorgung siehe Abschnitt 1.1.7.

Jede PCI-Konfiguration (zentrale Schaltmittel + angeschlossene Einrichtungen) muß für jeweils eine dieser Spezifikationen ausgelegt sein (“Mischkonfigurationen” sind unzulässig).

Wenn es sich um die Bestückung des Motherboards handelt, so ist es Sache des Herstellers, die passenden Schaltkreistypen auszuwählen. (Achtung: Beim Bestellen aufpassen, falls Sie sich doch an das Auswechseln von PCI-Schaltkreisen heranwagen!)

Steckkarten sind durch mechanische Verriegelung (genauer: durch Sperren im Slot-Steckverbinder und Kerben in der Leiterplatte) gegen fälschliches Stecken abgesichert (Abbildung 1.6).

Abbildung 1.6 PCI-Steckkarten und Signalpegel*Erklärung:*

- a) 5-V-Karten passen nur in 5-V-Slots (Kerbe/Sperre vorn¹⁾),
- b) 3,3-V-Karten passen nur in 3,3-V-Slots (Kerbe/Sperre hinten²⁾),
- c) Universalkarten (Dual Voltage Signaling Boards) passen in beide Arten von Slots (Kerben vorn und hinten). Sie stellen sich automatisch auf die jeweilige Umgebung (5 V oder 3,3 V) ein.

*) kennzeichnet Rückseite des PC (Anschlüsse für Interfacekabel),

- 1) "vorn" = der PC-Rückseite bzw. dem Slot-Abdeckblech (Bracket) der Karte abgewandt,
- 2) "hinten" = der PC-Rückseite bzw. dem Slot-Abdeckblech (Bracket) der Karte zugewandt (vgl. weiter unten Abbildung 1.12).

Hinweis zum Fach-Englisch:

Sammelbegriff für mechanische Maßnahmen gegen Falsch-Stecken: *Keying*. Die Kerben in der Leiterplatte heißen *Keyways*, die Sperren im Steckverbinder *Spacer*.

1.1.4.3. Steckverbinder

Die Steckverbinder-Technologie stammt vom MCA-Bussystem (Microchannel). Es handelt sich um direkte Leiterplatten-Steckverbinder mit einem Anschlußabstand (Pin Pitch) von 1,27 mm (0,05"). Wichtige technische Anforderungen an solche Steckverbinder sind aus Tabelle 1.4 ersichtlich.

Kennwert	Forderung
Kontaktmaterial	Phosphorbronze. Oberfläche mindestens 0,76 µm (0,03 mil) Gold über 1,27 µm (0,05 mil) Nickel oder Hauchvergoldung (Gold Flash) über 1 µm (0,04 mil) Palladium oder Palladium-Nickel
Lebensdauer	wenigstens 100 Steckzyklen ^{*)}
Steckkraft	1,7 N (180 g) je Kontaktpaar ^{*)}
Kontaktkraft	wenigstens 75 g
Kontaktwiderstand	anfänglich höchstens 30 mΩ, im Betrieb Erhöhung um maximal 10 mΩ
Isolationswiderstand	wenigstens 1000 MΩ
Kontaktkapazität	höchstens 2 pF bei 1 MHz
Strombelastbarkeit	1 A
Spannungsfestigkeit	125 V

*) : zum fachmännischen Stecken siehe weiter unten Abbildung 1.16

Tabelle 1.4 Anforderungen an PCI-Steckverbinder (nach: PCI SIG)

Es gibt 3 verschiedene Ausführungen (Tabelle 1.5). Zur Kontaktbelegung siehe weiter unten Tabelle 1.9.

Hinweis:

Die Ausgabe 2.2 der PCI-Spezifikation sieht weitere Steckverbinder vor, die zum Bestücken von Erweiterungskarten (Riser Cards) vorgesehen sind (Abbildung 1.9). Merkmale:

- etwa 2 mm höher,
- andere Lage der Fixierstifte (zwecks Vermeiden von Fehlbestückungen),
- ansonsten gleiche Ausführung.

Steckverbindertyp	Kontaktanzahl	Kontakte in jeder der Reihen A, B	Abbildung
32-Bit-Bus	120	1...62	1.7
64-Bit-Bus, 5 V	184	1...94	1.8a
64-Bit-Bus 3,3 V	184	1...94	1.8b

Tabelle 1.5 PCI-Steckverbindertypen

Abbildung 1.7 Der Slot-Steckverbinder des 32-Bit-PCI-Bus. R = PC-Rückseite (zur Orientierung); F = Fixierstifte; S = Sperre (Key). Oben: Lage in 5-V-System, unten (nur Draufsicht): Lage in 3,3-V-System. Unmaßstäblich

Erklärung:

Es gibt nur einen Steckverbinder für 5-V- und für 3,3-V-Systeme. Die Zweiseitenansicht entspricht der Bestückung eines 5-V-Systems. In 3,3-V-Konfigurationen wird der Steckverbinder um 180° gedreht bestückt.

Abbildung 1.8 Der Slot-Steckverbinder des 64-Bit-PCI-Bus. R = PC-Rückseite (zur Orientierung); F, F_E = Fixierstifte; S = Sperre (Key); E = zusätzliche Kontakte für 64-Bit-Erweiterung. Unmaßstäblich

Erklärung:

Der 120-polige Steckverbinder des 32-Bit-Bus ist um 64 Kontakte erweitert. Infolge dessen ist ein einfaches Wenden nicht mehr möglich. Vielmehr müssen 2 Ausführungen gefertigt werden:

- a) 5-V-Ausführung. Lage der Sperre S und der Fixierstifte F wie in Abbildung 1.7 oben.
- b) 3,3-V-Ausführung. Lage der Sperre S wie in Abbildung 1.7 unten. Fixierstifte F sinngemäß versetzt.

Abbildung 1.9 Steckverbinder zum Bestücken von Erweiterungskarten (Riser Cards). R = PC-Rückseite (zur Orientierung); F, F_E, F_R = Fixierstifte. Unmaßstäblich

Erklärung:

- a) Steckverbinder für 32-Bit-Bus
- b) Steckverbinder für 64-Bit-Bus, 5 V,
- c) Steckverbinder für 64-Bit-Bus, 3,3 V.

Beachten Sie (1) die Höhe von rund 17 mm und (2) die veränderte Lage einiger Fixierstifte (F_R) gegenüber der Darstellung in den Abbildungen 1.7 und 1.8.

Abbildung 1.10 Anschlußanordnung (1). a) 32 Bits, 5 V, b) 32 Bits, 3,3 V, c) 64 Bits, 5 V, d) 64 Bits, 3,3 V.
R = PC-Rückseite (zur Orientierung); F, F_E = Fixierbohrungen; S = Sperre.
Ansicht von oben (Bestückungsseite). Unmaßstäblich

Erklärung:

Die Abbildung zeigt die Orientierung und Zählweise der Anschlüsse anhand der Steckverbinder-Bohrungen auf dem Motherboard. Der Steckverbinder wird ausschließlich über die Lötverbindungen der Kontakt-Anschlüsse gehalten. (PCI-Leiterplatten sind grundsätzlich Mehrebenen-Leiterplatten, und die Bohrungen sind durchkontaktiert. Die 160 bzw. 184 Lötverbindungen sind somit in der Lage, die mechanischen Kräfte (vor allem beim Herausziehen der Steckkarte) aufzunehmen.) In die Fixierbohrungen F, F_E greifen die jeweiligen Fixierstifte (vgl. die Abbildungen 1.7...1.9) ein (keine Lötverbindungen - die Stifte bestehen aus dem Plastik-Material des Steckverbinder-Gehäuses).

Abbildung 1.11 Anschlußanordnung (2). Zur Zählweise. R = PC-Rückseite (zur Orientierung)

Erklärung:

- a) Ansicht von oben (Bestückungsseite; vereinfachte Darstellung entsprechend Abbildung 1.10),
- b) Ansicht von unten (Lötseite),
- c) Ansicht von oben (Bestückungsseite): Merkhilfe,
- d) Ansicht von unten (Lötseite): Merkhilfe,
- e) Draufsicht auf Slot-Steckverbinder (Merkhilfe),
- f) Draufsicht auf Steckkarte (Merkhilfe).

Merkhilfe

Die Abbildung soll das Zählschema veranschaulichen (Anwendung: falls doch einmal am PCI-Bus gemessen werden soll):

- Au = Reihe A¹⁾, ungerade Kontaktnummern (1, 3...),
- Ag = Reihe A¹⁾, gerade Kontaktnummern (2, 4...),
- Bu = Reihe B¹⁾, ungerade Kontaktnummern (1, 3...),
- Bg = Reihe B¹⁾, gerade Kontaktnummern (2, 4...),
- Zählrichtung: von der Rückseite aus nach vorn,
- in Steckverbindern²⁾ und auf Karten³⁾ wird, von hinten beginnend, in jeder Reihe fortlaufend gezählt (1, 2, 3...).

- 1) am Steckverbinder auf Lötseite des Motherboards,
- 2) beim Messen von oben im leeren Slot,
- 3) Zählen auf Karten:
 - A-Reihe = Lötseite,
 - B-Reihe = Bestückungsseite.

Hinweis:

Die Sperre (S) bzw. die entsprechende Kerbe in der Steckkarte unterbricht nicht die Kontaktzählung. Wo sich im Steckverbinder die Sperre befindet (vgl. Abbildung 1.10), fehlt in jeder der 4 Bohrungsreihen 1 Kontakt. Diese Kontaktpositionen sind aber jeweils mitzuzählen! Deshalb haben wir zwar nur 160 Kontakte, aber 164 Kontaktpositionen. Demgegenüber beginnt die Kontaktzählung der 64-Bit - Erweiterung mit den ersten Kontaktpositionen der Erweiterung (ungeachtet des Abstandes zwischen den Kontaktpositionen 61/62 und 63/64). Vgl. auch weiter unten Tabelle 1.6.

1.1.4.4. Steckkarten

Abmessungen

PCI-Steckkarten sind so spezifiziert, daß sie zusammen mit ISA- bzw.- EISA- oder MCA-Karten eingesetzt werden können. Die Abbildungen 1.12 und 1.13 geben einen Überblick über das Aussehen solcher Karten sowie über einige Grundmaße.

Hinweise:

1. Im Service müssen wir in der Lage sein, vorhandene Karten zutreffend zu erkennen. Hierzu genügen einige wenige Maßangaben.
2. Heutzutage wird überall gespart. Deshalb wendet man für Steckkarten oftmals nur soviel Leiterplattenmaterial auf, wie unbedingt notwendig. Wir finden deshalb oftmals kürzere und weniger hohe Karten vor.
3. Der PCI-Standard sieht weiterhin Haltebügel (Retainer) am anderen Ende der Karte vor. Auch diese werden üblicherweise weggelassen (weshalb auch wir nicht näher darauf eingehen - als Anregung zu Behelfszwecken siehe die Abbildungen 4.14 und 4.15 im Fehlersuchhandbuch).

Abbildung 1.12 Steckkarten für ISA/EISA-Systeme. Abmessungen in mm. Unmaßstäblich (nach: PCI SIG)

Erklärung:

- a) Draufsicht,
 - b) 5-V-Karte,
 - c) 3,3-V-Karte,
 - L Lötseite (zulässige Bauelementehöhe: 2,67 mm),
 - B Bestückungsseite (zulässige Bauelementehöhe: 14,48 mm),
 - SA Slot-Abdeckblech (Bracket), trägt ggf. die externen E-A-Anschlüsse,
 - KK kurze Karte (Richtwert für Länge),
 - LK lange Karte (größte zulässige Länge),
 - K₅ Kerbe für 5-V-Karten^{*)},
 - K₃ Kerbe für 3,3-V-Karten^{*)},
 - 1) Bezugskante für Bemaßung = Außenseite Abdeckblech,
 - 2) in Ausgabe 2.2 der PCI-Spezifikation Toleranz auf $\pm 0,25$ mm verringert.
- *) : Universalkarten haben beide Kerben.

Abbildung 1.13 Steckkarten für MCA-Systeme. Abmessungen in mm. Unmaßstäblich (nach: PCI SIG)

Erklärung:

- a) 5-V-Karte,
- b) 3,3-V-Karte.

Die weiteren Bezugszeichen entsprechen Abbildung 1.12.

Karten umbauen

PCI-Karten für ISA/EISA- und für MCA-Systeme unterscheiden sich typischerweise nur im Slot-Abdeckblech (Bracket). Damit ist es nicht aussichtslos, Karten umzubauen (Abdeckblech tauschen bzw. - als Behelf - einfach entfernen).

Geteilte Steckpositionen (Shared Slots)

Viele Motherboards haben eine "geteilte" Steckposition, die nebeneinander mit einem herkömmlichen (ISA, EISA oder MCA) und mit einem PCI-Steckverbinder belegt ist. Es kann stets nur eine Karte gesteckt werden. Damit dies funktioniert, werden PCI-Karten genau "anders herum" bestückt als herkömmliche (Legacy) Karten (Abbildung 1.14).

Abbildung 1.14 Geteilte Steckposition (Ausführungsbeispiel). Draufsicht

Erklärung:

- R PC-Rückseite,
- P PCI-Steckverbinder,
- I ISA-Steckverbinder,
- SA Slot-Abdeckblech (Bracket),
- B_p Bestückungsseite der PCI-Karte,
- B_i Bestückungsseite der ISA-Karte.

Rückseitige E-A-Anschlüsse

E-A-Steckverbinder werden am Slot-Abdeckblech montiert. Abbildung 1.15 veranschaulicht die Grundmaße.

Abbildung 1.15 Slot-Abdeckblech: Grundmaße für E-A-Anschlüsse (nach: PCI SIG). a) herkömmliche, b) moderne Auslegung

Erklärung:

Es ist ein "Fenster" definiert, in dem die E-A-Steckverbinder befestigt werden dürfen (I/O Window). Wesentlich ist das Höhenmaß H:

- herkömmliche Festlegung: H = 81,9 mm,
- moderne Festlegung (Ausgabe 2.2 der PCI-Spezifikation): H = 88,9 mm.

Die schraffierten Flächen (K) kennzeichnen die zulässige Außen-Kontur der Steckverbinder (diese dürfen oben 10 mm und unten 20 mm über das Abdeckblech hinausragen).

Karten richtig stecken und ziehen

Sowohl die Steckkräfte (vgl. Tabelle 1.4) als auch die Konstruktion der typischen PC-Gehäuse erlauben es nicht, die Karte vollkommen parallel zum Slot-Steckverbinder zu stecken bzw. herauszuziehen. Deshalb nach Abbildung 1.16 vorgehen:

- Stecken: Karte von der Rückseite an schräg ansetzen und in den Slot drücken,
- Ziehen: Karte zuerst vorn anheben (so daß sie zunächst mit den vorderen Kontakten frei - kommt)

Abbildung 1.16 Stecken und Ziehen einer PCI-Karte (nach: PCI SIG). 1 - Motherboard; 2 - Slot-Steckverbinder; 3 - Karte

1.1.5. Bussignale

1.1.5.1. Übersicht

Im folgenden geben wir zunächst einen Überblick darüber, welche Signale zum PCI-Bus gehören (Abbildungen 1.17, 1.18). In den folgenden Abschnitten 1.2 und 1.3 werden diese Signale zunächst einzeln und anschließend in ihrem funktionellen Zusammenwirken beschrieben.

Abbildung 1.17 PCI-Bussignale (1). Der 32-Bit-Bus

Erklärung:

- a) die unbedingt erforderlichen (required) Bussignale einer fest eingebauten ("planaren") Einrichtung, die nur als Target wirksam ist (Target-Only),
- b) soll eine Einrichtung auch als Master wirksam werden, kommen diese zwei Signale hinzu,
- c) die Signale der Fehlersignalisierung sind bei fest eingebauten ("planaren") Einrichtungen wahlfrei (optional), bei Slots aber erforderlich (required),
- d) diese Signale sind wahlfrei (optional),
- e) PCI-Einrichtungen können wahlweise mit einem Boundary-Scan-Testinterface ausgerüstet werden,
- f) diese Signale sind nur für Slots spezifiziert. Sie gehören nicht zum "eigentlichen" PCI-Bus.

*) *Achtung:* auch wenn auf die Fehlersignalisierung verzichtet wird - das Erzeugen des bzw. der Paritätsbits (beim Belegen der AD- und die C/BE-Leitungen) ist unbedingt erforderlich (required; vgl. das Signal PAR in Abbildung 1.17a).

Abbildung 1.18 PCI-Bussignale (2). Erweiterungen

Erklärung:

- a) zusätzliches Erlaubnissignal für 66-MHz-Betrieb (2.1¹),
- b) zusätzliches Taktsteuersignal, vorzugsweise für "mobile" Systeme (2.1¹)²),
- c) zusätzliches Signal zum Signalisieren von Stromspar-Ereignissen (Power Management Events; 2.2¹),

d) Signale der 64-Bit-Erweiterung.

- 1): bezeichnet die jeweilige Ausgabe (Revision) der PCI-Spezifikation),
- 2): das Signal ist nur für "planare" Einrichtungen bestimmt und nicht im Slot-Steckverbinder vorgesehen.

Hinweis:

Die Abbildungen 1.17, 1.18 veranschaulichen die PCI-Signale aus der Sicht der einzelnen Einrichtung am PCI-Bus. Nicht alle Signale sind aber "echte" Busleitungen (die von allen Einrichtungen gleichsam angezapft werden). Ein PCI-System ist vielmehr - auf Grund der zentralen Steuerung - ein "Gemisch" von Bus- und Einzelsignalen. Näheres in Abschnitt 1.2.

Erforderliche und wahlfreie bzw. zusätzliche Signale

Die - was den PCI-Anschluß angeht - einfachsten Einrichtungen (auf dem Motherboard, nur Targets) benötigen nur 45 Bussignal-Anschlüsse (Einrichtungen mit Busmaster-Funktionalität: 47 Anschlüsse).

Bei *fest eingebauten* ("planaren") Einrichtungen ist die Nutzung der zusätzlichen Bussignale freigestellt.

Slots müssen mit allen Signalen gemäß der jeweiligen Ausgabe des PCI-Standards belegt sein (gleichgültig, ob sie von den jeweiligen Steckkarten ausgenutzt werden oder nicht).

Auf *Steckkarten* ist die Nutzung der Signale gemäß Abbildung 1.17d, e sowie gemäß Abbildung 1.18 freigestellt.

Nähere Einzelheiten in Abschnitt 1.2.

1.1.5.2. Signalpegel und -flanken

Hinsichtlich der elektrischen Signalkennwerte unterscheidet sich der PCI-Bus in charakteristischer Weise von anderen Bussystemen:

- er ist als reiner CMOS-Bus ausgelegt,
- die Signalreflexion auf den Busleitungen wird ausgenutzt (die jeweils erste rücklaufende Wellenfront gehört mit zum Schaltvorgang; Reflected Wave Switching),
- damit dies funktioniert, sind die Busleitungen grundsätzlich *nicht* abgeschlossen (unterminated),
- die Treibfähigkeit der Buskoppelstufen im statischen Betrieb muß nicht besonders hoch sein,
- höhere Treiberströme werden praktisch nur beim Umschalten benötigt (zum Umladen der parasitären Kapazitäten). Hierfür hat man die Anforderungen an die "dynamische" Treibfähigkeit in besonderer Weise spezifiziert.

Wir brauchen nicht alle Einzelheiten, sollten uns aber eine gewisse Vorstellung von der elektrischen Auslegung des PCI-Bus bilden können. Die hierfür wesentlichen Kennwerte sind in den Tabellen 1.6 bis 1.8 zusammengefaßt.

Kennwert	Symbol	Minimalwert	Maximalwert
Speisespannung	V_{CC}	4,75 V	5,25 V
Low-Eingangsspannung	V_{il}	- 0,5 V	0,8 V
High-Eingangsspannung	V_{ih}	2,0 V	$V_{CC} + 0,5 V$
Low-Ausgangsspannung	V_{ol}		0,55 V ¹⁾
High-Ausgangsspannung	V_{oh}	2,4 V ²⁾	
Low-Ausgangsstrom ³⁾	I_{ol}	3 oder 6 mA ⁴⁾	
High-Ausgangsstrom ³⁾	I_{oh}	- 2 mA	
Eingangs-Leckstrom ⁵⁾	I_{il}, I_{ih}		$\pm 70 \mu A$ ⁶⁾
Schaltstrom, Übergang Low → High ⁷⁾	$I_{oh(AC)}$	<ul style="list-style-type: none"> ■ bei $0 V < V_{out} \leq 1,4 V$: - 44 mA, ■ bei $V_{out} = 3,1 V$: - 142 mA⁸⁾ 	
Schaltstrom, Übergang High → Low ⁶⁾	$I_{ol(AC)}$	<ul style="list-style-type: none"> ■ bei $V_{out} \geq 2,2 V$: 95 mA, ■ bei $V_{out} = 0,71 V$: 206 mA⁸⁾ 	
Flankensteilheit ⁹⁾	$slew_r,$ $slew_f$	1 V/ns	5 V/ns

1)...9): siehe Erklärung im Text

Tabelle 1.6 Signalkennwerte in 5-V-Systemen

Erklärung:

“5-V-Signalisierung” heißt praktisch: Eingangsverhalten wie TTL (Mindest-High-Pegel: 2,0 V), Ausgangsverhalten wie 5-V-CMOS (Schaltverhalten “rail to rail”).

- 1) bei statischem Ausgangsstrom $I_{ol} = 3$ oder 6 mA,
- 2) bei statischem Ausgangsstrom $I_{oh} = - 2$ mA,
- 3) statischer Ausgangsstrom; Wert kennzeichnet erforderliche Treibfähigkeit,
- 4) 3 mA Treibfähigkeit für Signale ohne und 6 mA Treibfähigkeit für Signale mit Pull-up-Widerständen (vgl. Abschnitt 1.2.1.2.),
- 5) gilt auch für die Leckströme von Tri-State-Ausgängen in hochohmigem Zustand,
- 6) positives Vorzeichen bei High-Pegel (I_{ih}), negatives Vorzeichen bei Low-Pegel (I_{il}),
- 7) zu Orientierungszwecken ausgewählte Werte; Näheres siehe Abbildung 1.19 (anschließend sind sämtliche Gleichungen angegeben, die Verlauf und Größenordnung der Schaltströme beschreiben),
- 8) Wert am “Testpunkt” (Abbildung 1.19),
- 9) betrifft sowohl die ansteigende als auch die abfallende Flanke. Es ist auch eine *maximale* Flankensteilheit definiert, um Störungen in Grenzen zu halten. Für Open-Drain-Ausgänge ist die Steilheit der Low-High-Flanke ($slew_r$) *nicht* spezifiziert.

Hinweis:

Die Kennwerte sind so festgelegt, daß 5-V-Hardware auch mit anderen Technologien als CMOS realisiert werden kann (d. h. praktisch: auf bipolarer (TTL-) oder BiMOS-Grundlage).

