

Heft

1

Jahrgang

9

PC-NEWS

Lfd.Nr.

36

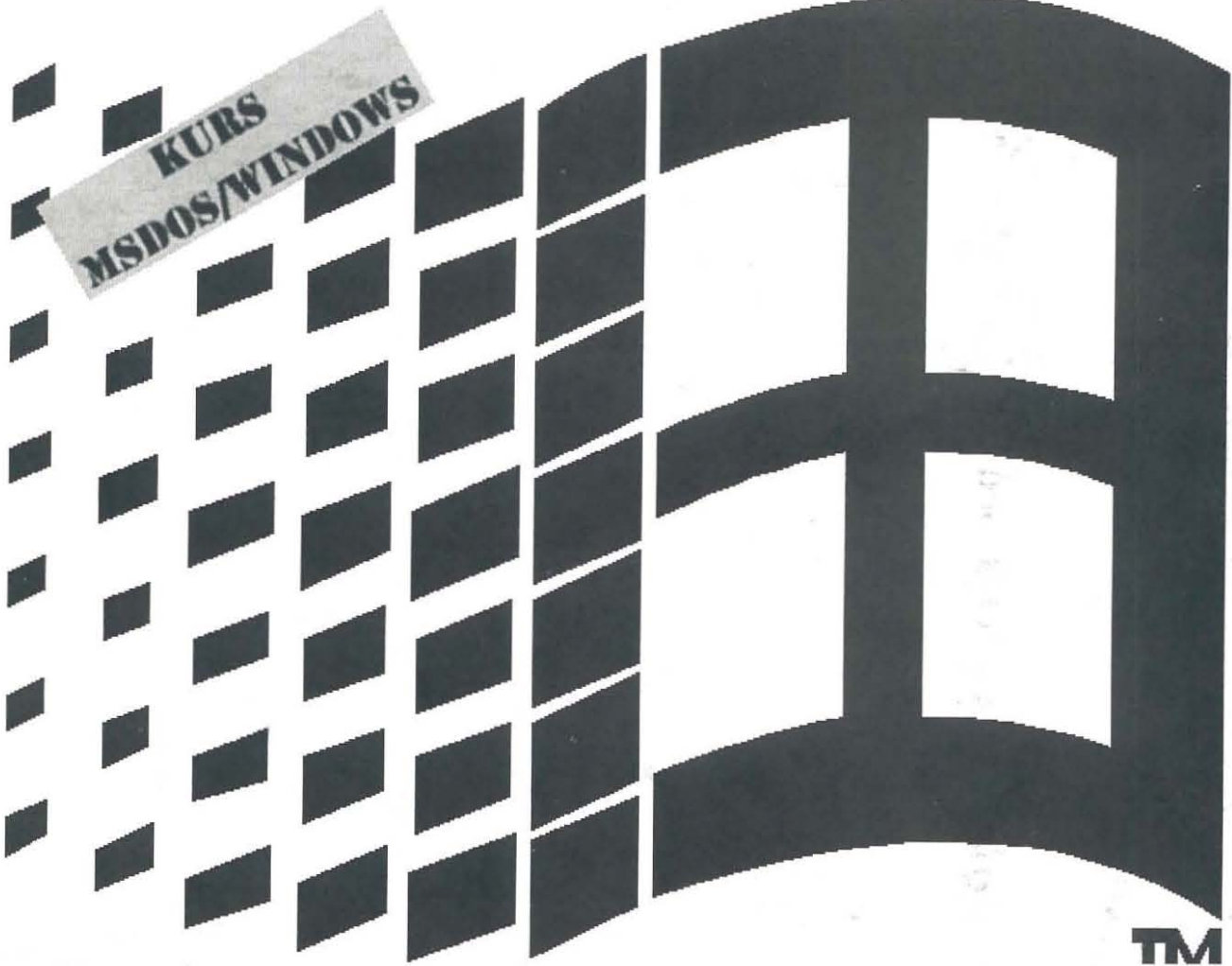
Februar

1994

Mitteilungen aus den Höheren Technischen Lehranstalten für aktive PC-Anwender

PC-NEWS Eigenverlag	ADIM	Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik	Einzelpreis öS 50,-	ISSN 1022-1611
	HMV	His Master's Voice-Mailbox des FIDO-Net		
Siccardsburg- gasse 4/1/22 1100 Wien	MCCA	BTX und Micro Computer Club Austria	Erscheint 5 mal / Jahr	P.b.b. Verlagspostamt 1100 Wien
	PCC-TGM	Personal Computer Club - Technologisches Gewerbe-Museum		
	PCC-S	Personal Computer Club - Salzburg		

Schwer-● -Programmierung



Inhalt

Allgemeines	angeTESTet	Schwer-●
Softwarerecht..... 16	Struktogrammgenerator.....44	WordBasic..... 57
Gewerberecht..... 16	Logitech AudioMan.....34	Visual-Basic..... 46
Microsoft® Select für Unterricht..... 14	HP-LJ-Drucker.....36-41	Windows/BORLAND..... 52
DSK/LIT/SON..... 9		Windows/MICROSOFT..... 58
DFÜ	Reihen	Mathematik
BTX: PAN, Opalis..... 27,28	ACCESS-Lehrgang Teil 2.....42	MathCad..... 19
TELEBOX-PC..... 26	C-Teil 7:Tonerzeugung im PC.....62	Mathematica..... 22
Mailbox-Umfage und Planung..... 8	DOS für Einsteiger.....31	
Clubs, Für alle	Betriebssysteme	Sammelbestellung
PCCTGM/MCCA..... 10	Was ist neu bei DOS 6.2?.....30	MS-DOS 6.2..... 6
PCCS/ADIM..... 11	OS/2-Hardwareanforderungen.....29	Zum Schmunzeln
Bazar..... 6	DOS für Einsteiger.....31	Schule,Real Programmers... 23,43,44, 61,69
Impressum..... 75		
Liebe LeserInnen..... 2		

Liebe LeserInnen!

Je größer die Zahl der installierten Einheiten, desto zäher ihre Umstellung auf eine neue Technik, ganz zu schweigen von der Umstellung der Benutzer. Die Schule ist ein gutes Beispiel. Während jeder neu einsteigende Schüler oder Lehrer ganz selbstverständlich mit Windows arbeitet und irgendwo auch das Symbol für die DOS-Box gesehen hat, gibt es noch ganze Lehrsäle, bei denen an den Betrieb von Windows noch einige Zeit nicht zu denken sein wird. Auch Lehrer, die ein großes Arbeitsvolumen in Vorbereitungen investiert und unter rauen Unterrichtsbedingungen getestet haben, stellen sich nur allmählich auf eine neue Ära ein.

Der Zug der Zeit: grafische Benutzeroberflächen, egal welcher Provenienz, stellen sich als die Hilfe dar, die es auch einer breiteren Schicht von Benutzern ermöglicht, ohne großartige Vorkenntnisse mit dem PC umzugehen. Ob es nun WINDOWS, OS/2, X-Windows oder DESQVIEW-X ist, alle kommen jenen entgegen, die nicht die Spezialisten sind; und Hand aufs Herz: auch die Spezialisten unter uns sind es ja nur in einem kleinen Gebiet. In allen anderen Bereichen tut es gut, sich durch ein ausgeklügeltes Bedienungsinterface durch das Leistungsangebot eines komplexen Programmes leiten zu lassen.

Welches der Fensterbetriebssysteme wir unterrichten werden? Wenn wir die überhaupt unterrichten müssen - denn die Schüler eignen sich ja die Handgriffe in kurzer Zeit selbst an -, ist die Auswahl - zumindest im TGM - zugunsten von **Windows** mit X-Windows-Anteilen ausgefallen; ohne große Evaluation, einfach dem Markt und der Verfügbarkeit für einen durchschnittlichen Lehrer-PC entsprechend. Daß einzelne Spezialisten und besondere Anwendungen, wie die Mailbox **His Master's Voice** auch **OS/2** verwenden, sollte sich in den **PC-NEWS** durch entsprechende Beiträge dieser Leser ausdrücken.

Es wird für Softwareentwickler und natürlich auch für unsere Schüler - speziell die Techniker unter ihnen - immer wichtiger, nicht nur gut funktionierende, sondern auch gut bedienbare Software herzustellen. Dabei entwickelt sich zwischen Lehrer und Schüler ein neues Lern- und Lehrerlebnis: ein Miteinander-Lernen, denn nichts von dem, was heute unterrichtet werden soll, haben Lehrer in ihrer Lehrzeit oder in ihrem Beruf mitbekommen, alles wird mehr oder weniger nacherfunden. Ist Programmierung allein schon ein Gebiet für eine Handvoll Tüftler, so stellt uns Programmierung mit Fenstern vor ganz neue Probleme:

Ein Vielfaches an Komplexität

Eine Komplexität, die etwa die Sprache C zum Assembler für Windows-Programmierung degradiert; C++ wird zum Muß; Bibliotheken ohne objektorientierten Ansatz gibt es praktisch nicht.

Glücklicherweise ist die Gründer-Zeit der Windows-Programmierung mit SDK und C-Kommandozeilen-Compiler schon vorbei, die neuen Compiler warten mit neuen Werkzeugen auf, die zumindest den Entwurf der Benutzeroberfläche zu einem Bildschirmspiel reduziert.

Weiters bietet sich als eine möglicherweise einfachere Einsteigersprache Visual-Basic an, die zwar nicht mit dem Tempo und mit den Datenstrukturen von C mithalten kann, dafür aber raschere Erfolge garantiert.

Ein Vielfaches an Ressourcen

DOS-Programme konnte man mit einfachsten Mitteln zur Not auch auf einem XT entwickeln. Damit ist es jetzt endgültig vorbei. Unter 8 MB Hauptspeicher ist nicht viel zu machen, eine volle Installation eines Compilers benötigt ab 50MB Festplattenplatz. Die Informationsflut erfordert mindestens ein CD-ROM-Laufwerk an einem Arbeitsplatz; in Lehrsälen, wo ein einzelner Schüler nicht immer unmittelbaren Zugang zur CD haben kann, wären mehrere gleichzeitig angeschlossene CD-ROM-Laufwerke anzuraten.

Leider gibt es aus dem Ressourcen-Engpaß (neben der Fensterverweigerung) keinen anderen Ausweg, außer den, in einer Niedrigpreis-phase RAM, ein schnelleres Motherboard und ein bißchen Festplatte zu kaufen.

Ein Vielfaches an Informationsmaterial

Könnte man die Programmierung unter DOS mit ein bis zwei kopierten Blättern über die DOS-Aufrufe unterrichten, wird es angesichts der sich stapelnden Windows-Book-Ware schwierig - speziell für Windows-Anfänger - ein sinnvolles Subset aus dem Material herauszuschälen.

Glücklicherweise kommt uns aber die sich rasch entwickelnde Technologie der Informationsdarstellung zu Hilfe; die Compiler haben die Handbücher on-line, kontextsensitive Hilfe macht Nachschlagen praktisch entbehrlich.

Und was soll dabei eine kleine Zeitung?

Es ist für eine kleine Zeitung unmöglich und auch für große Zeitschriften schwierig, ein großes Kapitel, wie es das der Windows-Programmierung ist, in einem Schwerpunkt einigermaßen geschlossen darzustellen. Immerhin widmen sich diesem Thema viele namhafte Zeitschriften. Wenn die Windows-Programmierung in Zukunft den Stellenwert haben wird, den heute die DOS-Programmierung noch hat, bedeutet unsere heutige Ausgabe lediglich einen Startschub für eine neue Phase der Hilfestellung durch die **PC-NEWS** für den EDV-Unterricht und für alle, die sich mit diesem Problem auseinandersetzen wollen und müssen. **Alle aktiven "Fenster"-Programmierer sind herzlich eingeladen, Beiträge und Tips an die Redaktion einzusenden; die Leser der PC-NEWS werden es ihnen danken. "Fenster" ist dabei stellvertretend für alle modernen grafischen Benutzeroberflächen.**

Wie also kann man sich einarbeiten?

Zunächst an Hand eines der großen Compiler: Schwerpunkt Windows-Programmierung wäre eigentlich ein Anlaß, die für diesen Zweck verfügbaren Compiler zu vergleichen. Welche sind das?

- **BORLAND:** BORLAND-C++ (noch 3.1 bald 4.0)
- **MICROSOFT:** Visual C++, 1.0
- **SYMANTEC:** C++, Version 6.0

Bisher hatte BORLAND die meisten Freunde bei den Schulen, wie etwa die Skripten der **ADIM** zeigen. Der Grund ist ziemlich sicher die einheitliche Benutzeroberfläche für PASCAL und C verbunden mit einer gut für den PC erweiterten Bibliothek. Je nachdem, ob in einer Schule eher PASCAL/C oder Visual-BASIC/C unterrichtet wird, kann den Ausschlag eher zu Gunsten des BORLAND- oder des MICROSOFT-Compilers ergeben, denn in den Schulen unterliegt die Bewertung der Compilerqualitäten anderen Maßstäben als in der Industrie. Ein hohes Maß an Einheitlichkeit bei der Bedienung verschiedener Compiler (BASIC-PASCAL-C) hilft, die Informationsflut zu bewältigen.

Nun, BORLAND-C++ Version 4.0, vielleicht als Weihnachtsgeschenk gedacht, ist noch nicht eingetroffen. Dazu kommt, daß die bestehenden Installationen nicht gerade nach Erneuerung schreien, denn die Zahl der Windows-Programmierer in den Schulen ist (noch) klein. Wenn man also vergleichen will, kann man das nur mit der alten Version des BORLAND-Compilers. Daher wollen wir über Compiler-Vergleiche, wenn überhaupt, in einer späteren Ausgabe berichten.

Wir müssen aber damit rechnen, daß bei Installation eines neuen Compilers auch noch für DOS entwickelt werden soll. Das läuft bei Visual-C unter der einheitlichen Windows-Oberfläche, bei Turbo ist bei Version 3.1 eine Trennung von DOS- und Windows-IDE vorgesehen; in der Version 4.0 des Compilers soll aber die DOS-Entwicklungsumgebung überhaupt fehlen und Turbo-Vision, die Fensterbibliothek für DOS, nur in einem Zusatzpaket zu erwerben sein. Man wird sehen.

Um einen einigermaßen fairen Vergleich durchzuführen, beschränken wir uns in der heutigen Ausgabe auf die Darstellung eines "HELLO-WORLD"-Programms für Windows unter Benutzung der Compiler VISUAL-C++ 1.0 und BORLAND-C++ 3.1, wobei jeweils zwei Methoden angewendet werden a) die DOS-kompatible (QuickWin/EasyWin) und b) die durch eine Klassenbibliothek unterstützte (Foundation-Classes/Object Windows). Daß der dritte Compiler (SYMANTEC) nicht verglichen wird, hat einen guten Grund: er benutzt ebenfalls die Foundation-Classes, der Unterschied liegt dann also bei der Benutzeroberfläche, die aber hier nicht verglichen werden soll.

