

Referat

zu dem Thema „Peripherie“

vom 30.10.1998

Was ist Peripherie.....	2
Das BIOS.....	2
Und ab die POST.....	3
Kommunikation mit Peripherie.....	4
Kommunikation im Polling- oder Interruptverfahren.....	5
Das Pollingverfahren.....	5
Das Interruptverfahren.....	5
Disketten und Festplatten.....	6
Aufbau einer Diskette.....	6
Aufbau von Festplatten.....	7
Aufzeichnung von Informationen (FM, MFM, RLL).....	9
Das FM-Verfahren.....	9
Das MFM-Verfahren.....	10
Das RLL-Verfahren.....	10
Controller.....	11
ST506 / 412.....	11
ESDI.....	11
SCSI.....	11
IDE / EIDE-Controller (auch ATA / ATAPI).....	12
Festplattenformate:.....	12
Der MBR.....	13
Der Bootsektor.....	13
Die FAT.....	13
Die unterschiedlichen BUS-Systeme:.....	14
Der AT-BUS / ISA-BUS.....	14
Von MCA, VLB und EISA.....	14
Der PCI-Bus.....	15
AGP.....	15
Graphikkarten.....	15
Die MDA-Karten.....	16
Die CGA-Karten.....	16
Die Hercules Graphics Card.....	17
Die EGA-Karte.....	17
Die VGA-Karte.....	18
Die Super-VGA-Karten.....	19
Die MCGA.....	19
Die 8514/A.....	20
TIGA.....	20
3Dfx / Voodoo / 3D-Beschleuniger-Karten.....	21
Literaturverzeichnis:.....	21

Was ist Peripherie

Ein Rechner besteht aus einer zentralen Komponente, dem sog. Mainboard oder Motherboard, auf welchem folgende Elemente untergebracht sind:

Heute:	Von Neumann:
Speicher	Speicher
Cache-Speicher	
CPU	Rechenwerk
FPU	
Busleitungen	Steuerwerk

Man faßt diese Komponentengruppe unter dem Begriff *Zentraleinheit* zusammen.

Nach der Von-Neumann-Architektur fehlen in der Aufzählung jedoch noch die Elemente *Eingabewerk* und *Ausgabewerk*, welche dann zu der Peripherie zählen.

Unter Peripherie versteht man die Zusammenfassung aller Geräte, die an die Zentraleinheit einer Rechanlage angeschlossen werden können. Man teilt Peripherie hinsichtlich ihrer Funktion ein in:

- *Speicherperipherie*: Zu dieser Gruppe der Peripherie gehören alle Massenspeichergeräte wie Bandlaufwerke, Festplatten, CD-ROM.
- *Ein/Ausgabepерipherie*: Zu dieser Gruppe zählt man alle Geräte, die zur Daten Ein- und Ausgabe geeignet sind, also Drucker, Tastatur, Bildschirm.
- *Steuerperipherie*: Zu dieser Gruppe zählt man alle Geräte, welche die Ein- und Ausgabepерipherie mit Steuersignalen versorgt sowie Geräte zur Steuerung des Datenflusses in Rechnernetzen. Dazu gehören auch die Disketten- und Festplattencontroller.

Das BIOS

Um mit Peripherie kommunizieren zu können, sind spezielle Programme notwendig, die auf das Zeitverhalten des Mainboards abgestimmt sind und daher auch mit diesem geliefert werden. Die Zusammenfassung dieser Programme befindet sich in einem EPROM auf dem Motherboard und wird BIOS genannt. (BIOS steht für Basic Input Output System.) Die Abarbeitung dieser Programme übernimmt ebenfalls die CPU.

Das BIOS ist die Standardschnittstelle zwischen Peripherie und Zentraleinheit und besitzt in seiner Funktion kleine Routinen die auf Standards zurückgreifen, um ein paar Zeichen auf dem Bildschirm zu schreiben, einen Sektor von der Platte zu lesen und so weiter.

Dabei spielt das Zeitverhalten eine wesentliche Rolle, was der Grund war, warum auf damaligen Rechnern pro Mainboard auch ein spezielles BIOS benötigt wurde. Die Kommunikation mit Peripherie geschieht über sog. Portbausteine. An einen Port kann man ein Byte anlegen und dann darauf hoffen, daß, nachdem der Baustein den Angriff verdaut hat, eventuell auch mal ein Antwortbyte zurückkommt. Die Wartezeit dazu ist leider nicht einfach zu messen, so daß man abhängig von dem Prozessortyp einfach

eine bestimmte Anzahl Taktzyklen warten muß. Da aber bei einem schnellen Prozessor ein Taktzyklus kürzer ist, als bei einem langsameren (->Taktzahl des Quarz), kann man hierfür kein einheitliches Programm entwerfen.

Das BIOS auf dem Mainboard ist aber nicht endgültig. Nach dem Booten (s.u.) werden viele Routinen des UrBIOS durch Routinen des nachgeladenen Betriebssystems ersetzt, ebenso ersetzen bereits Erweiterungskarten einige Funktionen des BIOS. Graphikkarten z.B. tun dies auf jeden Fall, da nur dann gewährleistet ist, daß alle Programme auch die volle Funktionalität und die Geschwindigkeit der Karte nutzen können.

Und ab die POST.

Nach dem Anschalten des Rechners nimmt die CPU sofort die Arbeit auf, und zwar bei INTEL 8088 oder seinen Nachfolgern an der Adresse F000h:FFF0h (= der absoluten Adresse FFFF0h). Sie ist Teil des BIOS-ROM und enthält in den meisten Fällen einen Sprungbefehl zu einer BIOS-Routine, die das System testet und die Initialisierung der verschiedenen Hardware-Komponenten vornimmt. Dieser Vorgang wird POST genannt, eine Abkürzung für „Power On Self Test“.

POST-Tests, die sich auf die zentrale PC-Hardware beziehen:

- Funktionsprüfung der CPU (Arithmetik, Real-Mode, Protected-Mode etc.)
- Bildung einer Checksumme über das BIOS-ROM
- Bildung einer Checksumme über das CMOS-RAM der batteriegepufferten Echtzeituhr und Vergleich mit der darin festgehaltenen Checksumme
- Test und Initialisierung des DMA-Controllers
- Test und Initialisierung des Tastatur-Controller
- Überprüfung der ersten 64 KByte des RAM-Speichers
- Test und Initialisierung des Interruptcontrollers
- Test und Initialisierung des Cache-Controllers (nur bei Ats)

POST-Tests der Systemerweiterungen:

- Videocontroller
- Hauptspeicher über 64 KByte
- Serielle und Parallele Schnittstelle
- Disketten- und Festplattencontroller

Danach werden noch ROM-Erweiterungen gesucht, die manche Erweiterungskarte benötigt. So ersetzen z.B. VGA-Karten die Standardfunktionen der BIOS-Funktionen hinter dem Interrupt 10h, da dieser nur auf CGA und MGA Graphikkarten eingestellt ist. SCSI-Controller verbiegen den Interrupt 13h auf ebenfalls verbesserte Routinen, da das BIOS selbstverständlich nicht mit SCSI vertraut ist und daher um diese Funktionalität erweitert werden muß.

Zu guter Letzt wird noch der auch unter dem Namen „bootstrap loader“ bekannte Interrupt 19h aufgerufen, der schließlich das Betriebssystem nachladen soll.

Dabei wird der 1. Sektor einer Diskette gelesen, falls eine eingelegt wurde. Ist dies nicht der Fall oder die Bootreihenfolge geändert worden, wird der 1. Sektor der 1. Partition der 1. Platte gelesen (->MBR).

Nachdem die Bootroutine aus dem ersten Sektor gelesen wurde, wird sie gestartet. Danach übernimmt diese Routine alles weitere, um den Rest des Betriebesystems nachzuladen und im Speicher zu installieren.