Kennwert	Symbol	Minimalwert	Maximalwert
Speisespannung	V_{CC}	3,0 V	3,6 V
Low-Eingangsspannung	V_{il}	- 0,5 V	$0,3 V_{CC}$ ($1 V^1$)
High-Eingangsspannung	V_{ih}	$0,5 V_{CC}$ ($1,65 V^1$)	$V_{CC} + 0,5 V$
Low-Ausgangsspannung	V_{ol}		$0,1 V_{CC}^2$ ($0,33V^1$)
High-Ausgangsspannung	V_{oh}	$0,9 V_{CC}^3$ ($3V^1$)	
Low-Ausgangsstrom ⁴⁾	I_{ol}	1,5 mA	
High-Ausgangsstrom ⁴⁾	I_{oh}	- 0,5 mA	
Pull-up-Spannung ⁵⁾	V_{ipu}	$0,7 V_{CC}$ ($2,3V^1$)	
Eingangs-Leckstrom ⁶⁾	I_{il}, I_{ih}		$\pm 10 \mu A^7$
Schaltstrom, Übergang Low \rightarrow High ⁸⁾	$I_{oh(AC)}$	<ul style="list-style-type: none"> ■ bei $0 V < V_{out} \leq 0,3 V_{CC}$: - $12 V_{CC} mA$ ($- 40 mA^1$), ■ bei $V_{out} = 0,7 V_{CC}$: - $32 V_{CC} mA^9$ ($- 106 mA^1$) 	
Schaltstrom, Übergang High \rightarrow Low ⁸⁾	$I_{ol(AC)}$	<ul style="list-style-type: none"> ■ bei $V_{CC} > V_{out} \geq 0,6 V_{CC}$: $16 V_{CC} mA$ ($53 mA^1$), ■ bei $V_{out} = 0,18 V_{CC}$: $38 V_{CC} mA^9$ ($125 mA^1$) 	
Flankensteilheit ¹⁰⁾	$slew_r,$ $slew_f$	1 V/ns	4 V/ns

1)...10): siehe Erklärung im Text

Tabelle 1.7 Signalkennwerte in 3,3-V-Systemen

Erklärung:

“3,3-V-Signalisierung” heißt praktisch: Ein- und Ausgangsverhalten “typisch CMOS”, wobei die Logikpegel auf die aktuelle Speisespannung bezogen werden (V_{CC} -relative Definition); (Mindest-High-Pegel am Eingang $0,5 V_{CC}$, ausgangsseitiges Schaltverhalten “rail to rail”).

- 1) Wert für $V_{CC} = 3,3 V$,
- 2) bei statischem Ausgangsstrom $I_{ol} = 1,5 mA$,
- 3) bei statischem Ausgangsstrom $I_{oh} = - 0,5 mA$,
- 4) statischer Ausgangsstrom; Wert kennzeichnet erforderliche Treibfähigkeit,
- 5) Mindest-High-Pegel, auf welchen das Signal durch den Pull-up-Widerstand gezogen werden muß,
- 6) gilt auch für die Leckströme von Tri-State-Ausgängen in hochohmigem Zustand,
- 7) positives Vorzeichen bei High-Pegel (I_{ih}), negatives Vorzeichen bei Low-Pegel (I_{il}),
- 8) zu Orientierungszwecken ausgewählte Werte; Näheres siehe Abbildung 1.19 (anschließend sind sämtliche Gleichungen angegeben, die Verlauf und Größenordnung der Schaltströme beschreiben),
- 9) Wert am “Testpunkt” (Abbildung 1.19),

- 10) betrifft sowohl die ansteigende als auch die abfallende Flanke. Es ist auch eine *maximale* Flankensteilheit definiert, um Störungen in Grenzen zu halten. Für Open-Drain-Ausgänge ist die Steilheit der Low-High-Flanke ($slew_r$) *nicht* spezifiziert.

Hinweis: 3,3-V-Hardware ist ausschließlich zur Realisierung in CMOS vorgesehen.

Kapazität	Symbol	Minimalwert	Maximalwert
Anschluß, allgemein	C_{in}		10 pF (“planare” Einrichtungen: 16 pF) ^{*)}
Takteingang (CLK)	C_{clk}	5 pF	12 pF
Eingang IDSEL	C_{IDSEL}		8 pF

*) : siehe Erklärung im Text

Tabelle 1.8 Parasitäre Kapazitäten

Erklärung:

Für Einrichtungen auf dem Motherboard wurden 16 pF zugelassen, um dort auch PGA-Gehäuse einsetzen zu können. Ansonsten lassen sich die Anforderungen nur durch Gehäuse mit besonders geringen parasitären Kapazitäten erfüllen (QFP, BGA usw.).

Schaltströme

Die Schaltströme bzw. die Anforderung an die Treibfähigkeit der PCI-Bustreiberschaltungen werden in Form von Strom-Spannungs-Kennlinien und Gleichungen spezifiziert (Abbildung 1.19).

Abbildung 1.19 Strom-Spannungs-Kennlinien der Schaltströme (Quelle: PCI SIG)

Erklärung:

- 5-V-Signalisierung; Schalten von Low nach High,
 - 5-V-Signalisierung; Schalten von High nach Low,
 - 3,3-V-Signalisierung; Schalten von Low nach High,
 - 3,3-V-Signalisierung; Schalten von High nach Low.
- *DC drive point:* kennzeichnet statischen Betriebszustand,
 - *AC drive point:* kennzeichnet erforderlichen Treiberstrom, um den Bus mit einer einzigen reflektierten Welle umzuschalten. Dieser Arbeitspunkt muß innerhalb der spezifizierten Ausgangsverzögerungszeit t_{VAL} (vgl. Abschnitt 1.2.2.4.) erreicht werden.
 - *Test point:* kennzeichnet den maximal zulässigen Stromfluß. Der “Testpunkt” entspricht näherungsweise einem Strom, der durch einen 22- Ω -Widerstand zu treiben ist. Dieser Wert entspricht wiederum näherungsweise dem Wellenwiderstand einer Leitung eines mit 10 “Buslasten” (Abschnitt 1.1.8.1.) voll ausgenutzten PCI-Bussystems (*Bestückungsbeispiel 1:* 6 Einrichtungen auf dem Motherboard, 2 auf Karten in Slots; *Bestückungsbeispiel 2:* 4 Einrichtungen auf dem Motherboard, 3 auf Karten in Slots).

Mit Ausnahme des statischen Arbeitspunktes (DC drive point) wird die Kurve nur während der Schaltvorgänge durchlaufen (d. h. typischerweise in wenigen ns). Zum Schalten der Signale siehe Abschnitt 1.2.2.4.

Es folgen die Gleichungen, die die Schaltströme in den einzelnen Abschnitten der Kennlinien beschreiben:

5-V-Signalisierung, Schaltstrom von Low nach High:

$$0 < V_{out} : -44 \text{ mA}$$

$$3,1 \text{ V} < V_{out} < V_{CC} : 11,9 \cdot (V_{out} - 5,25) \cdot (V_{out} + 2,45) \text{ mA}$$

$$1,4 \text{ V} < V_{out} < 2,4 \text{ V} : -44 + \frac{V_{out}}{0,024} \text{ mA}$$

5-V-Signalisierung, Schaltstrom von High nach Low:

$$V_{out} \geq 2,2 \text{ V} : 95 \text{ mA}$$

$$2,2 \text{ V} > V_{out} > 0,55 \text{ V} : \frac{V_{out}}{0,023} \text{ mA}$$

$$0,71 \text{ V} > V_{out} > 0 \text{ V} : 78,5 \cdot V_{out} \cdot (4,4 - V_{out}) \text{ mA}$$

3,3-V-Signalisierung, Schaltstrom von Low nach High:

$$0 \text{ V} < V_{out} < 0,3 V_{CC} : -12 V_{CC} \text{ mA}$$

$$0,3 V_{CC} < V_{out} < 0,9 V_{CC} : -17,1 \cdot (V_{CC} - V_{out}) \text{ mA}$$

$$0,7 V_{CC} < V_{out} < V_{CC} : \frac{98,0}{V_{CC}} \cdot (V_{out} - V_{CC}) \cdot (V_{out} + 0,4 V_{CC}) \text{ mA}$$

3,3-V-Signalisierung, Schaltstrom von High nach Low:

$$V_{CC} > V_{out} > 0,6 V_{CC} : 16 V_{CC} \text{ mA}$$

$$0,6 V_{CC} > V_{out} > 0,1 V_{CC} : 26,7 V_{out} \text{ mA}$$

$$0,18 V_{CC} > V_{out} > 0 \text{ V} : \frac{256}{V_{out}} \cdot V_{out} \cdot (V_{CC} - V_{out}) \text{ mA}$$

1.1.5.3. Schrittweise Signalaufschaltung: Address/Data Stepping

Die Anforderungen an das dynamische Schaltverhalten der PCI-Bustreiber sind beachtlich. Bedenken Sie, daß viele Signale gleichzeitig schalten müssen - und daß dies mit Nebenwirkungen verbunden ist (Übersprechen, Anhebung des Massepotentials (Ground Bounce)). Die PCI-Spezifikation erlaubt deshalb ein schrittweises bzw. zeitverzögertes Aufschalten, das mehrere Taktzyklen dauern darf. Die Alternativen:

- *Discrete Stepping*: in jeder der betreffenden Taktzyklen wird nur ein Teil der zu treibenden Signale aufgeschaltet,
- *Continuous Stepping*: es wird mit verminderter Anstiegszeit aufgeschaltet; die Signalflanken werden gleichsam über mehrere Taktzyklen gestreckt.

Auf dieser Grundlage kann man kostengünstigere Schaltkreise bauen (schwächere Treiberstufen, weniger Masse-Anschlüsse usw.) - allerdings zu Lasten der Datenraten und Latenzzeiten.

Das schrittweise Aufschalten ist für folgende Signale zulässig: AD31...0, AD63...32, PAR, PAR64, IDSEL.

Es handelt sich hierbei um Signale, deren Gültigkeit stets von weiteren Steuersignalen bestimmt wird (so werden die AD-Signale durch FRAME# (Adressierung) bzw. IRDY# und TRDY# (Datenübertragung) als "gültig" gekennzeichnet. Diese Steuersignale werden jeweils von derselben Einrichtung aktiviert, die auch die schrittweise aufzuschaltenden Signale treibt. Es ist also nicht besonders schwierig, das Erregen des jeweiligen Steuersignals bis zum Abschluß des schrittweisen Aufschaltens zu verzögern.

Hinweise:

1. Alle PCI-Einrichtungen müssen als Empfänger von Bussignalen die schrittweise Aufschaltung unterstützen (d. h., auf solche Signale korrekt reagieren).
2. Beim Aufschalten mit verminderter Anstiegszeit (*Continuous Stepping*) kann es vorkommen, daß Low-High-Taktflanken auf Signale treffen, die nicht stabil auf Low oder High liegen. Dies ist ein typischer Betriebsfall, der das Auftreten metastabiler Zustände begünstigt. Die PCI-Einrichtungen müssen auch dies aushalten.

1.1.6. Anschlußbelegung

Die Anschlußbelegung der PCI-Slots ist in Tabelle 1.9 dargestellt.

Anschluß	5-V-System		3,3-V-System	
	Kontaktreihe B	Kontaktreihe A	Kontaktreihe B	Kontaktreihe A
1	- 12 V	TRST#	- 12 V	TRST#
2	TCK	+12 V	TCK	+12 V
3	GND	TMS	GND	TMS
4	TDO	TDI	TDO	TDI
5	+ 5 V	+ 5 V	+ 5 V	+ 5 V
6	+ 5 V	INTA#	+ 5 V	INTA#
7	INTB#	INTC#	INTB#	INTC#
8	INTD#	+ 5 V	INTD#	+ 5 V
9	PRSNT1#	reserviert	PRSNT1#	reserviert
10	reserviert	+ 5 V (E-A)	reserviert	+ 3,3 V (E-A)
11	PRSNT2	reserviert	PRSNT2	reserviert
12	GND ³⁾	GND ³⁾	3,3-V-Kennung: Sperre in Slot, Kerbe in Karte	
13	GND ³⁾	GND ³⁾		
14	reserviert	res./3,3Vaux ²⁾	reserviert	res./3,3Vaux ²⁾
15	GND	RST#	GND	RST#
16	CLK	+ 5 V (E-A)	CLK	+ 3,3 V (E-A)
17	GND	GNT#	GND	GNT#
18	REQ#	GND	REQ#	GND
19	+ 5 V (E-A)	res./PME# ²⁾	+ 3,3 V (E-A)	res./PME# ²⁾
20	AD31	AD30	AD31	AD30
21	AD29	+ 3,3 V	AD29	+ 3,3 V
22	GND	AD28	GND	AD28
23	AD27	AD26	AD27	AD26
24	AD25	GND	AD25	GND
25	+ 3,3 V	AD24	+ 3,3 V	AD24
26	C/BE3#	IDSEL	C/BE3#	IDSEL
27	AD23	+ 3,3 V	AD23	+ 3,3 V
28	GND	AD22	GND	AD22
29	AD21	AD20	AD21	AD20
30	AD19	GND	AD19	GND

Anschluß	5-V-System		3,3-V-System	
	Kontaktreihe B	Kontaktreihe A	Kontaktreihe B	Kontaktreihe A
31	+ 3,3 V	AD18	+ 3,3 V	AD18
32	AD17	AD16	AD17	AD16
33	C/BE2#	+ 3,3 V	C/BE2#	+ 3,3 V
34	GND	FRAME#	GND	FRAME#
35	IRDY#	GND	IRDY#	GND
36	+ 3,3 V	TRDY#	+ 3,3 V	TRDY#
37	DEVSEL#	GND	DEVSEL#	GND
38	GND	STOP#	GND	STOP#
39	LOCK#	+ 3,3 V	LOCK#	+ 3,3 V
40	PERR#	SDONE	PERR#	SDONE
41	+ 3,3 V	SBO#	+ 3,3 V	SBO#
42	SERR#	GND	SERR#	GND
43	+ 3,3 V	PAR	+ 3,3 V	PAR
44	C/BE1#	AD15	C/BE1#	AD15
45	AD14	+ 3,3 V	AD14	+ 3,3 V
46	GND	AD13	GND	AD13
47	AD12	AD11	AD12	AD11
48	AD10	GND	AD10	GND
49	GND	AD09	GND/M66EN ¹⁾	AD09
50	5-V-Kennung: Sperre in Slot, Kerbe in Karte		GND ³⁾	GND ³⁾
51			GND ³⁾	GND ³⁾
52	AD08	C/BE0#	AD08	C/BE0#
53	AD07	+ 3,3 V	AD07	+ 3,3 V
54	+ 3,3 V	AD06	+ 3,3 V	AD06
55	AD05	AD04	AD05	AD04
56	AD03	GND	AD03	GND
57	GND	AD02	GND	AD02
58	AD01	AD00	AD01	AD00
59	+ 5 V (E-A)	+ 5 V (E-A)	+ 3,3 V (E-A)	+ 3,3 V (E-A)
60	ACK64#	REQ64	ACK64#	REQ64
61	+ 5 V	+ 5 V	+ 5 V	+ 5 V
62	+ 5 V	+ 5 V	+ 5 V	+ 5 V

Anschluß	5-V-System		3,3-V-System	
	Kontaktreihe B	Kontaktreihe A	Kontaktreihe B	Kontaktreihe A
	<i>- Ende des 32-Bit-Steckverbinders -</i> <i>- Sperre in 64-Bit-Slot, Kerbe in 64-Bit-Karte -</i> <i>- Beginn der 64-Bit-Erweiterung -</i>			
63	reserviert	GND	reserviert	GND
64	GND	C/BE7#	GND	C/BE7#
65	C/BE6#	C/BE5#	C/BE6#	C/BE5#
66	C/BE4#	+ 5 V (E-A)	C/BE4#	+ 3,3 V (E-A)
67	GND	PAR64	GND	PAR64
68	AD63	AD62	AD63	AD62
69	AD61	GND	AD61	GND
70	+ 5 V (E-A)	AD60	+ 3,3 V (E-A)	AD60
71	AD59	AD58	AD59	AD58
72	AD57	GND	AD57	GND
73	GND	AD56	GND	AD56
74	AD55	AD54	AD55	AD54
75	AD53	+ 5 V (E-A)	AD53	+ 3,3 V (E-A)
76	GND	AD52	GND	AD52
77	AD51	AD50	AD51	AD50
78	AD49	GND	AD49	GND
79	+ 5 V (E-A)	AD48	+ 3,3 V (E-A)	AD48
80	AD47	AD46	AD47	AD46
81	AD45	GND	AD45	GND
82	GND	AD44	GND	AD44
83	AD43	AD42	AD43	AD42
84	AD41	+ 5 V (E-A)	AD41	+ 3,3 V (E-A)
85	GND	AD40	GND	AD40
86	AD39	AD38	AD39	AD38
87	AD37	GND	AD37	GND
88	+ 5 V (E-A)	AD36	+ 3,3 V (E-A)	AD36
89	AD35	AD34	AD35	AD34
90	AD33	GND	AD33	GND
91	GND	AD32	GND	AD32

Anschluß	5-V-System		3,3-V-System	
	Kontaktreihe B	Kontaktreihe A	Kontaktreihe B	Kontaktreihe A
92	reserviert	reserviert	reserviert	reserviert
93	reserviert	GND	reserviert	GND
94	GND	reserviert	GND	reserviert

1): neu seit Ausgabe 2.1, 2): neu seit Ausgabe 2.2 der PCI-Spezifikation,
 3): *Universalkarten* haben auch hier eine Kerbe

Table 1.9 Anschlußbelegung (Pinout) der PCI-Slots. Besonderheiten durch Schraffur hervorgehoben (Erklärung in den nachfolgenden Einzelbeschreibungen)

Hinweise:

1. Die Anordnung in der Tabelle entspricht der tatsächlichen Kontaktanordnung, wenn man von oben auf den Steckverbinder blickt. Vgl. auch Abbildung 1.11 nebst Erklärung.
2. Die Speisespannungsanschlüsse dienen dann, wenn sie an sich nicht genutzt werden (z. B. 3,3-V-Anschlüsse in 5-V-Systemen) als zusätzliche "dynamische Masseverbindungen" (AC Return Paths) bei Schaltvorgängen. Sie sind deshalb über Kondensatoren (Richtwert: 0,01...0,1 µF) mit der Masse-Ebene verbunden. (Ein so geschalteter Kondensator stellt für eine *Stromänderung* - die in die Masse-Ebene des Motherboards eingeleitet werden muß - praktisch einen Kurzschluß dar und verringert somit die Impedanz der Masseverbindung zwischen Steckkarte und Motherboard.) Dies betrifft sinngemäß weitere Anschlüsse, die mit Festwerten belegt sind (PRSNT2#, 1# sowie M66EN; vgl. Abschnitte 1.2.14., 1.2.15. und 1.2.19.).

1.1.7. Speisespannungen

In den Slots sind folgende Speisespannungen vorgesehen: + 12 V, - 12 V, +5 V, + 3,3 V. Tabelle 1.10 nennt die geforderten Spannungstoleranzen und die je Slot entnehmbaren Ströme.

Speisespannung	Grenzwerte		Strombelastbarkeit je Slot
	minimal	maximal	
+ 5 V (± 5%)	4,75 V	5,25 V	max. 5 A ^{*)}
+ 3,3 V (± 0,3 V = 9,1%)	3,00 V	3,60 V	max. 7,6 A ^{*)}
+ 12 V (± 5%)	11,40 V	12,60 V	500 mA
- 12 V (± 10%)	- 10.80 V	- 13.20 V	100 mA

*): siehe Erklärung im Text

Table 1.10 Speisespannungen der PCI-Slots

Erklärung:

Eine Steckkarte darf eine maximale Verlustleistung von 25 W haben. Die angegebenen Ströme entsprechen dieser Verlustleistung, wenn die Karte ausschließlich über + 5 V oder + 3,3 V versorgt wird. Zur Signalisierung des Strombedarfs über die Anschlüsse PRSNT1#, 2# siehe Abschnitt 1.2.14.

Karten mit extremer Leistungsaufnahme

Die spezifizierten 25 W stellen (in einer PC-Umgebung) eine beachtliche Leistungsaufnahme dar. PCI-Karten, die mehr als 10 W verbrauchen, sollten deshalb einen Stromsparmodus haben, in dem die Leistungsaufnahme 10 W nicht überschreitet. Dieser Zustand muß nach dem Einschalten bzw. Rücksetzen automatisch eingenommen werden. Er muß die zum Starten erforderlichen Funktionen ermöglichen (u. a. Zugriffe auf den Konfigurationsadreßraum). Zum Stromsparen (Power Management) siehe weiterhin Abschnitt 1.6.5.

Einschaltreihenfolge

Es ist keine besondere Reihenfolge (Power Sequencing) beim Einschalten der Speisespannungen spezifiziert. Zum Einschaltrücksetzen siehe Abschnitt 1.2.3.

Speisespannungen in 5-V- und 3,3-V-Umgebungen

Tabelle 1.11 gibt einen Überblick über die in beiden Umgebungen genutzten Speisespannungen.

Systemumgebung (PCI-Signalpegel)	erforderliche Speisespannungen	E-A-Speisespannung ²⁾
5 V	+ 12 V, - 12 V, + 5 V ¹⁾	+ 5 V
3,3 V	+ 12 V, - 12 V, + 5 V, + 3,3 V	+ 3,3 V

1), 2): siehe Erklärung im Text

Tabelle 1.11 Speisespannungen in den PCI-Systemumgebungen

Erklärung:

- es ist zulässig, die 3,3-V-Speisespannung zusätzlich bereitzustellen. 5-V-Karten dürfen aber nicht davon abhängen, daß die 3,3 V am Slot anliegen (das gilt sinngemäß für Universalkarten). Wird die 3,3-V-Speisespannung benötigt, so muß diese auf der Karte erzeugt werden (Spannungsregler oder DC-DC-Wandler). Steht keine 3,3-V-Speisespannung bereit, so sollten die 3,3-V-Wege auf dem Motherboard vorhanden sein, und es sollte möglich sein, die 3,3-V-Versorgung nachzurüsten (eine Möglichkeit u. a. wäre eine Steckkarte, die aus den 5 V 3,3 V ableitet und in das 3,3-V-Leitungsnetz einspeist).
- diese Spannung ist an die Anschlüsse geführt, die in Tabelle 1.9 mit "+ 5 V E-A" bzw. "+ 3,3 V E-A" bezeichnet sind (in der PCI-Spezifikation: $+V_{IO}$). Hierüber werden die E-A-Stufen (Bustreiber und -empfänger) in den PCI-Schaltkreisen versorgt. Eine 5-V-Karte erhält demgemäß + 5 V, eine 3,3-V-Karte 3,3 V. Eine Universalkarte (Dual Voltage Card) erhält die Speisespannung, die der jeweiligen Umgebung entspricht (+ 5 V bei 5-V-Signalisierung, + 3,3 V bei 3,3-V-Signalisierung; die E-A-Stufen der Schaltkreise stellen sich gemäß der jeweiligen Spannung auf die Systemumgebung ein (Dual Voltage Buffers)).

Die neue Hilfsspannung: 3,3Vaux

Diese Hilfsspannung wurde in Ausgabe 2.2 der PCI-Spezifikation eingeführt. Der Zweck: eine Betriebsspannung soll auch dann noch verfügbar sein, wenn - zwecks Stromsparen - die anderen Speisespannungen von der Software abgeschaltet wurden. Damit sollen jene Schaltungen versorgt werden, die die Aktivitäten der Hardware überwachen und ggf. ein "Aufwecken" (= Verlassen des Stromsparmodus und Übergang zum Normalbetrieb) veranlassen. (Siehe auch Abschnitt 1.6.5.)