MS Word f. Windows 6.0
3,5", dt., Schulversion

PIRATENPREIS
2.590,-

SOFTWARE

Borland C++ 4.0
3,5", dt., Schulversion

PIRATENPREIS
2.990,-

ZU

Corel Draw 4.0
3,5", dt., Schulversion

PIRATENPREIS
3.990,-

PIRATEN-

Lotus Improv 2.0
3,5", dt., Schulversion

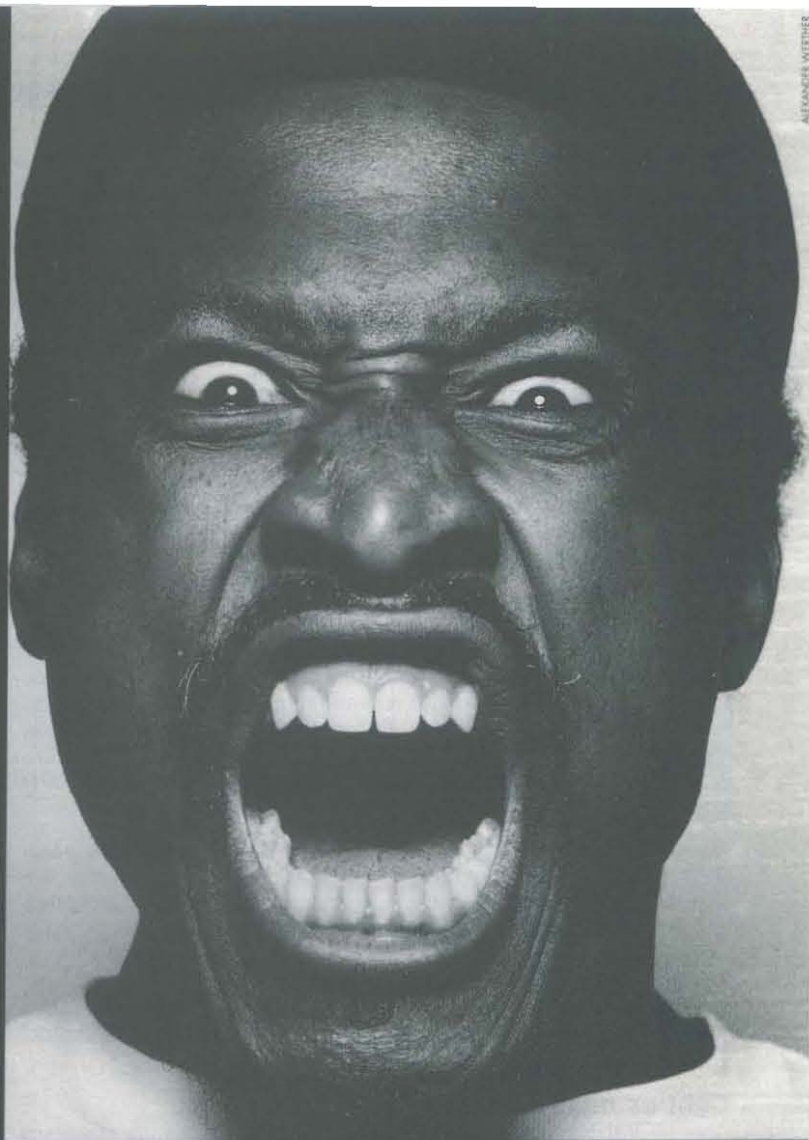
PIRATENPREIS
1.990,-

PREISEN!

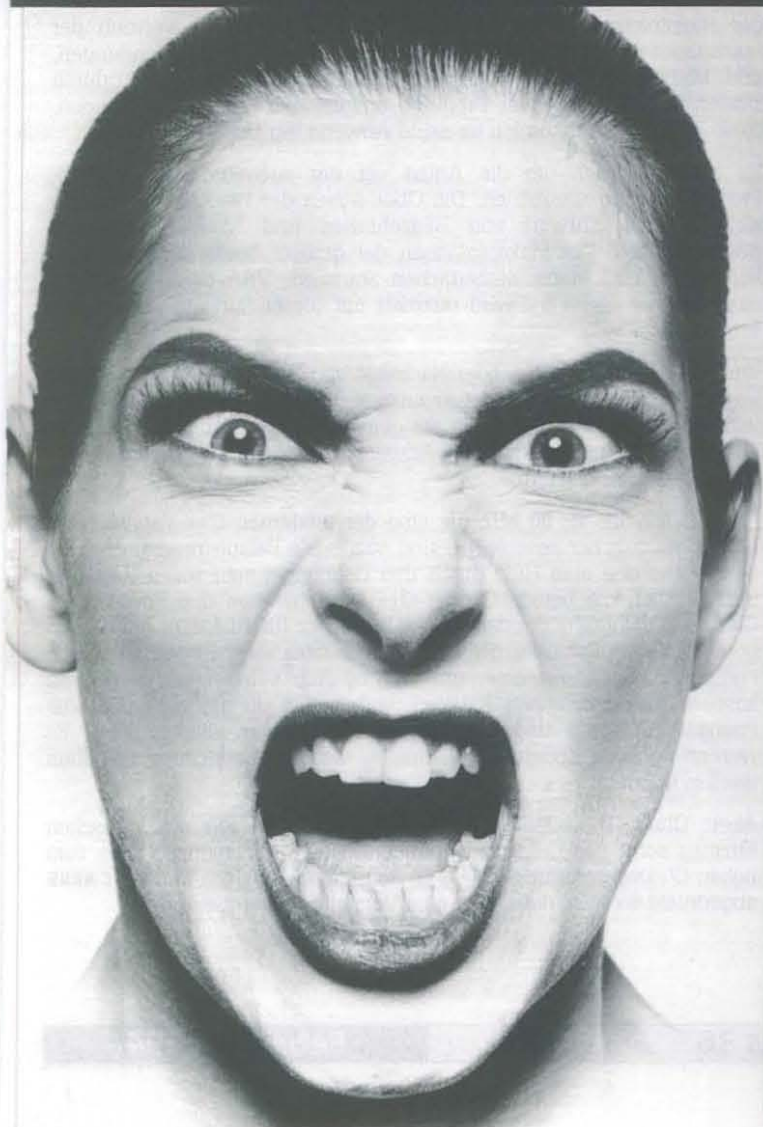
Designer 4.0
3,5", dt., Schulversion

PIRATENPREIS
3.490,-

Schulversionen bei Vorlage von Inskriptionsbestätigung und
Studentenausweis bzw. Schulbesuchsbestätigung erhältlich.



ALDOUSSEI WETTER



Die Computerpiraten

1090 Wien · Berggasse 5

Tel.: 0 222 / 317 44 42

8010 Graz · Schubertstrasse 2-4

Tel.: 0 316 / 38 46 70-3

COMP DELPHIN, ein Service der ÖH 

Großer Aufwand bei den Arbeitsplätzen

Der Aufwand für erforderliche Hard- und Software ist beträchtlich. Dabei ist Software das billigere Ende. Es sind ja noch nicht einmal alle Arbeitsplätze in der Lage Windows einigermaßen akzeptabel in Anwendungen zu fahren, geschweige denn als Arbeitsplätze zur Programmentwicklung. Natürlich könnte man alle Compiler in der Kommandozeilenversion verwenden; wer aber einmal eine moderne Entwicklungsumgebung benutzt hat, wird dieser Variante nichts mehr abgewinnen.

Die Bibliotheken machen den Unterschied

Das SDK-Entwicklungspaket (System Development Kit) war wohl etwas für die Pioniere der ersten Stunde. Es wird angesichts der neuen Bibliotheken in immer mehr reduzierter Form dokumentiert, wenn überhaupt. An seine Stelle treten die herstellerspezifischen Klassenbibliotheken, die die Komplexität der Fenstertechniken geschickt verbergen oder (wir wollen es nicht hoffen) durch neue Komplexität ersetzen.

Die Stärke eines Compilers liegt also nur zu einem Teil in der mehr oder weniger schnellen und effektiven Übersetzung der Sprache selbst; sie liegt auch im Angebot einer gut organisierten Klassenbibliothek für die Programmentwicklung allgemein. Daher wollen wir in unserer heutigen Ausgabe den Versuch unternehmen, das bekannte "Hello world"-Beispiel à la Windows mit den Foundation-Classes (MICROSOFT) einerseits und mit Object-Windows (BORLAND) andererseits zeigen.

Wer einmal begonnen hat, eine der beiden Bibliotheken zu verwenden, bleibt dabei, ja muß dabei bleiben, denn in C oder in C++ kann man mit einiger Erfahrung und vielleicht mit einer eigenen Standardbibliothek einigermaßen compilerunabhängig programmieren. Einmal mit einer herstellerspezifischen Bibliothek verheiratet, macht einen Umstieg zum Problem. (Aber es gibt ja auch glückliche Ehen!) Ein Versuch, eine Klassenbibliothek zu portieren ist aber schon geglückt: In seiner neuesten Ausgabe wird im "C-User's Journal" von einer Portierung der Foundation-Classes nach UNIX berichtet.

Geht es nicht einfacher?

Aber ja! Wir werden mit einer Sprache konfrontiert, die wir eigentlich schon beschrieben haben: **BASIC**, allerdings in neuem Gewand, in vielen Varianten (WordBasic, Access-Basic, Visual-Basic). Zwischen diesen Dialekten besteht eine enge Verwandtschaft, die in den nächsten Versionen der Anwenderprogramme durch den Nachfolger **VBA (Visual-Basic for Applications)** ersetzt werden wird. Diese wünschenswerte Einheitlichkeit ist durch die Vorreiterrolle von MICROSOFT in weiten Teilen des Softwaremarktes möglich. Aus didaktischen Gründen sollte man sich diesen Durchbruch wünschen, wenn auch die Akzeptanz der Sprache BASIC nicht in allen Fällen gegeben sein wird. Skeptikern sei berichtet, daß dieses BASIC mit seinen Ahnen durch die vielen Erweiterungen nicht mehr viel gemeinsam hat und durchaus salonfähig wurde.

Visual Basic for Applications bietet sich als Einsteigersprache an, denn diese Sprache wird jeder Anwendung als eine hochwertige Makrosprache beigegeben (Text, Tabelle und Datenbank). Als Erste wird EXCEL 5.0 damit ausgerüstet. Wir werden darüber berichten. Einen Vorgeschmack auf dieses BASIC kann aber bereits jetzt in den vielen Anwenderprogrammen und in Visual-BASIC finden. Kleine Anwendungen für WinBasic, sowie eine Einführung in Visual Basic finden Sie in in diesem Heft. Herr Nigischer stellt Ihnen die Arbeitsweise von Visual-Basic vor, unter dem Motto: Wer mit Windows umgehen kann, beherrscht auch Visual-Basic.

Wie kann man sich einarbeiten?

Seit einigen Jahren gibt es das **Microsoft System Journal**, welches sich in weiten Teilen mit Gebieten der Windows-Programmierung auseinandersetzt. Als Ausgangspunkt bietet sich eine Sonderausgabe über das **Programmieren in Visual-C++** mit einer Begleit-CD-ROM mit ca. 300 MB an Datenmaterial. In dieser Sonderausgabe gibt es auch einen Abschnitt mit einer Übersicht über einführende Literatur und unterstützende Programme. Preis: S 270,- Bezugsquelle: System Journal, Leserservice 7311, D-97064 Würzburg oder bei Konrad, Hergottschneitzergasse 8, 2344 Maria Enzersdorf, 022 36-27 371.

Eine weitere Quelle ist die **Microsoft Developer Network CD**, die eine Dokumentationssammlung und eine Sammlung von Hintergrundinformation über Windows-Programmierung ist. Sogar das Buch von Charles Petzold über Windows-Programmierung ist auf dieser CD enthalten. Da die Inhalte einer raschen Erneuerung bedürfen, gibt es die CD auch im Abo (4x jährlich). In den **PC-NEWS-32** (Windows), Seite 26 wurde darüber berichtet. Die CD darf interessanterweise nicht in einem Verleihsystem weitergegeben werden, wie man einem Beipacktext entnimmt, man kann sie aber weiterverkaufen oder schenken (und muß darüber Aufzeichnungen führen). Diese CD ist ein gelungenes Beispiel dafür, wie man große Informationsmengen durchaus verschiedener Provenienz in einer einheitlichen Bedieneroberfläche darstellen kann.

Und für die Profis gibt es seit Kurzem (ebefalls auf CD) das **Microsoft Developer Network Level Two**, das nicht nur die Dokumentation, sondern **auch die SDKs** selbst enthält; und zwar **alle**: 16 und 32 bit, englisch und international, SDK und DDK und auch das Betriebssystem W4WG 3.11. (Achtung: das sind keine Beschreibungen sondern Programme!)

Aus dem Inhalt: MSDOS 6.2, Windows 3.1 (viele Sprachen), Windows for Workgroups 3.11 (englisch), Windows NT 3.1 (viele Sprachen)(x86,Alpha-PC,MIPS), Financial SDK, MAPI, ODBC-SDK, OLE-SDK, OS/2 to Windows NT Porting-Kit, RLTool, Schedule-SDK, TCP/IP Sockets Developer Kit, Windows-3.1+NT3.2 (SDK+DDK), Win32 SDK. Aufgeteilt auf 4 CD's.

Bestellbar bei: Microsoft Subscription Centre, P.O. Box 5147, NL-2900 EC Capelle a/d IJssel, TEL: +31-10-258 88 64, FAX: +31 10 258 88-63. Preis: öS 4.250,-.

IBM hat die Bedeutung der Entwickler für den Erfolg eines Betriebssystems ebenso erkannt und startet mit dem **OS/2 Developer** um \$ 70,- ebenfalls eine Zeitschrift, die man bei OS/2 DEVELOPER, P.O.Box 1079, Skokie, IL, 60076-8079, USA, FAX: 1-415-905-2233 bestellen kann. Gleichfalls erhältlich sind Developer-CDs; Daten dazu sind der Redaktion nicht bekannt, bitte bei IBM anfragen.

Man sieht, der Erzeuger eines Betriebssystems hat als alleiniger Anbieter der Basisinformation einen großen Vorteil auch beim Vertrieb der "nachgeschalteten" Produkte, wie der Compiler und der Anwendungen, und Mitbewerber müssen diesen Vorsprung an Bekanntheit durch zusätzliche Features ihrer Produkte ausgleichen. Es ist am Kunden, diese Wettbewerbssituation für seine Anwendung richtig zu nutzen.

