

<b>Zeitschrift:</b>	Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association suisse des électriciens, de l'Association des entreprises électriques suisses
<b>Herausgeber:</b>	Schweizerischer Elektrotechnischer Verein ; Verband Schweizerischer Elektrizitätsunternehmen
<b>Band:</b>	81 (1990)
<b>Heft:</b>	17
<b>Artikel:</b>	IBM Micro Channel Architecture : wieso, wozu : Teil 1 : Konzept und Grundlagen
<b>Autor:</b>	Krähenbühl, Fred W.
<b>DOI:</b>	<a href="https://doi.org/10.5169/seals-903150">https://doi.org/10.5169/seals-903150</a>

### Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. [Mehr erfahren](#)

### Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. [En savoir plus](#)

### Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. [Find out more](#)

**Download PDF:** 24.02.2026

**ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>**

# IBM Micro Channel Architecture – Wieso, Wozu

## Teil 1: Konzept und Grundlagen

Fred W. Krähenbühl

**Selbst für den Ingenieur dürfte es nicht einfach sein, das heutige PC-Angebot richtig einzuschätzen. Die vorliegende Artikelserie bespricht nach einer theoretischen Einführung praxisnah und anhand von Schaltungen einige grundlegende Aspekte der Micro-Channel-Architektur. Ziel ist, dem Leser eine eigene Meinung im Wettkampf der Meinungen zu ermöglichen.**

**L'offre dans le domaine de la micro-informatique est tellement variée que même un ingénieur-système peut éprouver des difficultés à s'y retrouver. Cette série d'articles présente, après une brève introduction théorique, les aspects fondamentaux de l'Architecture Micro Channel. Des explications pratiques devraient permettre au lecteur de saisir ses enjeux stratégiques.**

### Adresse des Autors

Fred W. Krähenbühl, IBM Schweiz AG,  
General Guisan-Quai 26, 8022 Zürich

Anfang April 1987 wurde von IBM die Micro-Channel-Architektur angekündigt; kurz darauf wurden die ersten Maschinen ausgeliefert und Anfang 1990 waren weltweit über 2,5 Millionen Micro-Channel-Maschinen installiert. Für Ein-/Ausgabe- und Speicherfunktionen haben Dutzende von Herstellern bereits über 1000 Adapter angekündigt. Weitere IBM-Systeme, wie das Risc System/6000, sind mit Micro Channel ausgerüstet. 24 Hersteller haben sich bis heute für die Herausgabe eines oder mehrerer Computer auf der Basis von Micro Channel entschieden. Einige von ihnen sind heute bereits auf dem Markt.

Nach der Ankündigung wurden Stimmen laut, welche die Notwendigkeit einer neuen Architektur in Frage stellten. Man versuchte darzulegen, dass die mit dem IBM PC und dem IBM PC AT eingeführten Strukturen noch über Jahre hinweg genügen würden. Das mag im Ansatz sogar richtig sein. Zum Zeitpunkt der Einführung der ersten Maschinen mit Micro Channel waren die neuen Konzepte tatsächlich noch nicht so wichtig; die Maschinen waren nach Konzepten erarbeitet und gebaut worden, die erst noch wichtig werden sollten: Multiprogramming, Multitasking und Multiprocessing. In der ersten Phase konnte ein Benutzer tatsächlich «nur» von der erhöhten internen Geschwindigkeit der Maschine profitieren; seine DOS-Programme liefen unverändert, aber erheblich schneller. Die schönen Neuerungen waren für die Zukunft bestimmt. Kein Wunder, dass viele das Potential nicht gesehen haben.

In der Zwischenzeit ist viel passiert: Die Micro-Channel-Architektur ist ein Baustein aus einer ganzen Reihe, die dem Benutzer ermöglicht, aus der Datenverarbeitung in seiner Unternehmung viel mehr als heute zu machen. Ein zweiter wichtiger Baustein ist

OS/2, das Betriebssystem, welches nur mit dem Micro Channel zusammen seine gesamte Kraft spüren lässt. Man kann mit einem AT zwar OS/2 mit Multiprogramming fahren; die volle Wirkung aber erbringt nur ein vernünftiges Interrupt-Konzept. Selbstverständlich kann man mit einem PS/2 auch DOS und damit Applikationen laufen lassen, welche auf eine einzelne Arbeit ausgerichtet sind. Doch PS/2 kann mehr. PS/2 und OS/2 sind eben zusammen mehr als die Summe ihrer einzelnen Fähigkeiten. Dabei ist Multitasking bei weitem nicht das Ende der Entwicklung. Multiprocessing steht vor der Tür.

PS/2 und OS/2 können für viele Benutzer die richtige Wahl sein, weil diese heute eine hohe Leistung bieten und die Zukunft nicht verbauen. Maschinen mit OS/2 können aber auch die richtige Wahl innerhalb eines grösseren Konzeptes sein: SAA (Systems Application Architecture) definiert eine mögliche Art und Weise, wie Anwendungen, Datenbanken und Datenverbindungen konzipiert werden können, damit die Datenverarbeitung einer Unternehmung ein vollständiges und korrektes Abbild des Betriebes darstellt. PS/2 mit Micro Channel und OS/2 sind auf diese Herausforderung vorbereitet.

Umfassende Konzepte werden nicht an einem einzigen Tag realisiert. Einzelne Teile daraus müssen früher als andere erscheinen, damit andere darauf aufbauen können, selbst auf die Gefahr hin, dass die Tragweite eines neuen Konzeptes beim ersten Erscheinen noch nicht voll verstanden wird. Argumente... sind sie richtig? Sind sie verständlich? Kann man ohne weitere Kenntnis ermessen, was sie bedeuten? Kann man erkennen, ob sie den Mehraufwand für OS/2, für PS/2 und für Anwendungen, speziell für diese Umgebung erstellt, rechtfertigen?

Zurück zum Baustein IBM Micro Channel Architecture: Die vorliegende Arbeit will dem Leser durch Präsentation von technischen Details vertiefte Kenntnis über einige Zusammenhänge vermitteln, damit bei Bedarf ein sachlich fundiertes Urteil und damit im Einzelfall eine richtige Entscheidung möglich wird. Es werden dabei wiederum Fragen offen bleiben müssen; das Lesen einer Artikelserie vermag die eigene Erfahrung nicht zu ersetzen. Wenn sich aber jemand entscheidet, die angeführten Möglichkeiten auszuprobieren und praktisch anzuwenden, dann wird «be-greifen» zur Tatsache werden.

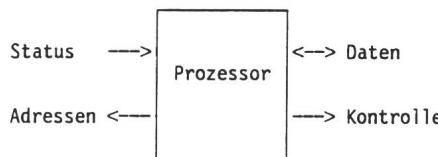
## Was ist ein Bus?

Der IBM PC ist wie auch der PC XT mit einem Prozessor des Typs Intel 8088 ausgerüstet. Dieser Prozessor liest die Instruktionen des Programms, eine nach der andern, aus dem Hauptspeicher und führt sie aus. Diese Instruktionen veranlassen den Prozessor, Daten von peripheren Einheiten (z.B. der Tastatur oder einer Diskette) in den Speicher zu lesen und zu verarbeiten (sammeln, addieren, rechnen) sowie die Resultate wieder an irgend eine periphere Einheit (z.B. den Drucker) auszugeben. Das Programm ist demnach für die gesamte Tätigkeit der Maschine verantwortlich. Der Prozessor selbst delegiert nur wenige Aufgaben an andere Systemkomponenten.

Die Adapter, also jene Teile der Elektronik, welche für die Verbindung der Peripherie mit dem Prozessor sorgen, sind meist reine Umsetzer, die oft nicht viel mehr als einzelne Zeichen (Bytes) zu verarbeiten vermögen. Der Taktgeber-Oszillator ist gleichsam der Antrieb des Prozessors. Seine Frequenz muss auf die Möglichkeiten des Prozessors und die Peripherie Rücksicht nehmen. Die Leistung des Prozessors bestimmt die Leistung des Gesamtsystems.

## Leitungen und Impulse auf dem Bus

Der Prozessor, der interne (Haupt-) Speicher, die Adapter sind untereinander verbunden. Physisch besteht ein Bus einfach aus einer Vielzahl von Leitungen, welche einen Prozessor mit den anderen Teilen des Computers verbinden. Damit man andere Teile nun auch anschliessen kann, sind in diesen Leitungen meist eine Anzahl Steckplätze für Adapter-Karten einge-



**Bild 1 Die verschiedenen Busleitungarten eines Prozessors**

bunden. Dadurch können Erweiterungen des Grundsystems – Speicherzusätze, sowie Ein- und Ausgabe-Anschlüsse – verwirklicht werden. Der Bus besteht aus folgenden Leitungstypen (Bild 1):

1. Datenleitungen: Je nach Implementation können 4 bis 64 Leitungen (bei Grosscomputern noch mehr) gleichzeitig je 1 Bit übertragen. Geht man von 8 Bit pro Zeichen (Byte) aus, können parallel also ein halbes bis 8 Byte über den Bus, z.B. vom Speicher zu einer Peripherieeinheit übermittelt werden.