1.1.8. Grenzen des PCI-Bus

Wieviele Einrichtungen dürfen an einen PCI-Bus angeschlossen werden, wie lang dürfen die Busleitungen sein?

1.1.8.1. Anschließbare Einrichtungen

Der maßgebende Kennwert heißt “Buslast” (Bus Load). 1 Buslast = 1 Schaltkreis gemäß PCI-Spezifikation oder 1 Steckverbinder (Slot). Eine Steckkarte mit PCI-Schaltkreis entspricht 2 Buslasten (Steckverbinder + Schaltkreis).

Richtwerte:

- ein PCI-Bus darf maximal 10 Buslasten umfassen,
- die Anzahl der zulässigen Buslasten darf erhöht werden, wenn (1) dadurch die anderweitigen PCI-Spezifikationen nicht verletzt werden bzw. wenn (2) die Bus-Taktfrequenz (Nennwert: 33 MHz) entsprechend vermindert wird. (Ein System gemäß (1) erfordert besonders sorgfältigen Entwurf (u. a. eine umfassende Analogsimulation und Erprobung). Beide Ansätze sind bei rein “planaren” Systemen eher praktikabel als bei solchen, die Steckkarten beliebiger Herkunft aufnehmen können.)
- größere PCI-Konfigurationen sind ausschließlich durch Zusammenschalten von spezifikationsgemäßen PCI-Bussystemen über Brücken-Schaltkreise (PCI-to-PCI Bridges) realisierbar (Näheres in Abschnitt 1.5.),
- der PCI-Bus auf Steckkarten: an die Steckkontakte der Karte darf lediglich ein PCI-Schaltkreis angeschlossen werden (es ist also unzulässig, den PCI-Bus auf der Karte gleichsam weiterzuschleifen). Sollen auf der Steckkarte mehrere PCI-Einrichtungen angeordnet werden, so ist auf der Karte ein weiteres Bussystem aufzubauen. Der den Steckkontakten nachfolgende Schaltkreis ist dann die zur Buskopplung erforderliche PCI-to-PCI-Bridge (vgl. Abbildung 1.100).
- 66-MHz-Systeme. Derartige Systeme sind typischerweise auf 4 Buslasten beschränkt (2 Schaltkreise auf dem Motherboard + 1 Slot). (Das ist keine grundsätzliche Einschränkung, sondern eine Konsequenz aus den PCI-Spezifikationen und dem aktuellen Stand der Schaltungstechnologie. Siehe auch Abschnitt 1.2.2.4., besonders Abbildung 1.31.)
- eine PCI-Konfiguration darf maximal 256 PCI-Bussysteme umfassen,
- weitere Grenzen sind durch die Spezifikation des Konfigurationsadreibraums gegeben (Abschnitt 1.4.).

1.1.8.2. Leitungslängen

Der maßgebende Kennwert heißt “Buslaufzeit” (Propagation Time t_{prop}). Dies ist praktisch die “Flugzeit hin und zurück” (Round Trip Delay; Stichwort: 1. reflektierte Wellenfront) einer Signalfanke über den gesamten Bus.

Festlegung:

- 33 MHz Bustakt: t_{prop} nicht größer als 10 ns (bei verringertem Taktversatz (Clock Skew) maximal 11 ns; siehe auch Abschnitt 1.2.2.),
- 66 MHz Bustakt: t_{prop} nicht größer als 5 ns.

Wird der Bus mit einer geringeren Taktfrequenz betrieben, sind entsprechend größere Buslaufzeiten zulässig (zu den Zusammenhängen siehe Abschnitt 1.2.2.4.). Abbildung 1.20 zeigt, wie t_{prop} definiert ist.

Abbildung 1.20 So wird die Buslaufzeit (t_{prop}) gemessen (nach: PCI SIG; Beschriftung vereinfacht)

Erklärung:

Diese etwas merkwürdig aussehende Zeichnung zeigt zwei Signalverläufe für beide Schaltvorgänge:

- a) Schalten von Low nach High,
 - b) Schalten von High nach Low.
- 1) Signalverlauf am Treiber-Ausgang,
 - 2) so “sieht” der Empfänger-Eingang das Signal: während des Anstiegs bzw. Abfalls läuft bereits eine Wellenfront über die Leitung und wird am Ende reflektiert. Da die Leitung nicht mit dem Wellenwiderstand abgeschlossen ist, ergibt sich ein Reflexionsfaktor von nahezu 1; die Wellenfront wird gleichsam 1:1 zurückgeliefert und überlagert die hinlaufende Welle. Nach 2 Signallaufzeiten sieht also der Empfänger ein Signal mit einer Amplitude, die hinreicht, den jeweiligen Schwellwert (Minimum-High bzw. Maximum-Low) zu über- bzw. zu unterschreiten (für High: V_{ih} , für Low: V_{il}).

t_{prop} entspricht somit 2 Signallaufzeiten. Der ungünstigste Fall (für den die t_{prop} -Spezifikation (10 bzw. 5 ns) zu erfüllen ist) ergibt sich, wenn Treiber und Empfänger dicht beieinander an einem Ende des Bus sitzen (Abbildung 1.21).

Hinweise:

1. Es muß gewährleistet sein (Auslegung der Treiberstufen sowie der Busleitungen auf dem Motherboard (Stichwort: Wellenwiderstand)), daß innerhalb der Zeit t_{prop} die jeweilige Strom-Spannungs-Kennlinie (Abbildung 1.19) durchlaufen wird.
2. Das Prinzip des Schaltens mit der 1. rücklaufenden Wellenfront hat eine Problemstelle: bei sehr steilen Flanken (= sehr intensiven Stromänderungen) kann es zu unzulässig hohem Überschwängen kommen (vgl. auch die PCI-Bussignale in Abbildung 1.20)*). Gegenmaßnahmen:
 - Spezifikation einer maximalen Flankensteilheit (vgl. die Tabellen 1.6 und 1.7),
 - Begrenzung des Treiberstromes (vgl. die “Testpunkte” in Abbildung 1.19),
 - Ausnutzung der in den Schaltkreisen eingebauten Klammerdioden zum Abschneiden von Über- und Unterschwingern,
 - falls notwendig, Anordnung besonderer Busabschlußdioden an den Leitungsenden.

*) das Signal darf nach Über- bzw. Unterschreiten des jeweiligen Schwellwertes (V_{ih} , V_{il}) maximal bis zu einem Spannungswert V_{ih} , V_{il} (vgl. die “Testpunkte” in Abbildung 1.19) schwingen, muß aber ober- bzw. unterhalb des jeweiligen Schwellwertes (V_{ih} , V_{il}) bleiben.

Abbildung 1.21 t_{prop} und Leitungslänge

Die PCI-Spezifikation nennt keine zulässige Gesamt-Leitungslänge. Diese ergibt sich vielmehr aus t_{prop} und der jeweiligen Signalausbreitungsgeschwindigkeit (bzw. aus deren Kehrwert τ).

Anhaltswerte:

Mit $\tau \approx 7$ ns/m ergeben sich rechnerisch für $t_{prop} = 10$ ns Leitungslängen von ca. 70 cm. Leitungslängen in der Praxis: ca. 30 cm, in 66-MHz-Konfigurationen höchstens die Hälfte.

Leitungslängen auf Steckkarten

Die Verbindung zwischen Slot-Kontakt und PCI-Schaltkreis hat den Charakter einer Abzweig- bzw. Stichleitung (Stub). Solche Stichleitungen sind - im Sinne der Leitungstheorie - Inhomogenitäten. Sie dürfen deshalb nicht allzu lang sein (Tabelle 1.12 + Abbildung 1.5).

Signale	Abbildung 1.5	maximale Leitungslänge ¹⁾
32-Bit-Bus	a	38 mm (1,5") ²⁾
64-Bit-Erweiterung	b	50 mm (2") ²⁾
Taktsignal (CLK)	c	61... 66 mm (2,5" \pm 0,1")

Tabelle 1.12 Zulässige Leitungslängen auf Steckkarten (nach: PCI SIG)

Erklärung:

- 1) Länge der Leiterzüge (Traces) vom Steckverbinder zum Schaltkreis, keine "Luftlinie"!
- 2) außer Interrupt-, Test- (JTAG-), Takt- und Rücksetzsignalen (CLK, RST#).

Richtwerte:

- Wellenwiderstand eines Leiterzuges: 60...100 Ω ,
- normierte Signallaufzeit (Kehrwert der Ausbreitungsgeschwindigkeit) τ : 150...190 ps/Zoll \triangle 59...75 ps/cm \triangle 6...7,5 ns/m.

1.1.9. Adressen, Daten, Kommandos**1.1.9.1. 32 und 64 Bits****Der 32-Bit-Bus**

PCI war von Anfang an als 32-Bit-Bus ausgelegt: 32 Bit Adreßlänge, Datenübertragung in 32-Bit-Worten.

Bit- und Byteanordnung

Hinsichtlich der Anordnung der Bits und Bytes in den jeweils längeren Datenstrukturen folgt PCI der Intel-Prozessorarchitektur (IA-32): es gilt Rechtsindizierung der Bits (Bit 0 ist das jeweils niedrigstwertige) und Rechtsadressierung der Bytes (eine Byteadresse zeigt auf das jeweils niedrigstwertige Byte in der jeweiligen Struktur).

Hinweise:

1. Die 32-Bit-Struktur heißt in der PCI-Terminologie *Doppelwort* (Double Word, DWORD oder DW), und zwar in Entsprechung zur Intel-Prozessorarchitektur IA-32.
2. Zur Bit- und Byteanordnung siehe weiter unten Abbildung 1.25.

64-Bit-Adressierung am 32-Bit-Bus

Bereits am 32-Bit-Bus kann man 64-Bit-Adressen verwenden. Solche Adressen werden zeitmultiplex in 2 aufeinanderfolgenden 32-Bit-Abschnitten übertragen (Dual Address Cycle).

Der 64-Bit-Bus

Die 64-Bit-Erweiterung sieht weitere 32 Adreß- und Datensignale sowie 4 ergänzende Byteerlaubnis-signale vor. Damit können 64 Datenbits (8 Bytes) parallel übertragen werden.

64-Bit-Adressierung am 64-Bit-Bus

Die zusätzlichen Adreß- und Datensignale (AD63...32) werden genutzt, um die höherwertigen 32 Adreßbits parallel zu den niederwertigen auf den Bus zu legen. Die 64-Bit-Adressierung erfordert aber nach wie vor einen Dual Address Cycle. Hierbei werden im zweiten Abschnitt die höherwertigen 32 Adreßbits nochmals über die niederwertigen Adreß- und Datenleitungen übertragen (somit sind auch 32-Bit-Einrichtungen am 64-Bit-Bus der 64-Bit-Adressierung zugänglich).

32-Bit-Adressierung am 64-Bit-Bus

Wird eine 32-Bit-Adresse auf den 64-Bit-Bus gelegt, so sind die höchstwertigen Adreßsignale AD63...32 mit Nullen zu belegen (Abschnitt 1.3.9.)

1.1.9.2. Adressierung**Adreßstrukturen**

Die Adresse dient (1) zur Auswahl der jeweiligen Einrichtung, auf die zugegriffen werden soll (des jeweiligen Targets) und (2) zur Adressierung der einzelnen Datenstruktur in der ausgewählten Einrichtung.

Die elementare adressierbare Datenstruktur ist das Byte (Byteadressierung). Das einzelne Byte wird im Zusammenwirken von Adreßangabe (AD-Signale) und Byteauswahlsignalen (C/BE-Signalen) ausgewählt.

Abbildung 1.22 veranschaulicht die verschiedenen Adreßstrukturen.

Abbildung 1.22 Adreßstrukturen. *): muß Null sein, wenn 64-Bit-Datenzugriffe über den 64-Bit-Bus geführt werden (integrale Adresse); **): Abbildung zeigt bevorzugte Nutzung (in PCI-Spezifikation: reserviert)

Adreßräume

Der PCI-Bus ermöglicht Zugriffe auf 3 Adreßräume (Tabelle 1.13):

1. auf den Speicheradreßraum,
2. auf den E-A-Adreßraum,
3. auf den Konfigurationsadreßraum.

Das jeweilige *Kommando* (Abschnitt 1.1.9.4.) bestimmt, auf welchen der Adreßräume jeweils zugegriffen wird.

Adreßraum	Adreßdecodierung	Adresse betrifft	adressierbar sind
Speicheradreßraum (Abbildung 1.22a)	Bits 31...2 (d. h. bis auf's Doppelwort) ¹⁾	1 Doppelwort (32 Bits); Byteauswahl über C/BE-Signale	1G Doppelworte = 4 GBytes ²⁾
E-A-Adreßraum (Abbildung 1.22b)	Bits 31...0 (d. h. bis auf's Byte)	1 Byte (ergänzende Byteauswahl über C/E-Signale)	4 GBytes ³⁾
Konfigurationsadreßraum (Abbildung 1.22c)	“geographische Adressierung” der Einrichtung (über IDSEL-Eingang), Auswahl des logischen Gerätes über Bits 10...8	1 Doppelwort (32 Bits); Byteauswahl über C/BE-Signale	64 Doppelworte = 256 Bytes je “logischem Gerät”

1): um Bits 63...31 erweiterbar (64-Bit-Adressierung); 2): bei 64-Bit-Adressierung: 2^{62} Doppelworte; 3): 64-Bit-Adressierung nicht vorgesehen

Tabelle 1.13 Die Adreßräume des PCI-Bus

Adreßdecodierung

Bei Speicher- und E-A-Zugriffen muß jede Einrichtung die Adreßbelegung auf dem Bus auswerten, um zu erkennen, ob sie als Target angesprochen wird. Hierbei ist die gesamte Adresse zu berücksichtigen.

Hinweis:

Da jede Einrichtung alle Adreßbits auswerten muß, gibt es am PCI-Bus keine Alias-Adressen.

Positive Adreßdecodierung

Hiermit bezeichnet man die sozusagen natürliche Form der Target-Auswahl: alle Einrichtungen beobachten die Adreßbelegung auf dem Bus und prüfen, ob diese ihren eigenen Adreßbereich betrifft. Ist das der Fall, so “erkennt sich” die betreffende Einrichtung als Target und nimmt demgemäß an dem Buszugriff teil. Die Adreßerkennung wird durch Erregen der Busleitung DEVSEL# angezeigt.

Subtraktive Adreßdecodierung

Am PCI-Bus kann eine einzige Einrichtung für subtraktive Adreßdecodierung ausgelegt werden. Diese Einrichtung “erkennt sich” immer dann als adressiert, wenn keine andere Einrichtung die Adreßerkennung gemeldet hat (Abbildung 1.22). Mit anderen Worten: der betreffenden (einzigen) Einrichtung wird gleichsam der ungenutzte Rest des Adreßraumes zugewiesen.

Es ist üblicherweise die PCI-to-PCI Bridge (in Aufwärtsrichtung; vgl. Abschnitt 1.5.4.), die als einzige Einrichtung für subtraktive Adreßdecodierung ausgelegt ist. Somit können alle Zugriffe auf Adressen, die von den anderen Einrichtungen nicht belegt sind, zum vorgeordneten PCI-Bus weitergereicht werden.

Abbildung 1.23 Positive und subtraktive Adreßdecodierung. A...D: Einrichtungen am PCI-Bus, A1, A2: beispielhafte Adressen

Erklärung:

Den Einrichtungen A, B und C ist jeweils ein Teil des Adreßraumes zugeordnet. Wird beispielsweise die Adresse A1 auf den Bus gelegt, so erkennt Einrichtung C, daß sie als Target angesprochen wurde und signalisiert dies durch Erregen der Busleitung DEVSEL#. Wird hingegen die Adresse A2 auf den Bus gelegt, so wird diese von keiner der Einrichtungen A, B, C erkannt, demzufolge wird DEVSEL# nicht aktiviert. Dies bewirkt, daß Einrichtung D (z. B. eine PCI-to-PCI Bridge) als Target wirksam wird.

Adreßzuordnung zu den Einrichtungen

Die Adreßbereiche der einzelnen Einrichtungen sollten grundsätzlich an integralen Doppelwortadressen beginnen, d. h. an Byteadressen, deren Bits 1, 0 gleich Null sind. Folglich kann der Adreßraum lediglich doppelwortweise zwischen den Einrichtungen aufgeteilt werden (DWORD Granularity). Für den Speicheradreßraum ist dies zwingend vorgeschrieben. Beim E-A-Adreßraum dürfen hingegen die Adreßbereiche der Einrichtungen an Bytegrenzen beginnen. Der E-A-Adreßraum läßt sich also bis auf's Byte zwischen den einzelnen Einrichtungen aufteilen (Byte Granularity). Der Grund: Abwärtskompatibilität zu herkömmlicher PC-Hardware.

Hinweis:

Aus den Strukturen der Konfigurationsangaben (Stichwort: Basisadreßregister) ergeben sich weitere Einschränkungen der Adreßaufteilung (Abschnitte 1.4 und 1.5).

Adressen für 64-Bit-Zugriffe über den 64-Bit-Bus

Diese müssen integrale Adressen auf 8-Byte-Strukturen (Quadwords) sein (Adreßbit 2 = 0; vgl. auch die Abbildungen 1.22 und 1.24).

Die Speicheradresse

Die Speicheradresse (Abbildung 1.22a) wählt ein Doppelwort im Speicheradreßraum aus. Die Adreßbits 31...2 dienen zur Doppelwortadressierung. In den Bits 1 und 0 ist die Art des jeweiligen Burst-Zugriffs codiert (Tabelle 1.14).

Adreßbits AD1, 0	Adreßfolge im Burstzugriff
0,0	lineare Aufwärtszählung
0,1	reserviert ^{1), 2)}
1,0	zyklischer Zugriff auf Cache-Eintrag (Cache Line Wrap Mode) ³⁾
1,1	reserviert ¹⁾

1)...3): siehe Erklärung im Text

Tabelle 1.14 Die Adreßbits 1 und 0 beim Speicherzugriff

Erklärung:

- 1) auf eine solche Bitbelegung hin muß die Target-Einrichtung nach der ersten Datenphase den Buszugriff durch Trennen (Disconnect) beenden,
- 2) in Ausgabe 2.0 der PCI-Spezifikation: Adreßzählweise der Intel-Prozessoren (Interleaved Burst Order). Wird nicht mehr unterstützt.

- 3) wie lang ein Cache-Eintrag ist, steht in einem besonderen Register des Konfigurationsadreibereiches (Cacheline Size Register; Abschnitt 1.4.3.3.). Eine Target-Einrichtung, die dieses Register nicht hat, muß nach der ersten Datenphase den Buszugriff durch Trennen (Disconnect) beenden.

Hinweise:

1. Speicherbereiche in den Einrichtungen müssen stets an integralen Doppelwortadressen beginnen.
2. Zum Adreßvergleich (für die positive Adreßdecodierung) werten die Einrichtungen die Adreßbits 31...2 aus.
3. Die Zugriffe selbst können bis auf's einzelne Byte (byteselektiv) geführt werden. Die Byteauswahl erfolgt hierbei über die Signale C/BE3#...0# (zuzüglich C/BE7#...4# bei 64-Bit-Datenübertragung). Ein Master muß also ggf. die intern vorliegende Byteadresse in eine Doppelwortadresse (bei 64-Bit-Zugriffen: Quadwortadresse) und in die Belegung der Byteerlaubnis-signale umschlüsseln.

Speicherzugriffe und 64-Bit-Adressierung

Der Speicheradreibereich kann durch 64-Bit-Adressierung über die Grenze von 4 GBytes hinaus erweitert werden. Eine 32-Bit-Adresse wird hierbei als Zugriff auf die niedrigsten 4 GBytes des Speicheradreibereiches angesehen. Somit gilt: eine 64-Bit-Adresse, die in ihren höherwertigen 32 Bits ausschließlich Nullen enthält, entspricht einer 32-Bit-Adresse (Abbildung 1.24).

Abbildung 1.24 64-Bit- und 32-Bit-Adressen. a) 64-Bit-Adresse mit Nullen in den höherwertigen 32 Bits, b) 32-Bit-Adresse. *): integrale Adresse für 64-Bit-Zugriff über 64-Bit-Bus

Es gilt folgende Festlegung: Adressen gemäß Abbildung 1.24a sind wie 32-Bit-Adressen zu behandeln. Ein Master, der mit einer solchen 64-Bit-Adresse auf den Bus zugreifen will, darf dann keinen 64-Bit-Adressierungszyklus (Dual Access Cycle) ausführen. Vielmehr ist der Zugriff mit einem 32-Bit-Adressierungszyklus (Single Access Cycle) zu beginnen. Sinngemäß muß sich in diesem Fall das Target, auch wenn es für 64-Bit-Adressierung ausgelegt ist, wie eine 32-Bit-Einrichtung verhalten (d. h., die Zugriffe auf die niedrigsten 4 GBytes des Speicheradreibereiches beschränken).

Die E-A-Adresse

Die E-A-Adresse (Abbildung 1.22b) wählt ein Byte im E-A-Adreibereich aus. Der E-A-Adreibereich umfaßt insgesamt 4 GBytes (es gibt keine 64-Bit-Adressierung). Beim Adreßvergleich müssen die Einrichtungen alle 32 Adreßbits auswerten. Die E-A-Adresse wirkt zwar bis auf's Byte, bei der Datenübertragung wird aber jedes Byte auf den Busleitungen erwartet, die seiner Adresse im Doppelwort entsprechen. Deshalb werden zusätzlich die Byteerlaubnis-signale (C/BE3#...0#) erregt. Tabelle 1.15 enthält die zulässigen Erregungsmuster.

Adreßbits AD1, 0	Byteerlaubnissignale			
	C/BE3#	C/BE2#	C/BE1#	C/BE0#
0,0 ¹⁾	x	x	x	0
0,1 ²⁾	x	x	0	1
1,0 ³⁾	x	0	1	1
1,1 ⁴⁾	0	1	1	1
x,x ⁵⁾	1	1	1	1

1)...5): siehe Erklärung im Text; x = beliebige Belegung

Tabelle 1.15 Zulässige Byteerlaubnisbelegungen bei E-A-Zugriffen (die Byteerlaubnissignale sind aktiv Low)

Erklärung:

- 1) Byteposition 0 adressiert; Zugriff auf 1, 2, 3 oder 3 Bytes (1110, 1100, 1000, 0000) ^{*)},
- 2) Byteposition 1 adressiert; Zugriff auf 1, 2 oder 3 Bytes (1101, 1001, 0001) ^{*)},
- 3) Byteposition 2 adressiert; Zugriff auf 1 oder 2 Bytes (1011, 0011) ^{*)},
- 4) Byteposition 3 adressiert; Zugriff auf 1 Byte,
- 5) kein Zugriff bei beliebiger Byteadresse.

*) : Belegung C/BE3#...0#.

E-A-Zugriffe mit anderen als den zulässigen Belegungen der Byteerlaubnissignale müssen vom Target durch Abbrechen (Target Abort) beendet werden.

64-Bit-Adressierung bei E-A-Zugriffen

Dies ist nicht vorgesehen; der E-A-Adreßraum ist auf 4 GBytes beschränkt.

Der 64-Bit-Bus bei E-A-Zugriffen

Die PCI-Spezifikation schließt nicht aus, Einrichtungen so zu entwerfen, daß sie E-A-Zugriffe mit 64-Bit-Daten unterstützen, es wird aber nicht empfohlen (hoher Aufwand bei eher geringem Nutzen).