Das Betriebssystem installiert neben vielen anderen Dingen auch weitere Routinen zur Kommunikation mit Peripherie. So werden z.B. Mouse-Treiber, Drucker-Treiber, Graphikkartentreiber und so weiter installiert, um eine typabhängige Kommunikation zu ermöglichen.

Leider ist mit der Vielfalt aller Peripheriegeräten das Problem der Inkompatibilität entstanden, da niemand sich darum kümmerte, was ein anderer Hersteller entwickelte. Dies machte es somit notwendig, sich mit den Gerätetreibern auszuhelfen, die das Problem der Standardisierung wieder behoben, indem man der Software, die mit einem Gerät kommunizieren möchte, ein API (Advanced Programmer Interface) anbietet, das standardisiert wurde. So kann die Software über eine weitere Software, also dem Treiber, der in der Regel vom Hersteller angeboten wird, mit der Hardware kommunizieren. (Beispiel Soundkarten)

Die Treiber übernehmen damit die Funktionalität des BIOS. Vor allem in Bereichen, die von dem Standardbios nicht abgedeckt wurden, da solche Geräte bei der damaligen Entwicklung des BIOS nicht bedacht wurden.

Gerätetreiber haben bei den multitaskingfähigen Betriebssystemen auch die Aufgabe, die Kommunikation mit Geräten zu steuern, damit z.B. nicht zwei Programme gleichzeitig versuchen, auf ein und das selbe Gerät zuzugreifen, und damit die Ausgabe eines Gerätes, z.B. der Tastatur, auch bei dem richtigen Programm ankommt.

Um unter Windows eine schnelle Graphikausgabe hinzubekommen, hat Microsoft einen neuen Treiberstandard entworfen, der sich DirectX nennt und eine direkte Kommunikation mit der Hardware einer Graphikkarte oder Soundkarte ermöglicht. Ohne diese neue Treiberarchitektur wären die verbesserten Leistungen im Multimedia- und Spielbereich undenkbar.

Kommunikation mit Peripherie

Wie oben bereits erwähnt gibt es im Bereich der Peripherie das große Problem der Kommunikation, da jeder Hersteller dazu geneigt ist, sein eigenes Süppchen zu kochen. Darunter leidet nicht nur der Konsument, der nicht weiß, auf welchem Standard morgen wieder gesetzt wird, sondern auch alle Programmierer, die diese Hardware steuern wollen.

Um nicht für jedes Programm das Rad neu zu erfinden liefern Betriebssysteme und Hersteller von Geräten spezialisierte Treiber aus, die im Gegensatz zu dem allgemein gehaltenen BIOS und dem erweiterten BIOS bei DOS ganz speziell für eine ganz bestimmte Hardware geeignet sind.

So kennt ein Druckertreiber den genauen Code zur Aktivierung des Unterstreichmodus oder Fettdruckes, wie es sein zugehöriger Drucker erwartet. Ein Soundkartentreiber entlockt nur der Soundkarte, für die er geschrieben wurde, den satten Sound (oder auch nicht...) und so weiter.

Kommunikation im Polling- oder Interruptverfahren

Die Treiber aber müssen irgendwann auf die Systemebene herunter und sich mit dem Gerät auseinander setzen. Dabei gibt es nun zwei Möglichkeiten, mit einem Gerät zu kommunizieren, dem *Pollingverfahren* und dem *Interruptverfahren*.

Wir wollen die Unterscheidung bei der Verfahren am Beispiel der parallelen Schnittstelle erläutern. Diese Schnittstelle ist für ihre negative Logik berühmt-berüchtigt und hat damit schon so manchen Hardwarefuzzy in den Wahnsinn getrieben.

Das Pollingverfahren

Im Normalbetrieb der parallelen Schnittstelle werden die zu übertragenden Bytes einzelnen an einem Port angelegt. Danach muß ein Bit im Statuswort des Druckers auf *low* gesetzt werden (-STROBE), also gelöscht werden, und danach wieder auf *high*. Dazwischen muß man mindesten eine Microsekunde warten, damit der Drucker das Byte auch abholen kann, aber auf der anderen Seite auch nicht zu lange, da der Drucker das Byte dann zweimal abholen wird.

Nun weiß der Drucker, daß für ihn ein Datenbyte anliegt und er holt es ab. Jetzt muß man warten, bis der Drucker sich meldet und die Bereitschaft für das nächste Byte signalisiert (-ACK). Dann kann man das nächste Byte anlegen und so weiter.

Das Interruptverfahren

Für multitaskingfähige Betriebssysteme ist das Pollingverfahren nicht oder nur bedingt geeignet. Schließlich muß der Port mit dem ACK-Bit von dem Treiber beim Druck ständig überwacht werden. Während dessen kann die CPU keine anderen Aufgaben erledigen, was ein Grund dafür sein dürfte, warum WINDOWS® keinen einwandfreien Hintergrunddruck hinbekommt.

Im Interruptverfahren wird der Port des Druckers so eingestellt, daß er immer dann, wenn er die Aufmerksamkeit des Treibers benötigt (Fehler, Bereitschaft,...) den korrespondierenden Hardwareinterrupt aufruft. Dort kann dann jegliche Information abgefragt (welcher Drucker war es, was hat er denn zu sagen?) und reagiert werden.

Oft werden Hardware-Interrupts auch dazu benutzt, um einem Gerät einen „Acknowledge“ zu senden, was normalerweise nur dem Auslesen eines Ports entspricht. Dann weiß aber das Gerät, daß es z.B. mit einer Ausgabe fortfahren kann. Das Prinzip des Acknowledge wird vor allem von Soundkarten verwendet.

Disketten und Festplatten

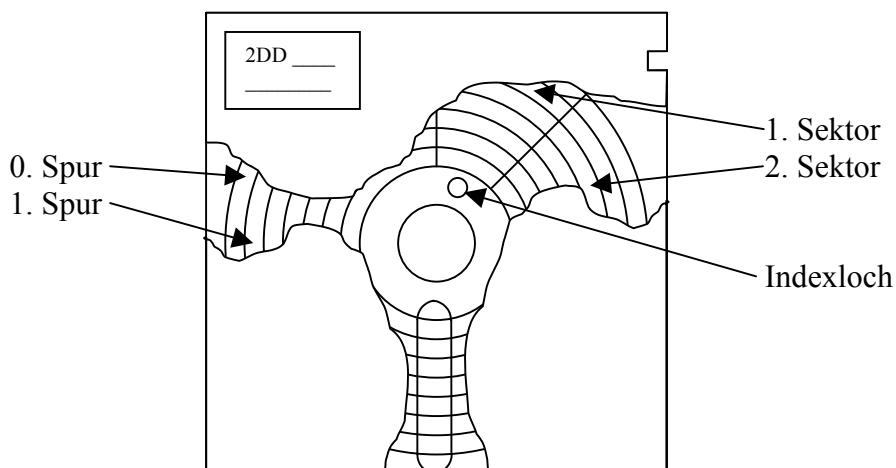
Was Disketten und Festplatten verbindet, ist ihr Aufbau. Disketten können als reduzierte (zweidimensionale) Festplatten aufgefaßt werden.

Aufbau einer Diskette

Disketten bestehen aus einer (mehr oder weniger) festen Hülle und einer darin befindlichen Scheibe mit einer magnetischen Oberfläche, ähnlich dem Material, daß von Musikkassetten bekannt ist.

Disketten werden grundsätzlich in einzelne Spuren unterteilt, die als konzentrische Kreise in gleichmäßigem Abstand über die Oberfläche der Magnetscheibe im Inneren einer Diskette verteilt sind. Diese Spuren werden mit Nummern von 0 bis n bezeichnet, wobei n die Gesamtzahl aller Spuren – 1 darstellt und von Format zu Format variiert. Die äußere Spur trägt aber grundsätzlich die Nummer 0, die darauf folgende die Nummer 1, und nach diesem Schema geht es bis zu innersten Spur weiter.