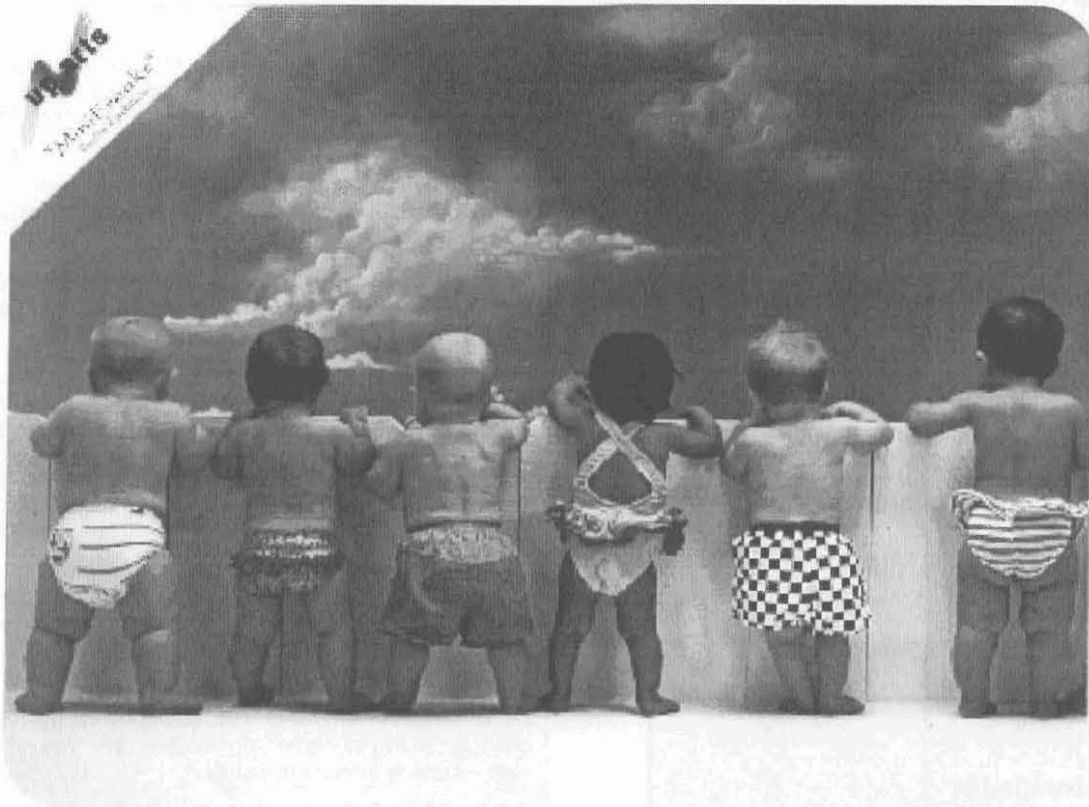
Es geschieht viel, um die Angst vor der aufwendigen Windows-Programmierung abzubauen. Die Oberflächen der Werkzeuge gleichen sich an, der Entwurf von Bildschirmen und Menüs wird zum Bildschirmspiel. Die Makrosprachen der großen Anwenderprogramme konvergieren zu einem einheitlichen Standard: VBA (Visual-Basic for Applications). Excel 5.0 wird erstmals mit dieser Sprache ausgerüstet sein.

Wir können über die Vor- oder Nachteile von BASIC diskutieren, in der Variante von Visual-Basic hat es einiges von seinem GOTO-Schrecken verloren und es bietet sich als *die* gesuchte Einsteigersprache an, die man im Anschluß an eines der Anwenderprogramme WinWord, Excel oder Access lernen kann.

Unter den mehr als 50 MB, die eine der modernen C++-Varianten an Ballast mit sich herumschleppt, sind auch viele Beispielprogramme, die dem Lernenden eine Hilfe durch den Dschungel sein sollen. Überläßt man Schüler, die bereits C oder C++ können, etwa den Foundation-Classes (MICROSOFT) oder Object-Windows (BORLAND), kann man hören, daß die Beispiele der Handbücher trotz ihrer Einfachheit noch immer zu kompliziert sind, um step-by-step verstanden werden zu können. Einen einfachen Einstieg, verschaffen die Beiträge über die Foundation-Classes und über Object-Windows in diesem Heft. Es wurden nur jene Kodeteile abgedruckt, die vom Benutzer eingegeben werden müssen.

Aber: Ohne C++ oder gar ohne C geht nichts! Für einen raschen Einstieg sorgt das C-Seminar mit dem Titel: "Der schnelle Weg zum hohen C". Das gesamte Seminar ist zu umfangreich, um in den **PC-NEWS** abgedruckt werden, daher hier eine Vorstellung: >>> Seite 6

Die andere Matte für Ihre Computer-Maus !



- Mousepad mit Spezial-Kunststoffoberfläche
- 4-Farben-Druck mit 5-Jahres-Garantie
- limitierte Auflage von 1000 Stück mit Echtheitszertifikat (numeriert)
- weitere limitierte Auflagen garantiert
- heißer Sammlertip für die Schnellen

Blitz-Bestellkarte

- Ja, ich bestelle Stk. Mousepad der Ausgabe "MiniFreaks" zu ÖS 189,-/Stk. inkl. MWSt. (zzgl. ÖS 25,- f. Versand)
- Ja, ich bin an weiteren Ausgaben von "up arts" Mousepads interessiert. Informieren Sie mich.
- Ja, ich bin an Mousepad Sammler-Informationen interessiert.

Name, Firma: _____

Straße: _____

PLZ, Ort: _____

Unterschrift: _____

Gewünschte Zahlungsweise:

- Geld liegt bei (Bar oder Scheck)
- per Nachnahme (+ ÖS 15,- Nachnahmegebühr)

einsenden oder faxen
an 02749/3326

2300

S 5.50
die sich
lohnen !

KÖNIG
MARKETING-SERVICES

Schulstraße 3
A-3385 Prinzersdorf
Tel. + Fax 02749/3326

up arts
Sammler-Mousepads

MICROSOFT DOS 6.2 Sammelbestellaktion

Was bietet MS-DOS 6.2 gegenüber DOS 6.0/DOS 5.0?

DBLSPACE - Festplattenverdopplung wurde überarbeitet und in der neuen Version wurde die Datensicherheit erheblich verbessert. **DBLSPACE** überprüft die Intaktheit der Daten und in Zusammenarbeit mit dem Programm **SCANDISK** von Peter Norton können Fehler rechtzeitig erkannt und beseitigt werden. **DISKCOPY** kann endlich (!!!) HD-Disketten ohne mehrfachen Diskettenwechsel kopieren. Der Befehl **COPY**, das Programm **XCOPY** und das Programm **MOVE** fragen vor dem Überschreiben von existierenden Dateien ab. **HIMEM.SYS** wurde um eine Testroutine erweitert, d.h. es testet jetzt jedes Mal den hohen Speicherbereich.

Die Version wird als Update in Buchform verkauft, ist aber **bootfähig(!)**.

In dem Hand-**BUCH** (!) (Hardcover) werden alle Befehle dokumentiert, und es werden für den Laien die Grundlagen des Handlings von DOS erklärt.

Das Update kostet ÖS 990.- (Listenpreis).

Durch eine Sammelbestellaktion kann der Preis auf ca. **850.- bis 900.-** sinken. (zzgl. Porto)

Interessenten senden bitte eine **Postkarte** an:

Fritz H. Zetik

**Himmelstraße 15
1190 Wien**

mit Name, Adresse und Telefonnummer (**TAGSÜBER** und **ABENDS**).

Bitte die Karten rechtzeitig bis zum **15. Februar 1994** absenden, ich rufe Sie dann an und teile Ihnen den endgültigen Preis mit, den Sie dann bitte auf mein Konto PSK.: 79 13 15 91 Blz.: 60000 lautend auf Fritz H. Zetik einzahlen. (Namen nicht vergessen!) Bitte auf der Post einzahlen, dann ist das Geld rechtzeitig verfügbar. Die Auslieferung bzw. Versendung erfolgt dann in der 3. Februarwoche.
(Siehe auch Beitrag **Was ist neu am DOS-6.2?**)

Der schnelle Weg zum hohen C

Zeigt an Hand ausgewählter Beispiele jene Bereiche von C und C++, die beim einführenden Unterricht derzeit noch zu kurz kommen, die aber für das Verständnis professioneller Programme unerlässlich sind. 60 Seiten A5, mit Begleiddiskette. Gearbeitet wird mit dem BORLAND-C++-Compiler Version 3.1. Aus dem Inhalt:

Teil 1: Installation des Compilers

Geht man bei der Installation eines modernen Compilers unbedacht vor, vergibt man bis zu 50MB Speicherplatz. Dieser Abschnitt zeigt, was ja und was eher nicht zu installieren ist.

Teil 2: Einstellungen und Optionen des Compilers

Mancher Ärger entsteht dadurch, daß man Fehler am falschen Platz sucht. Verstehen Sie die Einstellungen Ihres Compilers an Hand dieser Tabelle.

Teil 3: Grundlegendes über C und Pointer

Wichtige Kleinigkeiten über Pointer wollen wir nicht unbeachtet lassen.

Teil 4: Von der Struktur zu Klasse

Schritt für Schritt wird von einer Struktur zu einer Klasse übergeleitet und die neuen Elemente wie Konstruktoren und Destruktoren eingeführt. Als Beispiel dient ein Bildschirmpunkt.

Teil 5: Überladen von Funktionen und Operatoren

Nach der Demonstration der mitgelieferten Klassen COMPLEX und BCD wird das Überladen der Operatoren an der Punktklasse geprobt.

Teil 6: Ein vielgestaltiger Ball; Spiel (Polymorphie)

Wie man die Verwandtschaft wie Bildschirmabbildungen durch Vererbung festlegt und dabei eine Menge redundanten Codes spart.

Teil 7: Container-Klassen und Listen

Es wird versucht die Notwendigkeit der neueren Spracherweiterungen zu erklären und höherwertige Datenstrukturen wie Container zu entwickeln.

Teil 8: Fenster in DOS, Turbo-Vision

Es ist zwar - verglichen mit selbstprogrammierten Fenstern - ein geringer Aufwand, Arbeitsoberflächen mit Turbo-Vision zu erstellen aber einiges will beachtet werden.

Teil 9: Fenster in Windows, Object-Windows

Windows-Programmierung kann in dieser kurzen Darstellung nicht vermittelt werden, nur so viel als man benötigt eine eigene einfache Oberfläche zu schreiben.

Verstehen Sie diesen Sonderdruck nicht als ein vollständiges Skriptum; es ist vielmehr ein Versuch, an durchaus anspruchsvollen Beispielen, wichtige Probleme der Sprache C und deren Lösung auch durch die neuen Möglichkeiten von C++ zu besprechen. Ein auf Vollständigkeit bedachtes Skriptum oder Buch ist daneben durchaus anzuraten.

Die Schwerpunkte dieses Seminars sind die **Teile 4..7**. Sowohl die C-Grundlagen als auch die Fensterprogrammierung (**Teile 3, 8..9**) erfordern weitere Unterlagen.

Interessenten können das Skriptum zum Preis von S 80,- inklusive Diskette als **PC-NEWS**-Sonderdruck SON-3 bestellen.

Werbung für die PC-NEWS

Probeheft: Leider eignen sich die **PC-NEWS** wegen ihres Umfangs nicht sehr gut zum Verschenken als Probeheft. Daher werden Interessenten gebeten, im Sinne des Sharewaregedankens ein Probeheft bei Gefallen zu bezahlen und ansonsten an Freunde weiterzugeben.

Werbeblatt: Nebenstehend finden Sie die erste Seite der nächsten Ausgabe mit einem Werbetext auf der Rückseite. Alle Leser werden gebeten, dieses Blatt herauszutrennen und an mögliche Interessenten weiterzugeben.



Blatt bei Bedarf heraustrennen



Bazar

Biete **Motherboard 386DX-25** mit Coprozessor 387-33 und 4x1MB SIPP bestückt um VB 2.500,-, Tel.: (abends) (0222) 40 30 466 (Herr Premauer).

BORLAND-C++ 3.1, Original-Handbücher+Disketten: S 1000,- (+Versand oder Abholung). (0222) 604-50-70 (Fiala).

BORLAND-REFLEX-Datenbank mit Grafikeigenschaften, Original-Handbücher+Disketten: S 50,-(+Versand oder Abholung). (0222) 604-50-70 (Fiala).

BORLAND TURBO-Prolog 2.0, Handbücher+Disks + Prolog Toolbox S 100,- (+Versand oder Abholung). (0222) 604-50-70 (Fiala)

Festschrift 10 Jahre



MCCA Bestellkarte

- Senden Sie mir die Festschrift "10 Jahre MCCA" zum Preis vonS 80,-
- Ich werde MCCA-Mitglied
- als FirmaS 1.200,-
 - als EinzelmitgliedS 600,-
 - als Einzelmitglied mit Ermäßigung.....S 300,-
jeweils zuzüglich S 100,- Anmeldegebühr
- Ich möchte vom Clubangebot des ermäßigten Bezugs der **PC-NEWS** Gebrauch machen und bestelle das Abo der **PC-NEWS** zum MCCA-Jahrespreis von.....S 90,-
(statt S 150,-).

Datum: _____ Unterschrift: _____

Bezugsbedingungen für in den **PC-NEWS** beschriebene Dokumente (Disketten oder Literatur):

Alle Leser können die Disketten oder Literatur über die Mailbox **His Master's Voice** bestellen und senden die rechts abgebildete Bestellkarte an:

Werner Illsinger,
Flurschützstraße 36/12/5
1120 Wien

Mitglieder des **PCC-TGM** können auch über den Club bestellen und senden die Bestellkarte an:

PCC-TGM
Postfach 59
1202 Wien

Nicht vergessen, umseitig **richtige Adresse** einzutragen

DISKETTEN-DOWNLOAD

Mailbox **His Master's Voice** zum downloaden im Area PCNDISK.
Rufnummer siehe Seite mit Impressum.

Disketten/Literatur Bestellkarte

Gewünschtes Diskettenformat:

- 5,25",360k(40,-) 5,25",1.2MB(50,-) 3,5",720kB(50,-) 3,5",1.44MB(60,-)

Bitte geben Sie die gewünschte Nummer der **PCN-DSK**, bzw.

den Namen der CD-ROM, Unterverzeichnis, Dateinamen, Dateigröße an.

Verwenden Sie diesen Bestellschein auch für Zusatzliteratur **PCN-LIT** und Sonderdrucke **PCN-SON**.