Die Konfigurationsadresse

Die Konfigurationsadresse (Abbildung 1.22c) wählt ein Doppelwort (= Konfigurationsregister) im Konfigurationsadreßraum aus. Dieser ist auf 64 Doppelworte (= 256 Bytes) je "logischem Gerät" (Function) beschränkt (eine (physische) Einrichtung darf maximal 8 logische Gerätefunktionen enthalten; vgl. Abschnitt 1.1.3.).

Es gibt 2 Adreßstrukturen (Abbildung 1.22c):

- *Typ 0.* Eine solche Adresse dient zu Konfigurationszugriffen auf die Einrichtungen am selben Bus.
- *Typ 1.* Eine solche Adresse dient zu Konfigurationszugriffen auf Einrichtungen an anderen PCI-Bussystemen (in einer über Brücken-Schaltkreise verkoppelten Konfiguration).

Aufteilung der Adreßfelder:

- *Register*. Wählt eines der maximal 64 Konfigurationsregister (jeweils 1 Doppelwort) aus. Die Zugriffe selbst können bis auf's einzelne Byte (byteselektiv) geführt werden. Die Byteauswahl erfolgt hierbei über die Signale C/BE3#...0#.
- *logisches Gerät* (Function Number). Wählt die "logische" Gerätefunktion in der jeweiligen Einrichtung aus.
- *Einrichtung* (Device Number). Wählt die jeweilige "physische" Einrichtung (Schaltkreis auf dem Motherboard, PCI-Slot) aus.
- *Bus-Nr.* Wählt den PCI-Bus innerhalb der Systemkonfiguration aus.

Auswahl der (physischen) Einrichtung (Adresse vom Typ 0)

Die jeweilige Einrichtung wird durch "geographische" Adressierung ausgewählt, d. h. durch Erregen des jeweiligen IDSEL-Eingangs (damit wird genau ein bestimmter PCI-Schaltkreis oder Slot angesprochen, unabhängig von irgendwelchen in den Einrichtungen vorgegebenen Target-Adressen). Die PCI-Spezifikation schreibt nicht vor, auf welche Weise die IDSEL-Belegung von den zentralen Steuerschaltungen zu bilden ist (so könnte man sich einen programmierbaren Ausgabe-Port vorstellen, an den die IDSEL-Eingänge über Einzelleitungen angeschlossen sind). Üblich ist jedoch die Nutzung der Adreßbits 31...11. Diese werden bei Konfigurationszugriffen von den zentralen Steuerschaltungen (die meistens mit der Host-to-PCI Bridge in einem Schaltkreis untergebracht sind) im 1-aus-n-Code belegt. Es genügt dann, jeden IDSEL-Eingang mit jeweils einer anderen der AD-Leitungen 31...11 zu verbinden.

Hinweis:

Im PC-Bereich hat es eine gewisse Tradition, nur die Adreßbits 31...16 zur IDSEL-Ansteuerung auszunutzen. Typischerweise werden dann die Adreßbits von Bit 16 an auf die IDSEL-Eingänge der einzelnen Einrichtungen geführt (1. Einrichtung an AD16, 2. Einrichtung an AD17 usw.). Siehe weiterhin die Abschnitte 1.4.1. und 1.2.19.

1.1.9.3. Datenübertragung

Daten werden über die Busleitungen AD31...0 (beim 64-Bit-Bus zuzüglich AD63...32) übertragen (Abbildung 1.25). Die Gültigkeit der Datenbusbelegung wird byteweise über die Byteerlaubnisignale C/BE3#...0# (beim 64-Bit-Bus zuzüglich C/BE7#...4#) gesteuert:

- C/BE x # = 0 (aktiv): Datenbyte ist gültig (wird also von der jeweils empfangenden Einrichtung übernommen),
- C/BE x # = 1 (inaktiv): Datenbyte ist ungültig und wird ignoriert.

Abbildung 1.25 Datenanordnung

Erklärung:

- a) Datenanordnung im Doppelwort (32-Bit-Bus; Busleitungen AD31...0),
- b) Datenanordnung in der 64-Bit-Ergänzung (Busleitungen AD63...32),
- c) Byteadresse bei 32-Bit-Übertragung (Adreßbits 1,0),

- d) Byteadresse bei 64-Bit-Übertragung (Adreßbits 2,1,0),
- e) Zuordnung der Byteerlaubnissignale (C/BE-Leitungen).

Grundsatz:

Datenbytes werden auf dem Bus an der Position erwartet, der ihrer Byteadresse im Doppelwort (bzw. im 64-Bit-Wort) entspricht; ein Umordnen von Bytes (Byte Swapping) ist nicht vorgesehen. Die Byteerlaubnissignale werden jeweils so aktiviert, wie dies zum Übertragen der jeweiligen Daten erforderlich ist (Tabelle 1.16).

Zugriff auf	Byte- adresse	C/BE3#...0#*	Busbelegung			
			Byte 3	Byte 2	Byte 1	Byte 0
Byte	0	1,1,1,0				
	1	1,1,0,1				
	2	1,0,1,1				
	3	0,1,1,1				
16-Bit-Wort	0	1,1,0,0				
	1	1,0,0,1				
	2	0,0,1,1				
32-Bit-Doppelwort	0	0,0,0,0				

*) : 0 = aktiv (Datenbyte gültig), 1 = inaktiv (Datenbyte ungültig)

Tabelle 1.16 Erregung von Byteerlaubnissignalen

Erklärung:

Die Tabelle zeigt, wie die Byteerlaubnissignale des 32-Bit-Bus beim Übertragen typischer Datenstrukturen in Abhängigkeit von der Byteadresse belegt werden.

Hinweise:

1. Die Tabelle enthält nicht alle zulässigen Belegungen. So können Brücken-Schaltkreise mehrere Zugriffe, die dasselbe Doppelwort betreffen, zu einem Buszugriff zusammenfassen (Write Merging; vgl. Abschnitt 1.3.12.3.). Dann kann es durchaus vorkommen, daß z. B. C/BE3# und C/BE1# aktiv sind, C/BE2# und C/BE0# aber nicht (nämlich beim Zusammenfassen von Schreibzugriffen auf die Bytepositionen 3 und 1).
2. AD-Leitungen der Bytepositionen, die durch inaktive C/BE-Signale als ungültig gekennzeichnet sind, müssen trotzdem stabile Belegungen (Low oder High) tragen (die Belegung ist bedeutungslos, wird aber für die Paritätskontrolle benötigt).
3. Es gibt Fälle, in denen die Byteerlaubnissignale gar nicht berücksichtigt werden (Näheres siehe Abschnitt 1.3.12.).

1.1.9.4. Kommandos

In der Adreßphase des Buszugriffs tragen die Busleitungen C/BE3#...0# einen Kommandocode, der die Art des Buszugriffs bestimmt (Tabelle 1.17).

Hinweise:

1. Die eigentlichen Datenübertragungs-Kommandos werden in Form von Buszyklen ausgeführt. Diese Abläufe werden in Abschnitt 1.3. näher beschrieben.
2. Abschnitt 1.3.13. enthält eine Kurzbeschreibung der einzelnen Kommandos.

C/BE3#...0# ¹⁾	Kommando
0,0,0,0 (0H)	Unterbrechungsbestätigung (Interrupt Acknowledge)
0,0,0,1 (1H)	Sonderzyklus (Special Cycle)
0,0,1,0 (2H)	Eingabe (I/O Read)
0,0,1,1 (3H)	Ausgabe (I/O Write)
0,1,0,0 (4H)	reserviert
0,1,0,1 (5H)	reserviert
0,1,1,0 (6H)	Speicher lesen (Memory Read)
0,1,1,1 (7H)	Speicher schreiben (Memory Write)
1,0,0,0 (8H)	reserviert
1,0,0,1 (9H)	reserviert
1,0,1,0 (AH)	Konfiguration lesen (Configuration Read)
1,0,1,1 (BH)	Konfiguration schreiben (Configuration Write)
1,1,0,0 (CH)	Speicher sequentiell lesen ²⁾ (Memory Read Multiple)
1,1,0,1 (DH)	64-Bit-Adressierung, zeitmultiplex (Dual Address Cycle)
1,1,1,0 (EH)	Cache-Eintrag lesen (Memory Read Line)
1,1,1,1 (FH)	Cache-Eintrag schreiben (Memory Write and Invalidate)

1): bei Adreßübertragungen über den 64-Bit-Bus (Dual Address Cycles) wird der Kommandocode zusätzlich über die Leitungen C/BE7#...4# übertragen;

2): es wird mehr als ein Cache-Eintrag gelesen

Tabelle 1.17 Kommandocodes

Zu unterstützende Kommandos

Es gibt nur 2 Kommandos, die jede Einrichtung unterstützen muß: *Konfiguration lesen* und *Konfiguration schreiben*. Welche weiteren Kommandos implementiert werden, hängt von der Art der jeweiligen Einrichtung ab. Empfängt eine Einrichtung ein Kommando, das sie nicht ausführen kann, so muß sie den Buszugriff mit Abbruch (Target Abort) beenden (Abschnitt 1.3.5.).

1.1.9.5. Paritätskontrolle

Die AD- und C/BE-Signale werden mit einem Paritätsbit kontrolliert:

- AD31...0 + C/BE3#...0# durch Paritätssignal PAR,
- AD63...32 + C/BE7#...4# durch Paritätssignal PAR64 (64-Bit-Erweiterung).

Es wird auf gerade Parität (Even Parity) geprüft (die Anzahl der Einsen über alle Leitungen (AD + C/BE# + PAR) muß gerade sein).

Die Belegung des jeweiligen Paritätssignals folgt jener der AD- und C/BE-Signale mit einem Taktzyklus Verzögerung nach.

Das Paritätssignal wird gebildet:

- in Adreßphasen, beim Daten-Schreiben und in Sonderzyklen: vom Master,
- beim Daten-Lesen und bei der Unterbrechungsbestätigung: vom Target.

Zu weiteren Einzelheiten siehe Abschnitt 1.3.10.

1.2. Einzelbeschreibung der Bussignale

1.2.1. Signalanschaltung

1.2.1.1. Signaltypen

Die Signale des PCI-Bus werden, je nach Funktion, auf verschiedene Weise an den Bus angeschaltet. Zudem sind nicht alle PCI-Signale "echte" Busleitungen. Abbildung 1.26 veranschaulicht die verschiedenen Signaltypen.

Zur Bezeichnungswiese: Ein- und Ausgänge

In der PCI-Literatur (und auch hier) beziehen sich diese Bezeichnungen auf die einzelne Einrichtung am PCI-Bus, wobei die zentralen Steuerschaltungen eine Sonderstellung einnehmen (ein Eingangssignal führt von den zentralen Steuerschaltungen in die jeweilige Einrichtung hinein, ein Ausgangssignal führt aus der jeweiligen Einrichtung heraus zu den zentralen Steuerschaltungen).

Abbildung 1.26 Signaltypen: so werden Signale an den PCI-Bus angeschaltet

Erklärung:

- a) Eingang. Zweiwertiges Signal (nur Low oder High), das von der Einrichtung empfangen wird. (Als Quelle kommen nur zentrale Steuerschaltungen in Frage.)
- b) Ausgang. Zweiwertiges Signal (nur Low oder High), das von der Einrichtung getrieben wird. (Es handelt sich um Einzelsignale, die zu zentralen Steuerschaltungen führen.)
- c) TS-Busleitung. TS = Tri State (Low, High, hochohmig). Bidirektionale Busleitung. Darf jeweils nur von einer Einrichtung getrieben werden.

- d) STS-Busleitung. STS = Sustained Tri State (Low, High, hochohmig = High). Signal ist aktiv Low. Bidirektionale Busleitung, die im hochohmigen Zustand über einen Pull-up-Widerstand auf High-Pegel gehalten wird. Darf jeweils nur von einer Einrichtung getrieben werden.
- e) OD-Busleitung. OD = Open Drain. Zweiwertiges Ausgangssignal. Aktiv Low. Busleitung, die von den Einrichtungen zu den zentralen Steuerschaltungen führt und mit einem Pull-up-Widerstand beschaltet ist. Mehrere Einrichtungen können ein solches Signal gleichzeitig aktivieren (= auf Low ziehen; Prinzip des Wired Or).
- f) Festwert. Zweiwertiges Ausgangssignal. Wird zentral über Pull-up-Widerstand auf High gehalten (bei Steckkarten werden diese Anschlüsse als zusätzliche "dynamische Masseverbindungen" ausgenutzt; sie sind deshalb an den PCI-Slots über Kondensatoren (Richtwert: 0,01...0,1 μ F) mit der Masse-Ebene verbunden - vgl. weiter unten die Abbildungen 1.35 und 1.36). Eine Low-Belegung entsteht durch Festverbindung mit Masse (GND). Es genügt eine solche Festverbindung, um das Signal auf Low zu halten (Wired Or).
- 1) Eingangsstufe,
 - 2) Ausgangsstufen (Treiber),
 - 3) Lesesignalweg (zu entsprechenden Eingangsstufen),
 - 4) Lesesignalweg zum Zurücklesen. Manche Wired-Or-Signale können zurückgelesen werden. Damit kann die betreffende Einrichtung die Belegung des Signals abfragen. (Beispiel: eine für 66 MHz ausgelegte Einrichtung fragt das Signal M66EN ab, um zu bestimmen, ob der Bus mit 66 oder mit 33 MHz betrieben wird.)

Tabelle 1.18 gibt einen Überblick über die Signaltypen und Signale des PCI-Bus.

Spitzfindigkeiten

Es gibt Signale, die eigentlich Ein- oder Ausgänge sind, aber über Tri-State-Treiber angeschlossen werden, als Tri State (TS) spezifizierte Signale, die aber auf dem Motherboard mit Pull-up-Widerständen beschaltet sind usw. Diese Feinheiten erklären wir in den nachfolgenden Einzelbeschreibungen.

Signaltyp	Bussignale
Eingang	<ul style="list-style-type: none"> ▪ CLK, RST#, IDSEL, ▪ SBO#, SDONE ▪ TCK, TDI, TMS, TRST#
Ausgang	TDO
TS-Signal als Eingang	GNT#
TS-Signal als Ausgang	REQ#
Festwert-Ausgang	<ul style="list-style-type: none"> ▪ PRSNT2#,1# ▪ M66EN¹⁾
TS-Busleitung	AD31...0, C/BE3#...0#, PAR
TS-Busleitung mit Pull-ups	AD63...32, C/BE7#...4#, PAR64
STS-Busleitung	<ul style="list-style-type: none"> ▪ FRAME#, IRDY#, TRDY#, STOP#, LOCK#, DEVSEL#, PERR# ▪ REQ64#, ACK64#

Signaltyp	Bussignale
OD-Busleitung	SERR#, INTA#...INTD#, PME#, CLKRUN# ²⁾

1): ist auch Eingangssignal (vgl. Abbildung 1.36); 2): nur für Einrichtungen auf dem Motherboard

Tabelle 1.18 Signaltypen und Bussignale

Signalbezeichner

Wir verwenden die Bezeichnungen gemäß PCI-Spezifikation. Besonderheiten:

- Signale, die aktiv Low wirken, werden durch ein # gekennzeichnet (Beispiel: FRAME#),
- Durchnummerierungen: die PCI-Literatur verwendet eine eigene Darstellungsweise. Beispiel: AD[31::0] für die 32 Adreß- und Datenleitungen. Wir nutzen gelegentlich eine vereinfachte Darstellung (z. B. AD31...0).
- Bitnummern: es gilt - wie bei den Intel-Prozessoren - die Rechtsindizierung: Bit Nr. 0 ist das jeweils niedrigstwertige.

1.2.1.2. Pull-up-Widerstände

Verschiedene Signale müssen mit Pull-up-Widerständen beschaltet werden. Tabelle 1.19 nennt einschlägige Widerstandswerte.

E-A-Versorgungsspannung ¹⁾	Mindestwert ²⁾	typischer Wert ³⁾
5 V	963 Ω	2,7 kΩ ± 10%
3,3 V	2,42 kΩ	8,2 kΩ ± 10%

1)...3): siehe Erklärung im Text

Tabelle 1.19 Richtwerte für Pull-up-Widerstände

Erklärung:

- 1) vgl. Abschnitt 1.1.7.,
- 2) gilt für ein PCI-System mit 16 Einrichtungen (hierbei wird die Busbelastung über die (statischen) Lastströme eingerechnet, nicht über die kapazitive Belastung). Die 16 Einrichtungen werden gleichsam als schlimmster Fall angesehen.
- 3) gilt für ein PCI-System mit 10 Einrichtungen.

In Systemen mit weniger Einrichtungen dürfen die Widerstandswerte höher sein. *Berechnung:*

$$R_{\max} = \frac{V_{CC\min} - V_x}{I_{th} \cdot L}$$

Hierin bedeuten:

- $V_{CC\min}$: E-A-Versorgungsspannung. Vgl. Tabelle 1.11,
- 5-V-Signalisierung: $V_x = 2,7 \text{ V}$,
- 3,3-V-Signalisierung: $V_x = 0,7 V_{CC}$,

- I_{ih} : High-Eingangsstrom (Leckstrom); vgl. Tabellen 1.6 und 1.7,
- L = Anzahl der Einrichtungen.

Solche Pull-up-Widerstände sind für folgende Bussignale vorgesehen: FRAME#, IRDY#, TRDY#, STOP#, LOCK#, DEVSEL#, PERR#, CLKRUN#, SERR#, INTA#...INTD#, PME# sowie (wenn implementiert) für AD63...32, C/BE7#...4#, PAR64, REQ64#, ACK64#.

Ungenutzte Signale

PCI-Signale, die im betreffenden System nicht genutzt werden, sind gelegentlich mit Widerständen (Pull-up oder Pull-down) zu beschalten. Manchmal ist auch eine Beschaltung mit Kondensatoren vorgesehen. Näheres dazu in den jeweiligen Einzelbeschreibungen (Abschnitt 1.2.).

Unterbringung der Pull-up-Widerstände

Pull-up-Widerstände sind grundsätzlich nur auf dem Motherboard untergebracht, nicht auf Steckkarten.

1.2.1.3. Der Bus in Ruhe - Parken

Tri-State-Leitungen, die nicht angesteuert werden (und demzufolge hochohmig sind) nehmen, wenn nichts dagegen getan wird, mit der Zeit eine undefinierte Belegung ein. Beim PCI-Bus tut man aber etwas dagegen; "schwimmende" (floating) Signale gibt es grundsätzlich nicht. Man spricht hierbei vom "Parken" des Bus. Die Maßnahmen im einzelnen:

1. Ziehen auf High-Pegel über Pull-up-Widerstände,
2. "aktives" Parken durch eine Master-Einrichtung (Park Master).

Parken mittels Pull-up-Widerständen

Es liegt nahe, ein inaktives Signal mittels Pull-up-Widerstand auf einen definierten Pegel (hier: High) zu ziehen. Dies ist u. a. für die Signale vom Typ STS vorgesehen. Das Problem: ein vergleichsweise hochohmiger Widerstand (vgl. Tabelle 1.14) führt zu vergleichsweise langen Anstiegszeiten (da die parasitären Kapazitäten nur über den Widerstand umgeladen werden können). Die Lösung beim PCI-Bus: eine Einrichtung, die ein solches Signal treibt, darf sich nicht vom Bus trennen, sofern das Signal noch aktiv (Low) ist. Vielmehr muß es zunächst aktiv auf High getrieben werden - dann ist die Leitung, wenn sie hochohmig wird, bereits mit High belegt. Der Pull-up-Widerstand hat dann lediglich die Aufgabe, das Signal weiterhin auf diesem Pegel zu halten.

Parken durch aktive Ansteuerung

"Reine" Tri-State-Signale (Signaltyp TS) werden von einer Master-Einrichtung (Park Master) auf einem festen Pegel (Low oder High) gehalten. (Das betrifft die Signale AD31...0, C/BE3#...0# und PAR.)

Auswahl des Park Masters

Ein Park Master muß bestimmt werden, wenn sich der Bus im Ruhezustand befindet und keine Einrichtung die Busherrschaft anfordert. Hierzu ist folgendes festgelegt: eine Master-Einrichtung, die keine Busanforderung gestellt hat (d. h., deren REQ#-Ausgang inaktiv ist), wird durch Aktivieren ihres GNT#-Eingangs zum Park Master. Sie muß dann die betreffenden Signale (AD31...0, C/BE3#...0#, PAR) treiben. Die Auswahl des Park Masters ist den zentralen Steuerschaltungen freigestellt. Es sind 2 Fälle zu betrachten:

- eine Einrichtung ist bereits als Busmaster ausgewählt, stellt aber keine Busanforderung mehr. Dann wird diese Einrichtung zum Park Master (indem das betreffende GNT#-Signal aktiv gehalten wird).
- es ist gar kein Busmaster ausgewählt. Dann muß eine beliebige Einrichtung als Park Master bestimmt werden. Hierfür sind u. a. folgende Vorgehensweisen üblich:
 - die Host-PCI-Bridge (die typischerweise mit den zentralen Steuerschaltungen in einem Schaltkreis untergebracht ist; vgl. Abbildung 1.37 in Abschnitt 1.2.19.) wird zum Park Master.
 - diejenige Einrichtung wird als Park Master bestimmt, von der zu erwarten ist, daß sie demnächst wahrscheinlich wieder eine Master-Anforderung stellen wird.

Hinweise:

1. Ein Park Master kann zum "richtigen" Busmaster werden.
2. Zu Einzelheiten siehe Abschnitt 1.3.11.8.

1.2.2. CLK: Bustakt

PCI ist ein synchrones Bussystem. Alle anderen Bussignale mit Ausnahme von RST# und INTA#..INTD# werden auf die Low-High-Flanke des Bustaktes CLK bezogen.

Signaltyp: Eingang.

CLK wird von einem zentralen Taktgenerator erzeugt und an alle PCI-Einrichtungen geliefert.

1.2.2.1. Taktfrequenz und Signalverlauf

Es sind 2 Richtwerte spezifiziert: 33 MHz und 66 MHz. Die meisten PC-Konfigurationen sind 33-MHz-Systeme, ebenso die vom PCI-Bus abgeleiteten Bussysteme (LPC, Small PCI, CardBus usw.) . Die Auslegung für 66 MHz bedingt Einschränkungen hinsichtlich der zulässigen Leitungslängen und der Erweiterbarkeit des Bussystems (z. B. Beschränkung auf 4 Buslasten = 2 Schaltkreise auf dem Motherboard + 1 Slot). Das AGP-Interface (Kapitel 2) kann man als eine Ableitung vom 66-MHz-PCI-Bus ansehen. Abbildung 1.27 veranschaulicht den Verlauf des CLK-Signals, Tabelle 1.20 nennt die Zeitkennwerte.

Abbildung 1.27 Das CLK-Signal. a) 5-V-Signalisierung; b) 3,3-V-Signalisierung; 1 - Minimum-High 2,0 V (wie TTL); 2 - Minimum-High 0,4 V_{CC} (Quelle: PCI SIG)

Kennwert	Symbol	33 MHz		66 MHz	
		minimal	maximal	minimal	maximal
Taktzykluszeit	t_{CYC}	30 ns	$\infty^{1)}$	15 ns	30 ns ²⁾
Dauer der High-Phase	t_{high}	11 ns		6 ns	
Dauer der Low-Phase	t_{low}	11 ns		6 ns	
Anstiegsgeschwindigkeit (Slew Rate)	-	1 V/ns	4 V/ns	1,5 V/ns	4 V/ns

1), 2): siehe Erklärung im Text

Tabelle 1.20 Zeitkennwerte des CLK-Signals

Erklärung:

1. “ ∞ ” = statische Betriebsweise (“angehaltener Takt”, Taktfrequenz = 0 Hz),
2. $t_{\text{CYC}} \geq 30$ ns bedeutet, daß die Taktfrequenz ≤ 33 MHz ist. Dann gilt die 33-MHz-Spezifikation (somit müssen auch 66-MHz-Steckkarten statisch betreibbar sein).