Nun ein kleines Bildchen:



Jede Spur nimmt dabei eine konstante Anzahl Sektoren auf, welche die Spur in einzelne Abschnitte gleicher Größe unterteilen. Die Anzahl dieser Sektoren ist wiederum vom Disketten- und Laufwerksformat abhängig. Anders als die verschiedenen Spuren werden Sektoren nicht von 0, sondern von 1 bis k gezählt, wobei k die Anzahl der Sektoren pro Spur darstellt.

Da jede Spur die gleiche Anzahl Sektoren aufnehmen muß, werden diese Sektoren zum Zentrum der Diskette hin immer kleiner, so daß die Daten immer dichter gespeichert werden müssen. Die innersten Sektoren geben daher an, wie groß die Datenmenge in einem Sektor nur sein kann.

Jeder Sektor enthält 512 Byte an Daten und stellt somit die kleinste Zugriffseinheit dar, mit der ein Programm operieren kann. Es ist also nicht möglich, ein einzelnes

Byte von einem Datenträger zu lesen oder zu schreiben, sondern es muß immer ein ganzer Sektor gelesen werden.

Der Programmierer bemerkt diesen Umstand jedoch selten, da höhere Programmiersprachen Routinen zur Verfügung stellen, die das Lesen eines einzelnen Bytes ermöglichen.

Aufgezeichnet werde die Bitinformationen innerhalb eines solchen Sektors nach dem FM- oder MFM-Verfahren, das im Zusammenhang mit den Festplatten erklärt wird.

Gemäß der Aufteilung in Spuren und Sektoren berechnet sich die Kapazität einer Diskette nach der Formel:

$$\text{Spuren} * \text{Sektoren pro Spur} * 512 \text{ Byte pro Sektor}$$

Der dabei errechnete Wert beschreibt allerdings nur das Fassungsvermögen einer einzelnen Diskettenseite. Ist das Laufwerk mit zwei Lese-/Schreibköpfen ausgestattet und es werden somit beide Seiten einer Diskette beschrieben, muß dieser Wert noch verdoppelt werden.

Die Anzahl der Sektoren pro Spur bestimmt auch die sogenannte Datentransfer-Rate, die ein Maß für die Geschwindigkeit der Laufwerkselektronik und des mit ihr verbundenen Controllers darstellt, denn bei einer konstanten Umdrehungsgeschwindigkeit von 300 Umdrehungen in der Minute ziehen pro Zeiteinheit desto mehr Bits am Lese-/Schreibkopf vorbei, je mehr Sektoren auf einer Spur untergebracht sind.

Aufbau von Festplatten

Festplatten unterscheiden sich im grundsätzlichen Aufbau nur unwesentlich von Disketten. Man kann sagen, daß eine Festplatte eigentlich nur mehrere sich schnell drehende Diskettenscheiben sind, die übereinander auf einer Achse angeordnet werden. Die einzelnen Scheiben sind von höherer Aufzeichnungsdichte, da die Oberfläche „dichter gekörnt“ ist, und ebenso in Sektoren und Spuren aufgeteilt.

Als Trägermaterial verwendet man für die einzelnen Platten Aluminium oder Glas, da bei den hohen Rotationsgeschwindigkeiten einer Festplatte eine dünne Kunststoffscheibe flattern würde.

Da wie bereits erwähnt der innerste Sektor die maximale Größe an Bitinformationen festlegt, hat man sich bei Platten etwas einfallen lassen, um der Platzverschwendung auf den Außenseiten entgegenzuwirken. Bei Platten ist es durchaus möglich, unterschiedliche Sektoranzahlen pro Spur zu haben, so daß auf den Außenseiten mehr Sektoren sind, als innen. Dadurch kann man auf den Außenseiten wieder mehr Daten lassen und daher die Kapazität noch etwas erhöhen.

Da Festplatten aber mehrere Ebenen haben, braucht jede Scheibe zwei eigene Schreib-/Leseköpfe, da Ober- und Unterseite einer einzelnen Platte beschrieben werden. Die Köpfe aller Platten sind wie die Zähne eines Kamms alle miteinander verbunden, so daß ein Spurwechsel für alle Köpfe gleichzeitig vollzogen werden muß. Daher ist es sinnvoll, beim Lese-/Schreibvorgang erst alle Köpfe nacheinander zu durchlaufen, bis

die entsprechende Spur aller Platten gleichermaßen gelesen/beschrieben wurde, und erst dann die Spur zu wechseln, da ein Kopfwechsel schneller vollzogen ist, als ein Spurwechsel.

Um diese Bewegung von Kopf zu Kopf beschreiben zu können, hat man den Begriff des Zylinders eingeführt. Ein Zylinder umfaßt alle Sektoren mit gleicher Spurnummer, die sich aber alle auf unterschiedlichen Platten befinden.

Die Platten drehen sich wie bereits erwähnt mit ganz anderen Rotationsgeschwindigkeiten. Im Gegensatz zu Disketten mit ihren 300 Umdrehungen pro Minute drehen sich die Platten mit etwa 3.600 bis 5.000 Umdrehungen pro Minute. Heute sind auch bereits Platten mit annähernd 10.000 Umdrehungen pro Minute im Einsatz, die aber durch die hohe Rotation auch ziemlich laut werden können.

Diese hohen Rotationen bringen einige Probleme mit sich:

Die Berührung von Kopf- und Platte, der gefürchtete „Headcrash“, wäre bei diesen hohen Rotationsgeschwindigkeiten sowohl für die Plattenoberfläche als auch für den Kopf fatal. Beides würde unweigerlich zerstört, so daß man sich einen physikalischen Effekt zu nutze macht, den Bernoulli-Effekt. Zwischen dem Kopf und der Platte bildet sich ein Luftposter, auf dem der Kopf wie ein Luftkissenboot in einem Abstand von einem Mikrometer ($\frac{1}{1.000.000}$ Meter) reitet. Es wird dann auch klar, warum Staubkörner oder Haare fatal werden, sollten sie auf der Platte liegen, denn beides türmt sich dem Kopf wie ein Gebirge entgegen. Aus diesem Grund sind Festplatten auch nicht evakuiert, wohl aber staubfrei und luftdicht verschlossen.

Um dem Headcrash auch bei einem Transport entgegenzuwirken muß man die Köpfe einer Platte parken, daß heißt in die „Landing Zone“ bewegen. Heutige Festplatten machen das alleine, da sie nach dem Ausschalten die Schwungmaße der Platten und die daraus resultierende Restrotation nutzen, um damit Strom zu erzeugen und sich in einen Bereich zu fahren, in dem keine Daten gespeichert werden. Bei älteren Platten benötigt man dazu eine Software, die der Platte mitteilt, daß sie die Köpfe in einen datenfreien Bereich, also meistens über den innersten Sektor fahren soll.

Ein weiteres Problem ergibt sich bei dem Versuch, zwei logisch aufeinander folgende Sektoren auslesen zu wollen, denn durch die hohe Rotationsgeschwindigkeit ist der zweite Sektor bereits an dem Lesekopf vorbei gerauscht, bevor dieser überhaupt den ersten Sektor verdaut hatte. Er muß dann auf eine volle Umdrehung warten, bevor er wieder zuschlagen kann. Mit einem kleinen Trick hat man sich dann den vollen Umlauf erspart, indem man logische Sektoren nicht nebeneinander anordnete, sondern mehrere andere Sektoren dazwischen ansiedelte, so daß die Zeit genau ausreichte, um die Daten abzuliefern und wieder zum Lesen bereit zu sein, wenn der Sektor ankam. Diesen sogenannten Interleave mißt man mit dem „Interleave-Faktor“.

Bei einem Interleave-Faktor von 1:3 zum Beispiel werden zwischen zwei logischen Sektor 2 physikalische Sektoren zwischen gestreut. Damit sind genau drei Umdrehungen der Platte notwendig, wenn die gesamte Spur eingelesen werden muß.

Bei heutigen Platten allerdings ist ein Interleave von 1:1 üblich. Die Controller von Festplatten sind mittlerweile so schnell geworden, daß sowieso immer gleich die gesamte Spur eingelesen und zwischen gespeichert wird, ob angefordert oder nicht.