Datum: _____ Unterschrift: _____

PC-NEWS Bestellkarte

- Gesamtverzeichnis = PC-NEWS-34**

Schicken Sie mir ein Verzeichnis aller **PC-NEWS**, 44 Seiten mit Diskettenverzeichnis um S 20,- zuzügl. Porto.

- Probenummer**

Schicken Sie mir eine Probenummer. Bei Gefallen zahle ich einen Beitrag für Druck und Versand.

- Abo**

Ich bestelle ein Abo der **PC-NEWS** ab der nächsten Ausgabe. Ich erhalte 5 Hefte pro Jahr zum Preis von S 150,- inklusive Versand. (Ausland S 220,-). Ich kann das Abo jederzeit stornieren und erhalte dann den anteiligen Restbetrag zurück. Ich werde vor Ablauf zur Erneuerung eingeladen.

- XT/AT-Handbook**

Ich bestelle das XT/AT-Handbook, eine Sammlung von 64 Tabellen rund um die Hard- und Software im PC (englisch); ideal für Entwickler, Programmierer zum Preis von S 80,- zuzüglich Versand.

- Eine Ausgabe der PC-NEWS** (nur für Abonnenten)

zum Preis von S 40,-, drei Ausgaben S 100,-, fünf Ausgaben S 150,- zuzüglich Versand

- 1/92 2/92 3/92 4/92 1/93 2/93 2a/93 3/93 4/93 5/93 1/94

Datum: _____ Unterschrift: _____

Vorname / Firma	Name	Titel	
Straße		PLZ	Ort
VW/TEL	FAX	BTX	
FIDO/TELEBOX	CompuServe	Geb.Datum	
Firma	Name	Titel	
Straße		PLZ	Ort
VW/TEL/KLAPPE	FAX	BTX	

Bitte mit
S 5,50
frankieren

An den

MCCA

Traungasse 12
Postfach 143
1033 Wien

Vorname / Firma	Name	Titel	
Straße		PLZ	Ort
VW/TEL	FAX	BTX	
FIDO/TELEBOX	CompuServe	Geb.Datum	
Firma	Name	Titel	
Straße		PLZ	Ort
VW/TEL/KLAPPE	FAX	BTX	

Bitte mit
S 5,50
frankieren

Vorname / Firma	Name	Titel	
Straße		PLZ	Ort
VW/TEL	FAX	BTX	
FIDO/TELEBOX	CompuServe	Geb.Datum	
Firma	Name	Titel	
Straße		PLZ	Ort
VW/TEL/KLAPPE	FAX	BTX	

Bitte mit
S 5,50
frankieren

An den

PC-NEWS - *Eigenverlag*

Siccardsburggasse 4/1/22
1100 Wien

PCC-TGM Bestellkarte

Ich melde mich für eine Mitgliedschaft beim PCC-TGM (Personal Computer Club-Technologisches Gewerbemuseum) an. Die Mitgliedschaft ermöglicht:
Einkauf zu besonderen Bedingungen bei verschiedenen Händlern

- Teilnahme an Sammelbestellungen
- Jahresabo der PC-NEWS inkludiert
- Teilnahme an Seminaren und Clubabenden im TGM

Bei Eintritt ab September zahle ich für das nächste Jahr und erhalte dennoch alle Clubleistungen für das laufende Jahr. Ich erkläre mich mit der Weitergabe meiner Daten an Clubmitglieder und Lieferfirmen einverstanden ja nein.

Ich melde mich an als

- als Schüler (Schulbesuchsbestätigung liegt bei)S 200,-
 als Student (Inskriptionsbestätigung liegt bei)S 300,-
 jeder andereS 400,-

Datum: _____ Unterschrift: _____

DISKETTEN-DOWNLOAD in der Mailbox **His Master's Voice** zum im Area ADIM.
Rufnummer siehe Seite mit Impressum.

Die Hardwarepreise ändern sich in kurzer Zeit so rasch, daß der Bestellschein rasch ungültig würde. Wir organisieren Sammelbestellungen für Disketten, Modems und ähnliches Zubehör. Bitte fragen Sie in der telefonischen Sprechstunde der **ADIM** Wien oder fordern Sie eine aktuelle Liste (telefonisch oder schriftlich) an!

ADIM Bestellkarte

Nr=Bestellung von fertigen Skripten,
(Nr)= ~Vormerkung für geplante Skripten (bitte beachten Sie die BTX-Seite *56458# oder rufen Sie uns an):

Nr.	Produkt(Hersteller)	Version	ISBN		Auflage	Datum	nur Band		nur Disk		Band+Disk		Gesamtpreis
			ohne Disk	mit Disk			öS	Stk	öS	Stk	öS	Stk	
36	LOGO(IBM)	1.0	002-5	003-3	2.	Nov88	45		50		85		
38	Turbo-Pascal(Borland)	3.01	006-8	007-6	5.	Sep89	90		50		130		
39	RUN/CClassicABVERKAUF!	2.03	000-9	001-7	1.	Juli87	40		50		80		
40-4	Turbo-C(Borland)ABVERKAUF!	2.0	014-9	015-7	4.	Aug90	50		50		90		
40	Turbo-C(Borland)6226*	2.0	034-3	035-1	6.	Aug93	100		50		140		
41	Turbo/PowerBasic6451*	1/2	030-0	031-9	4.	Sep92	95		50		135		
(42)	C-Bibliothek(Turbo-C,Quick-C)				1.		~130		50		~170		
(43)	MS-DOS	6.0			1.		~100		50		~140		
(44)	Modula-2(Jensen&Partner)	3.1			2.	Nov93	~120		50		~160		
(46)	Word(MicroSoft)	6.0			1.		~100		50		~140		
47	Turbo-Pascal(Borland)**)6225*	7.0	032-7	033-5	6.	Okt93	120		50		160		
(48)	Quick-C(MicroSoft)	2.5			2.		~100		50		~140		
49	Quick-Basic(MicroSoft)	4.5	018-1	019-X	2.	Okt91	95		50		135		
50	C++(Borland)6450*	3.1	036-X	037-8	3.	Sep83	120		50		160		
98	PeterPfenicher:TurboPascal AnweisungssammlungmitBeispielen	6.0	ISBN3- 900985-00-6		4.	Okt91	130						
Gesamtanzahl der bestellte Skripten/Disketten													
Bandnummer für Freixemplare (für je 20 lieferbare Bände ein beliebiger Band)													0
Versandkostenanteil pro SKRIPTEN-Sendung (entfällt ab S 1000,- Bestellwert)													30
Endsumme (inklusive 10% bzw. 20% Umsatzsteuer)													

- *) Fachbuchnummer (für Höhere technische Lehranstalten auch über die Schulbuchaktion zu beziehen) Verlagsnr. 970
 **) Nur mehr Version 7.0 lieferbar. Der Band ist auch für die Versionen 4.0 bis 6.0 verwendbar. Alle Teile, die die älteren Sprachversionen betreffen, sind durch eine andere Schrift gekennzeichnet. Das neue Skriptum enthält zusätzliche Abschnitte und Beispiele.

Wien, Stand: 1. Februar 1993 (PC-NEWS-36)

Änderungen - insbesondere bei den Versandkosten - und Irrtum vorbehalten!

Datum: _____ Unterschrift: _____

>>>>>hier falten >>>>>

PC-NEWS - Sonderdrucke

PCN-SON-3: Der schnelle Weg zum hohen C

Eine papierene Wiedergabe des C/C++-Seminars beim PCC-TGM im Jahr 1993.

PC-NEWS - Literatur

PCN-LIT-66: Mathematica

Originalausdrucke des gleichnamigen Beitrags in den PC-NEWS-36. 23 Seiten deutsch.

PCN-LIT-67: TELEBOX-PC

Dokumentation zur Installation des TELEBOX-PC.

PCN-LIT-68: WORD für WINDOWS, Einführung

Mag. Eduard Fleck, 25 Seiten, deutsch, Begleitmaterial zum gleichnamigen Seminar des PCC-TGM.

PC-NEWS - Disketten

(1) 5-1/4", 360k, (2) 5-1/4", 1.2MB, (3) 3-1/2", 720 kB, (4) 3-1/2", 1.44 MB

PCN-DSK-374(4): Ashling Demo

PCN-DSK-375(4): Perestroika, Spiel

PCN-DSK-376(4): PC-AUSTRIA WINDOWS

Die folgenden Programme werden mit vielen anderen in der gleichnamigen Zeitschrift vorgestellt.

!INIMAN	EXE	Analysiert und editiert INI-Dateien
!WRU	EXE	"Where are You" sucht Dateien quer-disk-ein
DISKFACT	EXE	Kopiert und Formatiert Disketten im Hintergrund
MICRLATH	EXE	Rotationskörpermodelle mit Beleuchtungs- und Farbeffekten
UCLIP	EXE	Erweiterung der Windows-Zwischenablage
WEMPIRE	EXE	World-Empire, Spiel mit Maus-Bedienung
WFORM	EXE	Formulare für dBase-Dateien erstellen
WKIRK	EXE	Erweiterung des Windows-Dateimanagers
WKONTO	EXE	Verschiedene Bankkonten verwalten

PCN-DSK-377(3): ASHLING-Demos für 8051-Entwicklung

PathFinder for Windows, PathFinder DOS, Performance Analyser DOS Ver. 4.1.3

PCN-DSK-378(3): G.E.S.y II Graphical Editor for Structograms, Demo, Version 2.1

Siehe ausführliche Beschreibung in den PC-NEWS-36.

PCN-DSK-379(3): G.E.S.y II Graphischer Editor für Struktogramme, Demo, Version 2.1

Siehe ausführliche Beschreibung in den PC-NEWS-36.

PCN-DSK-380(3): MSDOS-KERMIT 3.13

PCN-DSK-381(3): MSDOS-KERMIT, Technical Documentation Disk 1(2)

PCN-DSK-382(3): MSDOS-KERMIT, Technical Documentation Disk 2(2)

PCN-DSK-383(3): C-KERMIT, Source 1(6)

Documentation, Makefiles, Sources

PCN-DSK-384(3): C-KERMIT, Source 2(6)

Generic Unix source files

PCN-DSK-385(3): C-KERMIT, Source 3(6)

Generic Unix source files

PCN-DSK-386(3): C-KERMIT, Source 4(6)

Generic Unix source files

PCN-DSK-387(3): C-KERMIT, Source 5(2)

VMS, OS/2, OS9 source files

PCN-DSK-388(3): C-KERMIT, Source 6(6)

DG, Atari, Amiga source files

PCN-DSK-389(3): ICON-HEAR-IT-Lite-Version

Demonstriert die Möglichkeiten von ICON-HEAR-IT. Beschrieben in PC-NEWS-37.

PCN-DSK-390(4): PCC-Magazin Extra 1993-94, Spiele

MENSCH.EXE	Mensch ärgere dich nicht
!WPOKER.EXE	Würfelpoker
ARIVA!.EXE	Linie durch Labyrinth führen
ARSMITH.EXE	Mauszeiger von Windows ändern
DAME41.EXE	Brettspiel Dame
MUHLE30.EXE	Brettspiel Mühle
FGOD.EXE	Jump and Run Spiel
KNIFFEL!.EXE	Würfelspiel
JET110.EXE	Jump and Run Spiel

PCN-DSK-391(4): PCC-Magazin 21, 11-12/93

!NOMEN.EXE	Datenbankprogramm unter Windows
!TRAINER.EXE	Maschinschreibtrainer unter Windows
DOK-PCC#.COM	Artikeldatei, die Sie lesen sollten
FD-FORM.EXE	Formatieren von Disketten mit mehr Kapazität
INI-MAN.EXE	Erklärt den Inhalt der INI-Dateien
TB-AV.EXE	Antivirenprogramm
WIN-FORM.EXE	Formulare erstellen und ausfüllen

PCN-DSK-392(4): Point-Installation für His Master's Voice (MS-DOS)

Nachfolgeversion zu PCN-DSK-344, Neuer Editor Gold-Ed.

PCN-DSK-393(4): Point-Installation für His Master's Voice (OS/2)

PCN-DSK-394(4): IST-Mail 2.0 Client

Bulletin-board-system im Auftrag des Bundesministerium für Unterricht und Kunst, Abt. II/12. Ziel dieser Mailbox ist es, ein Informationsmedium für den Lehrer in seiner professionellen Funktion zu sein. Der Server ist unter der Telefonnummer 0732/77 26 66 59 zu erreichen. Die Inhalte der PC-NEWS ab dieser Nummer und zunehmend auch rückwirkend sollen über diese Mailbox und über die Mailbox His Master's Voice abrufbar sein. Nähere Beschreibung dieser Box im nächsten Heft.

PCN-DSK-395(4): Microsoft ACCESS Companion Disk

PCN-DSK-396(4): Microsoft ACCESS Sample Code Disk

PCN-DSK-397(4): Microsoft ACCESS Seminar Disk

PCN-DSK-398(4): Microsoft VISUAL-C++ 1.0 Companion Disk

PCN-DSK-399(4): Microsoft VISUAL BASIC 2.0, Demo-Disk

PCN-DSK-400(4): Begleitdiskette C/C++-Seminar: Der schnelle Weg zum hohen C

PCN-DSK-401(4): Hardwarenahes Programmieren, Teil 7

PCN-DSK-402(4): Logitech AudioMan MS-DOS-Treiber sowie neue Windows-Treiber

PCN-DSK-403(4): TELEBOX-PC-Demo

PCN-DSK-404(4): Dateien zur PC-NEWS-36

PCN-DSK-405(4): Demoversion VISUAL-BASIC 1(2)

PCN-DSK-406(4): Demoversion VISUAL-BASIC 2(2)

PCN-DSK-407(4): Dateien des Visual-Basic-Beitrags in PC-NEWS-36 □

Liebe Mitglieder des PCC-TGM!

SEMINARE

Im November 1993 begann eine Seminarreihe für Mitglieder des **PCC-TGM** und des **MCCA**. Der bei den Seminaren angegebene Unkostenbeitrag ist bei der Anmeldung zu bezahlen und dient gleichzeitig als Platzsicherung. Falls ein Seminar nicht zustandekommen sollte, z.B. wegen zu geringer Teilnehmerzahl, wird dieser Betrag refundiert. Es gibt derzeit viele Anmeldeplätze, daher wird erwohnen, einzelne Seminare zu teilen.

Teilseminar S2:

INHALT: Einführung für Anfänger: Hardware, Betriebssystem DOS 5 und DOS 6, Hardware-Konfiguration und Setup, Virenschutz, WINDOWS 3.1, Dateimanager, Programmmanager, Installation.