Anhalten des Taktes; statische Betriebsweise

Aus der Spezifikation “ $t_{\text{CYC}} = \infty$ ” (Tabelle 1.19) ergibt sich die Forderung nach Betriebsfähigkeit bei einer Taktfrequenz von 0 Hz, d. h. bei “angehaltenem” Takt (statische Betriebsweise).

Einzelheiten:

- für PCI-Einrichtungen auf Steckkarten wird die statische Betriebsfähigkeit konsequent gefordert^{*)},
- eine PCI-Konfiguration darf auch mit einer geringeren Taktfrequenz als 33 oder 66 MHz betrieben werden,
- die Taktfrequenz darf während des Betriebes in den zulässigen Grenzen beliebig umgeschaltet werden (dabei dürfen aber keine zusätzlichen Taktflanken entstehen),
- bei jeder beliebigen Taktfrequenz (auch bei statischem Betrieb) sind die geforderte Flankenteilheit (Anstiegsgeschwindigkeit) sowie die Anforderungen an die Mindestdauer der High- bzw. Low-Phase (Tabelle 1.20) einzuhalten,
- der Takt darf nur in der Low-Phase angehalten werden,
- “planare” (auf dem Motherboard fest angeordnete) PCI-Einrichtungen müssen diese Forderungen nicht zwingend erfüllen (es ist hier zulässig, gewisse Mindest-Taktfrequenzen zu spezifizieren (kein statischer Betrieb möglich) und auch das Verändern der Taktfrequenz während des Betriebs auszuschließen).

^{*)}: die Forderung betrifft nur das PCI-Interface der Steckkarte, nicht die weiteren Funktionen. (Die Tatsache, daß die PCI-Spezifikation das Verändern bzw. Anhalten des Bustaktes erlaubt, sollte an sich Anlaß sein, die eigentlichen Funktionen der Steckkarte (zumindest, sofern sie zeitkritisch sind) nicht vom Bustakt abhängig zu machen. Es ist aber durchaus mit “Spar-Entwürfen” zu rechnen, die das CLK-Signal auch zu internen Taktierungszwecken ausnutzen. Eine solche Karte ist dann am PCI-Bus betriebsfähig, kann aber bei statischer Ansteuerung ihre eigentlichen Funktionen nicht erbringen.)

1.2.2.2. Zeitbezüge zu anderen Bussignalen

Maßgebend für die Signalspiele am PCI-Bus ist stets die Low-High-Flanke des Bustaktes: Ausgangsbelegungen erscheinen deshalb erst *nach* dem Auftreten der Taktflanke, Eingangsbelegungen werden mit der Taktflanke abgetastet, sie müssen also anliegen, *bevor* die Taktflanke wirksam wird (Abbildung 1.28, Tabelle 1.21).

Abbildung 1.28 Zeitbezüge von CLK zu anderen Bussignalen. a) Ausgangssignale; b) Eingangssignale; V - Signal gültig (valid); A - Signal aktiv; H - Signal hochohmig (Quelle: PCI SIG)

Erklärung:

- 1) Zeitverzögerung des Ausgangssignals. Dieses wird erst eine gewisse Zeit t_{val} nach der Taktflanke gültig (valid),
- 2) weitere Schaltzeiten von Tri-State-Signalen bei Zustandsänderung (zwischen aktiv (Low oder High) und hochohmig):
 - Verzögerungszeit bis zum Aktivieren (wenn infolge der Taktflanke das Signal aktiv wird): t_{on} ,
 - Verzögerungszeit bis zum Deaktivieren (wenn infolge der Taktflanke das Signal hochohmig wird): t_{off} (ein Ausgang gilt als hochohmig, wenn Ausgangsstrom \leq Leckstrom),
- 3) ein Eingangssignal muß um die Vorhaltezeit (Setup Time t_{su}) vor der Taktflanke stabil anliegen und so bis zum Ablauf der Haltezeit (Hold Time t_h) gehalten werden.

Kennwert	Symbol	33 MHz		66 MHz	
		minimal	maximal	minimal	maximal
Verzögerungszeit Takt zu gültiger Ausgangsbelegung (Bussignale)	t_{val}	2 ns	11 ns	2 ns	6 ns
Verzögerungszeit Takt zu gültiger Ausgangsbelegung (Punkt-zu-Punkt-Verbindungen) ¹⁾	$t_{val(ptp)}$	2 ns	12 ns	2 ns	6 ns
Verzögerungszeit hochohmig zu aktiv (Aktivierungszeit)	t_{on}	2 ns		2 ns	
Verzögerungszeit aktiv zu hochohmig (Deaktivierungszeit)	t_{off}		28 ns		14 ns
Eingangs-Vorhaltezeit (Bussignale)	t_{su}	7 ns		3 ns	
Eingangs-Vorhaltezeit (Punkt-zu-Punkt-Verbindungen) ¹⁾	$t_{su(ptp)}$	10 ns (GNT#), 12 ns (REQ#)		5 ns (GNT#, REQ#)	
Eingangs-Haltezeit ²⁾	t_h	0 ns		0 ns	

1), 2): siehe Erklärung im Text

Tabelle 1.21 Verzögerungs-, Vorhalte- und Haltezeiten

Erklärung:

- 1) Punkt-zu-Punkt-Verbindungen (Point-to-Point; ptp) sind hier lediglich die Signale REQ# und GNT#. Alle anderen Signale gelten in Bezug auf den Takt als Bussignale.
- 2) Haltezeit 0 bedeutet, daß die Eingangsbelegung zusammen mit der Taktflanke umschalten darf (eine in moderner Hardware oftmals übliche Spezifikation).

1.2.2.3. Taktversatz (Skew)

Auch die Ausbreitung des Taktsignals ist von Zeittoleranzen betroffen. Demzufolge wird eine Taktflanke an der einen PCI-Einrichtung eher eintreffen und an einer anderen später. Es ergibt sich somit ein Taktversatz (Clock Skew; Abbildung 1.29).

Abbildung 1.29 Taktversatz Clock Skew (Quelle: PCI SIG)

Zulässig sind:

- bei 33 MHz: t_{skew} maximal 2 ns,
- bei 66 MHz: t_{skew} maximal 1 ns.

1.2.2.4. Der Taktzyklus

Ein PCI-System können wir als ein vom Bustakt CLK gesteuertes Schaltwerk auffassen (Abbildung 1.30).

Abbildung 1.30 Der PCI-Bus als synchrones Schaltwerk

Erklärung:

1 - sendendes Register bzw. Flipflop; 2 - Ausgangs-Treiber; 3 - Schaltkreisanschluß (Pin); 4 - Eingangsstufe;

5 - empfangendes Register bzw. Flipflop. Die Abbildung soll veranschaulichen, woraus sich die minimale

Taktzykluszeit t_{CYC} ergibt:

- die Low-High-Flanke des Taktes trifft am sendenden Register (1) ein,
- nach Ablauf der Verzögerungszeit t_{val} erscheint das Signal am Schaltkreisausgang (3),
- das Signal muß sich über die Busleitung ausbreiten. Diese Buslaufzeit (Propagation Time t_{prop}) ergibt sich nach dem PCI-typischen Prinzip "einmal hin und zurück" (Round Trip Delay; vgl. Abschnitt 1.1.8.2.),
- am Eingang des empfangenden Schaltkreises (5) muß das Signal wenigstens für die Dauer der Mindest-Vorhaltezeit t_{su} anliegen, bevor die nächste Low-High-Taktflanke erscheinen darf,
- der Zeitversatz der Takte (Clock Skew t_{skew}) an beiden Registern (1, 5) ist zusätzlich einzurechnen.

Abbildung 1.31 veranschaulicht die Zeitspezifikationen für beide Taktfrequenzen (33 und 66 MHz).

Abbildung 1.31 Zeitanteile der Taktzyklen bei 33 und 66 MHz

Die Verzögerungszeit t_{val} und die Vorhaltezeit t_{su} sind Schaltkreis-Kennwerte, die Buslaufzeit t_{prop} und der Taktversatz t_{skew} ergeben sich hingegen aus den Leitungslängen der PCI-Konfiguration (beachten Sie die extrem knappen Vorgaben für 66 MHz).

1.2.2.5. Taktverteilung

Der geringe zulässige Taktversatz (Clock Skew) erfordert - vor allem in 66-MHz-Systemen - besondere Vorkehrungen, z. B. eine sternförmige Taktverteilung über Punkt-zu-Punkt-Verbindungen (Abbildung 1.32).

Abbildung 1.32 Taktverteilung in einem PCI-System

Erklärung:

1 - Taktgenerator; 2 - Takttreiberschaltkreis (Low Skew Clock Driver); 3 - PCI-Einrichtung auf dem Motherboard; 4 - Slot-Steckverbinder; 5 - Steckkarte; 6 - PCI-Einrichtung auf Steckkarte; 7 - Taktleitungen. Die Art der Leitungsführung soll andeuten, daß alle Taktleitungen nahezu gleich lang sind.

Hinweis:

Die Zeiten werden stets an den Anschlüssen der PCI-Schaltkreise gemessen (und nicht etwa an den Slots - auch der Takt-Leiterzug auf der Steckkarte ist bei der Leitungslänge zu berücksichtigen).

1.2.2.6. 66-MHz-Betrieb

Die PCI-Spezifikation (Ausgabe 2.1) regelt die Auf- und Abwärtskompatibilität:

1. 66-MHz-Einrichtungen können an einem 33-MHz-Bus betrieben werden,
2. ein Bussystem, das an sich für 66 MHz ausgelegt ist, wird auf 33 MHz zurückgeschaltet, sobald eine 33-MHz-Einrichtung angeschlossen ist.

Damit dies funktioniert, hat man verschiedene Maßnahmen vorgesehen:

- die für 66 MHz vorgesehenen PCI-Schaltkreise haben nahezu gleiche Zeitkennwerte wie die 33-MHz-Typen; es sind lediglich einige Spezifikationen "verschärft" worden,
- 66-MHz-Einrichtungen müssen auch gemäß der 33-MHz-Taktpezifikation funktionieren,
- im PCI-Zustandsregister (Konfigurationsadreibraum; vgl. Abschnitt 1.4.3.5.) wurde ein Steuerbit eingeführt, das anzeigt, ob die betreffende Einrichtung mit 66 MHz betrieben werden kann oder nicht (66 MHz Capable),
- der Taktgenerator eines 66-MHz-Systems muß auf 33 MHz umschaltbar sein,
- im Slot-Steckverbinder wurde das Signal M66EN eingeführt. Es ist auf dem Motherboard mit einem Pull-up-Widerstand beschaltet (gemäß Ausgabe 2.0 der PCI-Spezifikation war dieser Anschluß mit Masse belegt).

- das System hat 2 Möglichkeiten, die jeweils einzustellende Taktfrequenz zu erkennen: (1) softwareseitig durch Auswerten der Steuerbits 66 MHz Capable, (2) durch Abfragen des Signals M66EN:
 - 33-MHz-Karten ziehen M66EN auf Low,
 - 66-MHz-Karten fragen M66EN ab. Finden sie ein Low vor, so erkennen sie, daß sie mit 33 MHz betrieben werden (weil sie entweder in einem 33-MHz-System gemäß Ausgabe 2.0 stecken oder weil sich 33-MHz-Einrichtungen am selben Bus befinden).

1.2.3. RST#: Rücksetzen

Über die Leitung RST# wird allen Einrichtungen am PCI-Bus ein Rücksetzsignal zugeführt.

Signaltyp: Eingang.

RST# wird von den zentralen Steuerschaltungen erregt.

1.2.3.1. Auslösen des Rücksetzens

Das Rücksetzen ist auszulösen, sofern die Betriebsspannungen außerhalb der Spezifikation liegen. Des Weiteren kann es auf beliebige andere Weise "von außen" ausgelöst werden (z. B. über eine Rücksetzta-
ste).

Rücksetzen und Betriebsspannungen

Der PCI-Bus muß im Rücksetzen gehalten werden:

1. während des Einschaltens (so lange, bis alle Versorgungsspannungen "hochgefahren" sind; Stichwort: Power-Good-Bedingung),
2. beim Ausschalten (sobald die erste Versorgungsspannung ihren Minimalwert unterschreitet),
3. bei Stromversorgungsfehlern (Über- oder Unterspannung während des Betriebs).

Das Rücksetzen soll nicht nur für einen eindeutigen Anfangszustand sorgen (nach dem Einschalten), sondern auch dann, wenn die PCI-Einrichtungen nicht vollkommen betriebsfähig sind, vermeiden, daß 2 Fehlermechanismen wirksam werden: (1) Buskonflikte, (2) Kurzschlußströme durch Diodenstrukturen in den Ein- und Ausgangsstufen der Schaltkreise. (Solche Betriebsfälle treten sowohl beim Ein- als auch beim Ausschalten auf.)

Die PCI-Spezifikation nennt 2 Fehlerbedingungen der Stromversorgung. Für jede ist eine gewisse höchstzulässige Reaktionszeit t_{fail} spezifiziert:

Fehlerbedingung 1: eine beliebige Speisespannung unterschreitet ihren Minimalwert bzw. überschreitet ihren Maximalwert (gemäß den Tabellen 1.6., 1.7. und 1.10) um mehr als 500 mV. Maximale Reaktionszeit $t_{\text{fail}} = 500$ ns.

Fehlerbedingung 2: die 5-V-Speisespannung sinkt unter die aktuelle Spannung der 3,3-V-Versorgung um mehr als 300 mV. Maximale Reaktionszeit $t_{\text{fail}} = 100$ ns. (Dies ist ein kritischer Betriebsfall, in dem Kurzschluß-Stromwege über Diodenstrukturen bestehen. Ausgangsstufen, die nicht rechtzeitig deaktiviert werden, könnten somit zerstört werden.)

1.2.3.2. Rücksetzwirkung

Alle Bussignale sind in ihren inaktiven Zustand zu versetzen (im besonderen sind Tri-State-Signale hochohmig zu schalten und Open-Drain-Ausgänge zu deaktivieren).

Besonderheiten:

- REQ#, GNT# sind - als Tri-State-Signale - hochohmig zu schalten (sie werden über Pull-up-Widerstände hochgezogen),
- AD31...0, C/BE3#...0# und PAR sind während des Rücksetzens auf Low zu parken (Signale dürfen nicht High werden),
- die verbleibenden Signale werden über Pull-up-Widerstände auf High gezogen.
- REQ64# hat eine Sonderfunktion: ist die PCI-Konfiguration für den 64-Bit-Betrieb ausgelegt, so wird es von den zentralen Steuerschaltungen auf Low gehalten, während RST# von Low nach High wechselt (also beim Aufheben der Rücksetzwirkung). Damit wird PCI-Karten, die in einem 64-Bit-Slot stecken, angezeigt, daß das System den 64-Bit-Betrieb unterstützt (siehe weiterhin Abbildung 1.33 in Abschnitt 1.2.18.2.).

Hinweis:

In den folgenden Einzelbeschreibungen ist auch der jeweilige Rücksetzzustand angegeben.

1.2.3.3. Zeitvorgaben

RST# wirkt asynchron (ohne Zeitbezug zu CLK) und statisch (Rücksetzwirkung hält so lange an, wie RST# auf Low gehalten wird). Die Abläufe und Zeitkennwerte des Rücksetzens sind aus Abbildung 1.33 und Tabelle 1.22 ersichtlich.

Abbildung 1.33 Rücksetzabläufe (Quelle: PCI SIG)

- 1) Verlauf der Versorgungsspannungen,
- 2) Bezugspunkt: die letzte Versorgungsspannung überschreitet ihren Minimalwert ("Power Good"-Bedingung),
- 3) nach Eintreten der Power-Good-Bedingung muß RST # für wenigstens 1 ms aktiv sein (Tabelle 1.22),
- 4) neben den Versorgungsspannungen ist auch der Bustakt zu berücksichtigen: das Rücksetzen darf erst dann aufgehoben werden, wenn CLK stabil "eingeschwungen" ist (hier ist u. a. an PLL-Taktgeneratoren mit ihren längeren Einschwingzeiten zu denken). Zeitforderung (Tabelle 1.22): nach Einschwingen des Taktes RST# wenigstens noch 100 µs lang aktiv halten.
- 5) eine Rücksetzzeit von 100 ms ist ein praxisüblicher Wert,
- 6) Stromversorgungsfehler (oder Ausschalten); RST# wird nach Reaktionszeit t_{fail} aktiv,
- 7) Schalten von REQ# in Bezug auf die Low-High-Flanke von RST#,
- 8) Deaktivieren der Bussignale nach Aktivieren von RST# (mit einer maximalen Verzögerung von 40 ns),
- 9) Ende des Rücksetzens (Low-High-Flanke von RST#). RST# darf zwar asynchron zu CLK inaktiv werden, es muß sich aber um eine "saubere" Signalfanke handeln (prellfrei, PCI-typische Anstiegszeit).

Kennwert	Symbol	minimal	maximal
Reaktionszeit von Stromversorgungsfehler bis zum Aktivieren von RST# ¹⁾	t_{fail}		500 ns, 100 ns
Rücksetzdauer nach Anliegen der Speisespannung	t_{rst}	1 ms	
Rücksetzdauer nach Einschwingen des Bustaktes	$t_{\text{rst-clk}}$	100 μs	
Deaktivierungszeit der Ausgänge nach Aktivierung von RST#	$t_{\text{rst-off}}$		40 ns ²⁾
Vorhaltezeit von REQ64# vor Deaktivierung von RST#	t_{rrsu}	10 t_{CYC} ³⁾	
Haltezeit von REQ64# nach Deaktivierung von RST#	t_{rrh}	0 ns	50 ns ⁴⁾
Wartezeit bis zum 1. Buszugriff ⁵⁾	t_{rhff}	5 t_{CYC}	
Wartezeit bis zum 1. Konfigurationszugriff ⁵⁾	t_{rhfa}	$2^{25} t_{\text{CYC}}$	

1)...5): siehe Erklärung im Text

Tabelle 1.22 Zeitkennwerte des Rücksetzens

Erklärung:

- 1) siehe Abschnitt 1.2.3.1.
- 2) unabhängig vom Takt (RST# muß den Bus asynchron deaktivieren),
- 3) bei 33 MHz: 300 ns, bei 66 MHz: 150 ns,
- 4) REQ64# muß spätestens 50 ns nach dem Aufheben des Rücksetzens wieder deaktiviert werden,
- 5) Kennwerte gemäß Ausgabe 2.2 der PCI-Spezifikation.

Wartezeiten (Ausgabe 2.2 der PCI-Spezifikation)

PCI-Einrichtungen sind meistens recht komplizierte Gebilde, die nach Aufheben des Rücksetzens intern noch einiges zu tun haben, um voll betriebsfähig zu werden (hier ist z. B. an Initialisierungsvorgänge zu denken, die von internen Controllern bzw. Prozessoren gesteuert werden). Bisher gab es keine Vorgaben für das Verhalten des PCI-Bus in diesem Zeitabschnitt. Dies wird mit Ausgabe 2.2 abgestellt. Man hat 2 Zeitkennwerte definiert:

t_{rhff} - Wartezeit bis zum 1. Buszyklus

Bis zum Ablauf der t_{rhff} -Zeitvorgabe (d. h. während der ersten 5 Taktzyklen nach Aufheben des Rücksetzens) wird der Bus gleichsam in Ruhe gelassen; es finden keine Zugriffe statt. Genauer gesagt: es findet keine Master-Vermittlung (Arbitrierung) statt, und das FRAME-Signal wird in diesem Zeitintervall garantiert nicht erregt, die Einrichtungen bekommen also weder die Busherrschaft noch werden sie als Target angesprochen

t_{rhfa} - Wartezeit bis zum 1. Konfigurationszugriff

In dieser Zeit sollten wirklich alle Einrichtungen soweit sein, PCI-Zugriffe auszuführen (die spezifizierten $2^{25} \approx 32$ Millionen Taktzyklen entsprechen bei 33 MHz rund 1 Sekunde).

Diese Festlegungen beruhen auf der üblichen Reihenfolge, in der ein Computer-System seine Arbeit aufnimmt (Abbildung 1.34).

Abbildung 1.34 Abläufe nach dem Rücksetzen*Erklärung:*

- 1) vor Ablauf von t_{rhff} ist der Bus vollkommen in Ruhe. (Die spezifizierten 5 Taktzyklen genügen typischerweise, die State Machines der Bussteuerung soweit zu initialisieren, daß sie Target-Zugriffe steuern können.)
- 2) nach Ablauf von t_{rhff} sind die Einrichtungen für Target-Zugriffe zugänglich. Somit kann auch auf den Speicher zugegriffen werden, der die Kaltstart-Programme enthält (Programmstart vom Boot ROM).
- 3) die Software beginnt, das System zu initialisieren,
- 4) nach Ablauf von t_{rhfa} sind die Einrichtungen am PCI-Bus voll betriebsfähig, und die Software ist soweit, daß sie mit den Konfigurationszugriffen beginnen kann.

Es ist nicht vorgeschrieben, die Zeitintervalle unbedingt einzuhalten. Einrichtungen, die am Bus noch nicht betriebsfähig sind, müssen aber Buszugriffe auf reguläre Art abweisen (Master Abort oder Retry Termination; vgl. Abschnitt 1.3.5.).

Ausnahme: eine PCI-Einrichtung, die auf dem Zugriffsweg zum Boot ROM liegt (z. B. eine PCI-to - Legacy Bridge; vgl. Abschnitt 1.5.) bzw. die diesen Speicher selbst enthält, *muß* nach Ablauf von t_{rhff} für Target-Zugriffe betriebsfähig sein.

Ist bekannt, daß alle Einrichtungen bereits vor Ablauf von t_{rhfa} für Konfigurationszugriffe betriebsfähig sind, so darf die Software entsprechend eher mit dem Konfigurieren beginnen.

1.2.4. Adressen und Daten

1.2.4.1. AD31...0

Diese 32 Signale werden zeitmultiplex zum Übertragen von Adressen und Daten verwendet.

Signaltyp: Tri-State-Busleitung, bidirektional (TS). *Ohne* Pull-up-Widerstand.

Rücksetzzustand: die Signale werden von den zentralen Steuerschaltungen (in der Praxis: typischerweise vom Schaltkreis, der die Host-to-PCI Bridge enthält) auf Low getrieben.

Parken: die Signale werden vom jeweiligen Park Master (vgl. Abschnitt 1.2.1.3.) auf feste Pegel (Low oder High) getrieben. Wirken die zentralen Steuerschaltungen als Park Master, so werden sie auf Low getrieben.

Ruhezustand: die Signale werden vom jeweiligen Master auf stabile Pegel (Low oder High) getrieben.

Adreßübertragung: in der Adreßphase (im selben Takt, in dem FRAME# aktiviert wird) legt der Master die 32-Bit-Zugriffsadresse auf den Bus. Zu den Adreßbelegungen siehe Abbildung 1.22.