2. Adressleitungen: Auf diesen Leitungen laufen die Impulse, welche bei einem Transfer angeben, wo in der Gesamtheit des verfügbaren Speichers die Daten gelesen oder geschrieben werden sollen. Auch die Ein-/Ausgabestelle (Port) wird über diese Leitungen definiert. Mit  $n$  Leitungen können  $2^n$  Speicherplätze direkt adressiert werden. Während frühe Kleincomputer 16 Adressbits (Adressraum: 64 KByte), der PC und XT 20 Bit (Adressraum: 1 MByte) offerierten, können Maschinen mit modernen Prozessoren und 32 Leitungen einen Adressraum von 4 GByte ansprechen.

3. Kontrolleleitungen: Sie geben die verschiedenen Taktimpulse an die Peripherie ab und sorgen damit dafür, dass z.B. die Speicher anstehende Daten im richtigen Zeitpunkt übernehmen oder zur Verfügung stellen können.

4. Statusleitungen: Sie übermitteln Informationen über den Zustand der Maschine oder über Unterbrechungswünsche (Interrupt-Requests) der Peripherie. Mit Hilfe dieser Informationen steuert der Prozessor den Programmablauf.

Der unsichtbare Teil des Busses, das Protokoll, ist nicht weniger wichtig als die beschriebenen Leitungen. Das Protokoll definiert den zeitlichen Ablauf der Bussignale. Um z.B. Daten vom Hauptspeicher lesen zu können, müssen erst die Adress- und Kontrollssignale auf die Leitungen gegeben werden. Diese «öffnen» die entsprechenden Speicherstellen, deren Inhalt in einer

zweiten Phase auf den Datenleitungen erscheint. Die verschiedenen Elektronik-Teile müssen nun so konstruiert sein, dass sie die Steuerimpulse zur korrekten Verarbeitung der Datenimpulse verwenden können. Dies bedingt, dass das Timing bekannt ist und genau eingehalten wird.

Beim PC und PC AT-Bus ist das in den meisten Fällen recht einfach. Im wesentlichen ist dort der Bus einfach eine Verlängerung der Leitungen, die der Prozessor anbietet. Das Timing kann also gut aus dessen Spezifikationen ersehen werden. Das ist der Grund, warum die technischen Handbücher des IBM PC und PC AT keine Zeit-Diagramme liefern. Sie sind mit denen des Prozessors identisch. Von einer «Architektur» zu sprechen, ist deshalb reichlich viel Lob für den Bus des PC oder AT.

In der Hierarchie eines Busses kennt man den Bus-Master, der sämtliche Impulse für alle Datenübertragungen über den Bus liefert und die «Sklaven» (Slaves). Die letzteren werden so genannt, weil sie nur auf Befehle des Masters reagieren können. Im Normalfall gibt es in Maschinen der Klasse PC, XT oder AT nur einen Master, den Hauptprozessor.

## Bemerkungen zum PC-Bus

In der ersten Begeisterung – man hatte jetzt ja wirklich einen Computer, der sich aufs Pult stellen liess – während der ersten Jahre also, wurden einige Millionen Maschinen installiert. Viele neue Hersteller ergriffen die Gelegenheit, um in das neue Geschäft mit Hard- und Software einzusteigen. Programme kamen auf den Markt, welche zum Teil die Möglichkeiten des Personal Computers bis aufs Letzte ausnützten. Die so entstandenen Lösungen gingen oft über die Grenzen hinaus, für welche die Maschinen konstruiert worden waren. Die Sicherheit der Verarbeitung, die Kompatibilität mit späteren Entwicklungen, sogar die Datenintegrität wurden oft kurzfristigem Gewinn geopfert, ohne dass man den Benutzer darüber aufgeklärt hätte.

Dieser Vorgang hat aber – und das ist das Gute daran – aufgezeigt, in welche Richtung weitere Gedankenarbeit zu leisten war. Die weitere Entwicklung in möglichst geordneten Bahnen ablaufen zu lassen, war eine der wichtigsten Forderungen. Will man die Bedürfnisse der Benutzer mittel- und langfristig befriedigen können, sind weitergehende Konzepte nötig.

Im folgenden werden einige der immer wieder vorgebrachten Probleme und Forderungen skizziert und die jetzigen Lösungen mit den bestehenden Maschinen-, Bus- und Betriebssystemkonzepten dargestellt:

- In den letzten Jahren wurden mehr und mehr Anwendungen auf Kleincomputern installiert. Man möchte deshalb, ohne laufend aus- und einsteigen zu müssen, rasch von einer Applikation auf die andere wechseln können. Dies bedeutet, dass auf einer Maschine mehrere voneinander unabhängige Programme gleichzeitig laufen müssen (Multiprogramming). Auf einem PC kann dieses Problem über Software gelöst werden. Ein «kleines» Monitorprogramm, steuert die jeweiligen Anwendungsprogramme an und schaltet nach Bedarf zwischen ihnen um.

Der Prozessor Intel 8088 (im PC und XT verwendet) besitzt 20 Adressleitungen, mit denen ein Maximum von 1 048 576 Speicherplätzen (1 Megabyte = 1024 Kilobyte im 8-Bit-Modus) angesprochen werden. Da ein Teil dieses Bereichs für maschineninterne Zwecke (z.B. Video-Speicher) gebraucht wird, stehen dem Anwendungsprogramm, inklusive DOS, noch 640 KByte zur Verfügung. Diese einmal festgelegte Grenze wollte man aus Kompatibilitätsgründen nicht so einfach überschreiten. Somit können aus Platzgründen kaum gleichzeitig mehr als ein Anwendungsprogramm im Hauptspeicher resident sein.

Selbstverständlich wurden «Auswege» gefunden. Man baute einfach mehr Speicher in die Maschine ein. Die Adressierung wurde dann unter Umgehung des Betriebssystems durch die Anwendung selbst so gestaltet, dass jeweils ein nicht benötigter Teil des Speichers (seitenweise, je 64 KByte) ausgeblendet und dadurch ein anderer Teil adressierbar gemacht wurde.

Das eigentliche Problem wurde aber durch diese Massnahmen nicht gelöst. In diesem «realen» Adressiermodus kann durch den Prozessor (und damit durch jedes Programm) jede beliebige Speicherstelle angesprochen werden. Läuft eines der residenten Programme durch irgend einen Fehler (Merke: Jedes Programm ist nur solange fehlerfrei, bis der nächste Fehler gefunden wird ...) falsch, ist die Möglichkeit, dass es andere Programme beeinträchtigt, sehr gross. Daraus resultierende Abstürze sind zumindest unangenehm und verursachen meist viel Rekonstruktionsarbeit.

## Erklärung wichtiger Begriffe

### Adapterkarte:

Adapterkarten sind steckbare Elektronik-Schaltungen, welche an den Systembus (z.B. Micro Channel) angeschlossen werden können. Sie enthalten beispielsweise Speicherbausteine oder dienen als Interfaces des Computers zur Außenwelt, über welche Ein-/Ausgabegeräte (z.B. Drucker) angeschlossen werden.

### Adresse:

Nummer eines Speicherplatzes oder eines Ein-/Ausgabe Ports

### Addressierung:

Die Ausgabe von Adressen über die Adressleitungen – zusammen mit weiteren Kontrollsignalen – veranlasst eine Speicherstelle, bestimmte Daten über die Datenleitungen abzugeben (READ) oder anzunehmen (WRITE). Normalerweise übermittelt der Bus Master in jedem Bus-Zyklus eine Adresse.

### Addressierung bei der Streaming Data-Prozedur:

Konzepte innerhalb der Micro Channel-Architektur, welche einen erhöhten Daten durchsatz über den Kanal ohne Erhöhung des Grundtaktes der Maschine erlauben. Bei der Streaming Data-Prozedur wird durch den Master bei Übertragung einer Vielzahl von Zeichen (Burst) nur noch zu Beginn die Basisadresse ausgegeben. Der Slave erarbeitet sich die folgenden Adressen durch Hochzählen selbst. Das erlaubt die Übertragungsrate auf den Datenleitungen zu verdoppeln sowie die Adressleitungen zur zusätzlichen Übertragung von Daten zu verwenden, was den möglichen Durchsatz ein weiteres Mal verdoppelt.