TERMINE: Mi 12.1./19.1./26.1./2.2./16.2./23.2./2.3.1994

VORTRAGENDER: Ing. Syrovatka SAAL: 1400

UNKOSTENBEITRAG: S 350,-

Teilseminar S3:

INHALT: WORD für WINDOWS, Einführung für Anfänger: Text bearbeiten, formatieren und drucken. Textbausteine und Druckformate. Einbinden von Graphiken und Tabellen, Serienbriefe.

TERMINE: Mi 9.3./16.3./23.3./6.4./13.4./20.4.1994

VORTRAGENDER: Ing. Syrovatka SAAL: 1400

UNKOSTENBEITRAG: S 300,-

Teilseminar S4:

INHALT: EXCEL, Tabellenkalkulation unter WINDOWS, Einführung: Struktur einer Kalkulationstabelle. Hinzufügen, ändern und löschen von Daten. Graphische Darstellung von Berechnungen.

TERMINE: Di 15.3./22.3./12.4./19.4./26.4./3.5.1994

VORTRAGENDER: Dipl. Ing. Scharl SAAL: 1434

UNKOSTENBEITRAG: S 300,-

Teilseminar S5:

INHALT: MAILBOXEN, Zugang und Benutzung: Einloggen in die PCC-Mailbox "His Masters Voice". "Downloaden" von Programmen, Electronic-Mail.

TERMINE: Di 2 / 3 Abende WORKSHOP, Termine werden noch bekannt gegeben.

VORTRAGENDE: Ing. Zelinka, Ing. Illsinger SAAL: 1434

UNKOSTENBEITRAG: 100,-/150,-

Die Seminare finden zu den oben angeführten Terminen jeweils von 17 h 45 bis 20 h 55 am TGM - Wien 20., Wexstraße 19-23 - 14. Stock statt.

Alle Seminare finden ab einer Mindestzahl von 15 Teilnehmern statt. Sollte ein Seminar nicht zustandekommen, werden Sie rechtzeitig verständigt.

Anmeldungen: Sollten Sie an einem dieser Seminare Interesse haben, so bitten wir Sie um eine kurze schriftliche (Postkarte), telefonische (Tel 332 23 98 0 Anrufbeantworter) oder BTX-Mitteilung (*941# 912 222 584# bzw. BTX-Seite *56459#) oder mittels Einzahlung auf das Kto.Nr. 053-32338 bei 20111 Erste Österr. Spar-Casse. Der Zahlungsabschnitt gilt als Anmeldebestätigung.

Robert Syrovatka

Liebe Mitglieder und Freunde des MCCA

Das Clubjahr 1993 war ein ereignisreiches Jahr für die Mitarbeiter des **MCCA**!

Feierte der Club doch sein 10 jähriges Bestehen. Zum Sonderpostamt am 4.6.93 besuchten uns mehr als 300 Personen und nahmen Sonderpostbelege und Gedenkblätter, die mit dem Sonderstempel des **MCCA** versehen sind, mit nach Hause. Die Vorbereitungen für dieses Sonderpostamt waren gut geplant und in die Tat umgesetzt. Wenn Sie an diesem Tag keine Zeit hatten, uns zu besuchen, können Sie trotzdem über BTX auf Seite *25509011# (Preis 30,-, 60,-) in den Besitz dieser Belege kommen.

Um das Jubeljahr ausklingen zu lassen, veranstaltete der **MCCA** am 3.12.93 eine Party, zu der wir Sie herzlich eingeladen hatten. Nur ca. 35 Personen mit Begleitung nahmen leider die Gelegenheit wahr, bei guter Life-Musik (Entertainer) ein Tänzchen zu wagen und sich köstlich zu amüsieren.

Mit viel Fleiß gelang uns dann noch eine, so glaube ich, recht gut gelungene Festschrift herauszubringen, die wir den Clubmitgliedern gratis zukommen ließen. Durch unermüdete Anfragen an Banken und Firmen bekamen wir viele Werbeeinschaltungen als Jubiläumsgaben, die uns halfen, die Druckkosten einigermaßen in den Griff zu bekommen. Wenn Sie Interesse an dieser Dokumentation, die Einblicke in die BTX-Szene der letzten 10 Jahre gewährt, haben, bestellen sie diese über BTX auf Seite *255060# (Preis ÖS 80,-) Auch Beiträge namhafter Personen zu BTX durften nicht fehlen.

Gleichzeitig haben wir einen Fragebogen versendet, der dem **MCCA** die wahren Wünsche der Mitglieder an den **MCCA** und BTX aufzeigen soll. Dadurch helfen Sie den Mitarbeitern des **MCCA** den Verein neu zu strukturieren und die Ziele den Bedürfnissen der Mitglieder anzupassen. Vieles, was uns wichtig erschienen ist, wird dann vielleicht wegfallen, Neues wird in unser Programm aufgenommen werden. Auch die Post wird den **MCCA** mit seinen vielen Mitgliedern nicht ignorieren können und einige Vorschläge und Wünsche berücksichtigen müssen.

In guter Zusammenarbeit mit den **PC-NEWS**, in der sich jedesmal Artikel von Autoren des **MCCA** befinden, hat der **MCCA** einen wirklich interessanten Preis für ein Jahresabonnement dieser Fachzeitschrift, gekoppelt mit der **MCCA**-Mitgliedschaft, erreicht.

Dieses ABO für 5 Hefte im Jahr kostet ÖS 90,- und kann über BTX auf Seite *255091# bestellt werden. Diese Seite liegt aber in der geschlossenen Benutzergruppe des **MCCA** (GBG) und kann daher nur von Mitgliedern, die den Clubbeitrag 1994 entrichtet haben, aufgerufen werden.

Von ganzem Herzen möchte ich Ihnen und mit mir der Vorstand des **MCCA** ein erfolgreiches Jahr 1994 wünschen.

Ihr Helmuth Schlögl

Festschrift 10 Jahre MCCA

Der **MCCA** hat zum Anlaß seines zehnjährigen Jubiläums eine schöne und informative Festschrift herausgegeben. Sie enthält neben Vorworten von Dr. Sindelka (Generaldirektor der Österreichischen Bundespost) und Univ. Prof. Dr. Maurer ("Vater" des österreichischen BTX-Systems) Beiträge über den **MCCA**, seine Entwicklungsgeschichte und die interessanten Veränderungen der österreichischen BTX-Szene im Laufe der letzten zehn Jahre.

Bestellen können Sie die **MCCA**-Festschrift im BTX über die Seite *255060# oder über die beiliegende Antwortkarte.

Susanne Rupprecht, Georg Czedik-Eysenberg

Liebe Mitglieder des PCC-S!

PCC-Salzburg, ein Klub stellt sich vor

Am Freitag den 5. Februar 1988 19.00 Uhr war es soweit: Die Proponenten hoben den neuen Klub aus der Taufe. Die schon voll im Gange befindliche elektronisch-technische Revolution zwang einfach dazu, die Basis für eine private Bildungsoffensive zu legen. Da die klassischen Bildungseinrichtungen den Markt nicht genügend bedienen konnten, mußte eine Unterstützung für außerschulisches Lernen geschaffen werden. Der Beginn war furios: bei den bald beginnenden Kursen und Seminaren merkte man den beträchtlichen Nachholbedarf. Der Schwung dieser ersten Monate ist einem beschaulichen Klubleben gewichen, vor allem deswegen, weil das Klubziel in vielen Fällen erreicht worden ist: Viele Klubmitglieder sind gerätetüchtig bestens ausgerüstet und arbeiten problemlos mit den komplexesten Programmen! Vor allem der Spieltrieb der frühen Jahre ist einer gezielten Arbeitstechnik gewichen. Die in der Gründerzeit getätigten Hard- und Softwareaktionen haben eine Entwicklung angeworfen, die später durch die purzelnden Marktpreise unterstützt worden ist. So besteht unsere heutige Klubaktivität hauptsächlich darin, das Ohr am Puls der Entwicklung zu haben und Unterstützung zu bieten. Unser Kursprogramm spiegelt etwas von dieser Saturiertheit wieder. Wenn ich daran zurückdenke wie groß das Interesse an unseren Virus-Kursen gewesen ist und wie problemlos man heute mit diesen Plagegeistern umgeht, so zeigt das die Veralltägung der EDV.

Ein kleiner Ausschnitt aus den Statuten des **PCC-Salzburg** zeigen unsere Klubziele genauer:

- Förderung des Einsatzes des PC im Unterricht
- Lehrerfortbildung
- Erstellung von Lehrunterlagen
- Einsatz von CAD und CAE an der HTBLA
- Unterstützung fachlich interessierter Schüler
- Information der Mitglieder über neue Entwicklungen und Einsatzmöglichkeiten von PC und deren Peripheriegeräte
- Beratung der Mitglieder in allen mit dem Einsatz von PC zusammenhängenden Fragen
- Durchführung von Veranstaltungen für Mitglieder
- Zusammenarbeit mit Organisationen des In- und Auslandes, die ähnliche Zwecke verfolgen

In diesem Sinne erwarten wir von einem Schulerschluß der PC-Klubs einen gewichtigen Impuls für die zukünftigen Klubaktivitäten (vor allem im Bereich der elektronischen Telekommunikation).

Mit den besten Grüßen *Helmut Walters*

Kursprogramm Frühjahr 1994

Die angebotenen Kurse sind für Klubmitglieder kostenlos, die angegebenen Unkostenbeiträge gelten nur für interessierte Nichtklubmitglieder.

Um die Organisation zu erleichtern, bittet der Vorstand um eine baldige schriftliche Anmeldung, da die Kurse nur bei einer Mindestmeldung von 6 Teilnehmern abgehalten werden!! (Postfach Steindl oder Zimmer 303). Die genauen Termine hängen zum Teil von den schulischen Möglichkeiten ab und sind daher im Beiblatt zu den **PC-NEWS** (nur in der Version für den **PCC-Salzburg**) aufgeführt.

Februar 1994

GEDDY-CAD	4 Abende	400.- ÖS
DOS - Grundkurs	3 Abende	200.- ÖS
DOS - Erweiterung	3 Abende	200.- ÖS

März 1994

WINDOWS - Einführung	3 Abende	200.- ÖS
WINDOWS - Anwendungen	3 Abende	200.- ÖS
AUTOCAD - Grundkurs	3 Abende	300.- ÖS
GCS-CAD	3 Abende	300.- ÖS

Eine genauere Beschreibung der behandelten Lehrinhalte ist wieder aus der Beilage zu entnehmen. Die Kurszeiten sind Montag und Dienstag jeweils von 18.00 bis 19.30 Uhr. Kursunterlagen werden vom Klub kostenlos beigelegt.

Mit den besten Grüßen

Andreas Steindl

Neues von der ADIM

Modemaktion

Die Post hat die BTX-Auslandsübergänge (D, CH, L) zum Ortstarif (066811 bis 066813) aufgelassen. Statt dessen kann von der normalen Leitseite her direkt zu den Auslandsübergängen verzweigt werden. Als Vorteil wird dabei angegeben, daß dieser Übergang nun auch von schnellen Modems (2400 Baud über 06611 oder 14400 Baud über 06612) verwendet werden kann. Dem steht der nicht so gerne erwähnte Nachteil gegenüber, daß diese Zugänge jetzt mehr als 3 Schilling pro Minute kosten.

Mein privater Kommentar dazu: für ein lockeres Blättern in einem anderen BTX-System sind die Auslandsübergänge damit zu teuer und uninteressant geworden. Vertretbar ist nur mehr die gezielte Suche nach bestimmten Informationen.

Die alten Übergänge waren nur mit dem V.23-Modus zu verwenden. Somit besteht keine Notwendigkeit mehr, Modems mit V.23-Modus zu verwenden. Da inzwischen Modems von den verschiedensten Firmen angeboten werden, werden Modems von der **ADIM** nur mehr auf Bestellung besorgt. Wir werden aber weiterhin neue, interessante Geräte erproben und in den **PC-NEWS** vorstellen.

Neue Skripten

Ich bitte um weitere Stellungnahmen zu einem Skriptum "Mini-Pascal". (Wie schon erwähnt sollte es etwa den halben Umfang des Bandes 47 haben und nur auf diejenigen Sprachelemente eingehen, die im Unterricht dann sicher umgesetzt werden können, wenn hinreichend viel Zeit für die andere Themen, wie zum Beispiel Anwendersoftware bleiben soll.)

Vorteile eines Mini-Pascal-Skriptums: dünner; billiger; kein Stoff enthalten, für den "sowieso keine Zeit bleibt".

Nachteile: keine Anregungen für das Selbststudium; keine besonderen Anregungen für sehr interessierte Schüler; keine vollständige Darstellung der Sprache; wegen der zu erwartenden geringen Auflage keine zur Seitenzahl proportionale Verbilligung.

AutoCad-Skripten

Derzeit wird mit großem Eifer an einem AutoCAD-Band (53) für die 2-D-Konstruktion gearbeitet. Das neue Skriptum ist nicht als Nachschlagewerk gedacht; vielmehr werden die Elemente von AutoCAD in einer didaktisch sinnvollen Reihenfolge besprochen. Der neue Band wird viele Zeichnungen enthalten, etwa im März lieferbar sein und 150 bis 160 Schilling kosten. Natürlich gibt es auch wieder eine Diskette dazu: die Diskette ist besonders nützlich, wenn mit den Beispielzeichnungen gearbeitet werden soll. Ein **ADIM**-Band (54) für die 3-D-Konstruktion und ein weiterer (54) für AutoLisp sind ebenfalls in Planung.

Martin Weissenböck

Neues aus der Mailbox His Master's Voice

In den **PC-NEWS-35** vom November habe ich angekündigt, eine Befragung der Benutzer der Mailbox durchzuführen. Diese Befragung lief von 30. Oktober bis 26. Dezember. Schwerpunkte der Befragung waren die Zufriedenheit der Benutzer mit der Mailbox, die Finanzierungsmöglichkeiten und auch die Erweiterung der Dienste der Mailbox. Ausgewertet wurden 537 eingelangte Antworten.

Zufriedenheit

Die Benutzer der Mailbox wurden gebeten, die Mailbox auf einer Skala von 1 bis 5 (1 = sehr gut, 5 = nicht genügend) zu beurteilen. Der Notenschnitt dafür lag bei 1,8.

Es wurden viele Dinge angeregt, die hinzugefügt werden sollten, um die Qualität der Mailbox zu heben. Einige davon sollen hier aufgezählt werden:

- mehr Übersichtlichkeit (16x)
- mehr OS/2 (16x)
- mehr Windows (16x)
- C't Magazin Files (9x)
- Source Code (6x)

Mit den seit Jänner neuen Menüs soll die Übersichtlichkeit etwas besser werden - vor allem durch die Leitfarben (blau = Dateien, grün = Mitteilungen) und durch die vollständige Hilfe (auch in deutscher Sprache). Die C't-File-Area (Listings der Zeitschrift C't) wurde hinzugefügt und die in den letzten **PC-NEWS** vorgestellte Source Code CD-ROM wird so bald wie möglich (derzeit fehlt nach der nötigen Platz auf der Festplatte) zum Download angeboten werden.