Datenübertragung beim Schreiben: Master aktiviert IRDY# und legt gleichzeitig Daten auf den Bus. Target übernimmt Daten vom Bus, wenn IRDY# und TRDY# aktiv sind.

Datenübertragung beim Lesen: Target aktiviert TRDY# und legt gleichzeitig Daten auf den Bus. Master übernimmt Daten vom Bus, wenn IRDY# und TRDY# aktiv sind.

Unterbrechungsbestätigung (Abschnitt 1.3.13.3.): AD31...0 sind mit dem Interruptvektor belegt.

Sonderzyklen (Abschnitt 1.3.13.4.):

- AD15...0: Nachrichtencode,
- AD31...16: beliebige stabile Belegung oder zusätzliches Datenfeld, das die Nachricht näher kennzeichnet.

Gültigkeit der Datenbelegung: diese wird byteweise durch die Byteauswahlsignale C/BE3#...0# gesteuert. Ein Datenbyte ist gültig (valid), wenn das zugehörige Erlaubnissignal aktiv (Low) ist.

Bei bestimmten Speicherzugriffen sind - unabhängig von der C/BE-Belegung - *alle* Datenleitungen gültig (Abschnitt 1.3.12.).

Ungültige Datenbytes: diese müssen während der Datenübertragung mit stabilen Werten belegt werden (beim Schreiben: vom Master, beim Lesen: vom Target).

Bit- und Byteanordnung: siehe Abbildungen 1.22 (Adreßbelegungen) und 1.25 (Datenbelegung).

1.2.4.2. 64-Bit-Signale: AD63...32

Diese 32 Signale werden zeitmultiplex zum Übertragen von Adressen und Daten verwendet. Sie sind Teil der 64-Bit-Erweiterung des PCI-Bus und werden nur dann genutzt, wenn 64-Bit-Datenübertragungen stattfinden.

Signaltyp: Tri-State-Busleitung, bidirektional (TS). Mit Pull-up-Widerstand beschaltet.

Rücksetzzustand, Parken: die Signale werden über die Pull-up-Widerstände auf High gehalten.

Belegung in 32-Bit-Buszyklen

Die Signale werden nicht genutzt und über die Pull-up-Widerstände auf High gehalten.

Hinweis:

Zu 64-Bit-Karten in 32-Bit-Slots siehe Abschnitte 1.2.18.4. und 1.3.9.

Belegung in 64-Bit-Buszyklen

Ruhezustand: die Signale werden über die Pull-up-Widerstände auf High gehalten. Alternativ dazu kann der Master die Signale auf High treiben.

Adreßübertragung bei 32-Bit-Adressierung: die Signale werden über die Pull-up-Widerstände auf High gehalten oder vom Master mit stabilen Werten belegt.

Adreßübertragung bei 64-Bit-Adressierung: 64-Bit-Adressen werden in Dual Address Cycles übertragen. Während eines solchen Buszyklus legt der Master die höherwertigen 32 Adreßbits (63...32) auf die Leitungen AD63...32. Siehe weiterhin Abschnitt 1.3.9.

Hinweis:

Die 64-Bit-Adressierung wird von 64-Bit-Mastern durch Erregen von REQ64# angezeigt.

Datenübertragung beim Schreiben: Master aktiviert IRDY# und legt gleichzeitig die höchstwertigen 32 Datenbits (63...32) auf den Bus. Target übernimmt Daten vom Bus, wenn IRDY# und TRDY# aktiv sind.

Datenübertragung beim Lesen: Target aktiviert TRDY# und legt gleichzeitig die höchstwertigen 32 Datenbits (63...32) auf den Bus. Master übernimmt Daten vom Bus, wenn IRDY# und TRDY# aktiv sind.

Hinweise:

1. Die 64-Bit-Datenübertragung ist nur bei Speicherzugriffen von Bedeutung; die Datenzugriffe in allen anderen Buszyklen sind 32-Bit-Zugriffe.
2. Eine 64-Bit-Datenübertragung kommt nur dann zustande, wenn ein 64-Bit-Master mit einem 64-Bit-Target zusammenarbeitet, d. h., wenn die Aktivierung von REQ64# vom Target durch Aktivierung von ACK64# bestätigt wird.
3. Wird eine vom Master begonnene 64-Bit-Übertragung nicht bestätigt (ACK64# bleibt inaktiv), so wird die Datenübertragung mit 32-Bit-Zugriffen weitergeführt.
4. Wenn der Master einen 64-Bit-Schreibzugriff startet, so belegt er zunächst AD63...32 mit den entsprechenden Daten. Wird die 64-Bit-Übertragung vom Target nicht bestätigt, so muß der Master die Signale auf stabilen Pegeln halten oder auf High treiben und dann freigeben (nachfolgend wird die High-Belegung über die Pull-up-Widerstände gehalten).

Gültigkeit der Datenbelegung: diese wird byteweise durch die Byteauswahlsignale C/BE7#...4# gesteuert. Ein Datenbyte ist gültig (valid), wenn das zugehörige Erlaubnissignal aktiv (Low) ist.

Bei bestimmten Speicherzugriffen sind - unabhängig von der C/BE-Belegung - *alle* Datenleitungen gültig (Abschnitt 1.3.12.).

Ungültige Datenbytes: infolge der Beschaltung mit Pull-up-Widerständen ist es nicht erforderlich, die entsprechenden AD-Signale mit stabilen Werten zu belegen.

Bit- und Byteanordnung: siehe Abbildungen 1.22 (Adreßbelegungen) und 1.25 (Datenbelegung).

1.2.5. Kommandos und Byteauswahl

1.2.5.1. C/BE3#...0#

Diese 4 Signale werden zeitmultiplex zum Übertragen von Kommandocodes und von Gültigkeitsangaben für Datenbytes (Byte Enables) verwendet.

Signaltyp: Tri-State-Busleitung, bidirektional (TS). *Ohne* Pull-up-Widerstand.

Rücksetzzustand: die Signale werden von den zentralen Steuerschaltungen (in der Praxis: typischerweise vom Schaltkreis, der die Host-to-PCI Bridge enthält) auf Low getrieben.

Parken: die Signale werden vom jeweiligen Park Master (vgl. Abschnitt 1.2.3.1.) auf feste Pegel (Low oder High) getrieben. Wirken die zentralen Steuerschaltungen als Park Master, so werden sie auf Low getrieben.

Ruhezustand: die Signale werden vom jeweiligen Master auf stabile Pegel (Low oder High) getrieben.

Adreßübertragung: in der Adreßphase (im selben Takt, in dem FRAME# aktiviert wird) legt der Master den Kommandocode auf den Bus. Zu den Kommandocodes siehe Tabelle 1.17.

Datenübertragung: die Signale werden vom Master getrieben, um die Gültigkeit der Datenbytes anzuzeigen (Beispiel: C/BE3# = Low: Datenbyte 3 (Bits 31...24) ist gültig, C/BE3# = High: Datenbyte 3 ist ungültig). Ungültige Datenbytes werden nicht vom Bus genommen (d. h. sie werden nicht geschrieben bzw. gelesen).

Gültigkeitsangabe bei Speicherzugriffen: die zuvor übertragene Adresse betrifft das Doppelwort (Auswertung der Adreßbits 31...2), die C/BE-Signale wählen die Bytes im Doppelwort aus.

Gültigkeitsangabe bei E-A-Zugriffen: die zuvor übertragene Adresse wirkt byteselektiv (es werden alle Adreßbits 31...0 ausgewertet). Die Erregung der C/BE-Signale muß der Byteauswahl im Doppelwort, d. h. der Belegung der Adreßbits 1, 0 entsprechen. Beispiel: Belegung der Adreßbits 1, 0 = 01. Damit ist Byte 1 im Doppelwort ausgewählt. Demzufolge darf C/BE0# nicht aktiv werden. (Vgl. auch Tabelle 1.15.)

Unterbrechungsbestätigung (Abschnitt 1.3.13.3.):

- Adreßphase: Kommandocode 0H (Interrupt Acknowledge Cycle),
- Datenphase: Gültigkeit der Datenbytes, die mit dem Interruptvektor belegt sind.

Sonderzyklen (Abschnitt 1.3.13.4.): Kommandocode 1H (Special Cycle).

Bit- und Byteanordnung: siehe Abbildungen 1.22 (Adreßbelegungen) und 1.25 (Datenbelegung).

1.2.5.2. C/BE7#...4#

Diese 4 Signale werden zeitmultiplex zum Übertragen von Kommandocodes und von Gültigkeitsangaben für Datenbytes (Byte Enables) verwendet. Sie sind Teil der 64-Bit-Erweiterung des PCI-Bus und werden nur dann genutzt, wenn 64-Bit-Datenübertragungen stattfinden.

Signaltyp: Tri-State-Busleitung, bidirektional (TS). Mit Pull-up-Widerstand beschaltet.

Rücksetzzustand, Parken: die Signale werden über die Pull-up-Widerstände auf High gehalten.

Belegung in 32-Bit-Buszyklen

Die Signale werden nicht genutzt und über die Pull-up-Widerstände auf High gehalten.

Hinweis:

Zu 64-Bit-Karten in 32-Bit-Slots siehe Abschnitte 1.2.18.4. und 1.3.9.

Belegung in 64-Bit-Buszyklen

Ruhezustand: die Signale werden über die Pull-up-Widerstände auf High gehalten. Alternativ dazu kann der Master die Signale auf High treiben.

Adreßübertragung bei 32-Bit-Adressierung: die Signale werden über die Pull-up-Widerstände auf High gehalten oder vom Master mit stabilen Werten belegt.

Adreßübertragung bei 64-Bit-Adressierung: 64-Bit-Adressen werden in Dual Address Cycles übertragen. Während eines solchen Buszyklus legt der Master den Kommandocode (Tabelle 1.17) auch auf die Leitungen C/BE7#...4#.

Hinweis:

Die 64-Bit-Adressierung wird von 64-Bit-Mastern durch Erregen von REQ64# angezeigt.

Datenübertragung beim Schreiben: die Signale werden vom Master getrieben, um die Gültigkeit der Datenbytes anzuzeigen (Beispiel: C/BE6# = Low: Datenbyte 6 (Bits 55...48) ist gültig, C/BE6# = High: Datenbyte 6 ist ungültig). Ungültige Datenbytes werden nicht vom Bus genommen (d. h. sie werden nicht geschrieben bzw. gelesen).

Hinweise:

1. Die 64-Bit-Datenübertragung ist nur bei Speicherzugriffen von Bedeutung; die Datenzugriffe in allen anderen Buszyklen sind 32-Bit-Zugriffe.
2. Eine 64-Bit-Datenübertragung kommt nur dann zustande, wenn ein 64-Bit-Master mit einem 64-Bit-Target zusammenarbeitet, d. h., wenn die Aktivierung von REQ64# vom Target durch Aktivierung von ACK64# bestätigt wird.
3. Wird eine vom Master begonnene 64-Bit-Übertragung nicht bestätigt (ACK64# bleibt inaktiv), so wird die Datenübertragung mit 32-Bit-Zugriffen weitergeführt.
4. Wenn der Master einen 64-Bit-Schreibzugriff startet, so belegt er zunächst C/BE7#...4# mit den entsprechenden Gültigkeitsangaben. Wird die 64-Bit-Übertragung vom Target nicht bestätigt, so muß der Master die Signale auf stabilen Pegeln halten oder auf High treiben und dann freigeben (nachfolgend wird die High-Belegung über die Pull-up-Widerstände gehalten).

Bit- und Byteanordnung: siehe Abbildungen 1.22 (Adreßbelegungen) und 1.25 (Datenbelegung).

1.2.6. Paritätskontrolle**1.2.6.1. PAR**

Dieses Signal dient zur Paritätskontrolle. Es ergänzt die Belegung der Signale AD31...0 und C/BE3#...0# auf gerade Parität (die Anzahl der Einsen auf AD31...0 + C/BE3#...0# + PAR muß gerade sein, sonst Fehler).

Signaltyp: Tri-State-Busleitung, bidirektional (TS). *Ohne* Pull-up-Widerstand.

Rücksetzzustand: PAR wird von den zentralen Steuerschaltungen (in der Praxis: typischerweise vom Schaltkreis, der die Host-to-PCI Bridge enthält) auf Low getrieben.

Parken: PAR wird vom jeweiligen Park Master (vgl. Abschnitt 1.2.1.3.) auf einen festen Pegel (Low oder High) getrieben. Wirken die zentralen Steuerschaltungen als Park Master, so wird PAR auf Low getrieben.

Nacheilende Wirkung: Die entsprechende Belegung von PAR eilt der Belegung der AD- und C/BE-Leitungen um einen Taktzyklus nach (d. h., die PAR-Belegung erscheint einen Taktzyklus später auf dem Bus).

Erregung von PAR:

PAR wird vom *Master* in folgenden Buszyklen erregt:

- in der Adreßphase,
- im Ruhezustand^{*)},
- in der Datenphase beim Schreiben,
- in Sonderzyklen.

^{*)}: im Ruhezustand wird PAR vom Master auf einen stabilen Pegel (Low oder High) getrieben. Dabei ist aber nicht gefordert, die Parität (d. h. die Anzahl der Einsen auf den AD- und C/BE-Leitungen) zu berücksichtigen.

PAR wird vom *Target* in folgenden Buszyklen erregt:

- in der Datenphase beim Lesen,
- in Unterbrechungsbestätigungszyklen.

Hinweis:

Eine Paritätsprüfung, die “über alles” wirkt, hat eine Problemstelle: die byteselektiven Zugriffe. Beim PCI-Bus hat man das Problem folgendermaßen “entschärft”:

- das Paritätsbit dient lediglich zur Überwachung der Datenübertragung. An den PCI-Bus angeschlossene Speicher müßten hingegen auf die Paritätsprüfung verzichten, eine byteweise Paritätsprüfung haben oder byteselektive Zugriffe mit internen Steuermitteln (so daß der PCI-Bus davon nichts merkt) als Read-Modify-Write-Zugriffe ausführen.
- damit die Paritätsprüfung “über alles” funktioniert, wird Wert darauf gelegt, ungenutzte Signale auf stabile Pegel zu treiben. Damit haben alle zu kontrollierenden Signale definierte Belegungen, über die das Paritätsbit gebildet wird.

1.2.6.2. PAR64

Dieses Signal dient zur Paritätskontrolle. Es ergänzt die Belegung der Signale AD63...32 und C/BE7#...4# auf gerade Parität (die Anzahl der Einsen auf AD63...32 + C/BE7#...4# + PAR64 muß gerade sein, sonst Fehler). Es ist Teil der 64-Bit-Erweiterung des PCI-Bus und wird nur dann genutzt, wenn 64-Bit-Datenübertragungen stattfinden.

Signaltyp: Tri-State-Busleitung, bidirektional (TS). Mit Pull-up-Widerstand beschaltet.

Rücksetzzustand, Parken: PAR64 wird über den Pull-up-Widerstand auf High gehalten.

Belegung in 32-Bit-Buszyklen

Das Signal wird nicht genutzt und über den Pull-up-Widerstand auf High gehalten.

Hinweis:

Zu 64-Bit-Karten in 32-Bit-Slots siehe Abschnitte 1.2.18.4. und 1.3.9.

Belegung in 64-Bit-Buszyklen

Adreßübertragung bei 32-Bit-Adressierung: das Signal wird über den Pull-up-Widerstand auf High gehalten oder vom Master mit einem stabilen Wert belegt.

Erregung von PAR64:

PAR64 wird vom *Master* in folgenden Buszyklen erregt:

- in der Adreßphase bei 64-Bit-Adressierung¹⁾,
- im Ruhezustand²⁾,
- in der Datenphase beim Schreiben³⁾.

PAR64 wird vom *Target* in der Datenphase beim Lesen erregt.

- 1) 64-Bit-Adressen werden in Dual Address Cycles übertragen. Die 64-Bit-Adressierung wird von 64-Bit-Mastern durch Erregen von REQ64# angezeigt.
- 2) das Signal wird über den Pull-up-Widerstand auf High gehalten. Alternativ dazu kann der Master das Signal auf High treiben.
- 3) Siehe die folgenden Hinweise.

Nacheilende Wirkung: Die entsprechende Belegung von PAR64 eilt der Belegung der Leitungen AD63...32 und C/BE7#...4# um einen Taktzyklus nach (d. h., die PAR64-Belegung erscheint einen Taktzyklus später auf dem Bus).

Hinweise:

1. Eine 64-Bit-Datenübertragung kommt nur dann zustande, wenn ein 64-Bit-Master mit einem 64-Bit-Target zusammenarbeitet, d. h., wenn die Aktivierung von REQ64# vom Target durch Aktivierung von ACK64# bestätigt wird. PAR64 führt bei Datenübertragungen nur dann ein gültiges Paritätsbit, wenn sowohl REQ64# als auch ACK64# aktiv sind.
2. Wird eine vom Master begonnene 64-Bit-Übertragung nicht bestätigt (ACK64# bleibt inaktiv), so wird die Datenübertragung mit 32-Bit-Zugriffen weitergeführt.
3. Wenn der Master einen 64-Bit-Schreibzugriff startet, so belegt er zunächst PAR64 mit einem korrekten Paritätsbit. Wird die 64-Bit-Übertragung vom Target nicht bestätigt, so muß der Master PAR64 auf einem stabilen Pegel halten oder auf High treiben und dann freigeben (nachfolgend wird die High-Belegung über den Pull-up-Widerstand gehalten).

1.2.7. Zyklussteuerung

1.2.7.1. FRAME#

Das Signal kennzeichnet den Beginn eines Buszyklus..

Signaltyp: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS).

Rücksetzzustand, Parken, Ruhezustand: FRAME# wird über den Pull-up-Widerstand auf High gehalten.

Beginn des Buszyklus (Adreßphase): der Master aktiviert FRAME# und legt gleichzeitig die Target-Adresse auf die AD-Leitungen sowie den Kommandocode auf die C/BE-Leitungen.

Belegung in der Datenphase: der Master nutzt FRAME# zusammen mit IRDY#, um die Art der Datenübertragung zu signalisieren (Tabelle 1.23).

Schaltfolge	Datenübertragung
gleichzeitig werden FRAME# inaktiv und IRDY# erstmalig aktiv	Einzelübertragung (Single Access Cycle)
FRAME# wird nach der ersten Aktivierung von IRDY inaktiv	Mehrfachübertragung (Burst Access Cycle)

Tabelle 1.23 Datenübertragungssteuerung über FRAME# und IRDY#

Mindestdauer der Aktivierung:

- 2 Taktperioden: bei 64-Bit-Adressierung (d. h. in einem Dual Address Cycle),
- 1 Taktperiode: sonst.

1.2.7.2. TRDY#

Das Signal kennzeichnet, daß das Target bereit ist, den Buszyklus bzw. die Datenübertragung zu beenden (Target Ready).

Signaltyp: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS).

Rücksetzzustand, Parken, Ruhezustand: TRDY# wird über den Pull-up-Widerstand auf High gehalten.

Beginn des Buszyklus (Adreßphase): TRDY# wird über den Pull-up-Widerstand auf High gehalten.

Belegung in der Datenphase: das Target aktiviert TRDY#, um zu signalisieren, daß die Datenübertragung (Buszyklus oder Mikrozugriff) beendet werden kann:

- Lesen: TRDY# zeigt an, daß die Lesedaten auf den AD-Leitungen liegen,
- Schreiben: TRDY# zeigt an, daß das Target bereit ist, Schreibdaten von den AD-Leitungen zu übernehmen,

- Unterbrechungsbestätigung: TRDY# zeigt an, daß der Interruptvektor auf den AD-Leitungen liegt.

Hinweise:

1. Ein Buszyklus kann erst dann tatsächlich beendet werden, wenn TRDY# und IRDY# gleichzeitig aktiv sind.
2. Beendet das Target den Buszyklus durch Erregen von STOP#, so wird TRDY# verwendet, um die jeweilige Endebedingung zu kennzeichnen (vgl. Tabelle 1.24 in Abschnitt 1.2.7.3.).

Belegung in Sonderzyklen (Special Cycles): TRDY# wird über den Pull-up-Widerstand auf High gehalten.

1.2.7.3. IRDY#

Das Signal kennzeichnet, daß der Master bereit ist, den Buszyklus bzw. die Datenübertragung zu beenden (Initiator Ready).

Signaltyp: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS).

Rücksetzzustand, Parken, Ruhezustand: IRDY# wird über den Pull-up-Widerstand auf High gehalten.

Beginn des Buszyklus (Adreßphase): IRDY# wird über den Pull-up-Widerstand auf High gehalten.

Belegung in der Datenphase: der Master aktiviert IRDY#, um zu signalisieren, daß die Datenübertragung (Buszyklus oder Mikrozugriff) beendet werden kann:

- Lesen: IRDY# zeigt an, daß der Master bereit ist, Lesedaten von den AD-Leitungen zu übernehmen,
- Schreiben: IRDY# zeigt an, daß die Schreibdaten auf den AD-Leitungen liegen,
- Unterbrechungsbestätigung: IRDY# zeigt an, daß der Master bereit ist, den Interruptvektor von den AD-Leitungen zu übernehmen.

Hinweis:

Ein Buszyklus ist erst dann tatsächlich beendet, wenn IRDY# und TRDY# gleichzeitig aktiv sind.

Belegung in Sonderzyklen (Special Cycles): IRDY# wird über den Pull-up-Widerstand auf High gehalten.

1.2.7.4. STOP#

Mit diesem Signal zeigt das Target dem Master an, daß der aktuelle Buszyklus sofort zu beenden ist (Target Initiated Termination).

Signaltyp: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS).

Rücksetzzustand, Parken, Ruhezustand: STOP# wird über den Pull-up-Widerstand auf High gehalten.

Beginn des Buszyklus (Adreßphase): STOP# wird über den Pull-up-Widerstand auf High gehalten.

Belegung in der Datenphase: das Target aktiviert STOP#, um zu signalisieren, daß der Buszyklus sofort zu beenden ist. Es sind 3 verschiedene Endebedingungen definiert, die durch STOP# zusammen mit TRDY# und DEVSEL# gekennzeichnet werden (Tabelle 1.24).

Endebedingung	Schaltfolge
Wiederanlauf (Retry)	in der ersten Datenphase wird STOP# aktiviert. Dabei bleibt TRDY# inaktiv. Es findet keine Datenübertragung statt
Trennen (Disconnect)	<ul style="list-style-type: none"> ▪ Trennen mit Zu-Ende-Führen der aktuellen Datenübertragung: STOP# und TRDY# werden gleichzeitig aktiviert ▪ Trennen ohne Datenübertragung: STOP# wird aktiviert. Dabei ist TRDY# inaktiv
Abbruch (Target Abort)	gleichzeitig werden STOP# aktiviert und DEVSEL# deaktiviert

Tabelle 1.24 Vom Target signalisierbare Endebedingungen (Target Initiated Termination)

Belegung in Sonderzyklen (Special Cycles): STOP# wird über den Pull-up-Widerstand auf High gehalten.

1.2.7.5. DEVSEL#

Mit diesem Signal zeigt die betreffende Einrichtung an, daß sie als Target am Buszyklus teilnehmen wird (Device Select). Grundlage hierfür ist die Decodierung der auf dem Bus liegenden Adresse (unter Berücksichtigung des Kommandocodes).

Signaltyp: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS).

Rücksetzzustand, Parken, Ruhezustand: DEVSEL# wird über den Pull-up-Widerstand auf High gehalten.