Aufzeichnung von Informationen (FM, MFM, RLL)

Um Informationen auf einem magnetischen Medium zu speichern, könnte man schlaue einfach für jede Eins eine magnetische Signatur hinterlassen und für eine Null entmagnetisieren. Doch das ist so einfach leider nicht möglich, da der Lesekopf bei einer Folge von Einsen nie sagen könnte, ob in dem Bereich magnetisierter Fläche drei oder fünf Einsen encodiert sind.

Man benötigt also ein Verfahren, mit dem man genau die Länge einer magnetisierten Fläche, die für ein Bit notwendig ist, angeben kann. Es scheint nötig, eine Art Rhythmus einzuführen, einen Taktgeber, der die Länge in Abhängigkeit der Rotationsgeschwindigkeit festlegt. Immer wenn der Taktgeber schlägt, ist die Zeit für ein neues Bit angebrochen.

Ein reiner Rhythmus ist aber auch nicht möglich, da die Rotationsgeschwindigkeit in bestimmten kleinen Bereichen variiert und sich somit die Längen immer ändern würde. Zudem kann man nicht ein einzelnes Magneteilchen ausrichten, sondern nur eine ganze Gruppe, die ebenfalls in ihrer Länge variiert. Zudem können magnetische Informationen nicht allzu dicht auf einem magnetischen Medium gespeichert werden, da sich die Teilchen gegenseitig beeinflussen würden.

Was man aber speichern kann, sind sogenannte Flußwechsel. Die Änderung der Magnetisierung erzeugt im Schreib/Lesekopf für einen elektronischen Impuls der für die Decodierung von Nullen und Einsen verwendet wird.

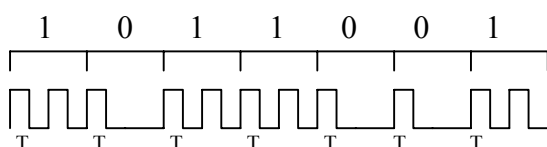
Und die Anzahl dieser Flußwechsel ist es auch, die immer kompakter werden soll, denn je mehr Flußwechsel pro Quadratmillimeter gespeichert werden, gibt Auskunft über die Speicherkapazität eines Sektors und damit der ganzen Platte. Bei gleicher Größe können dann mehr Bits und Bytes gespeichert werden.

Das FM-Verfahren

Das einfachste Verfahren, um Nullen und Einsen mit Hilfe der Flußwechsel zu speichern besteht in der Aufzeichnung eines Flußwechsels für eine Eins und dem Weglassen eines Flußwechsels für eine Null. Damit wiederholen sich aber die Probleme, die oben bereits angesprochen wurden, denn auch hier muß ein Takt vorgegeben werden.

Bei der Frequenzmodulation hat man sich daher überlegt, bereits einen Takt aufzuzeichnen, zwischen dem die Flußwechsel für die Datenbits Platz finden. In zeitlich regelmäßigen Abständen werden daher Flußwechsel aufgezeichnet, auf den sich die Steuerelektronik synchronisieren kann.

Dieses Verfahren ist allerdings nicht optimal, da für jedes Datenbit zwei Flußwechsel benötigt werden und damit die mögliche Kapazität der Platte halbiert wird.

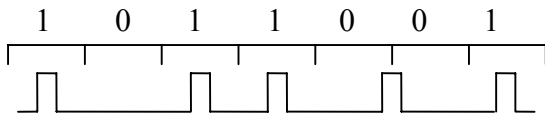


Das MFM-Verfahren

Das MFM-Verfahren machte mit dieser Platzverschwendung Schluß. Es stellt eine Modifikation des FM-Verfahrens dar, was ihm den Namen „Modified Frequency Modulation“ einbrachte.

Hier können die Takt-Flußwechsel als Daten-Flußwechsel mißbraucht werden, indem sie gegenüber dem normalen Takt verschoben werden. Zunächst bleibt allerdings alles beim Alten, denn eine Null besteht nach wie vor aus einem Takt-Flußwechsel, dem bis zum nächsten Takt-Flußwechsel kein weiterer folgt. Eine Eins wird jedoch nur noch als ein Flußwechsel gespeichert, und zwar an genau der Position, wo er auch bei dem FM-Verfahren zu finden ist. Der vorausgehende Takt-Flußwechsel fehlt jedoch.

Bei diesem Verfahren sollte die Steuerelektronik und die Aufzeichnung gut sein, denn hier können natürlich mehr Fehler auftreten.



Das RLL-Verfahren

Bei dem RLL-Verfahren (run length limited) werden Einsen als Flußwechsel und Nullen als das Fehlen eines Flußwechsels gespeichert, ohne allerdings einen Takt-Flußwechsel festzuhalten. Die Laufwerkselektronik soll den Takt nun selber „zählen“, was durch die stetige Verbesserung der Rotationsgleichmäßigkeit und der Plattenelektronik nur möglich wurde.

Da aber die Abstände zwischen Flußwechsel nicht zu lang werden dürfen, da sich die Elektronik bei bestem Willen einfach verzählen wird, kodiert man hier die Information auf einen etwa doppelt so langen Code, der aber die Forderung nicht zu langer Null- oder Einsfolgen sicherstellt.

Mit diesem Verfahren können noch mal bis zu 50% mehr Informationen auf eine Platte gespeichert werden, als bei dem MFM-Verfahren.

Controller

Jede Festplatte und jedes Diskettenlaufwerk benötigt einen Controller, um mit dem Datenbus auf dem Mainboard in Verbindung treten zu können. Bei den Disketten hat sich da nichts mehr seit den Anfängen geändert, wohl aber bei den Festplatten. Es gibt mittlerweile eine Vielzahl an System, die sich in ihren Unterstützten Kapazitäten und dem Datendurchsatz unterscheiden. Diese sind:

ST506, ESDI, SCSI, IDE und EIDE

ST506 / 412

Dieses Interface wurde nach der ersten Seagate-Festplatte verwendet, der ST506 (ST=Seagate-Technology), die diese Schnittstelle verwendete. Je nach Aufzeichnungsverfahren waren Datenübertragungsraten von 5 MBit bei MFM bis zu 7,5 MBit bei RLL pro Sekunde möglich, in der Praxis aber nie erreicht. Dort lag der Durchsatz bei etwa 200 bis 600 KByte pro Sekunde. Der Controller verwendete ein 20poliges Datenkabel und ein 34poliges Steuerkabel. Die Datenübertragung erfolgte seriell und war daher nicht besonders schnell.

ESDI

Das Enhanced Small Device Interface (ESDI) ist als Weiterentwicklung des ST506/412 zu sehen. Es arbeitet mit den gleichen Kabelverbindungen und ebenfalls seriell. Die Übertragungsraten liegen hier mit 10 bis 15 MBit schon deutlich höher. ESDI und ST506 sind PC-spezifische Entwicklungen und erlauben daher auch nur zwei Festplatten gleichzeitig.

SCSI

Das Small Computer System Interface ist die bisher schnellste und vor allem vielseitigste Schnittstelle in diesem Bereich, da nicht nur Festplatten, sondern auch Scanner, Brenner und CD-ROMs und Bandlaufwerke (Streamer) angeschlossen werden können. Adaptec hat mit seinem 2940UW oder dem günstigeren 2940AU heutzutage eine Marktführung übernommen und bietet mit Controllern im Bereich von 300,- DM für den privaten Nutzer reizvoll Update-Möglichkeiten. Auch finden sich bereits SCSI-Controller (von Adaptec) fest installiert auf dem Mainboard, wie man es bisher nur von dem EIDE-Controller gewohnt war.