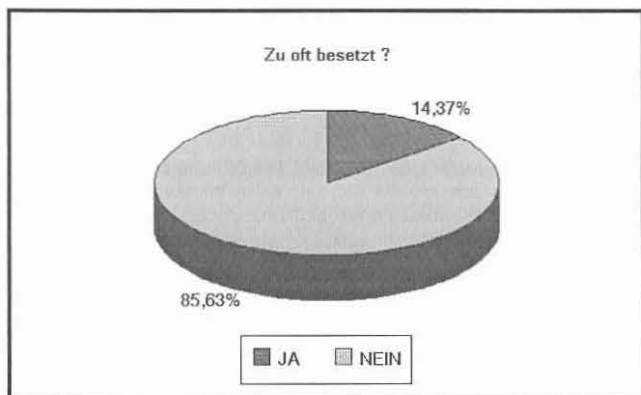
Nach den Antworten der Benutzer im Fragebogen sollte eher nichts weggelassen werden, eher mehr hinzugefügt, da eine größere Auswahl das Angebot bereichert. Folgende Dinge wurden angeregt wegzulassen:

- GIF Bilder (13x)
- OS/2 Programme (12x)
- DOS/Windows Programme (12x)
- alte Programme (10x)

Wie man sieht, sind für OS/2-Benutzer die DOS- und Windows-Programme eher uninteressant und für die DOS-Anwender die OS/2-Programme, daher wird es in nächster Zeit für den Benutzer möglich sein, gewisse Bereiche "auszublenden". Dadurch, daß für den Benutzer subjektiv uninteressante Bereiche nicht mehr sichtbar sind, sollte auch wieder die Übersichtlichkeit steigen.

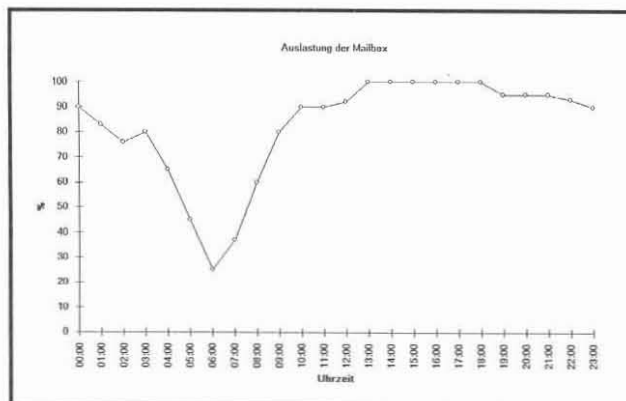
Besetzungsfälle

Eine weitere Frage war: "Ist die Mailbox zu oft besetzt?" 85 % der Anrufer finden, daß die Mailbox nicht zu oft besetzt ist, was ein Zeichen dafür ist, daß (noch) keine weiteren Telefonleitungen benötigt werden.



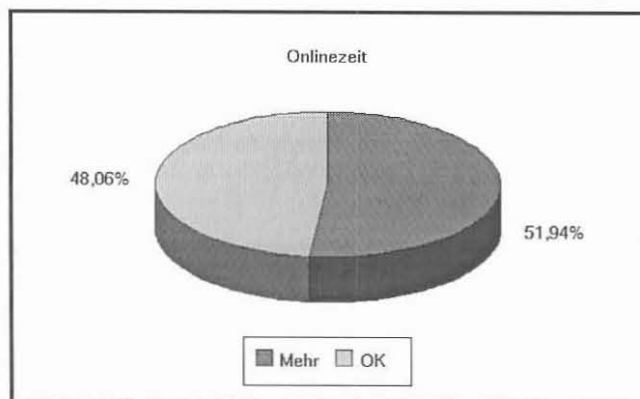
Auslastung

Dieses Ergebnis ist verwunderlich, nachdem die Auslastungskurve der Mailbox zeigt, daß vor allem am Nachmittag die Box so gut wie immer besetzt ist:



Ich nehme daher an, daß diese Aussage im Vergleich zu anderen Mailboxen zu sehen ist, die nur eine Telefonleitung haben und daher praktisch immer besetzt sind.

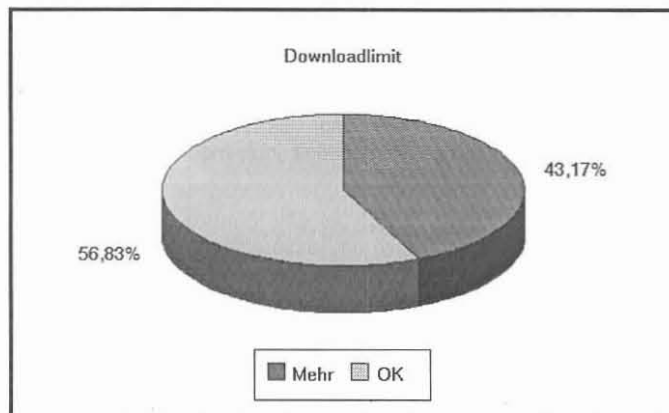
Ungefähr die Hälfte der Benutzer hätten gerne mehr Onlinezeit. Durch die Erhöhung der Onlinezeit wäre aber die Box öfter besetzt als jetzt.



Auch unter diesem Aspekt wären mehr Telefonleitungen durchaus zu rechtfertigen.

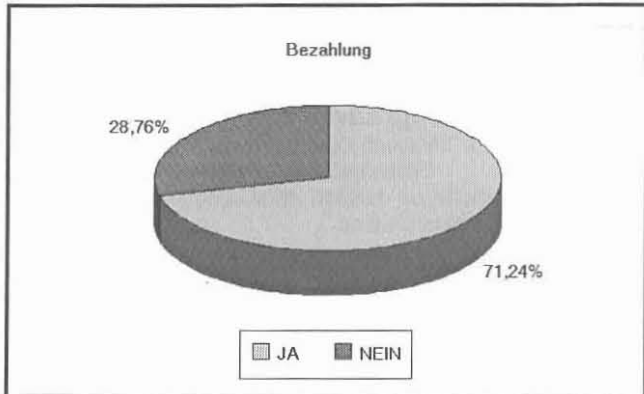
Downloadlimits

Die Mehrzahl der Benutzer ist mit den derzeitigen Downloadlimits zufrieden.

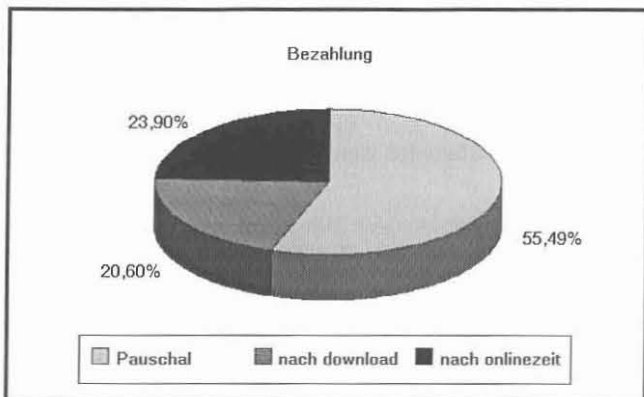


Bezahlung

Die Mehrzahl der Benutzer (ca. 3/4) ist bereit, für die Dienste der Mailbox etwas zu bezahlen.

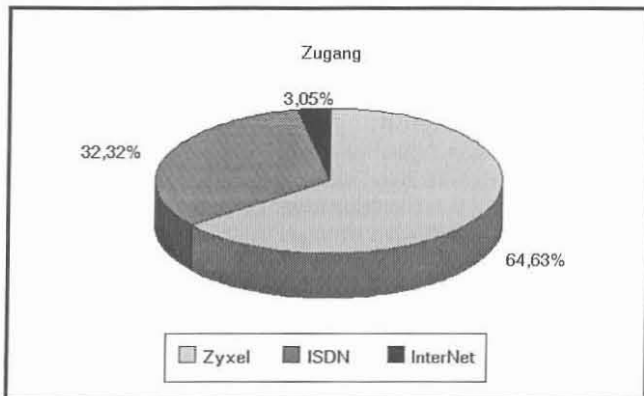


55 % der Befragten würden eine pauschale Bezahlung am gerechtesten finden, nur ca. je 20% würden lieber nach Download, bzw. nach Onlinezeit bezahlen (leistungsorientierte Bezahlung).



Zugänge

Ungefähr 70% der Benutzer, die sich andere Zugänge zur Mailbox wünschen, hätten gerne einen Zugang über ein Zyxel Modem, da dies 19.200 Bps Übertragungsgeschwindigkeit bietet. 32 % würden einen ISDN-Anschluß begrüßen und 3 % hätten gerne, daß die Mailbox über InterNet (FTP) erreichbar wäre.

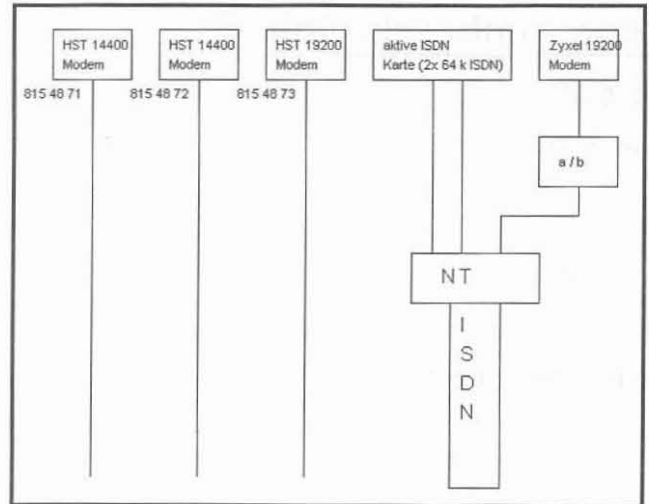


Da ein Zugang zum InterNet derzeit noch mit sehr hohen Kosten verbunden ist, ist wohl dieser Zugang ein Fernziel. Die fast 70-prozentige Forderung nach einem Zyxel Zugang ist nicht zu übersehen. Der große Zuspruch zu ISDN kann wohl derzeit nicht durch die große Verbreitung von ISDN zustande gekommen sein, sondern zeigt eher das Interesse der Benutzer, in Zukunft so einen Anschluß zu bekommen.

Ein ISDN Anschluß würde vor allem für die Telefonrechnung der Mailbox entscheidende Vorteile bringen, da - wenn alle anderen Mailboxen ebenfalls über ISDN erreichbar wären -, bei gleichem Mailaufkommen die Telefonrechnung um drei Viertel sinken müßte.

Planung der Zugänge für 1994

Daher sieht die Planung für die Erweiterungen 1994 wie folgt aus:



Zusätzlich zu den drei Telefonleitungen wird eine ISDN-Leitung angeschafft (bereits angemeldet), an der eine ISDN Karte angeschlossen wird, die auf zwei ISDN-Kanälen je 64kBit zur Verfügung stellt. Zusätzlich wird ein a/b-Adapter an das ISDN angeschaltet, der es möglich macht, konventionelle Telefonendgeräte am ISDN zu betreiben. An diesem Anschluß wird ein Zyxel 19k2 (Zyxel Modus) Modem angeschlossen, um den Wünschen der Benutzer nachzukommen.

Außerdem wird das derzeit auf Leitung drei (815 48 73) befindliche HST-Modem auf 21k6 (V32terbo) upgegradet, um bei Anrufen zu Mailboxen, die keinen ISDN-Anschluß besitzen, Kosten zu sparen.

Point Installation

Die Point Installationsdiskette wurde weiter ausgefeilt. Es ist jetzt je eine Diskette für

- OS/2 PCN-DSK-393
- DOS PCN-DSK-392

verfügbar. Die Installation wurde wesentlich vereinfacht, da man nicht mehr in den einzelnen Konfigurationsdateien alle Adressen ändern muß, sondern nur noch eine Installationsprozedur aufrufen muß. Die Installationsprozedur der DOS Version ist "Command Line" orientiert, die der OS/2 Version ist menügeführt.

PCAD Bereich

Die PCAD-Files von Hr. Ing. Zelinka sind von so großem Interesse, daß sie bereits nach Vorarlberg weitergeleitet werden. Hr. Franz Ehgartner (Fido 2:318/1.0) war gerne bereit, die Dateien in Vorarlberg für HTL-Lehrer verfügbar zu machen.

PCCTGM.AUS Echomail

Das PCCTGM.AUS Area ist nun ebenfalls in Vorarlberg verfügbar. Falls Sie Interesse daran haben, diese Echomail-Area können Sie von

- 2:310/1 His Master's Voice BBS (Wien)
(0222)-815-48-71
- 2:316/3 The DTP BBS (Graz)
(0316)-461-88-66
- 2:318/1 Silvretta BBS (GAIS)
(05525)-31-36-83

beziehen. Falls Sie in einem anderen Bundesland wohnen, richten Sie Anfragen bitte an die Mailbox **His Master's Voice**. □

Werner Illsinger

Microsoft® Select für Unterricht, Forschung & Lehre

Programmbeschreibung

Microsoft

Grundlagen

Zwischen dem Bundesministerium für Unterricht und Kunst und Microsoft gibt es eine Vereinbarung für Select für Unterricht, Forschung & Lehre (folgend Select für F & L) für 2 Jahre gültig ab 15. Oktober 1993. Alle Schulen mit Öffentlichkeitsrecht (Bundesschulen, Schulen anderer Gebietskörperschaften, Privatschulen) sowie deren Lehrer und Institutionen der Lehrerbildung und Lehrerweiterbildung sind berechtigt, an Microsoft Select für F & L teilzunehmen.

Programmelemente

Das *Microsoft F & L Select* Programm stellt folgende Möglichkeiten zum Bezug und zur Wartung der Software zur Verfügung: günstige Preisgestaltung, variable Lizenzierung, Software-Update-Programm.

1. Kopierlizenzen (Microsoft Variable License Paks for Education (MVLPEs))

Bei dem MVLPE handelt es sich um einen Lizenzvertrag, der es den Teilnehmereberechtigten ermöglicht, innerhalb der Vertragsdauer von 2 Jahren eine variable Anzahl von Kopierlizenzen zu erwerben. Das MVLPE wird ausschließlich von einem F & L Handelspartner abgewickelt.