### Adressraum:

Maximale Anzahl der Speicherplätze oder Ein-/Ausgabe-Ports, die durch einen speziellen Prozessor (oder einen anderen Bus Master) angesprochen werden können. Die Micro Channel-Architektur definiert 32 Adressleitungen, mit denen  $2^{32}$  Adressen zur Verfügung stehen. Nicht in jeder Maschine mit Micro Channel ist der gesamte durch die Architektur vorgesehene Adressraum implementiert.

### Direct Memory Access (DMA):

Fähigkeit des Computers, Ein-/Ausgabe-Einheiten über ihre Adapter unter Umgehung des Hauptprozessors direkt auf den Speicher zugreifen zu lassen. Das Konzept kommt in vielen Ausprägungen vor.

### Multiprocessing:

Fähigkeit eines Computers, in Zusammenarbeit mit einem geeigneten Betriebssystem gleichzeitig mit mehreren Prozessoren Leistung zu erbringen. Die zusätzlichen Prozessoren übernehmen oft spezielle Funktionen.

### Multiprogramming:

Fähigkeit eines Computers, in Zusammenarbeit mit einem geeigneten Betriebssystem mehrere voneinander unabhängige Anwendungen, welche sich gleichzeitig im Speicher befinden, gemäss einer Prioritätenordnung «gleichzeitig» abarbeiten zu können. Das Konzept benutzt «Multitasking des Prozessors».

### Multitasking des Prozessors:

Fähigkeit des Prozessors, zwischen mehreren Programmen, welche sich gleichzeitig im Speicher befinden, hin und her zu schalten und die Speicherbereiche, welche zu den jeweiligen Programmen gehören, gegen Eingriffe der anderen Programme zu schützen. Diese Fähigkeit eines Mikroprozessors wird durch die Betriebssysteme zu unterschiedlichen Funktionen benutzt.

### Multitasking des Systems:

Fähigkeit eines Computers, in Zusammenarbeit mit einem geeigneten Betriebssystem und einer entsprechend geschriebenen Anwendung innerhalb eines Programmes «gleichzeitig» mehrere anstehende Aufgaben zu erledigen (z.B. die gleichzeitige Übermittlung von Daten von verschiedenen Quellen, welche anschliessend in einer Auswertung zu vereinigen sind).

### Parallelprocessing:

Der Computer vervielfacht seine Leistung, indem er ein einzelnes kompliziertes und zeitaufwendiges Programm in mehrere Prozesse aufteilt und diese an mehrere Prozessoren zur parallelen und gleichzeitigen Ausführung übergibt.

### Programmable Option Select (POS):

Die eingebauten Optionen einer Adapterkarte können über Software aktiviert werden (z.B. die Basisadresse für Speicherbereiche auf dem Adapter). Es müssen keine DIP-Schalter mehr eingestellt werden.

### System-Platine (auch Mother Board oder Systemkarte):

Elektronische Schaltung, welche den zentralen Teil des Rechners sowie die Steckplätze (Slots) zur Aufnahme der Adapterkarten enthält.

- Läuft eine Arbeit, welche den Computer sehr lange belegt, aber dabei nur einen kleinen Teil von dessen Ressourcen benötigt, so sollten diese gleichzeitig für andere Arbeiten eingesetzt werden können (Thema Multiprogramming). Als Beispiel diene eine Datenübermittlung im asynchronen Modus, wie sie auf mehr oder weniger jedem Kleincomputer möglich ist. Bei der durchaus üblichen Geschwindigkeit von 1200 Bit pro Sekunde dauert eine Übermittlung von 100 000 Zeichen (etwa 40 Seiten Text auf Papier des Formates A4) volle 14 Minuten. Die Belastung eines Systems der PC AT-Klasse beträgt dadurch aber nur etwa 2,5%; die restliche Leistung von 97,5% wird mittels Programmschlaufen vernichtet!

Eine Verbesserung dieser Situation kann nur mit Hilfe der Hardware erreicht werden. Der Adapter sollte einen Teil der Arbeit übernehmen. Dieser könnte beispielsweise eine ganze Zeichenkette empfangen (oder senden) und das Ende der Sequenz über eine Unterbrechung (Interrupt) dem Prozessor melden. Der Prozessor wird damit während der eigentlichen Übertragung frei für andere Arbeiten. Da der PC per definitionem als eine Maschine für eine einzelne Arbeit eines einzelnen Benutzers konzipiert wurde, konnten die meisten Adapter keine derartigen Interrupts liefern, obwohl der Prozessor selbst durchaus für diese Möglichkeit ausgerüstet war.

- Wird die Geschwindigkeit in immer kürzerer Zeit verdoppelt und wieder verdoppelt, so gelangt man sehr bald an eine Grenze, die nicht so einfach zu umgehen ist: Jedes System, das mit hochfrequenten Schwingungen arbeitet, strahlt Energie ab. Vorschriften bestimmen, wieviel das sein darf. Ein Computer darf keine anderen Geräte, seien dies nun Fernseher der Nachbarn oder medizinische Geräte im Spital, stören. Bis zu einem Takt von etwa 20 MHz lässt sich die Abstrahlung durch eine abschirmende Verkleidung noch gut in den geforderten Grenzen halten. Die Emission steigt aber im Quadrat der Taktgeschwindigkeit an. Die abgestrahlte Energie kommt ja auch nicht aus dem Nichts. Auch von einer anderen Seite zeigen sich Grenzen. Es sind die an sich richtigen Impulse, welche die Strahlungsenergie liefern. Irgendwann wird soviel Energie abgestrahlt, dass die Impulse auf den Leitungen verfälscht werden.

Eine Erhöhung der Taktrate – die

bequemste und billigste Reaktion auf die Forderung nach schnellen Maschinen – bringt also auf lange Sicht zusätzliche Probleme. Man sieht auch bei den anderen Forderungen, dass Flickwerk nur kurzfristig einzelne Probleme lösen kann. Es kommt also unweigerlich der Punkt, wo der ganze Komplex neu zu überdenken ist. Das Resultat einer solchen Gedankenarbeit: die Micro Channel Architecture.

## Entwurfskriterien und -ziele der Micro Channel Architecture

Die folgenden Ziele ergaben sich aus den im vorhergehenden Abschnitt dargelegten Einsichten für die Konzeption der IBM-Micro-Channel-Architektur:

- Die Geschwindigkeit des Gesamt-Systems soll der wachsenden Geschwindigkeit des Prozessors angepasst werden können.
- Ein System, das auf dem Micro Channel basiert, soll mit 8-, 16 und 32-Bit-Prozessoren von unterschiedlicher Machart und mit unterschiedlicher Peripherie ausgerüstet werden können. Der Kanal soll also völlig unabhängig vom Aufbau seines Prozessors sein. Diese Flexibilität ist nötig, um dort, wo nur eine Datenbreite von 8 Bit (bei den meisten heutigen Ein-/Ausgabe-Adaptoren) benötigt wird, nicht unnötig Geld für Elektronik investieren zu müssen. Sie ist aber auch dort nötig, wo eine Maschine mit einem spezialisierten Prozessor spezielle Aufgaben erfüllen und trotzdem auf Adapter aus dem Angebot für Micro-Channel-Maschinen zurückgreifen möchte.
- Immer intelligentere Subsysteme sollen unterstützt werden können. Diese sollen zeitweise frei über den Bus verfügen (Multiprocessing) und so den Prozessor wirkungsvoll entlasten können.
- Der Bus muss Betriebssysteme und Anwendungen unterstützen, auf welche mehrere Programme und mehrere Tasks parallel zugreifen können. Diese Tasks sollen auch unabhängig voneinander in einem oder verschiedenen Prozessoren laufen können.
- Mehr-Benutzer-Anwendungen (Multi-User-Anwendungen) sollen auf einem oder auf mehreren Prozessoren möglich sein.
- Die Adapterkarten sollen identifi-

zierbar sein, damit das System mit seinen Programmen weitestgehend die Kontrolle über die gesamte Installation ausüben kann. Damit kann ein leidiges Problem der PC-Welt – die Notwendigkeit, mit Schaltern auf den Adapterkarten Parameter einzustellen zu müssen – eliminiert werden. Diese Flexibilität wird zudem verlangt, weil auch die Adapter immer mehr Möglichkeiten unterstützen müssen.

- Die Produkt-Zuverlässigkeit und die Daten-Integrität müssen verbessert werden, und die durch Kunden getätigten früheren Investitionen müssen möglichst weitgehend geschützt sein.