Die SCSI-Schnittstelle bietet zudem durch ihre Konstruktion einen entscheidenden Vorteil. Alle Geräte, die an einer SCSI-Schnittstelle angeschlossen werden, müssen ihre eigene Logik mitbringen, da der SCSI-Controller nur für den Datentransfer und die Kommunikation mit dem PC zuständig ist. Dadurch kann jeder Hersteller natürlich die Elektronik auf seiner Platte den gegebenen Erfordernissen anpassen und ist zudem auch absolut kompatibel zu allen Systemen, die SCSI einsetzen können. (86000er, MAC., etc.)

Leider sind die Platten dadurch auch extrem teuer, da jede Platte ihren eigenen halben Controller huckepack trägt.

Damit die einzelnen SCSI-Geräte, die alle über ein einziges Flachbandkabel (als Strang bezeichnet) angesprochen werden, auch immer wissen, daß sie gemeint sind, vergibt man jedem Gerät eine Nummer von 0 bis 6. Über diese Nummer kann der SCSI-Controller genau festlegen, welches Gerät er meint. Sollten mehrere SCSI-Controller ihren Dienst im Rechner tun, so bekommen auch diese jeweils eine Nummer zugewiesen, so daß auch hier keine Probleme auftreten.

IDE / EIDE-Controller

Der IDE-Controller (Integrated Device Electronics, kurz AT-BUS-Controller, bzw. ATA) bietet annähernd die gleichen Geschwindigkeiten wie SCSI für einen wesentlich niedrigeren Preis, nicht nur für den Controller. Selbst die Platten sind ungefähr nur halb so teuer, wie eine vergleichbare SCSI-Platte. Dieser Umstand ließ die IDE-Controller einen wahren Siegeszug erleben. Sie haben die Controller ST506 und ESDI vollständig abgelöst.

Durch die Entwicklung des EIDE (Enhanced Integrated Device Electronics, auch ATAPI) wurden auch Beschränkungen in der Mediengröße und im Medienbereich beseitigt. Am EIDE lassen sich Platten und CD-ROMs anschließen, und zwar vier Stück, zwei Geräte pro Strang. Diese Anzahl genügt im privaten Bereich völlig.

Da aber nun zwei Platten sich einen Strang teilen müssen, ist es notwendig, beide auseinander zu halten. Dazu muß man eine Platte auf Master und die andere auf Slave einstellen. Diese Einstellung berührt die Geschwindigkeit einer Platte nicht und dient nur dazu, eine eindeutige Identifikation zu ermöglichen. Wie die Einstellung genau auszusehen hat, erfährt man im Begleitheft zu der Platte oder aber durch einen Aufkleber auf dem Gehäuse.

Festplattenformate:

Wer eine Diskette für den Rechner lesbar machen will, muß diese zu erst einmal formatieren, also die Einteilung in Spuren und Sektoren vornehmen. Diese Prozedur ist bei Festplatten ebenfalls durchzuführen, jedoch gibt es hier noch einen Schritt mehr zu tun: die Partitionierung. Dabei ist die Partitionierung bereits der zweite Teil eines dreistufigen Prozesses, mit dem Platten für einen Computer vorbereitet werden.

Der erste Schritt ist die Low-Level-Formatierung, die auch von dem BIOS des Rechners vorgenommen werden kann (SOLLTE MAN ABER NIE MACHEN!). Festplatten sind sozusagen Softsektored und müssen daher eine Einteilung in Sektoren und Spuren erhalten, die den korrekten Interleave angibt. Dazu werden Kennungen auf die Platte geschrieben, die den jeweiligen Sektoren entspricht. Diese Prozedur wird aber von den Plattenherstellern übernommen und sollte auch nie wiederholt werden. Bei Disketten ist diese Einteilung bereits vorhanden und es damit nicht mehr notwendig, dieses nachzuholen.

Bei dem Prozeß der Partitionierung wird eine Platte in mehrere Bereiche unterteilt, die eigene Laufwerksbuchstaben erhalten, sofern man überhaupt eine Einteilung wünscht. Ursprünglich sollten so mehrere Betriebssysteme parallel auf einer Platte unterzubringen sein, wozu es auch heute noch benutzt wird.

Man konnte aber auch große Platten übersichtlich in kleine Happen unterteilen. Bei alten BIOS-Versionen mußte man das sogar, da die Routinen des BIOS nicht genügend Adressraum zur Verfügung hatten, um große Formate zu lesen, die somit in kleinere Bereiche unterteilt werden mußten.

Zudem wird auch die FAT (File Allocation Table) festgelegt, mit der die Platte arbeiten soll. Um nicht zu große Adressen für Sektoren und Spuren zu erhalten, hat man die Datenmenge innerhalb eines Sektors vergrößert, da auch dann noch große Platten zu verwalten waren (DOS : bis zu 32KB pro Sektor und vielleicht auch noch mehr...).

Dadurch belegte jede auch noch so kleine Datei mindestens 32KByte auf der Platte. Das war selbstverständlich nicht wünschenswert, da man sicher mehrere MByte pro Platte verschenkte, nur weil die Sektoren nicht gefüllt waren.

Bei Win'95b war es dann endlich möglich, statt der (uralten) 12-Bit oder 16-Bit FAT eine 32-Bit FAT anzulegen, so daß die Sektorgröße auf 4KB schrumpfen konnte und trotzdem noch große Platten gefahren werden konnten.

Eine zu kleine Stückelung der Sektoren würde jedoch bedeuten, daß die FAT sich wesentlich vergrößern würde, da die Belegung von wesentlich mehr Sektoren gespeichert werden müßte

Der MBR

Im MBR (Master Boot Record oder Partitionssektor) ist bei Festplatten der Startcode hinterlegt. Anders als bei Disketten muß nämlich auf Festplatten erst noch die Partition gefunden werden, in welcher der Bootsektor zu finden ist. Dieser Sektor wird mit der Hilfe einer Partitionstabelle ermittelt, dann der Bootcode des dortigen Bootsektors nachgeladen und schließlich das gesamte Betriebssystem.

Der Bootsektor

Der eigentliche Bootsektor, wie er am Beginn der Bootpartition und Disketten vorkommt, lädt das Betriebssystem in seiner Gänze nach, da schließlich kein ganzes Betriebssystem in die 512 Byte eines Diskettensektors passen kann, und startet es.

Die FAT

In der File Allocation Table ist festgelegt, welcher Sektor belegt, welcher defekt und welcher (immer gut!) frei ist. Von dieser FAT gibt es immer eine Kopie, die z.B. bei einem Chkdsk oder bei einem Scandisk auf Gleichheit abgetestet wird.

Belegungsfehler in dieser FAT, die Sektoren als belegt markieren, die aber durch keinen einzigen Verzeichniseintrag und auch keiner einzigen Dateikette benötigt werden, werden als sogenannte „Verlorenen Dateisegmente“ gemeldet und können in der Regel entfernt werden, da es mit ziemlicher Sicherheit Relikte aus einem Löschvorgang sind.

Die unterschiedlichen BUS-Systeme:

Wer schon einmal ein Motherboard betrachtet hat oder aber eine Erweiterungskarte einbauen wollte, der ist mit ihm in Brührung gekommen, dem Daten-Bus.

Um Steckkarten mit dem Computer zu verbinden, hat man einen Steckplatz konzipiert, der in der Lage war, mit den Steckkarten eine kommunikative Verbindung zwischen der CPU und der Steckkarte herzustellen. Im Prinzip handelte es sich dabei um eine Öffnung des Datenbusses, die von der IEEE genormt wurde und sich *AT-BUS* nannte.

Der AT-BUS / ISA-BUS

Da in den Anfangstagen der Datenbus nur 8-Bit transportieren konnte, war dieser AT-Bus auch durchaus legitim, aber bald wurden die CPUs verbessert und auf 16-Bit erweitert.

Da mußte natürlich der Datenbus mitziehen, so daß alsbald der AT-Bus auf 16-Bit erweitert wurde. Dabei hatte man auf Abwärtskompatibilität geachtet und so konnten 8- und 16-Bit-Karten gemeinsam auf einem System ihren Dienst tun.