Beginn des Buszyklus (Adreßphase): DEVSEL# wird über den Pull-up-Widerstand auf High gehalten.

Belegung in der Datenphase: die betreffende Einrichtung aktiviert DEVSEL#, um zu signalisieren, daß sie die in der Adreßphase auf dem Bus liegende Adresse erkannt hat und als Target am Buszyklus teilnehmen wird. DEVSEL# bleibt bis zum Ende der Datenphase aktiv.

Hinweise:

1. DEVSEL# darf nicht später erregt werden als TRDY#, STOP# oder PERR#.
2. Um eine 64-Bit-Datenübertragung zu bestätigen, muß das Target DEVSEL# und ACK64 # gleichzeitig aktivieren.
3. Beendet das Target den Buszyklus durch Erregen von STOP#, so kennzeichnet das Deaktivieren von DEVSEL# eine Abbruchbedingung (Target Abort; vgl. Tabelle 1.24 in Abschnitt 1.2.7.4.).

4. Ist eine Einrichtung angeschlossen, die für subtraktive Adreßdecodierung ausgelegt ist (z. B. eine PCI-to PCI Bridge), so muß diese DEVSEL# beobachten. Wird DEVSEL# nicht erregt, so ist diese Einrichtung als Target ausgewählt (sie aktiviert dann ihrerseits DEVSEL#, um sich als Target zu erkennen zu geben).
5. Der Master beobachtet DEVSEL#. Wird DEVSEL# nach einer Adreßphase gar nicht erregt, so wird eine Abbruchbedingung (Master Abort) wirksam.

Belegung in Sonderzyklen (Special Cycles): DEVSEL# wird über den Pull-up-Widerstand auf High gehalten.

1.2.7.6. IDSEL

Das Signal dient zur Auswahl des Konfigurationsadreßraums der entsprechenden Einrichtung (“geographische Adressierung”).

Signaltyp: Eingang. *Ohne* Pull-up-Widerstand.

Rücksetzzustand: IDSEL wird von den zentralen Steuerschaltungen mit einem stabilen Wert (High oder Low) belegt.

Parken: IDSEL wird von den zentralen Steuerschaltungen bzw. vom jeweiligen Park Master mit einem stabilen Wert (High oder Low) belegt.

Adreßübertragung: in der Adreßphase ist IDSEL lediglich unter folgenden Bedingungen von Bedeutung:

1. Zugriff auf den Konfigurationsadreßraum (Kommandos: Konfiguration lesen oder Konfiguration schreiben),
2. die Konfigurationsadresse (auf AD31...0) ist vom Typ 0 (Belegung der Signale AD1, 0 = 00; vgl. Abbildung 1.22c in Abschnitt 1.1.9.2.).

Belegung in allen anderen Busphasen: IDSEL ist bedeutungslos. Die Belegung entspricht typischerweise jener der zwecks geographischer Adressierung angeschlossenen AD-Leitung. Andernfalls wird IDSEL vom Master bzw. von den zentralen Steuerschaltungen mit einem stabilen Wert (High oder Low) belegt.

Erregung von IDSEL: jede Einrichtung hat einen IDSEL-Eingang, der gesondert zu erregen ist. Die PCI-Spezifikation schreibt keine Schaltungslösung vor. So könnte man die IDSEL-Eingänge aller Einrichtungen an ein zentral angeordnetes Steuerregister anschließen, das vor dem jeweiligen Zugriff vom Prozessor aus entsprechend geladen wird.

Vorzugslösung: Nutzung der Adreßleitungen AD31...11. Hierbei ist der IDSEL-Eingang jeder Einrichtung an jeweils eine AD-Leitung angeschlossen (z. B. (wie in vielen PCs üblich) Einrichtung 1 an AD16, Einrichtung 2 an AD17 usw.). Siehe weiterhin Abschnitt 1.4.1. (vgl. aber auch Abschnitt 1.2.19.).

Hinweis:

Das Anschließen von IDSEL an eine AD-Leitung stellt eine zusätzliche Belastung des jeweiligen AD-Signals dar. Die typische Abhilfe: Zwischenschaltung eines Serienwiderstandes (Richtwert: 100 Ω ; vgl. die Abbildungen 1.39 und 1.40 in Abschnitt 1.2.19.). Da dies die IDSEL-Erregung verlangsamt (RC-Glied aus Widerstand und parasitären Kapazitäten verschleift die Signalfanken), muß bei Konfigurationszugriffen die Adresse voreilend aufgegeben werden (Address Pre-Driving; vgl. Abschnitt 1.3.4.1., Abbildung 1.50). Gelegentlich verzichtet man auf den Serienwiderstand. (In vielen Konfigurationen hat der Bus genügend "Reserven" hinsichtlich der Treibfähigkeit, so daß die zusätzliche Buslast an einzelnen AD-Leitungen nicht schadet. Dies wird bei der Systementwicklung üblicherweise durch Analogsimulation der Buskonfiguration überprüft.)

1.2.8. Mastervermittlung

Die nachfolgend beschriebenen Signale REQ# und GNT# sind nur in Master-Einrichtungen von Bedeutung. Jede Master-Einrichtung hat einen REQ-Ausgang und einen GNT-Eingang. Diese Anschlüsse sind mit den Vermittlungseinrichtungen der zentralen Steuerschaltungen verbunden. Die Signale schalten synchron, aber unabhängig von den Buszyklen. Die Signale werden üblicherweise wie die jeweiligen Einrichtungen durchnummeriert (REQ1#, GNT1#, REQ2#, GNT2# usw.).

1.2.8.1. REQ#

Das Signal dient zum Stellen von Master-Anforderungen (Request).

Signaltyp (logisch): Ausgang. Jede Master-Einrichtung hat einen REQ-Ausgang, der mit den zentralen Steuerschaltungen verbunden ist.

Signaltyp (technisch): Tri-State-Signal. Mit einem hochohmigen Pull-up-Widerstand (Weak Pull-up) beschaltet (typisch: 100 k Ω).

Rücksetzzustand: solange RST# aktiv ist, dürfen die zentralen Steuerschaltungen REQ# nicht auswerten.

Hinweise:

1. Der Zweck der Tri-State-Spezifikation besteht darin, die üblichen PCI-Treiberstufen (die Tri-State-Charakteristik haben) einsetzen zu können.
2. Nach dem Rücksetzen wird das Signal ständig getrieben (es führt also stets High- oder Low-Pegel).

Stellen einer Master-Anforderung: will die Einrichtung am Bus als Master wirksam werden (mit anderen Worten: die Busherrschaft erlangen), so aktiviert sie REQ#.

REQ# wird wieder deaktiviert, wenn die Einrichtung die Busherrschaft nicht mehr benötigt (Abschnitt 1.3.11.).

1.2.8.2. GNT#

Das Signal dient zum Bestätigen der Busherrschaft (Grant).

Signaltyp (logisch): Eingang. Jede Master-Einrichtung hat einen GNT-Eingang, der mit den zentralen Steuerschaltungen verbunden ist.

Signaltyp (technisch): Tri-State-Signal. Mit einem hochohmigen Pull-up-Widerstand (Weak Pull-up) beschaltet (typisch: 100 k Ω).

Rücksetzzustand: solange RST# aktiv ist, darf die Einrichtung GNT# nicht auswerten.

Hinweise:

1. Der Zweck der Tri-State-Spezifikation besteht darin, die üblichen PCI-Treiberstufen (die Tri-State-Charakteristik haben) einsetzen zu können.
2. Nach dem Rücksetzen wird das Signal ständig getrieben (es führt also stets High- oder Low-Pegel).

Bestätigen der Busherrschaft: zunächst fordert die Einrichtung den Bus durch Erregen von REQ# an. Wird ihr im Verlauf der Master-Vermittlung der Bus zugesprochen (granted), so erregen die zentralen Steuerschaltungen die zugehörige GNT-Leitung. Im folgenden Takt darf die Einrichtung dann als Master den ersten Buszyklus starten (mit der Adreßphase).

Entziehen der Busherrschaft: soll der Einrichtung die Busherrschaft entzogen werden, so wird GNT# von den zentralen Steuerschaltungen deaktiviert. Daraufhin gibt die Einrichtung den Bus frei (der laufende Buszyklus wird aber noch zu Ende geführt).

Parken: die zentralen Steuerschaltungen können GNT# aktiv lassen, auch wenn die Einrichtung den Bus nicht mehr benötigt (erkennbar am inaktiven REQ#). Dann wird der Bus auf dieser Einrichtung geparkt; die Einrichtung wird zum Park Master.

1.2.9. Fehlersignalisierung

1.2.9.1. PERR#

Das Signal dient zum Anzeigen von Paritätsfehlern während der Datenübertragung (Parity Error). In der Adreßphase und in Sonderzyklen (Special Cycles) wird PERR# nicht verwendet.

Signaltyp: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS).

Rücksetzzustand, Parken, Ruhezustand: PERR# wird über den Pull-up-Widerstand auf High gehalten.

Beginn des Buszyklus (Adreßphase): PERR# wird bei Adreß-Paritätsfehlern nicht erregt. In einer Adreßphase kann PERR# aber aktiv sein, und zwar als Folge eines Paritätsfehlers in der vorausgegangenen Datenphase.

Datenphase: die jeweils empfangende Einrichtung wertet die Daten- und Byteerlaubnisignale aus

(Paritätskontrolle über AD31...0 + C/BE3#...0# + PAR (bei 64-Bit-Datenübertragung zusätzlich über AD63...32 + C/BE7#...4# + PAR64). Bei ungerader Parität (= Paritätsfehler) wird PERR# aktiviert.

Aktivierung beim Schreiben: vom Target. Das Target darf PERR# erst dann aktivieren, wenn es sich als solches zu erkennen gegeben hat (d. h., frühestens 2 Takte nach Erregen von DEVSEL#).

Aktivierung beim Lesen und bei der Unterbrechungsbestätigung: vom Master.

Achtung:

Es gibt "planare" Einrichtungen, die gar keine Fehlersignalisierung haben. Empfängt eine solche Einrichtung fehlerhafte Daten, so wird PERR# nicht erregt.

Zum Zeitverhalten:

- die zu einer bestimmten Datenübertragung gehörende PERR-Belegung wird 2 Takte später wirksam: 1. Takt: Datenübertragung, 2. Takt: die zugehörigen Paritätssignale (PAR, ggf. PAR64) erscheinen auf dem Bus, 3. Takt: die Paritätskontrolle wurde ausgeführt, PERR# wird entsprechend belegt (High = o. k., Low = Paritätsfehler).
- PERR# muß wenigstens über eine Taktperiode aktiv gehalten werden.

Reaktion auf den Paritätsfehler: diese ist den zentralen Steuerschaltungen freigestellt. Eine übliche Reaktion: (1) Erregen von SERR#, (2) Auslösen eines NMI im Prozessor.

1.2.9.2. SERR#

Das Signal dient zum Anzeigen von Paritätsfehlern in der Adreßphase und in Sonderzyklen sowie als allgemeines PCI-Fehlersignal.

Signaltyp: Open-Drain-Busleitung mit Pull-up-Widerstand, die von jeder Einrichtung erregt werden kann. Auswertung: durch die zentralen Steuerschaltungen.

Rücksetzzustand, Parken, Ruhezustand: SERR# wird über den Pull-up-Widerstand auf High gehalten.

Beginn des Buszyklus (Adreßphase): SERR# dient zum Anzeigen von Paritätsfehlern der Adreß- und Kommandosignale (Paritätskontrolle über AD31...0 + C/BE3#...0# + PAR (bei 64-Bit-Datenübertragung zusätzlich über AD63...32 + C/BE7#...4# + PAR64)). Bei ungerader Parität (= Paritätsfehler) wird SERR# aktiviert. Hierbei kann eine beliebige Einrichtung die Paritätskontrolle durchführen und ggf. SERR# erregen.

Hinweise:

1. Tatsächlich ist während der Adreßphase in jeder Einrichtung die Paritätskontrolle "scharf" (mit Ausnahme jener "planaren" Einrichtungen, die gar keine Fehlersignalisierung haben). Jede Einrichtung, die einen Paritätsfehler erkennt, kann dann SERR# erregen.
2. Ein möglicher Fehlermechanismus, der ebenfalls zur Erregung von SERR# führen kann: in einer Einrichtung sind Buskoppelstufen defekt, so daß die intern empfangenen Signale inkorrekte Parität haben, obwohl die Signalbelegung des Bus an sich o.k. ist.

Sonderzyklen: SERR# dient zum Anzeigen von Paritätsfehlern der Datenbelegung (Paritätskontrolle über AD31...0 + C/BE3#...0# + PAR).

Andere Fehlermeldungen: SERR# kann von jeder Einrichtung jederzeit erregt werden, um beliebige Fehler anzuzeigen. Verschiedene Fehlermeldungen können sich überlagern oder aufeinander folgen.

Dauer der Erregung: der einzelne Fehler wird signalisiert, indem SERR# eine Taktperiode lang aktiviert wird.

Schalten von SERR#: SERR# schaltet taktsynchron ein, aber asynchron aus (und zwar infolge des Open-Drain-Prinzips - die Dauer des Low-High-Übergangs wird praktisch allein vom Pull-up-Widerstand bestimmt). *Richtwert:* wenn SERR# für eine Taktperiode aktiviert wird, so sehen die auswertenden Schaltungen eine Erregung, die 2 bis 3 Taktperioden dauert.

Reaktion auf die Fehlermeldung: diese ist den zentralen Steuerschaltungen freigestellt. Eine übliche Reaktion besteht darin, im Prozessor einen NMI auszulösen.

Hinweis:

Die PCI-Spezifikation spricht von schwerwiegenden Fehlern (Catastrophic Errors). Man kann hierunter alle Fehlersituationen rechnen, unter denen der weitere Betrieb des PCI-Bus unmöglich oder nicht mehr sinnvoll ist. SERR# löst lediglich eine pauschale, harte Fehler-Reaktion aus - mehr sieht die Spezifikation nicht vor. Damit kann man weder auf Fehler selektiv reagieren noch genügend Information zusammentragen (im Sinne eines Maschinenfehler-Logouts), um die Fehlerursache genauer zu bestimmen. Aus anderer Sicht: Ein Entwickler, der SERR# nutzen will, muß damit rechnen, daß SERR# lediglich einen NMI oder eine andere harte Reaktion auslöst und den totalen Zusammenbruch des Bus-Betriebs zur Folge hat. Eine ggf. erforderliche "feinfühligere" Fehlerbehandlung (Beispiele: ECC-Fehlermeldungen von Speichereinrichtungen auf PCI-Karten, Meldungen von Netzwerkfehlern, FIFO-Überlaufbedingungen usw.) müssen auf andere Weise implementiert werden (z. B. durch Interruptauslösung oder nach dem Abfrageprinzip (Polling)).

1.2.10. Zugriffssteuerung: LOCK#

Über dieses Signal kann ein Master ausschließlichen Zugriff auf ein bestimmtes Target erlangen. Solche ausschließliche (unteilbare, verriegelte, exklusive) Zugriffe sind notwendig, um bestimmte Formen der Software-Organisation zu implementieren.

Signaltyp: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS).

Rücksetzzustand, Parken: LOCK# wird über den Pull-up-Widerstand auf High gehalten.

Die Verriegelungswirkung (Lock-Funktion) am PCI-Bus: ein einziger Master hat ausschließlichen Zugriff auf ein einziges Target. Die Verriegelung gilt nur für Speicherzugriffe. Der Master, der durch Erregen von LOCK# diesen ausschließlichen Zugriff erlangt hat, heißt Lock Master. Es gibt zwei Varianten der Zugriffssteuerung:

1. *Busverriegelung (Complete Bus Lock CBL):* bei aktivem LOCK wird der Bus nicht weiter vermittelt; der Lock Master hat also die ausschließliche Busherrschaft (die naheliegende, einfachste Lösung, die aber, da alle anderen Einrichtungen vom Buszugriff ausgeschlossen werden, mit Leistungsminderung verbunden ist).

2. *Ressourcenverriegelung (PCI Resource Lock PRL)*: andere Einrichtungen können zwischenzeitlich die Busherrschaft erlangen. Das vom Lock Master angesprochene Target weist aber Zugriffsversuche anderer Einrichtungen ab (indem es eine Wiederanlauf-Endebedingung signalisiert).

Die Implementierung (CBL oder PRL) ist freigestellt; es hängt also von den zentralen Steuerschaltungen ab, wie die Verriegelungsfunktion realisiert ist.

Die Verriegelungswirkung kommt durch besondere Signalprotokolle der Busvermittlung und der Target-Adressierung zustande (weitere Einzelheiten in Abschnitt 1.3.7.):

- Busvermittlung (Arbitrierung): eine Einrichtung, die Lock Master werden will, beantragt die Busherrschaft erst dann (REQ#-Signal), wenn kein anderer Lock Master aktiv ist (mit anderen Worten: wenn LOCK# frei verfügbar ist). Hat diese Einrichtung die Busherrschaft erhalten (GNT#-Signal) und war dabei LOCK# frei verfügbar gewesen, so hat sie die volle Kontrolle über den Bus einschließlich des LOCK-Signals. Somit kann sie zum (einzigen) Lock Master werden.
- Verriegelung des Targets: der Zugriff auf das jeweilige Target wird verriegelt, indem LOCK# in der ersten Taktperiode nach der Adreßphase aktiviert wird. Damit "weiß" das Target, daß es der Verriegelung unterworfen ist.
- Zugriffe anderer Master auf ein verriegeltes Target werden mit einer Wiederanlauf-Endebedingung abgewiesen (bei PRL),
- der Lock Master hält LOCK# so lange aktiv, wie die Verriegelungswirkung anhalten soll, und zwar auch dann, wenn er nicht die Busherrschaft hat (bei PRL),
- greift der Lock Master auf das verriegelte Target zu, so "pulst" er das LOCK-Signal (indem er es während der Adreßphase auf High schaltet). Das Target kann also Zugriffe des Lock Masters (die es nicht abweisen darf) daran erkennen, daß LOCK# in der Adreßphase inaktiv ist und gleich darauf wieder aktiv wird. Zugriffsversuche anderer Master (die abgewiesen werden müssen) sind daran erkennbar, daß LOCK# während der Adreßphase aktiv bleibt.

1.2.11. Interrupts: INTA#, INTB#, INTC#, INTD#

Die Signale dienen zum Auslösen von Interrupts im zentralen Prozessor.

Signaltyp: Open-Drain-Busleitung mit Pull-up-Widerstand, die von jeder Einrichtung erregt werden kann. Auswertung: durch die zentralen Steuerschaltungen.

Rücksetzzustand: die INT-Signale werden über die Pull-up-Widerstände auf High gehalten.

Die INT-Signale wirken asynchron und sind von den eigentlichen Buszyklen vollkommen unabhängig; es handelt sich lediglich um ein Durchreichen von Unterbrechungsanforderungen zu einem zentralen Interrupt-Controller.

Die Open-Drain-Auslegung ermöglicht es, mehrere Einrichtungen auf eine INT-Leitung zu schalten. Die Interruptsignalisierung selbst erfolgt pegelgesteuert (level sensitive) - im Gegensatz zur flanken-gesteuerten Signalisierung beispielsweise am ISA-Bus.

Es ist Angelegenheit der Software, herauszufinden, welche Einrichtungen eine Unterbrechung ausgelöst haben (z. B. durch Abfragen entsprechender Anforderungsregister in den Einrichtungen). Eine Einrichtung, die ein INT-Signal aktiviert hat, muß diese Aktivierung so lange halten, bis sie durch Software zurückgesetzt wird.

1.2.12. Cache-Steuerung

Die Vorkehrungen zur Cache-Steuerung betreffen das Problem der Übereinstimmung von Cache-Inhalt und Speicher-Inhalt (Cache-Kohärenz).

Hinweise:

1. Nähere Einzelheiten werden in Abschnitt 1.3.8. erklärt.
2. Ist der Speicher am PCI-Bus vollständig vom "Caching" ausgeschlossen (Non-Cacheable Memory), so ist es nicht erforderlich, die Signale SBO# und SDONE anzusteuern. (Es genügt dann, sie mit Pull-up-Widerständen auf High zu halten.)
3. In PCs ist "Cacheable Memory" am PCI-Bus typischerweise *nicht* vorgesehen (vgl. auch Abschnitt 1.2.19.).

1.2.12.1. SDONE

Mit diesem Signal zeigt die Cache-Steuerung an, daß eine Fremdadreßprüfung des Cache (Address Snoop) ausgeführt wurde (Snoop Done).

Signaltyp: Eingang. *Ohne* Pull-up-Widerstand. Das Signal wird nur von Einrichtungen ausgewertet, die Speicherbereiche enthalten (gleichgültig, ob "cacheable" oder nicht).

Rücksetzzustand: SDONE wird von den zentralen Steuerschaltungen mit Low belegt.

Parken, Ruhezustand: SDONE wird von den zentralen Steuerschaltungen mit einem stabilen Wert (High oder Low) belegt.

Wirkungsweise: SDONE wird nur in Speicherzugriffen ausgewertet. Es ist dazu vorgesehen, den Buszyklus erforderlichenfalls zu verlängern, bis die Adreßprüfung (mit ggf. erforderlichem Ändern bzw. Ungültigmachen des Cache-Eintrags) abgeschlossen ist. *Prinzip:* Das Target darf erst dann TRDY# signalisieren (um das Ende des Zugriffs anzuzeigen), wenn SDONE aktiv ist bzw. für die Dauer eines Taktzyklus aktiviert wurde. Zu Einzelheiten siehe Abschnitt 1.3.8.

Festbeschaltung: unter folgenden Bedingungen ist es nicht erforderlich, das Signal anzusteuern (es wird dann über einen Pull-up-Widerstand (Richtwert: 5 k Ω) fest auf High gehalten): (1) das System enthält gar keinen Cache, (2) Cacheable Memory am PCI-Bus wird grundsätzlich nicht unterstützt.

1.2.12.2. SBO#

Mit diesem Signal zeigt die Cache-Steuerung an, daß der Zugriff ein Treffer (Cache Hit) auf einen veränderten (modifizierten) Cache-Eintrag war und daß es demzufolge erforderlich ist, vor Ausführung des Zugriffs den Cache-Eintrag in den Speicher zurückzuschreiben (Snoop Backoff). SBO# ist nur bei Write-Back-Caches von Bedeutung.

Signaltyp: Eingang. Ohne Pull-up-Widerstand. Das Signal wird nur von Einrichtungen ausgewertet, die Cacheable Memory enthalten.

Rücksetzzustand: SBO# wird von den zentralen Steuerschaltungen mit High belegt.

Parken, Ruhezustand: SBO# wird von den zentralen Steuerschaltungen mit einem stabilen Wert (High oder Low) belegt.

Wirkungsweise: SBO# wird nur in Speicherzugriffen ausgewertet. Sind SDONE und SBO# gleichzeitig aktiv, so beendet das Target den Buszugriff mittels einer Wiederanlauf-Endebedingung und gibt somit den zentralen Steuerschaltungen Gelegenheit, den Cache-Eintrag in den Speicher zurückzuschreiben. (Hierzu muß die einschlägige Hardware - z. B. die Host-to-PCI Bridge - ihrerseits als Master am PCI-Bus wirksam werden.) Zu Einzelheiten siehe Abschnitt 1.3.8.3.