Daraus ergaben sich die Entwurfsprioritäten wie folgt:

- An erster Stelle steht die physische Sicherheit des Benutzers. Ein einziger Unfall auf Grund einer (richtigen oder falschen) Handhabung der Maschine wirkt sich extrem auf das ganze Geschäft aus.
- Daten- und System-Integrität stehen an zweiter und dritter Stelle. Das bedeutet, dass das ganze System und damit der Bus als integrierter Bestandteil des Systems dafür zu sorgen hat, dass die Daten und Auswertungen des Benutzers in Ordnung sind.
- Was nützt ein Computer, der alle Fehler in bester Weise handhabt, aber jeden Tag einmal völlig versagt? Die Zuverlässigkeit der Maschine muss ein sinnvolles Arbeiten gewährleisten. Dabei steigen die Anforderungen in dieser Hinsicht laufend.
- Über den Begriff «Kompatibilität» kann man Bände schreiben. Die Software ist heute unbestreitbar derjenige Systemteil, der die höchsten Kosten verursacht. Dies bedeutet, dass getätigte Investitionen in dieser Sparte in erster Linie zu schützen sind. Die Frage, ob nicht auch wenigstens ein Teil der vorhandenen Hardware wieder verwendbar gemacht werden könnte, ist verständlich. Einen teuren Video-Adapter z.B. würde man eigentlich ganz gerne ins neue System retten. Andererseits kann man die alte Maschine ohne Video-Adapter nicht einmal dem Sohn zum Lernen oder spielen geben. Zudem könnte der alte Adapter von den neuen Möglichkeiten des neuen Bus-Designs nicht profitieren.
- Erst jetzt werden die nötigen Funk-

tionen definiert und die Leistung festgelegt. An letzter Stelle der wichtigen Prioritäten stehen die Kosten, der Grund, dass IBM-Maschinen noch nie zu den Billig-Produkten gehört haben.

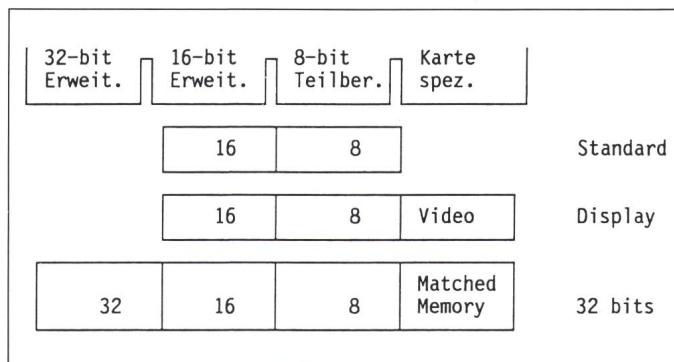
## Neue Schnittstellen zur Außenwelt

Maschinen mit Micro Channel haben eine völlig neu gestaltete System-schnittstelle. Die Steckplätze wurden neu definiert, und es wurde eine neue Standard-Adapter-Karte geschaffen. Ihr schmales Profil ermöglicht eine grazile Gehäuseform. Ein Arbeitsgerät soll schliesslich nicht die gesamte Arbeitsfläche belegen. Nachfolgend werden die übrigen wichtigen Interface-Merkmale kurz zusammengefasst.

- Maschinen, die auf einem Prozessor mit einer externen Datenbreite von 16 Bit aufbauen (Intel 80286, 80386SX), haben entsprechende 16-Bit-Steckplätze, in die auch 8-Bit-Karten eingesteckt werden können; Maschinen mit 32-Bit-Prozessoren (Intel 80386, 80486) haben zusätzliche 32-Bit-Steckplätze, in welchen auch 8- und 16-Bit-Karten verwendet werden können. Alle 32-Bit-Steckplätze sind mit dem Zusatz für das *Matched-Memory-Protokoll* ausgerüstet. Adapter, welche schnell genug sind, können deshalb mit weniger Zyklen Daten übertragen.
- Ein 16-Bit-Steckplatz in jedem System weist eine zusätzliche Schnittstelle zum Video-Subsystem auf. Hersteller von speziellen Videokarten können somit, soweit für die Funktion der neuen Karte sinnvoll, auf Dienste des Video-Subsystems des Micro Channels zurückgreifen. Auf der System-Platine vorhandene Elektronik muss deshalb auf dem Adapter nicht noch einmal implementiert zu werden.
- Die Architektur umfasst auch einen Audioteil, der mit einem programmierbaren Tongenerator verbunden ist. Adapterkarten können nun vor dem Endverstärker eigene Signale (Sprache, Tonsignale) einspeisen und so den Systemlautsprecher für die Kommunikation mit dem Bediener benützen.

Das physische Layout der Micro-Channel-Steckplätze (Datenleitungen und entsprechende Kontrollsignale) zeigt das Bild 2.

**Bild 2**  
Micro Channel-  
Steckplätze im  
Überblick



## Elektromagnetische Kompatibilität

Messungen an PCs und ATs haben gezeigt, wo und wie Störstrahlungen auftreten. Will man die Taktrate erhöhen, ohne dass die Störstrahlung ein untragbares Mass annimmt, sind Massnahmen nötig, welche darauf abzielen, die Strahlung möglichst gar nicht entstehen zu lassen. IBM-Entwickler haben die PS/2 und insbesondere die Kontakte ihrer Steckplätze so konzipiert, dass – sobald die Chip-Technologie das zu vernünftigen Preisen erlaubt – Taktraten bis zu 100 MHz möglich sind.

## Die Schalter der Adapter werden elektronisch gestellt

Wofür Schalter? Nehmen wir an, ein Adapter enthalte 256 Byte Speicherkapazität, die vom Prozessor mit normalen Adressen angesprochen werden sollen. Solange kein zweiter Adapter desselben Typs oder kein anderer Adapter mit denselben Adressen im gleichen System eingebaut ist, entsteht kein Problem. Ändert sich aber diese Situation, muss ein Adapter mit seinen Adressen ausweichen können. Jeder Adapter soll ja von den Programmen unabhängig angesprochen werden können. Bei PC- und AT-Adaptoren wird die Basisadresse über Schalter eingestellt. Wer die Betriebsanleitung gut liest, dürfte mit der Schalttereinstellung gut zurecht kommen. Probleme entstehen erst, wenn eine weitere Adapterkarte in eine bestehende Maschine eingebaut werden soll. Meist findet man dann die alte Betriebsanleitung nicht mehr, und damit auch nicht mehr die eingestellten Basisadressen. Vom System erfährt man leider auch nichts...

Programmable Option Select (POS) löst alle diese Probleme auf einen Streich: Jeder Adapterkartentyp für Micro Channel trägt eine durch den Computer lesbare – auf der ganzen Welt eindeutige – 2 Byte lange Identifikationsbezeichnung. Damit ein möglichst weitgehender Schutz der Hersteller gegen Duplizität gewährt werden kann, administriert IBM als kostenlose Dienstleistung weltweit alle Identifikationen. Mit allen Adaptoren für den Micro Channel wird eine Optionen-Diskette mitgeliefert, auf der alle möglichen Einstellungen gespeichert sind, für die der Adapter konstruiert worden ist. Bei der Installation im PS/2 wird nun der Inhalt dieser Datei mit den im System gespeicherten Daten von bereits früher installierten Geräete-Teilen verglichen und dem neuen Adapter ein Arbeitsbereich zugeteilt. Diese Zuteilung wird im System permanent gespeichert. Bei jedem Aufstarten werden nun die Konfigurationsdaten an die Adapter übermittelt und dort in den sogenannten POS-Registern gespeichert. Jede Änderung kann mittels der Programme auf der Referenzdiskette auf dem Bildschirm kontrolliert und bei Bedarf geändert werden: Jedes PS/2 mit Micro Channel kennt deshalb seine eigene Konfiguration und eliminiert so jedes Problem im Zusammenhang mit Neuinstalltionen oder Änderungen. Das Bild 3 zeigt, wie die POS-Register definiert sind.

Die POS-Bytes jedes Adapters können von jedermann versuchsweise gelesen werden. Beim PS/2-Modell 80-71 sind dazu die folgenden Schritte nötig (bei einem anderen System kontrolliere man die Angaben im entsprechenden technischen Handbuch): Über den Ein-/Ausgabe-Port 096H (H für hexadezimal), wird ein Register auf der Systemplatine angesprochen.