2. Softwaredokumentationen und Disketten erhältlich über Select F & L Handelspartner.

Kunden, die Lizenzen von Microsoft im Rahmen von Select für F & L erwerben, können die Handbücher und Diskettensätze gesondert über den Select F & L Handelspartner beziehen. Preisauskunft bekommen Sie bei einem F & L Handelspartner. Handbücher und Diskettensätze dürfen nur bei dem F & L Handelspartner gekauft werden, bei welchem auch die Lizenzen erworben wurden.

Pro gekaufter Lizenz darf maximal 1 Handbuch und 1 Diskettensatz erworben werden.

Achtung:

Handbücher und Diskettensätze können längere Lieferzeiten haben. Bitte um rechtzeitige Disposition.

3. Software-Update-Programm (Microsoft Variable License Pak Maintenance for Education (MVLPE Maintenance))

Eine weitere Komponente des *Microsoft F & L Select* Programms ist das MVLPE Software-Update-Programm (Wartung) für 2 Jahre. Das Software-Update-Programm ist optionell zu MVLPE.

- Die Gebühr in der Preisliste bezieht sich auf 1 Jahr/Produkt
- Update von bereits bestehenden Microsoft Lizenzen: müssen bei Vertragsbeginn bekanntgegeben werden
- Update von neugekauften Lizenzen: es muß beim Kauf bekanntgegeben werden, ob diese in das Software-Update-Programm aufgenommen werden sollen.
- Jahresgebühr wird geviertelt und quartalsmäßig fakturiert
- Software-Update-Programm kann nur jährlich gekündigt werden und während des Jahres nicht stückmäßig reduziert werden

Der Kunde erhält **einen** kostenlosen Diskettensatz. Mit diesem ist er berechtigt, die Software sooft zu vervielfältigen, wie er Produkte im Software-Update-Programm hat.

Auch das MVLPE Software-Update-Programm ist **ausschließlich** über Select F & L Handelspartner zu beziehen.

Microsoft Office Upgrade for Education

Das neue Upgrade für Microsoft Office ermöglicht es dem Kunden, der eine Microsoft Textverarbeitungssoftware oder eine Tabellenkalkulationssoftware besitzt, diese bequem und einfach auf Microsoft Office zu standardisieren.

4. Select-F & L Handelspartner

Ein Select F & L Handelspartner ist ein durch Microsoft autorisierter Händler, der bestimmte Anforderungskriterien erfüllen muß, um an dem Select Programm teilnehmen zu können. Diese Kriterien werden von Microsoft festgelegt. Microsoft Select für F & L wird ausschließlich über diese Händler erhältlich sein.

Voraussetzungen

- **Schulen:**
Es muß pro Schule mindestens **1 Schul-Vollprodukt** oder mindestens **1 Diskettensatz und 1 Handbuch** vorhanden sein, um im Rahmen von Select Lizenzen bestellen zu können. Wenn nicht vorhanden, muß dieses bei einem Select F & L Handelspartner bezogen werden.
- **Lehrer:**
Wenn ein Lehrer im Zuge von Select Lizenzen kaufen möchte, muß die Schule an der er unterrichtet Lizenzen von diesem Produkt besitzen. Sollte es sich um ein neues Produkt handeln, welches die Schule noch nicht besitzt und der Lehrer evaluieren möchte, ist er berechtigt, eine Lizenz im Rahmen von Select zu beziehen. In diesem Fall muß er allerdings Handbuch und Diskettensatz mitbeziehen. Lehrer sind berechtigt, pro Microsoft-Produkt maximal je 1 Handbuch, 1 Diskettensatz und 1 Lizenz zu erwerben.

Abwicklung für Teilnehmereberechtigte

Die Teilnehmereberechtigten bestellen mit **festgelegtem Bestellformular** bei einem Select F & L Handelspartner. **Der gelbe Durchschlag der Bestellung muß von der Schule bzw. dem Lehrer direkt an Microsoft geschickt werden.**

Daraufhin wird vom F & L Handelspartner eine Rechnung ausgestellt, welche auch den Lizenznachweis darstellt.

Produkte, die im Rahmen von Select getrennt geliefert werden:

Kopierlizenz: Eine Kopierlizenz berechtigt, die Software auf einen PC zu installieren. Es wird kein Produkt geliefert! Kopierlizenzen werden über die Select F & L Handelspartner gekauft.

Handbücher: Handbücher können zusätzlich bei den Select F & L Handelspartnern gekauft werden

Diskettensatz: Diskettensätze können zusätzlich bei den Select F & L Handelspartnern gekauft werden. Diese Diskettensätze beinhalten keine Kopierlizenz.

Bei Unklarheiten wenden Sie sich bitten an einen Select F & L Handelspartner. (Liste Umseitig) □



PABLITOS®
SOFTWARE

A-8063 EGGERSDORF 367, TEL. 03117/3251

Die Fa. PABLITOS ist durch Eigenentwicklungen - professionelle Literaturverwaltung für den wissenschaftl. Anwender, praktisches Bibliotheks-System - vor allem im Bereich Wissenschaft und Forschung bekannt geworden.

in den letzten Jahren hat sich das Software-Angebot ständig erweitert, neben Standard-Produkten bietet die PABLITOS Software GmbH zahlreiche spezielle Produkte an, wie z.B. Software für Statistik, Mathematik, Chemie, Physik, Datenanalyse, Meßwerterfassung, Textverarbeitung mit fremdsprachigen Zeichensätzen, Konvertierungsprogramme und andere spezielle Utilities, Netzwerkprodukte, u.v.m..

Seit Herbst 1993 ist die Fa. PABLITOS als MS-Select Partner auch Select-Betreuer zahlreicher Schulen im Raum Steiermark, Kärnten und Burgenland.

Alle Kunden und Interessenten werden 3x jährlich in einer Aussendung über spezielle Produkte und aktuelle Preise informiert (siehe auch Inserat auf Seite 21). Dieses Informationsmaterial, Informationen zu MS Select oder zu speziellen Software-Produkten kann jederzeit bei PABLITOS Software angefordert werden. □

Achtung HTL-Newcomer: Neues Berufsausbildungsgesetz

Fußnoten (zu Beitrag auf Seite 16)

- 1) In der Folge BAG.
- 2) § 33 Abs. 1 BAG.
- 3) § 13 Abs. 5 BAG.
- 4) § 28 Abs. 3 BAG.
- 5) Siehe dazu auch Abschnitt 2.4.
- 6) § 33 Abs. 1 a BAG.
- 7) Siehe dazu die Abschnitte 2.3 und 2.4.
- 8) § 23 Abs. 1 lit. c BAG.
- 9) Siehe dazu Abschnitt 2.3.
- 10) § 23 Abs. 3 lit. a BAG. Es handelt sich dabei um die "alten" Verordnungen des Bundesministers für wirtschaftliche Angelegenheiten zum § 28 BAG über den Ersatz der Lehrzeiten (und Lehrabschlußprüfungen).
- 11) § 23 Abs. 8 BAG.
- 12) § 34 a BAG.
- 13) § 35 Abs. 1 BAG.
- 14) § 35 Abs. 1 Z. 2 BAG.
- 15) In der Folge GewO.
- 16) § 23 GewO.
- 17) § 18 Abs. 1 Z. 4 GewO sowie § 8 Abs. 2 Unternehmerprüfungsordnung (BGBl.Nr. 453/1993 vom 9. Juli 1993).
- 18) Verordnung des Bundesministers für wirtschaftliche Angelegenheiten über die Durchführung der Unternehmerprüfung (Unternehmerprüfungsordnung), BGBl.Nr. 453/1993. In der Folge UPO.
- 19) § 3 Abs. 1 UPO.
- 20) Ebda.
- 21) § 3 Abs. 3 und 4 UPO
- 22) Lt. Auskunft des Bundesministeriums für wirtschaftliche Angelegenheiten wird durch eine Verwaltungsanweisung des Bundesministers für wirtschaftliche Angelegenheiten an die Gewerbebehörden geregelt werden, welche Schulen von Z. 3 erfaßt sind. Es ist beabsichtigt, u.a. folgende Schulformen aufzunehmen:
 - Höhere Lehranstalt für elektronische Datenverarbeitung und Organisation (einschließlich Sonderformen).
 - Höhere Lehranstalt für Fremdenverkehrsberufe (einschließlich Sonderformen)
 - Höhere Lehranstalt textilkaufmännischer Richtung.
 - Höhere Lehranstalt für Textilwirtschaft.
- 23) Siehe dazu auch Abschnitt 2.3.
- 24) Der Ersatz gilt für diese Schulformen gem. § 8 Abs. 3 nur bis zum Schuljahr 1993/94. Absolventen, die im Schuljahr 1994/95 oder später mit der Schulausbildung beginnen, wird die Unternehmerprüfung nur bei erfolgreich bestandener Abschlußprüfung ersetzt werden. Durch Verordnungen gemäß § 28 BAG über den Ersatz von Lehrabschlußprüfungen (und Lehrzeiten) aufgrund schulischer Ausbildung.
- 25)

Microsoft® Select

F & L Handelspartner:

(Auskunft Microsoft-Info-Service 0660/6520).

COM-CAT HandelsgmbH

Griesauweg 35a, 6020 Innsbruck
TEL: 0512/492 400 FAX: 0512/492 400 20

Comp Delphin

Berggasse 5, 1090 Wien
TEL: 0222/310 53 56/13 FAX: 0222/319 70 50

Computer Center Lorentschitsch

Sperlingweg 23, 5023 Salzburg
TEL: 0662/66 05 05-0 FAX: 0662/66 05 05-21

COMPUTER COMPANY

Grillparzerstr. 1, 4020 Linz
TEL: 07 32/667373-0 FAX: 07 32/66 73 78

Edlinger GesmbH

Vollrathgasse 11, 3130 Herzogenburg
TEL: 027 82/34 93-0 FAX: 027 82/34 93 80

ELBE-Elektronische Büroeinrichtung

Villacher Straße 29, 9800 Spittal
TEL: 047 62/28 25 FAX: 047 62/28 25-17

Hapra GesmbH

Gewerbepark Urfahr 6, 4040 Linz
TEL: 0732/700 425 FAX: 0732/700 425-50

MES

Görglstraße 17, 3500 Krems
TEL: 027 32/74 193 FAX: 027 32/74 193

MIDAS Mikro Datensysteme GmbH

Frankenbergg. 12, 1040 Wien
TEL: 0222/505 01 75-0 FAX: 0222/505 94 71

MIDAS Mikro Datensysteme GmbH.

Elisabethstr.35, 8010 Graz
TEL: 03 16/376 32 FAX: 03 16/384 634

Pablitos

Eggersdorf 367, 8063 Eggersdorf bei Graz
TEL: 031 17/32 51 FAX: 031 17/32 51-90

SPP HandelsGmbH

Lainzerstraße 41, 1130 Wien
TEL: 0222/877 26 87 FAX: 0222/877 26 80

Ulbel & Freidorfer GmbH & Co KG

Reichsstraße 66, 8045 Graz
TEL: 0316/69 29 11-0 FAX: 0316/69 29 11-77

- 26) § 18 GewO
- 27) § 18 Abs. 1 Z. 4 GewO.
- 28) § 18 Abs. 1 Z. 5 GewO.
- 29) § 18 Abs. 3 Z. 2 GewO.
- 30) § 94 GewO. Bei den "Gruppen von Gewerben" handelt es sich ebenfalls um Handwerk!
- 31) § 19 Abs. 2 GewO.
- 32) § 19 Abs. 3 GewO.
- 33) § 21 Abs. 1 und 2 GewO.
- 34) § 126a Abs. 1 GewO
- 35) §§ 126 und 128 GewO.
- 36) § 22 GewO.
- 37) Gemäß § 22 Abs. 3 GewO.
- 38) Zur Frage des Ersatzes der Unternehmerprüfung siehe Abschnitt 2.2.

Software: Registrierungskarte einsenden oder nicht?

Walter J. Jaburek, Diebold

§ In vielen Diskussionen um den Softwareschutz nach dem neuen Urheberrecht taucht auch die Frage ab, ob Lizenzkarten (Registrierkarten) nach dem Kauf von Standardsoftware vor allem im PC-Bereich ausgefüllt und abgesendet werden müssen, und welche Folgen dies hat.

Das Urheberrechtsgesetz kennt keine Registrier- oder Lizenzkarten. Urheberrechtlich ist es also nicht notwendig, Registrierkarten einzusenden.

Vertragsrechtlich versuchen die Softwarelieferanten durch Aufdrucke auf der Packung, dem Anwender einzureden, durch Öffnen der Packung entstehe eine "Lizenzvertrag", der sie verpflichte, auch die Registrierkarte auszufüllen und einzusenden. Dies ist eine Irreführung des Anwenders. Denn der erwähnte Vertrag kommt erst durch die

Unterschrift auf die Registrierkarte zustande. Denn auf dieser befindet sich meist der Satz: "Durch meine Unterschrift erkenne ich das XXX-Lizenzabkommen an.". Jetzt haben wir auch gerade gesehen, wozu die Registrierkarte dienen soll: Der Anwender soll vertragliche Beschränkungen der Nutzung der Software durch seine Unterschrift rechtlich verbindlich anerkennen!

Daher: Registrierkarten am besten nicht einmal ignorieren.

Ausnahme sind dabei einige wenige Hersteller von Netzwerkprodukten und Softwareentwicklungssystemen, die zum Beispiel erst nach Erhalt einer Registrierkarte die fehlerfreie Version einiger Softwaremodule losschicken und die registrierten Benutzern auch eine bestimmte Zeit lang (meist ein Jahr) auf Anfrage auch neue Versionen zusenden und eine wirkliche Telefonauskunft bieten. □

Achtung HTL-Newcomer: Neues Berufsausbildungsgesetz

K. Schlick, Abt. II/7, BMUK

Fußnoten siehe Seite 15

§ Novellierung des Berufsausbildungsgesetzes und der Gewerbeordnung sowie die Annexmaterien Unternehmerprüfung und Ausbilderprüfung - Eine Erläuterung der neuen Rechtslage

1. Berufsausbildungsgesetz (BGBl.Nr. 23/93 vom 14. 1. 1993)

Für die meisten Schüler, die sich derzeit in der Ausbildung befinden, ergeben sich durch die am 1. Juli 1993 in Kraft getretene Novelle keine Änderungen hinsichtlich des Ersatzes der Lehrzeiten bzw. der Lehrabschlußprüfung. Erst für im Schuljahr 1993/94 in berufsbildende mittlere und höhere Schulen eintretende Schüler ist ein Ersatz der Lehrabschlußprüfung im neuen Berufsausbildungsgesetz¹ nicht mehr vorgesehen. Der Zugang zum Beruf wird in Zukunft direkt durch die Gewerbeordnung bzw. die darauf basierenden Verordnungen geregelt.