Die 16-Bit-Karten waren natürlich schneller, da sie nicht zweimal 8-Bit Daten senden konnten, sondern auf einmal ganze 16-Bit. Um noch etwas mehr Geschwindigkeit herauszuholen, hatte man die Standard-Taktfrequenz des Bus-Systemes dann von 4,77 MHz auf 8 MHz angehoben. Heutige BIOS unterstützen aber auch eine variable Einstellung bis zu 16 MHz, was jedoch mit Vorsicht zu genießen ist, da manche alten Karten auf den genormten 8 MHz bauen und bei höheren Frequenzen durchbrennen könnten.

Leider ist diese Frequenz nicht mit den Prozessoren mitgewachsen, so daß schließlich andere Bus-Systeme Einzug erhielten. Als dies passierte, ersann man auch einen neuen Namen für den AT-BUS, nämlich den *Industry-Standard-Architecture-Bus*, kurz ISA-Bus.

Der ISA-Bus hatte einen gravierenden Nachteil, der nicht nur in der starren Frequenz lag, sondern auch in der unangenehmen Tatsache, daß mit einem 24-Bit breiten Datenbus nicht mehr als 16 MB an Speicher adressiert werden konnten. Man mußte regelrechte Klimmzüge unternehmen, um einen größeren Adressraum zu erreichen. Doch auch heute gibt es dieses Bus-System noch und es wird mit Sicherheit noch eine ganze Weile existieren.

Von MCA, VLB und EISA

Als erstes versuchte IBM mit seinem Micro Channel Marktmacht auf seine Seite zu ziehen, da diese neue Architektur nur in den edlen PS/2 Modellen zu erhalten war und die neuen VGA-Karten nur auf dieser Architektur laufen sollten.

Der Micro Channel konnte sich aber aus mehreren Gründen nicht durchsetzen und so begann schließlich im Home-Bereich der Kampf zwischen VLB und EISA-Bussystemen.

Sowohl der *Vesa Local Bus* (VLB), als auch der *Extended Industry Standard Architecture-Bus* (EISA) konnten mit 32-Bit Datenbreite bereits noch einmal die

Datentransferrate verdoppeln. Doch da der VLB die wesentlich günstigere Alternative darstellte, konnte er sich eine Zeit lang gegen den EISA durchsetzen.

Der PCI-Bus

Alle Bussysteme wurden schließlich durch den von INTEL entwickelten *Peripheral Component Interconnect* (PCI) Bus abgelöst, der nicht nur mit Prozessortakt (dennoch zwischen 25 und 33 MHz) läuft, sondern auch noch in weiter Zukunft mit jeder Art von CPU kommunizieren können soll. Zu diesem Zweck führte man eine *PCI-Bridge* ein, eine Datenbrücke, über die der PCI-Bus mit der CPU den Datenaustausch vollzieht. Ebenfalls wird mit einer PCI-ISA-Bridge das alte aber immer noch aktuelle ISA-Bus-System an den PCI-Bus angebunden. Ferner gibt es extra eine SCSI-PCI-Schnittstelle und eine verbesserte Schnittstelle für Netzwerkkarten.

AGP

Die AGP-Schnittstelle ist nur für Graphikkarten konzipiert worden und findet sich auf den Motherboards für die Pentium-2-Generation von INTEL. Es soll eine beschleunigte Kommunikation stattfinden, wie dieser Bus aber genau funktioniert, ist mir nicht bekannt.

Graphikkarten

Als primäres Ausgabemedium stellt der Bildschirm die Verbindung zwischen dem Anwender und den verschiedenen Programmen her, die ihre Ausgabe über die sogenannten Videokarten an den Bildschirm weiterleiten. Ein nicht unwesentlicher Teil des Entwicklungsaufwandes fließt daher bei den meisten Programmen in die Erstellung von Routinen, mit denen Informationen auf dem Bildschirm sichtbar gemacht werden können.

Nicht immer hat man es dabei direkt mit den Videokarten zu tun, stellt doch das BIOS zahlreiche vorgefertigte Routinen zum Bildschirmzugriff zur Verfügung. Doch den meisten Programmierern sind diese Routinen zu langsam, so daß sie doch die Ansteuerung der Videokarten wieder selbst in die Hand nehmen. Das ist mitunter gar nicht so einfach, sieht sich der Programmierer doch einer Vielzahl unterschiedlicher Karten verschiedenster Hersteller gegenüber, die sich alle in Feinheiten unterscheiden. So gibt es bisher sechs verschiedene Standards (3D-Karten nicht eingerechnet), die sich in ihren Leistungsmerkmalen stark unterscheiden.

Am Beginn des Zeitalters der Graphikkarten war der Monitor etwa 12“-13“ groß und stellte noch keine Farben dar. Es war bisher auch noch nicht notwendig geworden, Graphik darzustellen, so daß man sein Augenmerk auf die rein Textdarstellung lenken mußte. Das mag auch der Grund sein, warum gerade der Textmodus bei allen Graphikkarten einem Standard unterliegt und immer gleich ist.

Die MDA-Karten

Der *IBM Monochrome Display Adapter*, kurz MDA, stellt zusammen mit der CGA-Karte den ältesten Grafikadapter dar, der für den PC verfügbar war. 1981 wurde er zusammen mit dem ersten IBM Personal Computer der Öffentlichkeit vorgestellt und galt über Jahre hinweg als Standard im Bereich der monochromen Graphikkarten.

Die MDA unterstützte lediglich einen Betriebsmodus, in dem 25 Zeilen zu je 80 Spalten mit Textzeichen auf dem Bildschirm erschienen. Im Gegensatz zu vielen anderen Graphikkarten verfügte sie dabei über sehr wenig Video-RAM, so daß nur eine Textseite auf einmal im Speicher festgehalten werden konnte.

Zwar konnten auf diesen Karten keine Graphiken dargestellt werden, doch zogen trotzdem viele Anwender die MDA der einzigen Alternative, der CGA-Karte, vor, da die MDA über eine bessere Auflösung im Textmodus verfügte und daher die Augen des Anwenders lange nicht so strapazierte, wie das bei der CGA-Karte der Fall war.

Heute sind reine MDA-Karten nicht mehr im Einsatz, da es bereits viele Farbgraphikkarten gibt, die allen den monochromen Standard unterstützen und dabei wesentlich höhere Auflösungen darstellen können.

Die CGA-Karten

Ebenfalls aus dem Jahre 1981 datiert der CGA-Standard, dessen Akronym für *Color Graphics Adapter* steht. Als Alternative zu den MDA-Karten bot diese Karte den Anwendern bereits in den Anfangstagen der PCs die Möglichkeit zur Erzeugung von Graphiken – wengleich zu einem für heutige Verhältnisse unglaublichen Kaufpreis.

Wer mit dem Erwerb einer CGA-Karte sein Portemonnaie geschröpft hatte, konnte dafür aber auch auf die Anschaffung eines Monitors verzichten, da die CGA-Karten über einen speziellen Anschluß verfügten, mit dem man sie an einem normalen Fernseher anschließen konnte. Darüber hinaus verfügten sie auch über einen RGB-Ausgang, also eine Leitung, welche das Videosignal neben ein paar Synchronisationssignalen in die Farbanteile Rot, Grün und Blau unterteilte. Das erzeugte Bild wies gegenüber der MDA-Karte jedoch eine mindere Qualität auf, was nicht nur auf die geringere Auflösung, sondern auch auf den größeren Punkteabstand im CGA-Monitor zurückzuführen war.

Die CGA-Karte stellt im Textmodus genau wie die MDA-Karte 25 Zeilen und 80 Spalten dar, wobei die Zeichen auf einer feineren Punktmatrix basieren, als dies bei der MDA-Karte der Fall war.

Dazu kam noch ein Graphikmodus mit einer Auflösung von 320x200 Punkten und vier Farben, sowie ein weiterer Modus mit 640x200 Punkten mit nur zwei Farben.

Die CGA-Karten sowie die MDA-Karten basierten trotz ihrer Unterschiede doch auf dem Selben Chip, dem MC6845 von Motorola.