I/O Adresse	Byte-Bezeichnung
0100	Karten-Identifikation (niederwertiges Byte, obligatorisch). Register 0
0101	Karten-Identifikation (höherwertiges Byte, obligatorisch). Register 1
0102	Byte 1 für die Optionen Auswahl. Register 2
0103	Byte 2 für die Optionen Auswahl. Register 3
0104	Byte 3 für die Optionen Auswahl. Register 4
0105	Byte 4 für die Optionen Auswahl. Register 5
0106	Unter-Adressierung* / Statusmeldung. Register 6
0107	Unter-Adressierung* / Statusmeldung. Register 7

\* Unter-Adressierung erlaubt die Angabe eines Bereichs von adapter-residentem Speicher. Damit können über POS vom Hauptsystem z.B. Adapter-Programme oder Grunddaten geladen werden.

Selektierte Kanal Position	Bit	7	6	5	4	3	2	1	0
Keine	0	0	0	0	0	X	X	X	X
1	0	0	0	0	1	0	0	0	0
2	0	0	0	0	0	1	0	0	1
3	0	0	0	0	1	0	1	0	0
4	0	0	0	0	0	1	0	1	1
5	0	0	0	0	0	1	1	0	0
6	0	0	0	0	0	1	1	0	1
7	0	0	0	0	1	1	1	0	0
8	0	0	0	0	1	1	1	1	1

Bild 4 Einteilung Ein-/Ausgabe-Port 096H

Durch geeignetes Setzen seiner einzelnen Bits kann das System in den Setup-Modus gesetzt werden. Danach können die POS-Informationen jedes Steckplatzes einzeln gelesen oder geändert werden. Durch jedes nachfolgende Ansprechen der Ein-/Ausgabe-Ports 0100H bis 0107H werden die POS-Bytes des spezifizierten einzelnen Adapters angesprochen. Dabei ist folgendes zu beachten (Bild 4):

- Bit 7 muss 0 sein; andernfalls werden die POS-Bytes von Teilen der System-Platine angesprochen.
- In die Bits 4, 5, 6 wird 0 geschrieben, das System gibt aber immer 1 zurück.
- Für die normale Arbeit ist 00H in den E/A-Port 096H zu schreiben.

Mit dem Basic-Programm (Bild 5) können die Identifikationen sämtlicher Adapter im erwähnten Modell 80-71 gelesen werden. Da die Datenleitungen unangesteuert über Widerstände auf + 5 Volt liegen, wird ein leerer

Bild 3  
POS-Register

Installationsvorgang festgelegt. Die Darstellung in Bild 6 stellt deshalb eine Möglichkeit dar, wie die POS-Bytes 1 bis 4 verwendet werden können.

## Interrupts

Wie weiter oben dargelegt, muss der Prozessor nicht für jede Arbeit im Computer zuständig erklärt werden. Adapter können einen Teil der Arbeit übernehmen. Sobald diese ihren Teil beendet haben (zum Beispiel das Drucken einer Zeile), können sie mittels eines Unterbrechungssignals die Aufmerksamkeit des Prozessors auf sich lenken. Die «Interrupt Service Routine» des Device Driver Programms kann sodann dafür sorgen, dass in unserem Beispiel eine neue Zei-

Bild 5  
Programm zum  
Lesen der  
Adapter-Identifika-  
tionen

```

10 FOR X=&H8 TO &HF      'Zähler X von 1000B bis 1111B
15 OUT &H96,&H0      'Port 096: 0000 0000B
20 OUT &H96,INP(&H96) OR X  'Port 096: setzen der Kanalnummer
30 PRINT X-&H7,HEX$(INP(&H101));HEX$(INP(&H100)) 'Ausgabe Id.
40 NEXT X      'Schlaufen-Ende
50 OUT &H96,&H0      'SETUP-Ende

```

Steckplatz mit 0FFFFX ausgewiesen. Es sind nur wenige Bits durch die Architektur in ihrer Bedeutung vorgegeben. In den Registern können deshalb sehr viele adapterabhängige Informationen definiert werden. Diese werden vom «Adapter Definition File» als Möglichkeiten offeriert und durch den

le zum Drucken in den Speicher des Adapters transferiert wird.

In der «alten» PC-Welt reagiert ein Prozessor auf Veränderungen der Spannung (Flankentriggierung) auf einer Interrupt-Leitung (Bild 7). Das bedeutet, dass

- ein Störimpuls auf der Leitung von

Bild 6  
Mögliche Einteilung  
der POS-Register

0102	R	O	M	S	I	O	A	Cen	← Card Enable Bit
0103	F	U	N	F	A	R	B	L	
0104	-----	f	r	e	e	f	o	r	m-----
0105	CCK	S	---	f	r	e	e	f	m---

ROMS: 1 von 16 alternativen Basisadressen für ROM/RAM auf dem Adapter (innerhalb 0C0000 bis ODFFFF des System-Speicherbereichs). Soweit benötigt.  
IOA: 1 von 8 Ein-/Ausgabe-Adressen, auf welche der Adapter reagieren soll.  
Cen: Card-Enable/N\_Sleep: 1B für "Adapter in Betrieb", 0B, wenn Adapter ausser Betrieb (1)  
FUN: Optionen für die Karten-Funktion  
F: Fairness: 1 eingeschaltet, 0 ausgeschaltet  
ARBL: Arbitration-Level; Prioritäts-Steuerung  
CCK: Channel Check: Wenn 1, verwendet der Adapter diese Möglichkeit  
S: Wird CCK verwendet: 0, wenn in POS Registern 6 und 7 ein Status (oder seine Adresse) gespeichert wird (2)  
(1) Bit ist in Architektur beschrieben (obligatorisch)  
(2) Wird Channel Check auf dem Adapter implementiert, sind für CCK und S Register 5, für den Status selbst Register 6 und 7 zu verwenden.

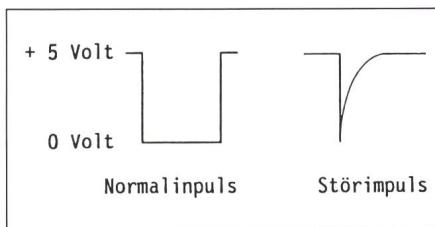


Bild 7 Impulsformen auf PC Interrupt-Leitungen

aussen einen irrtümlichen Interrupt auslösen kann

- ohne spezielle Schaltungs-Techniken eine Interrupt-Leitung nur durch einen einzigen Adapter benutzt werden kann.
- Systeme, welche mehr Durchsatz bringen sollen, Möglichkeiten aufweisen müssen, die eine gleichzeitige, problemlose Benutzung einer Leitung durch mehrere Adapter ermöglichen.

Die Micro-Channel-Architektur definiert niveausensitive Unterbrechungen (Level Sensitive Interrupts). Der Interrupt gilt als pendent, solange die Leitung am Controller oder Prozessor aktiv (low) ist (Bild 8). Die Hardware des Adapters und die Software des Betriebssystems müssen so gestaltet sein, dass die Quelle der Unterbrechung in jedem Fall eruiert werden kann. Jeder Adapter, der Interrupt signalisiert, hält hardwaremäßig die Interrupt-Leitung auf low, bis die Software diesen Interrupt Request durch Überschreiben des entsprechenden Registers zurücksetzt. Eine spezielle Software (Device Driver) kann nun auf die Register des Adapters zugreifen und

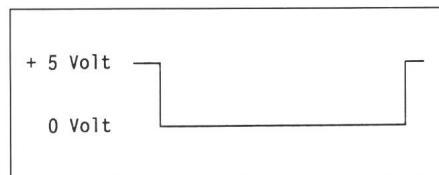


Bild 8 Interruptsignal gemäss Micro Channel Definition

eruiert, welcher Adapter den Interrupt ausgelöst hat. Dies geschieht unter OS/2 wie in Bild 9 gezeigt ist:

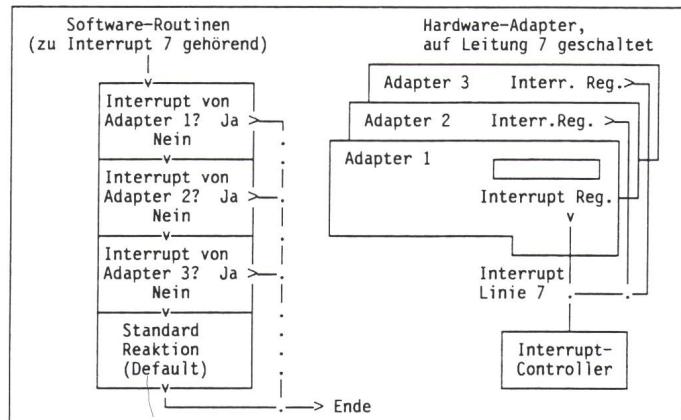
1. Ein oder mehrere Adapter setzen ihre Interrupt Linie auf low (0 V). In unserem Beispiel ist dies die gemeinsame Leitung 7.
2. Der Interrupt Controller empfängt den Interrupt und veranlasst,

dass das Prozessorprogramm zum ersten Mitglied in der Kette jener Unterbrechungsroutinen (hier Adapter 1) verzweigt, welche zum Interrupt 7 gehören.