Festbeschaltung: unter folgenden Bedingungen ist es nicht erforderlich, das Signal anzusteuern (es wird dann über einen Pull-up-Widerstand (Richtwert: 5 k Ω) fest auf High gehalten): (1) Write-Thru-Cache, (2) das System enthält gar keinen Cache, (3) keine PCI-Einrichtung enthält Cacheable Memory.

1.2.13. Testinterface: TDI, TDO, TCK, TMS, TRST#

PCI-Einrichtungen können wahlweise mit einem Boundary-Scan-Testinterface gemäß JTAG/IEEE 1149.1 ausgerüstet werden. Tabelle 1.25 gibt einen Überblick über die Signale. Gleichzeitig ist angegeben, wie die entsprechenden Leitungen zu beschalten sind, wenn das Testinterface nicht implementiert wird.

Signal	Funktion	Beschaltung bei Nichtnutzung
TDI	Daten-Eingang (Test Data Input)	Pull-up-Widerstand ^{1),2)}
TDO	Daten-Ausgang (Test Data Output)	offen lassen ²⁾
TMS	Zustandssteuerung (Test Mode Select)	Pull-up-Widerstand ¹⁾
TCK	Takt (Test Clock)	Pull-down-Widerstand ¹⁾
TRST#	Rücksetzen (Test Reset)	Pull-down-Widerstand ¹⁾

1): Richtwert: 5 k Ω ; 2) auf PCI-Steckkarten, die Boundary Scan nicht nutzen, müssen die TDI- und TDO-Anschlüsse des Steckverbinders miteinander verbunden sein

Tabelle 1.25 Boundary-Scan-Signale

Hinweise:

1. TDI, TDO, TMS, TCK sind stets erforderlich, um Boundary Scan zu implementieren, TRST# ist wahlfrei (optional).
2. TMS, TCK, TRST# sind an alle betreffenden PCI-Einrichtungen geführt. Typischerweise sind mehrere Boundary-Scan-Schaltkreise über TDO und TDI zu einer Schiebekette (Test Ring) zusammengeschaltet.

3. Das Zusammenschalten von Schaltkreisen mit Boundary-Scan-Vorkehrungen zu Test-Schiebeketten ist dem Hardware-Entwickler freigestellt. (Beispielsweise bilden die Boundary-Scan-Schaltkreise eines Motherboards oder einer Steckkarte jeweils eine Schiebekette.)
4. Die Boundary-Scan-Vorkehrungen sind vom eigentlichen PCI-Bus vollkommen unabhängig. Während des normalen Betriebs werden sie nicht genutzt.

1.2.14. Anwesenheits- und Strombedarfsanzeige: PRSNT2#, PRSNT1#

Mit diesen Signalen zeigen PCI-Steckkarten zweierlei an: (1) daß sie im PCI-Slot stecken (also anwesend (*present*) sind) und (2) wieviel Strom sie verbrauchen.

Signaltyp: Festwert-Ausgang.

Signalcodierung: die Anwesenheits- und Stromverbrauchsangabe ist gemäß Tabelle 1.26 codiert.

Beschaltung		Signalbelegung PRSNT2#, 1#	Bedeutung
PRSNT2#	PRSNT1#		
offen	offen	1, 1	keine Steckkarte anwesend
offen	Masse	1, 0	Steckkarte anwesend, Stromaufnahme maximal 25 W
Masse	offen	0, 1	Steckkarte anwesend, Stromaufnahme maximal 15 W
Masse	Masse	0, 0	Steckkarte anwesend, Stromaufnahme maximal 7,5 W

Tabelle 1.26 Codierung der Signale PRSNT2#, 1#

Hinweise:

1. Die Stromaufnahme betrifft die "eigentliche" Speisespannung (5 V oder 3,3 V), und zwar bei *voll bestückter* Steckkarte (wichtig, wenn die Steckkarte Erweiterungs-Vorkehrungen hat (z. B. für SIMMs, Speicherschaltkreise usw.) - dann gilt die Stromaufnahme des "Vollausbaus").
2. Die PRSNT-Signale der einzelnen Slots (auf dem Motherboard) dürfen nicht zusammenschaltet werden.
3. Die Auswertung der Signale liegt außerhalb des PCI-Bus. (Sie können z. B. an programmseitig abfragbare Eingangs-Ports auf dem Motherboard angeschlossen sein.)
4. Abbildung 1.35 zeigt eine typische Beschaltung auf dem Motherboard. (Der Zweck der Kondensatoren: sie dienen als "dynamische Masseverbindungen" (AC Return Paths (siehe Hinweis 2 unter Tabelle 1.9)).
5. Vielfach werden sie gar nicht genutzt. Sie müssen dann einzeln mit einem Kondensator (Richtwert: 0,01...0,1 µF; Zweck: siehe Hinweis 4) nach Masse beschaltet werden (vgl. Abschnitt 1.2.19.)

Abbildung 1.35 Beschaltung der PRSNT-Signale

1.2.15. 66-MHz-Betrieb: M66EN

Mit diesem Signal zeigen PCI-Einrichtungen an, für welche Betriebsfrequenz (33 oder 66 MHz) sie ausgelegt sind (66 MHz Enable). Das Signal wurde in Ausgabe 2.1 der PCI-Spezifikation definiert.

Signaltyp: Festwert-Ausgang, der auch zurückgelesen werden kann.

Wirkung: in einem 66-MHz-System beeinflusst M66EN unmittelbar die PCI-Takterzeugung. Signalbelegung:

- Low = 33-MHz-Betrieb,
- High = 66-MHz-Betrieb.

Beschaltung (Abbildung 1.36):

- ein 33-MHz-Motherboard beschaltet das Signal fest mit Masse (Low),
- ein 66-MHz-Motherboard beschaltet das Signal mit einem Pull-up-Widerstand,
- eine 33-MHz-Einrichtung beschaltet das Signal fest mit Masse (Low),
- eine 66-MHz-Einrichtung treibt das Signal nicht (offen).

M66EN ist demzufolge nur dann High, wenn “alles” (Motherboard + angeschlossene/eingesteckte PCI-Einrichtungen) für 66 MHz ausgelegt ist.

66-MHz-Einrichtungen können das Signal abfragen, um zu erkennen, ob sie mit 66 oder mit 33 MHz betrieben werden.

Tabelle 1.27 nennt die Kombinationsmöglichkeiten der Taktfrequenz-Auslegung.

Abbildung 1.36 Beschaltung von M66EN

Erklärung:

- 1) Taktgenerator für PCI-Takt (CLK),
- 2) PCI-Einrichtung auf Motherboard. Fragt M66EN ab, um zu erkennen, welcher Bustakt t anliegt.
- 3) Slot,
- 4) Steckkarte,
- 5) funktionelle Logik. Fragt M66EN ab, um zu erkennen, welcher Bustakt anliegt.
- 6) Beschaltung auf 66-MHz-Motherboard,
- 7) Beschaltung auf 33-MHz-Motherboard.

Bussystem (Motherboard) für 66 MHz ausgelegt (66 MHz Capable)	Einrichtung für 66 MHz ausgelegt (66 MHz Capable)	resultierende Belegung von M66EN	Bestückungs-zustand	wirksamer Bustakt
nein: legt M66EN auf Masse (Low)	nein: legt M66EN auf Masse (Low)	Low	33-MHz-Einrichtung an 33-MHz-Bus	33 MHz
nein: legt M66EN auf Masse (Low)	ja: läßt M66EN offen	Low	66-MHz-Einrichtung an 33-MHz-Bus	33 MHz ^{*)}
ja: beschaltet M66EN mit Pull-up-Widerstand (= High)	nein: legt M66EN auf Masse (Low)	Low	33-MHz-Einrichtung an 66-MHz-Bus	33 MHz ^{*)}
ja: beschaltet M66EN mit Pull-up-Widerstand (= High)	ja: läßt M66EN offen	High	66-MHz-Einrichtung an 66-MHz-Bus	66 MHz

*) : siehe den folgenden Hinweis 3 im Text

Tabelle 1.27 Taktfrequenzsteuerung über M66EN

Hinweise:

1. Nur 3,3-V-Systeme können mit 66 MHz betrieben werden.
2. 66-MHz-Karten finden in PCI-Systemen gemäß Spezifikation 2.0 M66EN = Low vor (infolge der festen Massebeschaltung des Anschlusses).
3. Gelegentlich fragt die Konfigurationssoftware die (im Konfigurationsadreibraum spezifizierte) Betriebsfähigkeit (Bitposition "66 MHz Capable") und die Belegung von M66EN ab. Die in Tabelle 1.27 mit *) bezeichneten Konfigurationen werden dann als unzuweckmäßig erkannt, und es wird eine entsprechende Warnung angezeigt.

1.2.16. Taktsteuerung: CLKRUN#

Dieses Signal dient der Taktsteuerung (d. h. dem Anhalten und Starten des Taktes), insbesondere in mobilen Systemen (Clock Running). Es ist nicht an die Slots geführt, sondern nur für fest eingebaut e ("planare") Einrichtungen vorgesehen.

Hintergrund: im statischen Betrieb (wenn keine Schaltvorgänge stattfinden) haben CMOS-Schaltungen eine sehr geringe Stromaufnahme ("fast nichts"). Ist es wichtig, Strom zu sparen, so liegt es nahe, den Takt anzuhalten, falls gerade nichts zu tun ist.

Signaltyp:

- für die zentralen Steuerschaltungen: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS),
- für die anderen Einrichtungen: Open-Drain-Ausgang + Eingang (zum "Zurücklesen" der Belegung).

Wirkung: die Taktsteuerung (Anhalten/Laufenlassen) obliegt den zentralen Steuerschaltungen. Es sind 3 Zustände zu unterscheiden:

1. Takt läuft: die zentralen Steuerschaltungen aktivieren CLKRUN#. Die anderen Einrichtungen beobachten CLKRUN#, um ggf. einen Übergang nach Zustand 2 zu erkennen.
2. Takt läuft, könnte aber (aus Sicht der zentralen Steuerschaltungen) angehalten werden: die zentralen Steuerschaltungen schalten CLKRUN# für eine Taktperiode auf High (dies zeigt den anderen Einrichtungen den Zustandsübergang an), geben dann aber das Signal frei (Übergang in den hochohmigen Zustand) und beobachten, welcher Pegel sich einstellt. Die einzelnen Einrichtungen prüfen, ob sie zur Zeit den Takt noch brauchen oder nicht. Einrichtungen, die den Takt noch benötigen, aktivieren ihre Open-Drain-Treiber, so daß CLKRUN# wieder Low wird. Der Takt kann somit noch nicht angehalten werden (Rückkehr zu Zustand 1). Erlauben hingegen alle Einrichtungen das Anhalten des Taktes, so wird CLKRUN# durch den Pull-up-Widerstand nach High gezogen. Daraufhin halten die zentralen Steuerschaltungen den Takt an.
3. Takt ist angehalten. Falls eine Einrichtung etwas zu tun bekommt (angeregt durch Tastenbetätigung, Anforderungen von Schnittstellen usw.), muß der Takt wieder gestartet werden. Um dies anzufordern, zieht die betreffende Einrichtung CLKRUN# auf Low. Daraufhin schalten die zentralen Steuerschaltungen den Takt wieder ein (Rückkehr zu Zustand 1).

Hinweise:

1. Zu weiteren Einzelheiten siehe Abschnitt 1.6.5.
2. Anstelle des "harten" Anhaltens kann auch die Taktfrequenz verlangsamt werden (wenn im Stromsparbetrieb doch noch gewisse "synchrone" Aktivitäten in den Einrichtungen ablaufen sollen (Überwachen von Interfaces, Auszählen von Zeitintervallen, DRAM-Refresh usw.)).

1.2.17. Stromspar-Ereignissignalisierung: PME#

Mit diesem Signal können PCI-Einrichtungen ein Stromsparereignis (Power Management Event) anfordern. Das Signal wurde in Ausgabe 2.2 der PCI-Spezifikation definiert.

Signaltyp: Open-Drain-Busleitung mit Pull-up-Widerstand, die von jeder Einrichtung erregt werden kann. Auswertung: durch die zentralen Steuerschaltungen.

Rücksetzzustand: PME# wird über den Pull-up-Widerstand auf High gehalten.

Wirkung: PME# schaltet asynchron. Es kann von jeder Einrichtung erregt werden, die ein Stromsparereignis auslösen möchte. PME# bewirkt lediglich eine Art Interrupt. Die Behandlung dieses Ereignisses ist Sache der Software (die hierzu die näheren Umstände, den aktuellen Betriebszustand usw. untersuchen muß). Eine Einrichtung kann ein solches Ereignis in beiden Situationen anfordern:

1. sie hat nichts mehr zu tun und wünscht deshalb, in einen Stromsparzustand überführt zu werden,
2. es ist wieder etwas zu tun, deshalb muß der Stromsparzustand aufgehoben werden.

Stromsparen kann man u. a. durch Verringern oder gänzlich Abschalten der Speisespannung. Unterwirft man (naheliegenderweise) nur jene Einrichtungen dem Stromsparen, die gerade nichts zu tun

haben, so liegt ein kritischer Betriebsfall vor: ausgeschaltete Hardware in einer eingeschalteten Umgebung (Partial Power Down). PME# verbindet Einrichtungen in verschiedenen Stromsparzuständen (Ein, reduzierte Speisespannung, Aus). Deshalb sind an die PME-Treiberstufen ausgeschalteter (bzw. mit verminderter Spannung betriebener) Einrichtungen folgende Forderungen zu stellen: der Treiber darf (1) keinen Schaden nehmen, wenn die PME-Leitung High-Pegel führt und er darf (2) keinen Stromweg nach Masse schalten.

1.2.18. 64-Bit-Erweiterung

1.2.18.1. Adressen, Daten, Kommandos, Byteauswahl, Paritätsbit

Die 64-Bit-Erweiterung umfaßt die Signale AD63...32, C/BE7#...4# und PAR64. In PCI-Slots belegen diese Signale die 64-Bit-Erweiterung des Steckverbinders (vgl. Tabelle 1.9). Die Signale sind in den Abschnitten 1.2.4.2., 1.2.5.2. und 1.2.6.2. näher beschrieben.

1.2.18.2. REQ64#

Mit diesem Signal zeigt der Master an, daß er eine 64-Bit-Datenübertragung ausführen kann (Request 64 Data Bit Access). REQ64# ist nur in Speicherzugriffen von Bedeutung und wird ansonsten ignoriert.

Signaltyp: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS).

Rücksetzzustand: REQ64# wird über den Pull-up-Widerstand auf High gehalten.

Sonderwirkung beim Rücksetzen: über REQ64# wird angezeigt, daß das System den 64-Bit-Betrieb unterstützt (indem die zentralen Steuerschaltungen REQ64# aktivieren; vgl. die Abschnitte 1.2.3.2. und 1.2.3.3.).

Parken: REQ64# wird über den Pull-up-Widerstand auf High gehalten.

Ruhezustand: ein 64-Bit-Master treibt REQ64# nach High. Ansonsten wird REQ64# über den Pull-up-Widerstand auf High gehalten.

Beginn des Buszyklus (Adreßphase): ein 64-Bit-Master aktiviert REQ64#, wenn er einen 64-Bit-Speicherzugriff ausführen will.

Belegung in der Datenphase: wurde in der Adreßphase REQ64# aktiviert, so bleibt es auch in der Datenphase aktiv.

Hinweis:

Das Schaltverhalten von REQ64# entspricht dem von FRAME# (Abschnitt 1.2.7.1.).

1.2.18.3. ACK64#

Mit diesem Signal reagiert das Target auf eine 64-Bit-Anforderung des Masters (über REQ64#); es bestätigt, daß es die geforderte 64-Bit-Übertragung ausführen kann (Acknowledge 64 Data Bit Access). ACK64# ist nur in Speicherzugriffen und nur als Reaktion auf REQ64# von Bedeutung. Ansonsten wird das Signal nicht erregt (und über den Pull-up-Widerstand auf High gehalten).

Signaltyp: Tri-State-Busleitung mit Pull-up-Widerstand, bidirektional (STS).

Rücksetzzustand, Parken, Ruhezustand: ACK64# wird über den Pull-up-Widerstand auf High gehalten.

Beginn des Buszyklus (Adreßphase): ACK64# wird über den Pull-up-Widerstand auf High gehalten.

Belegung in der Datenphase: wurde das Target mit aktivem REQ64 angesprochen und ist es in der Lage, 64-Bit-Speicherzugriffe auszuführen, so aktiviert es ACK64#.

Hinweis:

Das Schaltverhalten von ACK64# entspricht dem von DEVSEL# (Abschnitt 1.2.7.5.).

1.2.18.4. 64-Bit-Karten in 32-Bit-Slots

Adressen, Daten, Kommandos, Byteauswahl, Paritätsbit

In dieser Betriebsweise hängt die 64-Bit-Erweiterung des Karten-Steckverbinders gleichsam in der Luft. "Offene" CMOS-Eingänge führen aber zu undefiniertem Verhalten. Deshalb muß dieser Betriebszustand verhindert werden. Was hierfür *nicht* erlaubt ist, sind an die Signalleitungen *direkt* angeschlossene Pull-up-Widerstände auf der Steckkarte^{*)}. Eine mögliche Lösung besteht darin, die Signale als Ausgänge zu betreiben und so auf definierten Pegeln zu halten.

^{*)}: gegen *geschaltete* Pull-ups (über CMOS-Schalterbauelemente (Crossbar, QuickSwitch usw.) hat die Spezifikation offensichtlich nichts einzuwenden.

REQ64# und ACK64#

Diese Signale sind im 32-Bit-Slot geführt. In einem 32-Bit-System ist je Slot und Signal ein gesonderter Pull-up-Widerstand vorgesehen. (Die Signale werden hier je Slot einzeln auf High gezogen; sie sind nicht als Busleitungen zusammengeschaltet.)

1.2.19. Ausführungsbeispiel

Anhand der Abbildungen 1.37 bis 1.41 wollen wir einen Überblick über den Aufbau eines PCI-Subsystems in einem typischen PC geben.

Hinweise:

1. Die Abbildungen stammen aus einem Schaltplan, der den Aufbau eines Motherboards in allen Einzelheiten darstellt. Solche Schaltpläne werden - als Beispiele und Entwurfs-Hilfe - von verschiedenen Halbleiter-Herstellern veröffentlicht (Internet).
2. Einige Signalbezeichner tragen hier den Vorsatz-Buchstaben P, um die Zugehörigkeit zum PCI-Bus zu kennzeichnen.

Abbildung 1.37 Blockschaltbild eines Motherboards mit PCI-Bus (Quelle: Intel)

Erklärung:

- 1) Steuerschaltkreis (A.G.P. Controller, PAC). Enthält u. a. das Interface zum Prozessor (Pentium II Systembus), die Arbeitsspeichersteuerung und die Host-to-PCI Bridge sowie die zentralen Steuerschaltungen des PCI-Bus.
- 2) Speichersubsystem (Arbeitsspeicher). Hier mit SDRAM-Moduln ausgeführt.
- 3) PCI-Bus. 32-Bit-Bus gemäß Spezifikation 2.1, aber erweitert um Signal PME#.
- 4) PCI-Slots (4 Stück),
- 5) Steuerschaltkreis (PCI-to-ISA/IDE Xcelerator, PIIX4). Enthält die PCI-to-ISA Bridge, 2 USB-Interfaces, 2 IDE-Anschlüsse (jedem dieser Interface-Anschlüsse entspricht eine PCI-Funktion) sowie weitere PC-typische Funktionseinheiten (u. a. einen DMA-Controller, einen Interrupt-Controller, die Stromversorgungssteuerung usw.). Dieser Schaltkreis stellt somit eine "Mehrgerätesteuereinheit" (Multi-Function Device) dar, d. h. es handelt sich um eine (physische) Einrichtung (Device) am PCI-Bus, die mehrere PCI-Funktionen enthält (wovon einige als Master arbeiten können).

Abbildung 1.38 Der PCI-Anschluß am Steuerschaltkreis 8244BX (Intel)*Erklärung:*

- 1) die PCI-Busleitungen,
- 2) die REQ-Eingänge zu den Arbitrierungsschaltungen. Es sind 5 Signale PREQ#4...PREQ#0 vorgesehen, davon werden 4 von den PCI-Slots genutzt.
- 3) die GNT-Ausgänge der Arbitrierungsschaltungen. Es sind 5 Signale PGNT#4...PGNT#0 vorgesehen, davon werden 4 von den PCI-Slots genutzt.
- 4) diverse Signale der Stromsparsteuerung,
- 5) die Signale PHLD#, PHLDA# sind Masteranforderungs- und Bestätigungssignale vom 82371EB (da es in diesem Schaltkreis mehrere PCI-Masterfunktionen gibt, hat man dort eine Vor-Vermittlung angeordnet und stellt somit eine Sammel-Anforderung an die PCI-Steuerung).

Abbildung 1.39 Der PCI-Anschluß am Steuerschaltkreis 82371EB (Intel)*Erklärung:*

- 1) die PCI-Signale,
- 2) Signal CLOCKRUN# (= CLKRUN#) ist vorhanden, wird aber nicht genutzt,
- 3) PHLD# ist das Master-Anforderungssignal, PHLDA# das Bestätigungssignal für die Sammel-Anforderung der Masterfunktionen des Schaltkreises,
- 4) die PCI-Anforderungssignale der 4 Slots sind hier angeschlossen, um (zwecks Aktivitäts-Erkennung) von der Stromsparhardware überwacht zu werden,
- 5) Adreßbit 18 dient hier als IDSEL-Signal zur Auswahl der Einrichtung bei Konfigurationszugriffen.

Abbildung 1.40 Einzelheiten eines PCI-Slots (Intel)

Erklärung:

Die Abbildung zeigt - als Ausführungsbeispiel - den 2. Slot von Abbildung 1.37. Hier sind vor allem die Festbeschaltungen dargestellt (wobei teilweise die Schaltmittel für 2 Slots gezeigt werden):

- 1) in typischen PCs gibt es keinen Cacheable Memory am PCI-Bus. Deshalb Festbelegung von SDONE und SBO# durch einzelne Pull-up-Widerstände.
- 2) die Anwesenheits- und Strombedarfsanzeige wird nicht genutzt. PRSNT2#, 1# dienen lediglich als "dynamische Masseverbindungen" (AC Return Paths). Deshalb ist je 1 Kondensator nach Masse geschaltet.
- 3) es ist ein 32-Bit-Bus. Deshalb Festbelegung von REQ64# und ACK64# durch einzelne Pull-up-Widerstände.
- 4) zur Slot-Auswahl bei Konfigurationszugriffen ("geographische Adressierung") dient hier Adreßbit 27. Deshalb ist IDSEL über einen Serienwiderstand an AD27 angeschlossen.
- 5) das Testinterface wird nicht genutzt. Deshalb Festbelegung über Pull-up- und Pull-down-Widerstände (vgl. Abschnitt 1.2.13.).
- 6) das zusätzliche Signal PME#,
- 7) die Interrupt-Anforderungssignale (hier mit PIRQ#A...PIRQ#D bezeichnet) sind an entsprechende Interruptcontroller-Eingänge des 82371EB angeschlossen.

Abbildung 1.41 Pull-up-Widerstände (Widerstandsnetzwerke) an Busleitungen (Intel)

Erkennbar ist die Beschaltung der PCI-Signale mit Pull-up-Widerständen von 2,7 k Ω und 3,2 k Ω (in Form von Widerstandsnetzwerken).