Die Hercules Graphics Card

Ein Jahr nach Markteinführung der PCs trat die bis dahin völlig unbekannte Firma Hercules mit einer neuen Grafikkarte für den PC auf den Markt und erntete riesigen Erfolg. Dabei basierte auch diese Karte auf dem bereits genannten Motorola-Controller und war weitgehend zu der MDA-Karte von IBM kompatibel. Über die Fähigkeiten dieser Karte ging Hercules jedoch weit hinaus, denn neben dem Textmodus kann die HGC-Karte auch zwei Graphikseiten mit einer Auflösung von 720x348 Punkten verwalten und auf den Bildschirm bringen. Sie kombinierte dabei die hervorragende Lesbarkeit einer MDA-Karte mit den Graphikfähigkeiten einer CGA-Karte und ging dabei noch über deren Auflösungsvermögen hinaus.

Die Herculeskarte wurde jedoch nicht von dem BIOS unterstützt, da es IBM unvernünftigerweise immer abgelehnt hatte, Graphikkarten von Fremdherstellern zu unterstützen. Das tat dem damaligen Siegeszug der HGC aber keinen Abbruch, da sie zu der MDA kompatibel und daher zumindest der Textmodus über das BIOS ansprechbar war.

Der Graphikmodus wurde aber weiterhin nicht vom BIOS unterstützt, und zwar weder die Initialisierung noch das setzen einzelner Bildpunkte. Dies stellte jedoch kein großes Problem dar, da die entsprechenden BIOS-Routinen aufgrund ihrer mangelnden Geschwindigkeit onehin oft umgangen wurden und sich der Zugriff auf die Punktinformationen bei einer HGC noch dazu recht simpel gestaltet.

Die EGA-Karte

Nachdem es ein anderer Anbieter gewagt hatte, in den bisher von IBM beherrschten Markt einzudringen, mußte IBM reagieren und brachte daher 1985 eine neue Graphikkarte auf den Markt, welche die CGA-Karte ablösen und die hervorragenden Eigenschaften der HGC in Punkto Auflösung noch schlagen konnte, den *Enhanced Graphics Adapter*, kurz EGA genannt.

Diese Graphikkarte war voll abwärtskompatibel zu den MDA- und CGA-Karten und konnte daher weiterhin alle Textauflösungen von 80x25 Zeichen und dazu die CGA-Graphikaufösungen von 320x200 Pixel in vier Farben und 640x200 Pixel in 2 Farben. Darüber hinaus war die EGA-Karte in der Lage, wie die Herculeskarte bei monochromen Auflösungen an einem Monochrommonitor betrieben zu werden.

Doch im Bereich der Hardware hatten sich zwischen 1981 und 1985 viele Innovationen getan und so konnte man bei der EGA-Karte nicht nur Standards vereinen, sondern auch noch eine neue Auflösung anbieten, die leider auch die EGA-Karte für damalige Verhältnisse ausgesprochen teuer werden ließ, so daß es sehr lange dauerte, bis private Anwender sich diese Karte leisten konnten.

Neu war nun eine Auflösung von 640x350 Pixel, was nicht wesentlich höher war, als bei den CGA-Karten, doch war in diesem Modus eine Farbzahl von 16 Farben aus einer Palette von 64 Farben möglich. Diese Farbvielfalt macht auch eine Vergrößerung des Video-RAM nötig, die bei EGA-Karten bis 256 KByte gehen konnte, um mehreren Graphikseiten Platz zu bieten.

Das alles konnte aber nur durch eine Abkehr vom MC6845 erreicht werden, der bei diesen Leistungsanforderungen nicht mehr mitkam. Statt dessen wechselten man bei den EGA-Karten zu den VLSI-Chips über, die alle Aufgaben der Bilderzeugung übernahmen. Vor allem die Art und Weise, wie Bildinformationen im Video-RAM gespeichert werden, unterschied sich damals dramatisch von der bisherigen Vorgehensweise und stellte daher einen wichtigen Einschnitt in der Programmierung von Video-Karten dar.

Durch die Wahl eines geringen Punkteabstandes bei dem EGA-Monitor brillierte die EGA-Karte gegenüber der CGA-Karte durch ein wesentlich schärferes Bild. Auch in anderer Hinsicht konnte die EGA-Karte mit entscheidenden Verbesserungen aufwarten. So war es nun möglich, per Software eigene Schriftsätze auf die Karte zu laden, und es fanden sich erst bei der EGA-Karte Performancegrößen, die für die Animation benötigt wurden. Das ist auch der Grund, warum die ersten wirklich sehenswerten Action-Spiele zu erst auf der EGA-Karte möglich wurden.

Da aber diese ganzen Verbesserungen von dem alten ROM-BIOS nicht unterstützt wurden, mußte diese Karte ihr eigenes BIOS mitbringen, daß wie bereits oben beschrieben bei dem POST die alten ROM-BIOS-Routinen ablöste.

Die VGA-Karte

Die VGA-Karte (*Video Graphics Adapter*), die zusammen mit dem neuen PS/2 von IBM im Frühjahr 1987 vorgestellt wurde, knüpfte nahtlos an die Tradition der EGA-Karte an. Die VGA-Karte war voll abwärtskompatibel zu allen Vorgängern, hatte mehr Farben, höhere Auflösungen und eine verbesserte Textdarstellung.

Die VGA-Karten waren ursprünglich für die Micro Channel-Architektur der PS/2-Maschine konzipiert, doch da VGA-Monitore immer günstiger wurden und sich der Micro Channel nicht durchzusetzen vermochte, stießen schnell andere Anbieter mit VGA-Karten für den ISA-Bus auf den Markt.

Von den EGA-Karten unterschieden sich die VGA-Karten bereits durch eine wesentlich höhere Integrationsdichte, welche die Unterbringung der gesamten Steuerelektronik in nur einem Baustein ermöglichte. Ferner unterschied sich die VGA-Karte von der Vorgängerin aber auch darin, daß nicht mehr ein digitales Signal an der Monitor gesendet wurde, sondern ein analoges. Dadurch konnten VGA-Karten bis zu 260.000 verschiedene Farben erzeugen, von denen je nach Modus 2, 4, 16 oder 256 aktiv waren.

Der höchste Modus einer VGA-Karte war 640x480 Pixel mit 2, 4 oder 16 Farben. Besonders vielfältig war dagegen der Modus 320x200 mit 256 Farben, in welchem für damalige Verhältnisse Photorealismus erreicht wurde. Da Video-RAM immer noch teuer war, schaffte man auch schlicht keine höheren Auflösungen mit entsprechender Farbenvielfalt.

Da aber jeder Anbieter mit immer besseren VGA-Karten auf den Markt stieß und dabei auch immer sein eigenes Süppchen kochte, mußten Programmierer immer ihre Programme auf ganz bestimmte Karten abstimmen.

Die Super-VGA-Karten

Die Super-VGA-Karten entsprechen im Grunde dem Aufbau der vorherigen VGA-Karten, sind jedoch schneller, um in der gleichen Zeit wesentlich mehr Punkte auf den Bildschirm zu bringen, und damit eine höhere Auflösung zu erzielen. Zudem konnten diese Karten sämtliche Graphikmodi der VGA-Karte und darüber hinaus noch höhere Farbenvielfalt bei höheren Auflösungen erzeugen.

So war nicht nur bei 320x200 eine Farbanzahl von 256 Farben möglich, sondern auch bei anderen Modi (640x200, 640x350, 640x480). Dazu erschloßen sie höhere Auflösungen von 800x600 und 1024x786. Heutige Karten schaffen auch Auflösungen von 1280x1024 oder 1600x1200 in True Color (24-Bit oder 32-Bit Farbtiefe), die aber nur noch von sehr leistungsfähigen VGA-Monitoren oder Multiscan-Monitoren geleistet werden können. Dazu konsumieren diese Karten auch ein nicht unerhebliches Maß an Video-RAM.