3. Diese Interrupt-Routine liest als erstes das Interrupt Register «ihres» Adapters, um herauszufinden, ob der eigene Adapter Urheber der Unterbrechung war. Wenn ja, wird die Unterbrechung bearbeitet und anschliessend das Interrupt-Register auf dem Adapter gelöscht. Dadurch deaktiviert der Adapter seinen Interrupt Request. Das Programm kehrt unmittelbar anschliessend zum normalen Verarbeitungsprogramm zurück. Reagiert der Adapter negativ (sein Interrupt Register ist nicht gesetzt), so erhält die nächste Routine der Kette die Kontrolle. Sollte kein Adapter positiv rea-

Die späteren Prozessoren (bei Intel ab 80286) sind denn auch für Multitasking eingerichtet. Das bedeutet nichts anderes, als dass der Prozessor zwischen den im Speicher residenten Programmen zu bestimmten, vom Betriebssystem vorgegebenen Augenblicken, umschalten und dort weitermachen kann, wo er nach Ablauf des vorangegangenen Zeitabschnitts aufgehört hat. Multitasking ist also nichts anderes als eine intelligente Art und Weise, die vorhandene Leistung des Prozessors nach Prioritäten auf die anstehenden Arbeiten aufzuteilen. Wie das Betriebssystem diesen Mechanismus einsetzt, bleibt ihm überlassen. So ist DOS ein Single-Tasking-Betriebssystem, das einen allfällig vorhandenen Schaltmechanismus des Prozessors nicht auszunützen vermag. OS/2

Bild 9  
Interrupt Service



gieren, wird OS/2 durch eine «Default-Unterbrechungs-Routine» die Situation bereinigen und die Kontrolle dem unterbrochenen Anwendungsprogramm zurück geben.

### Mehr Freiheit für Sklaven

Weiter oben wurde dargelegt, dass in einer Maschine vom PC-Typ der Prozessor eine permanente Kontrolle über die gesamten Maschine ausübt. Er ist der Chef, die anderen sind Sklaven. Danach wurde am Beispiel einer asynchronen Datenübertragung gezeigt, dass die Belastung der Maschine haarsträubend tief sein kann und dass eine Verbesserung zu erreichen wäre, wenn man über einen Interrupt jeweils das Ende einer durch die Peripherie zu leisenden Teilarbeit signalisieren könnte.

stellt dem Benutzer über ein ausgeklügeltes System sowohl Multiprogramming (die Fähigkeit also, mehrere voneinander unabhängige Arbeiten «gleichzeitig» abzuarbeiten), als auch Multitasking (die Fähigkeit, innerhalb eines Programmes verschiedene Teile «gleichzeitig» zur Ausführung zu bringen) zur Verfügung. OS/2 ist ein Betriebssystem für einen einzelnen Benutzer (Single User, Multiple Tasks). IBM AIX-PS/2, eine Unix-Implementation, wendet den durch die Hardware zur Verfügung gestellten Schaltmechanismus auch dazu an, am gleichen System mehrere Benutzer zu betreuen (Multi User System). Man beachte, dass das Wort «gleichzeitig» eigentlich immer in Anführungszeichen zu setzen wäre, weil die vermeintliche Gleichzeitigkeit auf einem raffinierten Schaltalgorithmus beruht. Ein

einzelner Prozessor kann zu einem gegebenen Zeitpunkt natürlich nur ein Programm bearbeiten.

Will man diesbezüglich weitere Verbesserungen erzielen, müssen mehrere Prozessoren sinnvoll so eingesetzt werden können, dass sie einerseits unter der Kontrolle des Hauptprogramms stehen, andererseits aber Teile der Arbeit selbstständig ausführen und den Hauptprozessor wirklich entlasten können: Das ist Multiprocessing.

## Multiprocessing

Multiprocessing ist eines der Hauptanliegen der Micro-Channel-Architektur. Die Konzepte der «Arbitration» und «Fairness» setzen sie in die Lage, aus einer Maschine ein eigentliches Multiprozessorsystem zu machen, wobei natürlich auch das Betriebssystem seinen Teil zu übernehmen hat.

Als Beispiel diene das folgende Szenario: In einem System zur Ausgabe von Audio-Visions-Sequenzen sei ein Adapter eingebaut, der erstens gespeicherte Sätze in synthetischer Sprache über einen Lautsprecher ausgeben kann, zweitens in der Lage ist, unbewegte Bilder (z. B. Dias), welche in Pixel-Form gespeichert sind, auf den Bildschirm zu bringen. Der Hauptprozessor vermag die geforderte Geschwindigkeit nicht zu entwickeln. Spezialisierte Prozessoren, welche nur Video-Bilder erzeugen oder Sätze formulieren können, sind nötig. Diese Chips sind auf Grund ihrer Spezialisierung um Größenordnungen schneller, als ein Tausendsassa-Prozessor, wie ihn die Zentraleinheit benötigt. In einem Computer macht Sinn, was eine gut geführte Unternehmung auszeichnet: Das Erkennen und optimale Ausnützen der Fähigkeiten ihrer Mitarbeiter.

Das Hauptprogramm wird im Hauptspeicher des Systems die elektronische Darstellung der benötigten Bilder und Sätze bereitstellen. Bei einem echten Multiprocessing übernimmt der Adapter jetzt die Daten; der Hauptprozessor kann anderweitige Arbeit ausführen. Sein Auftrag an den Adapter lautet: Hole die Daten ab und präsentiere sie mittels Bild und Ton.

Der Adapter meldet sich als erstes beim Micro Channel (Bild 10) und stellt seine Forderung für eine Übertragung; er setzt seine PREEMPT-Leitung auf Low (0 V). Dies veranlasst den Central Arbitration Control Point, die ARB-/GNT (Arbitration/

-Grant) Leitung auf High (+5 V) zu setzen und dadurch den Kanal für den Wettbewerb um den nächsten Transfer frei zu geben. Jeder Adapter, ob einfacher DMA-Sklaven-Adapter (DMA Slave) oder komplexer Bus Master, der einen Transfer anstehen hat, gibt nun seine – bei der Installation mittels dem Programmable Option Select, POS, eingestellte – 4-Bit-Arbitration-Identifikation auf die gemeinsamen Leitungen. Diese signalisiert mit der Identifikation zugleich seine Kanal-Priorität.

Jeder Adapter (und *nicht* der Kanal!) prüft nun selbst, ob er die höchste Priorität (sie entspricht der niedrigsten Arbitration Identification) unter den Bewerbern hat. Alle verlierenden Adapter ziehen sich sofort vom Wettbewerb zurück und warten den nächsten «Arbitration Cycle» ab. Der gewinnende Adapter kann nun ein, oder mehrere Zeichen (Bursts) über den Micro Channel übertragen, sei dies mit Unterstützung des DMA-Controllers (für DMA-Sklaven), sei dies indem der Adapter als nunmehriger Bus Master sämtliche Adress-, Daten- und Kontrollsiegel generiert und auch die Status-Meldungen des Partners entgegennimmt und verwertet.

Die Dauer, während der ein Adapter den Kanal benutzen darf, hängt von vielen Faktoren ab. Hat ein weiterer Adapter durch Aktivierung seiner PREEMPT-Leitung seinen Bedarf angemeldet, darf jener noch während höchstens 7,8 µs den laufenden Transfer abschliessen. Kommt er dieser Vorgabe nicht nach, wird der Arbitration Control Point einen neuen Arbitrations-Zyklus einleiten und aktiv halten. Ein spezieller Interrupt wird signalisiert und die entsprechende Service-Routine kann nun den fehlerhaften Adapter ausser Betrieb setzen (Bild 6: Cen = 0). Ohne Erweiterung des Konzeptes könnte ein Adapter mit hoher Priorität den Kanal monopolisieren, weil er bei jeder Arbitration auf Grund seiner Priorität den Kanal wieder gewinnen müsste. Für Adapter mit

niedrigerer Priorität wäre Datenverlust (Überlauf) oft nicht vermeidbar.

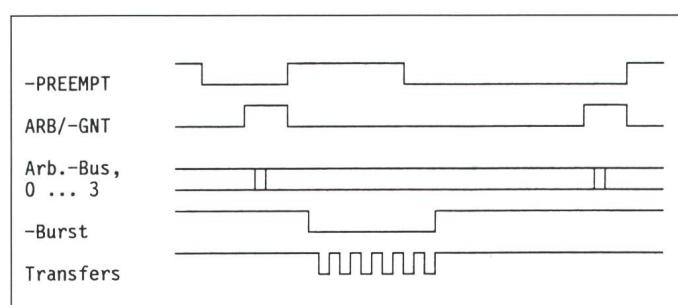
Das Konzept der *Fairness*, dem jeder Arbiter folgen sollte, verlangt, dass ein Adapter, welcher Daten übertragen hat, sich nicht mehr um den Kanal bewirbt, bis jeder angemeldete Adapter einmal den Kanal benutzen konnte. Diese Fähigkeit kann über POS ein- und ausgeschaltet werden. Das Konzept der Arbitration wird auch für die Speicher-Auffrischung (Memory Refresh) benutzt. Durch die Vergabe von *Überklassen* bei den Arbitration Levels (-2 für Refresh) kann sichergestellt werden, dass lebenswichtige Prioritäten nicht unterbrochen werden

## Übertragungsprotokolle

Bereits bei der Besprechung des Arbitrations-Konzeptes wurde festgestellt, dass ein Adapter (entweder DMA oder Bus-Master Adapter) selbst herausfinden muss, ob und zu welchem Zeitpunkt er über den Bus verfügen kann. Damit wurde ein erstes Konzept dargelegt, das die Unabhängigkeit des Adapters vom Gesamt-System aufzeigt: Ein intelligenter Adapter soll Teile der Aufgabe selbst übernehmen können. Ein Sklaven-Adapter auf der anderen Seite, muss auf Aufrufe reagieren können. Aufruf und Antwort müssen transparent und unabhängig davon erfolgen, ob der Sklave vom Haupt-Prozessor, von einem intelligenten Adapter oder vom DMA-Controller angesprochen wird.

Die Micro-Channel-Architektur definiert deshalb einen Zyklus, nach dem sämtliche daran angeschlossenen Geräte zu arbeiten haben. Ausnahmen sind hier die Regel: Die an einem Transfer Beteiligten – immer ein Master (Prozessor, DMA-Controller, Bus Master) und ein Sklave (Speicher, Ein-/Ausgabe Adapter) – werden den Übertragungsmodus unter sich vereinbaren. Diese Vereinbarung geschieht über Protokolle, mittels derer die Fä-

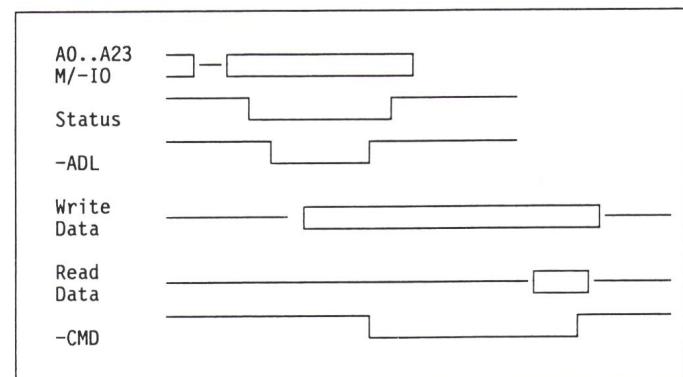
Bild 10  
Arbitration und  
Burst-Transfer



higkeiten der Partner abgeglichen und der effizienteste gemeinsame Modus gewählt wird. Hier liegt einer der grössten Vorteile des ganzen Konzeptes: Die Definitionen machen den Kanal unabhängig von jedem Prozessor und von jedem Adapter. Nur so ist es verständlich, dass nicht nur die IBM PS/2, sondern auch die Risc-Systeme/6000 über diesen Kanal verfügen können. Hier zeigen sich nun die umfassenden Möglichkeiten für Hersteller von Adapterkarten: Zusatzkarten, für PS/2 hergestellt, laufen unverändert auch auf den Risc-Systemen. Die Architektur des Kanals beschreibt die Prozeduren, nach welchen gearbeitet werden kann. Beide Partner eines Datenaustausches handeln über definierte Signale ohne Intervention des Prozessors oder des Kanals miteinander aus, wie sie am besten über den Kanal miteinander kommunizieren. Die Raffinesse: Wenn ein Teilnehmer zu Beginn eines Transfers seinem Partner keine Information über den Transfer-Modus gibt, wird ein sogenannter *Default Cycle* initialisiert. Dieser läuft nach einem Minimal-Protokoll ab, nach dem ein Systemteil zu arbeiten imstande sein muss. Die Intelligenz liegt also nicht beim Kanal, sondern bei seinen Benutzern. Damit ergibt sich ein unerhört grosser Spielraum für Hersteller von Zusatzgeräten. Sie bauen ihre Geräte so, wie es ihr Projekt erfordert, im Minimum aber so, dass ein voller «Default Cycle» gefahren werden kann. In Bild 11 ist ein vereinfachter Basis-Übertragungs-Zyklus dargestellt.

Ein Zyklus beginnt, indem der Master (z.B. der Prozessor) die Adresse und gleichzeitig das Kontrollsiegel M-/IO (1 für Speicher-, 0 für den Ein-/Ausgabe-Zyklus) setzt. Anschliessend gibt er über die Status-Signale bekannt, ob dies eine Lese- oder eine Schreiboperation ist. Das Signal -ADL (Address Decode Latch) zeigt an, dass die Adresse und die weiteren Kontrollsiegel stabil sind und durch den Partner benutzt werden können. Da jeder Micro-Channel-Zyklus die Datenphase mit der Adressphase des nächsten Zyklus überlappen lässt, ist dies der Moment, an dem jene Signale auf dem Adapter gespeichert werden müssen. -ADL liefert die nötige Zeitinformation. -CMD wird jetzt aktiv und gibt den Beginn des Schreibimpulses an. Zum Ende dieses Impulses ist der gesamte Zyklus beendet. Die zweite Flanke von -CMD kann demnach gebraucht werden, um Daten, welche der

**Bild 11**  
**Basis-Übertragungs-Zyklus (Auszug)**



Prozessor (z.B. aus einem Speicherplatz) liest, in die Register aufzunehmen.

Sieht man sich die Zeitvorgaben der Definition in den technischen Handbüchern zu den einzelnen Systemen der PS/2-Familie an, kann man sehen, dass mit dem traditionellen Default Cycle, der eine Länge von 200 ns hat, 5 Millionen Transfers pro Sekunde durchgeführt werden können. Legt man 32 Bit Datenbreite zu Grunde, können zwei Partner mit dem Default Cycle 20 Millionen Zeichen (20 MByte) pro Sekunde austauschen.

Die IBM Micro Channel-Architektur definiert eine Vielzahl von Leitungen zum Austausch von Protokollen zwischen den Adapters, welche jedem Partner ermöglichen, mit dem anderen die für einen Transfer geltenden Regeln auszuhandeln. Einzelne Parameter (z.B. die Datenbreite von 8, 16 und 32 Bit, die Verlängerung des Zyklus durch Warte-Zyklen) gehören zum Default Cycle, andere sind als Erweiterungen dazu aufzufassen. Im folgenden sind die wichtigsten davon dargestellt:

**1. Datenbreite:** Die Architektur definiert Datenbreiten von 8, 16, 24 und 32 Bit. Somit können über die Datenleitungen maximal 4 Bytes parallel übertragen werden. Die dazugehörenden Protokollsignale geben an:

- CD DS 16 (Card Data Size 16): 16 Bit
- CD DS 32 (Card Data Size 32): 32 Bit parallel

Es besteht zudem ein wichtiges Protokoll, das die Verbindung zwischen Partnern jeder beliebigen Datenbreite ermöglicht. Dadurch ist ein Transfer zwischen z.B. einem 32-Bit Master und einem 8-Bit Ausgabe-Adapter ohne spezielle Elektronik möglich. Adapter älterer Bauart (sie tendieren zu einfacherem Design) werden auch in zukünftigen Systemen mit all ihren Möglichkeiten laufen, sofern ihre Hersteller sich an die Definitionen der Architektur gehalten haben.

**2. Übertragungs-Geschwindigkeit:** Es

gibt Adapter und Maschinen (und das sind aus Kostengründen heute die Mehrzahl), welche nicht die volle Geschwindigkeit des Default Cycle zu leisten vermögen. Die Micro-Channel-Architektur definiert für beide Richtungen Varianten, deren Anwendung durch Signalprotokolle zwischen den beiden Partnern vereinbart werden kann. So kann ein Sklave seinem Master seine Langsamkeit durch CD\_CHRDY signalisieren. Der Zyklus kann dadurch auf maximal 3,5 µs verlängert werden, was einer niedrigen Datenrate von noch knapp 300 kbyte/s entspricht.