Sollen z.B. bei einer Auflösung von 1600x1200 noch 32-Bit Farbinformationen pro Pixel codiert werden, benötigt man $1600 * 1200 * 4$ ($32 = 4 * 8$) Byte = 7.680.000 Byte = 7.500 KByte = 7,32 MByte, also 8 MByte. Man kann dann aber auch nur eine Seite auf einmal darstellen. Möchte man dazu noch mindestens zwei Seiten aufbauen, benötigt man am Besten 16 MByte an Video-RAM.

Da es aber immer noch keine Einigung darüber gab, wie solche Graphikmodi auf der Hardware-Ebene über die Register einer solchen Karte initialisiert und angesprochen werden sollten, haben sich 1990 die wichtigsten Graphikhersteller von VGA-Kompatiblen Graphikkarten (Tseng, Paradise und Video Seven) in einem Konsortium zusammen gefunden, dem *Video Electronic Standard Association* (VESA, die übrigens auch den Vesa Local Bus verbrochen hatten).

Die MCGA

Da IBM seine VGA-Karte damals nur für die gehobene Klasse der PS/2 vorgesehen hatte, waren die anderen Modelle von IBM mit der MCGA ausgestattet. Dabei handelt es sich um ein Surrogat aus fast allen vorhergehenden Standards, das bisher niemanden so richtig begeistern konnte. Der Name MCGA steht für *Memory Controller Gate Array*, das noch nicht einmal so richtig nach Graphikkarte klingt.

Was den Textmodus angeht verhält sich diese Karte exakt wie eine CGA-Karte, also stellt einen 80x25 Zeichen-Modus bereit, wobei Vorder- und Hintergrundfarbe aus einer Palette von 16 Farben gewählt werden konnte. Sie sind aber im Gegensatz zu der CGA-Karte nicht fest vorgeschrieben, sondern können wie bei einer VGA-Karte aus einer Palette von 262.000 Farben gewählt werden.

Im Gegensatz zu einer CGA-Karte beträgt die vertikale Auflösung in diesem Modus auch nicht 200, sondern 400 Punktklinien, wodurch eine wesentlich höhere Darstellungsqualität erreicht wurde.

Ebenfalls wie ein Zwitter verhält sich die Karte in seinen Graphikmodi. Neben zwei VGA-Kompatiblen Modi werden auch die zwei CGA-Modi von 320x200 und 640x200 unterstützt. Da die Karte aber in jedem Modus 400 vertikale Pixelzeilen darstellen muß, werden die einzelnen Punktzeilen dieser beiden Auflösungen

verdoppelt, was zwar kein besseres Bild, dafür aber nur die Hälfte der möglichen Auflösung bringt.

Noch übler sieht es mit den VGA-Modi aus, die zwar die gewohnte VGA-Auflösung erreichen, aber in Bezug auf die Farbenvielfalt stark eingeschränkt sind. Grund dafür ist die Bestückung der MCGA mit nur 64 KByte Video-RAM, aus dem sich bei entsprechenden Auflösungen natürlich nur begrenzte Farben herausholen lassen. (Zitat M.Tischer: „Mich erinnert das an den Besitzer einer teuren Limousine, der zu geizig ist, um für sein Auto Benzin zu kaufen.“. Offensichtlich war der Autor mit dieser Karte ebenfalls nicht glücklich...)

Die 8514/A

IBM hat aber nicht nur schlechte Sachen gebaut. Im Jahre 1987 präsentierte IBM einen würdigen Thronfolger der VGA-Karten, der auf den etwas kryptischen Namen 8514/A hörte. Während bisher alle Graphikkarten relativ dumme Controller waren, hatte man der 8514/A einen Prozessor spendiert, der einfache Berechnungen durchführen sollte.

Der Vorteil liegt klar auf der Hand. Nun sollte nicht mehr die CPU die Berechnung für ein Polynom anstellen, sondern nur noch den Befehl dafür an den Prozessor der Karte schicken. Dieser würde dann alle Berechnungen durchführen und das Polynom zeichnen.

Die Idee war gut, doch wollte IBM nicht, daß es andere Anbieter geben würde, und so hielten sie die technischen Merkmale dieser Karte geheim. Zudem wurde sie nur für die (wie bereits schon erwähnt) supertolle PS/2-Modellreihe gebaut, also auf MCA-Bus. Damit war die Hoffnung gestorben, einmal eine vergleichbare Karte auf ISA-Basis zu erhalten.

Weiterhin sprach der übermäßig hohe Preis, die Tatsache, daß die Karte einen speziellen Monitor benötigte und die allgemein belächelte Softwareschnittstelle gegen die Karte, womit IBM ihr Todesurteil unterschrieben hatte.

TIGA

Mit den TIGA-Karten sind Karten gemeint, die wie die 8514/A einen Graphikprozessor haben, aber sich dem Standard von Texas Instruments unterworfen haben. TIGA steht für *Texas Instruments Graphics Architecture* und beschreibt einen Satz von Funktionen, die sich auch leicht aus Hochsprachen heraus nutzen lassen und geeignet sind, um dem Graphikprozessor auf diesen Karten Befehle zu erteilen.

Genau wie bei dem 8514/A-Standard lassen sich auch mit Hilfe dieser Prozessoren Graphikkarten entwickeln, die parallel zu der eigentlichen Programmausführung durch den Prozessor Grafiken erstellen und dabei selbst so komplexe Aufgaben wie z.B. das Ray-Tracing zur Berechnung von Spiegeleffekten wahrnehmen. Der Einsatz des Prozessors ist dabei nicht an eine bestimmte Bildschirmauflösung gebunden, sondern kann von VGA-kompatiblen Modi bis hin zu Bildschirmauflösungen eingesetzt werden, die man im CAD-Bereich antrifft und die derzeit noch jenseits des Vorstellungsvermögens eines PC-Anwenders liegen.

3Dfx / Voodoo / 3D-Beschleuniger-Karten

Bei den 3Dfx Karten handelt es sich um Beschleuniger-Karten, welche wie die TIGA und 8514/A einen eigenen Graphikprozessor besitzen. Dabei wird besonderes Augenmerk auf die Darstellung von drei-Dimensionalen Objekten, Clipping, 3D-Effekten wie Nebel und vor allem das Belegen von Flächen mit einem Bitmap (einer Oberfläche, die sich um die Linien des Objektes spannen soll) gelegt.

Der Voodoo-Chipsatz hat in diesem Bereich einen Standard geschaffen, der sowohl leistungsfähig, wie auch erschwinglich ist.

Diese neue Generation von Graphikkarten sind in der Regel nicht als Stand-Alone-Karten konzipiert, sondern arbeiten mit einer Graphikkarte zusammen. Dabei wird die Graphikkarte an der 3D-Karte mit einem (möglichst dicken, kurzen und gut abgeschirmten) Kabel verbunden, wobei das Signal der Graphikkarte durch die 3D-Karte geschleift wird, wenn keine 3D-Funktionalität benötigt wird.

Wird dann die 3D-Karte aktiviert, schaltet sie das Signal der Graphikkarte ab und sendet ihr eigenes Signal an den Monitor. Selbstverständlich gibt es bereits symbiotische Karten auf dem Markt, die beide Funktionen in sich vereinen, also sowohl 3D-Beschleunigung, als auch die normale Funktionalität einer Graphikkarte.

Wie die Graphikkarten leben auch die 3D-Karten von ihrem Speicher, dessen Größe ein Indiz für die Leistungsfähigkeit der Karte ist. Auflösung, Farbvielfalt und Geschwindigkeit hängt von diesem Speicher ab.

Literaturverzeichnis:

1. Michael Tischer,
PC-Intern 3.0, 1.Auflage,
ISBN 3-89011-591-8
2. H.R. Hansen,
Wirtschaftsinformatik I, 7. Auflage,
ISBN 3-8252-0802-8
3. DUDEN Informatik, 2. Auflage,
ISBN 3-411-05232-5