Obwohl heute aus Kostengründen die meisten Geräte nicht einmal die Datenrate des Default Cycle ausnützen, definiert die Micro-Channel-Architektur die verschiedensten Protokolle und zeigt auf, dass die Kapazität der sie anwendenden Maschinen auf Jahre hinaus nicht erschöpft sein wird.

Der *Matched Memory Cycle* ermöglicht schnellen 32-Bit Geräten, den Zyklus von 200 auf 125 Nanosekunden zu verkürzen. Dadurch wird die normale Datenrate von 20 auf 32 Mbyte/s erhöht. Wie weiter oben dargelegt wurde, zeigte die Erhöhung der Taktrate eines Computers nur in wenigen Fällen befriedigende Resultate, wenn es darum geht, die Leistung der Maschine insgesamt zu erhöhen. Micro Channel Architecture definiert deshalb Protokolle, welche eine drastische Erhöhung des gesamten Durchsatzes ermöglichen, ohne die Taktrate anzutasten. Noch einmal: Jedes Partner-Paar definiert «seinen» Transfer nach der effizientesten, durch beide Partner beherrschten Methode.

Die Darstellung in Bild 12 gibt eine

Adress-Bus	Adresse	Adresse	Adresse
Daten-Bus	Daten	Daten	Daten
Adress-Bus	Adresse		
Daten-Bus	Daten	Daten	Daten
Adress-Bus	Adresse	Daten	Daten
Daten-Bus	Daten	Daten	Daten

**Bild 12**  
Übertragung über  
den Kanal im  
Basis-Zyklus

**Bild 13**  
Streaming Data  
Prozedur, 32 Bit

**Bild 14**  
Streaming Data  
Prozedur, 64 Bit

Übersicht über den Verkehr auf den Daten- und Adressleitungen. Jeder Zyklus ist 200 ns lang. Im ersten Teil werden auf den Adressleitungen die Adressen, im zweiten auf den Datenleitungen die Daten (je bis zu 32 Bit) übertragen. Das ist der Default-Zyklus. Die erreichbare Geschwindigkeit: 20 Mbyte/s. Wird zwischen den Partnern einer Übertragung die «Streaming Data Procedure» vereinbart (Bild 13), so wird nur eine einzige Basisadresse an die Adressleitungen angelegt. In der Folge werden in der Adressphase des Zyklus statt einer (je-weils um 1 erhöhte) Adresse, zusätzliche Daten übertragen. Der Sklave hat also nach der Übernahme der Basisadresse alle Adressen zur Übernahme von Daten während der gesamten Übertragung selbst zu liefern. Damit werden ab dem 2. Zyklus pro 100 Nanosekunden 32 Bit übertragen, was die erreichbare Geschwindigkeit auf 40 Mbyte/s erhöht.

Da bei Ausführung des Streaming Data-Protokolls die 32 Adressleitungen nach Absenden der ersten Adresse keine Informationen mehr transportieren müssen, besteht kein Grund diese Ressource brach liegen zu lassen. Es kann durchaus Hardware definiert werden, die ab dem 2. Zyklus (der erste ist immer ein Default Cycle, während dem das Protokoll für die nachfolgenden Cycles ausgehandelt wird) auch auf den Adressleitungen Daten überträgt (Bild 14). Diese Technik erlaubt bei einer Datenbreite von 64 Bit (8 Byte) eine Durchsatzrate von 80 Mbyte/s. Dies sind keine Zukunftsvisionen. Da die Partner an einem Transfer selbst bestimmen, wie rasch und wieviele Bytes parallel übertragen werden, sind hauptsächlich sie für den Gesamt-Durchsatz des Systems verantwortlich.

Eine noch höhere Datenrate wird dann erreichbar sein, wenn Systeme

auf dem Markt sind, welche jeden Zyklus statt in 200 in 100 Nanosekunden abzuarbeiten vermögen. Micro Channel Architecture kann auch mit den Systemen von morgen umgehen. Die potentielle Geschwindigkeit liegt dort bei max. 160 MByte/s.

## Sicherheit

Die Frage, wie ein Transfer über den Bus geprüft werden kann, ist auf mehreren Ebenen zu beantworten:

Solange der Prozessor der einzige Master im gesamten Computersystem ist, können Fehlermeldungen jedes Adapters durch Programmteile des Betriebssystems ausgewertet werden. Verschiedene Programme stehen dabei zur Verfügung:

- Installation: Ist ein Adapter im System eingesetzt, aber nicht installiert, so wird er bereits während des Systemstarts ausser Betrieb gesetzt. Er wird erst berücksichtigt, wenn seine korrekten Daten im POS bekannt sind.
- Der Default Cycle definiert, dass jede Karte innerhalb einer sehr kurzen Zeit nach der Adressierung ein *Card Select Feedback*-Signal liefert. Dies gestattet dem Device Driver-Programm, das Vorhandensein «seines» Adapters zu prüfen.
- Auf jedem Adapter für den IBM Micro Channel muss *Channel Check* installiert werden (POS Byte 5). Zeigt sich bei einem Adapter ein Fehler (z.B. Parity Check bei einem Speicher), so setzt er dieses Bit. Über POS Byte 5 wird zudem definiert, ob eine Statusmeldung (in diesem Fall eine Fehlerartmeldung des Adapters) erhältlich ist und aus den POS Bytes 6 und 7 gelesen werden kann. Der Channel Check löst einen NMI (Non Maskable Interrupt) aus, dessen Service-Routine den entstandenen Fehler bearbeiten muss.

Sobald ein Bus Master im Micro Channel selbständig Übertragungen vornimmt, steht keine Interrupt-Routine des Hauptprozessors zur Bearbeitung von Fehlermeldungen zur Verfügung. Deshalb ist der Bus Master als Subsystem auch zuständig dafür, von seinem Partner gemeldete Fehlersituationen zu bearbeiten und allfällige Wiederholungen selbständig durchzuführen. Erst, wenn die Intelligenz des Bus Masters nicht ausreicht, den Fehler zu bereinigen, meldet er dies dem Hauptsystem (Channel Check und Status).

Die Micro-Channel-Architektur stellt für die Fehlererkennung durch Subsysteme ein weiteres Konzept zur Verfügung, das synchron zu jeder Übertragung mitläuft und die laufende Prüfung der Datenintegrität ermöglicht, den *Synchronous Channel Check*.

Jede Übertragung über den Micro Channel (Adressen und Daten) kann zudem auf korrekte Parität überprüft werden. Die Architektur weist Leitungen und Protokolle auf, welche für jedes der bis zu 8 parallel übertragenen Bytes je ein Paritätsbit liefert. Der Sender generiert die Bits, der Empfänger überprüft sie und stellt auf diese Weise die korrekte Übertragung weitestgehend sicher. Bus Master haben die Kapazität, Datenmengen in der Größenordnung von MByte selbständig, ohne Mithilfe des Hauptprozessors, zu übertragen. Eine Kontrolle ist deshalb auf dieser Ebene nach jedem Zeichen unbedingt notwendig.

Teil 2 folgt in Heft 23/90

## Literatur

- [1] IBM PS/2 Seminar Proceedings, Vol. 5 Nr. 3, May 1987.
- [2] IBM Personal System/2 Models 50, 60, 80 Micro Channel Architecture, Hardware Features and Design Considerations, IBM Nr. G360-2637.
- [3] IBM PS/2 Technical Reference Manuals, all Micro Channel Systems.
- [4] A Closer Look at Micro Channel Architecture, IBM Nr. GK14-9748.
- [5] IBM Personal Systems Technical Journal, Issue 4, 1989, Special Micro Channel Issue, IBM Nr. G325-5004.
- [6] IBM Micro Channel Architecture Bus Master, IBM Nr.: GG24-3477.
- [7] Chips and Technologies, Inc., 82C614 Data Book. March 1 1990.
- [8] Chips and Technologies, Inc., Application Note: 82C614 Bus Master Micro Chip 20 Mbyte/s Micro Channel Bus Master SCSI Adapter Designed with the 82C5086 SCSI-Bus Interface Controller, March 1 1990.
- [9] Intel 82325 Bus Master Interface Controller Chip Set for Micro Channel Architecture, January 1990, Order Number 290 271-001.
- [10] Intel Datenbücher zu den Mikroprozessoren 8088, 8086, 80286, 80386 SX, 80386 DX, 80486.