1.1 Anrechnung des Schulbesuchs auf die Lehrzeit (Ersatz von Lehrzeiten)

Alle bisherigen Verordnungen über den Ersatz der Lehrzeit aufgrund schulmäßiger Ausbildung bleiben solange als Bundesgesetz aufrecht, bis sie durch eine neue Verordnung des Bundesministers für wirtschaftliche Angelegenheiten ersetzt werden.² Der Lehrzeitersatz kann allerdings im Einzelfall und bei Vorliegen sachlich gerechtfertigter Gründe durch eine individuelle Vereinbarung zwischen dem Lehrberechtigten und dem Lehrling um maximal ein Jahr verringert werden.³ Dafür ist ein innerhalb von vier Wochen zu erstellendes Gutachten des Landes-Berufsausbildungsbeirates erforderlich.

Eine individuelle Anrechnung von Schulzeiten auf die Lehrzeit wird dann vorgenommen, wenn etwa ein Schulabbrecher nach einem Lehrplan unterrichtet wurde, der nicht von einer Verordnung über den Lehrzeitersatz erfaßt ist. Konkret sind bei höchstens dreijähriger Lehrzeit bis zu 1½ Jahre, bei mehr als dreijähriger Lehrzeit bis zu zwei Jahre auf die Lehrzeit anzurechnen. Diese Anrechnung ist in Verbindung mit der Anmeldung des Lehrvertrages zu beantragen und bedarf eines binnen vier Wochen zu erstattenden Gutachtens des Landes-Berufsausbildungsbeirates.⁴

Da die Lehrpläne der berufsbildenden mittleren und höheren Schulen laufend novelliert werden, ist damit zu rechnen, daß sich der Anwendungsbereich der "alten" Verordnungen zum § 28 BAG hinsichtlich des Ersatzes der Lehrzeiten in nur wenigen Jahren stark ausdünnen wird. Gleichzeitig ist noch nicht geklärt, ob das Bundesministerium für wirtschaftliche Angelegenheiten die gem. § 33 Abs. 2 BAG vorgesehenen "neuen" Verordnungen über den Ersatz von Lehrzeiten erlassen wird, da nach Ansicht des Bundesministeriums für wirtschaftliche Angelegenheiten eine einzige neue Verordnung alle derzeit bestehenden "alten" Verordnungen (Lehrzeitersatz und Ersatz der Lehrabschlußprüfungen) außer Kraft setzen würde. Dies würde eine Neuverhandlung aller bestehenden Lehrpläne hinsichtlich der Ersätze

von Lehrzeiten durch Schulbesuch sowie eine sofortige Novellierung der Befähigungsnachweisverordnungen für all jene gebundenen Gewerbe erforderlich machen, für die unsere Schüler ausgebildet werden.⁵

1.2 Der Zugang zum Gewerbe (Ersatz von Lehrabschlußprüfungen)

Die derzeit bestehenden Verordnungen des Bundesministers für wirtschaftliche Angelegenheiten über den Ersatz von Lehrabschlußprüfungen aufgrund schulischer Ausbildungen bleiben für die Schüler aufrecht, die spätestens im Schuljahr 1992/93 den Schulbesuch an einer berufsbildenden mittleren oder höheren Schule aufgenommen haben und nach einem von diesen Verordnungen erfaßten Lehrplan unterrichtet werden.⁶

Über umgekehrt formuliert: Die neue Rechtslage wird für die Absolventen von berufsbildenden mittleren und höheren Schulen Anwendung finden, die

- a) im Schuljahr 1993/94 erstmals eine solche Schule besuchen oder
- b) nach einem Lehrplan unterrichtet werden, zu dem keine Verordnung gem. § 28 BAG erlassen wurde.⁷

Der Zugang zum Gewerbe wird künftig direkt auf Grundlage der Gewerbeordnung erfolgen.

1.3 Antreten zu Lehrabschlußprüfung

Das bereits im alten BAG⁸ vorgesehene Antreten zur Lehrabschlußprüfung ist eher als theoretische Möglichkeit zu betrachten, die für den einzelnen die Betätigungsmöglichkeiten, die sich durch die neue Gewerbeordnung ergeben (Überstiegsmöglichkeiten zu verwandten Handwerken sowie innerhalb der Gruppen von Gewerben⁹), zusätzlich erweitert. Dem Antrag ist als Nachweis das Zeugnis einer Schule anzuschließen, deren Besuch die Lehrzeit ersetzt.¹⁰ In solchen Fällen entfällt bei der Lehrabschlußprüfung der theoretische Teil der Prüfung.¹¹

1.4 Zulassung zur Ausbilderprüfung

Die Bestimmung des § 29c Abs. 1 Z. 1 BAG, wonach zur Ausbilderprüfung zuzulassen ist, "wer durch Zeugnisse ... die erfolgreiche Ablegung der Lehrabschlußprüfung, den Ersatz der Lehrabschlußprüfung gemäß § 8 Abs. 7 oder den erfolgreichen Besuch einer Schule, durch den die Lehrabschlußprüfung auf Grund von Vorschriften gemäß § 28 ersetzt wird, und eine ... zweijährige berufliche Praxis ..." nachweist, gilt noch für all jene Absolventen, die spätestens im Schuljahr 1992/93 mit dem Besuch einer berufsbildenden mittleren oder höheren Schule begonnen haben und für deren Lehrpläne Verordnungen des Bundesministers für wirtschaftliche Angelegenheiten über den Ersatz von Lehrabschlußprüfungen existieren.

Den zukünftigen Absolventen, die ab dem Schuljahr 1993/94 in eine berufsbildende mittlere oder höhere Schule eintreten, wird hingegen aufgrund der Novellierung des Berufsausbildungsgesetzes nicht mehr die Lehrabschlussprüfung ersetzt. In Zukunft wird daher die Bestimmung des § 29c Abs. 1 Z. 1 BAG nicht mehr zur Anwendung gebracht werden können, da ab dem Ende des Schuljahres 1997/98 generell, in zahlreichen Fällen (mangels Verordnung über den Ersatz von Lehrabschlussprüfungen) bereits früher kein Ersatz von Lehrabschlussprüfungen aufgrund Schulbesuchs stattfinden wird.

Was bedeutet dies nun für die Zulassung zur Ausbilderprüfung?

§ 34a BAG bestimmt u.a., daß "für den Bereich der beruflichen Qualifikationen ... das Prüfungszeugnis, mit dem der erfolgreiche Abschluß einer mindestens dreijährigen berufsbildenden mittleren Schule, einer berufsbildenden höheren Schule oder deren Sonderformen einschließlich der Schulversuche nachgewiesen wird, zumindest als Nachweis einer mit einer facheinschlägigen Lehrabschlussprüfung abgeschlossenen beruflichen Ausbildung" gilt. Mit anderen Worten: Das Prüfungszeugnis, mit dem der erfolgreiche Abschluß einer derartigen Schule nachgewiesen wird, ist als äquivalent mit einer Lehrabschlussprüfung in dem Lehrberuf anzusehen, der der schwerpunktmäßigen Ausbildung dieser Schule entspricht.

Im Lichte der neuen Rechtslage ab 1. Juli 1993 ist daher die Bestimmung des § 29c Abs. 1 Z. 1 folgendermaßen zu interpretieren:

Zur Ausbilderprüfung ist zuzulassen, wer durch Zeugnisse nachweist:

1. die erfolgreiche Ablegung der Lehrabschlussprüfung und eine nachfolgende mindestens zweijährige berufliche Praxis,
2. den erfolgreichen Besuch einer Schule, durch den gemäß § 33 Abs. 1a BAG die Lehrabschlussprüfung aufgrund der im § 33 Abs. 1 angeführten und als Bundesgesetz aufrechterhaltenen Vorschriften ersetzt wird ("alte" Verordnungen des Bundesministers für wirtschaftliche Angelegenheiten über den Ersatz von Lehrabschlussprüfungen vor der Gewerberechtsnovelle 1992) und eine nachfolgende mindestens zweijährige berufliche Praxis (gilt zum letzten Mal für die Schüler, die 1992/93 den Besuch einer berufsbildenden mittleren oder höheren Schule begonnen haben) oder
3. die erfolgreiche Abschlußprüfung bzw. Reifeprüfung einer mindestens dreijährigen berufsbildenden mittleren Schule, einer berufsbildenden höheren Schule oder deren Sonderformen einschließlich der Schulversuche, womit gem. § 34a BAG, entsprechend der schwerpunktmäßigen beruflichen Ausbildung der Schule, das Niveau einer facheinschlägigen Lehrabschlussprüfung nachgewiesen wird, und eine nachfolgende mindestens zweijährige berufliche Praxis.

Diese im § 29c vorgeschriebene berufliche Praxiszeit von wenigstens zwei Jahren ist bereits durch den Befähigungsnachweis erbracht.

Aufgrund der Bestimmung des § 34a BAG ergibt sich allerdings Handlungsbedarf insofern, als Absolventen von berufsbildenden mittleren Schulen, deren Besuch derzeit nicht durch eine Abschlußprüfung abgeschlossen wird (z.B. Handelsschulabsolventen ab 1996), keinen Zugang zur Ausbilderprüfung haben werden.

1.5 Arbeitsrechtliche Auswirkungen

Arbeits- und kollektiwertstragsrechtlich gilt der erfolgreiche Abschluß einer berufsbildenden mittleren oder höheren Schule zumindest als Nachweis einer mit einer Lehrabschlussprüfung abgeschlossenen beruflichen Ausbildung.¹²

1.6 Vollziehung des Gesetzes

Für die Vollziehung des BAG ist der Bundesminister für wirtschaftliche Angelegenheiten zuständig.¹³ Hinsichtlich der Erlassung der Verordnungen gem. § 28 (Ersatz von Lehrzeiten) ist das Einvernehmen mit dem Bundesminister für Unterricht und Kunst herzustellen.¹⁴

2. Gewerbeordnung (BGBl.Nr. 29/1993 vom 15. 1. 1993)

Die Gewerberechtsnovelle 1992, ebenfalls am 1. Juli 1993 in Kraft getreten, bietet Absolventen berufsbildender mittlerer und höherer Schulen neue und liberalisierte Zugangsbedingungen zu Gewerben, d.h. zur selbständigen Ausübung von reglementierten Berufen.

2.1 Einteilung der Gewerbe

Die neue Gewerbeordnung¹⁵ unterscheidet Handwerke, gebundene Gewerbe und freie Gewerbe; für letztere ist die Erbringung eines Befähigungsnachweises nicht erforderlich. Die ehemaligen konzessionierten Gewerbe sind großteils in den gebundenen Gewerben aufgegangen, die sich ihrerseits in bewilligungspflichtige und nicht bewilligungspflichtige Gewerbe unterteilen.

Die Unterscheidung zwischen Handwerk und gebundenem Gewerbe gründet sich auf der unterschiedlichen Weise, in der der Befähigungsnachweis erbracht wird. Im Falle der Handwerke ist es die Meisterprüfung gem. § 18 GewO, im Falle der gebundenen Gewerbe der Befähigungsnachweis gem. § 22 GewO.

2.2 Unternehmerprüfung

Für die selbständige Ausübung eines Handwerks oder gebundenen Gewerbes ist neben dem Nachweis der handwerklichen Fertigkeiten bzw. facheinschlägigen Kenntnisse auch der Nachweis der betriebswirtschaftlichen und rechtlichen Kenntnisse in Form einer Unternehmerprüfung zu erbringen.¹⁶ Diese Unternehmerprüfung kann als Prüfungsteil der Meisterprüfung oder als Prüfungsteil des Nachweises der Befähigung für ein gebundenes Gewerbe (aber auch als Einzelprüfung) erbracht werden. Das Antreten zu der Unternehmerprüfung ist an keine Zulassungsvoraussetzungen gebunden.

Allen Absolventen berufsbildender höherer Schulen, die den Befähigungsnachweis für ein einschlägiges Handwerk erbringen, wird die Unternehmerprüfung jedenfalls ersetzt werden.¹⁷ Für die Fälle, in denen die schulische Ausbildung nicht einem Handwerk zugeordnet ist, sondern zu einem gebundenen Gewerbe führt, hat der Bundesminister für wirtschaftliche Angelegenheiten mit Verordnung festgelegt, welchen Absolventen mindestens dreijähriger berufsbildender Schulen die Unternehmerprüfung ersetzt wird.¹⁸

Ziel der Unternehmerprüfung ist es festzustellen, "ob der Kandidat die Zusammenhänge der Bereiche eines Unternehmens versteht und dieses Wissen bei der Gründung eines Unternehmens und bei der Bewältigung der häufigsten Aufgaben und Problemsituationen in einem Unternehmen anwenden kann."¹⁹ Als Themenbereiche sind vorgesehen:

1. Kommunikation und Verhalten innerhalb des Unternehmens und gegenüber nicht dem Unternehmen angehörigen Personen und Institutionen (Lieferanten, Kunden, Kreditinstituten, Behörden ua.),
2. Marketing,
3. Organisation,
4. unternehmerische Rechtskunde,
5. Rechnungswesen,
6. Mitarbeiterführung und Personalmanagement.²⁰

Die Unternehmerprüfung besteht aus einem schriftlichen (Projektarbeit, Verständnisfragen, Fallbeispiele; Dauer: 4 Stunden) und einem mündlichen Teil (Verständnisfragen, dazu mindestens ein Fallbeispiel; Dauer: 20 bis 40 Minuten).²¹

Gem. § 8 Abs. 2 UPO entfällt der Prüfungsteil Unternehmerprüfung für Absolventen folgender Schulformen:

1. Handelsakademien sowie deren Sonderformen,
2. Höhere Lehranstalten für wirtschaftliche Berufe und deren Sonderformen,
3. nicht unter Z. 1, 2 oder 4 fallende berufsbildende höhere Schulen, deren schwerpunktmäßige Ausbildung im betriebswirtschaftlich-kaufmännischen Bereich liegt,²²
4. berufsbildende höhere Schulen, sofern durch das Zeugnis über deren erfolgreichen Besuch sowie eine mindestens dreijährige fachliche Tätigkeit der Befähigungsnachweis für ein Handwerk erbracht wird (die diesbezügliche Verordnung des Bundesministers für wirtschaftliche Angelegenheiten ist derzeit noch in Ausarbeitung und soll im Herbst 1993 in Kraft treten),²³
5. dreijährige Handelsschule,²⁴
6. dreijährige Fachschule für wirtschaftliche Berufe,²⁴
7. Hotelfachschule, Gastgewerbefachschule und Tourismusfachschule,
8. Werkmeisterschule (unter gewissen Voraussetzungen).

§ 3 (Inhalte und Struktur der Unternehmerprüfung) tritt mit 1. August 1994 in Kraft. Alle anderen Bestimmungen der Verordnung sind am 10. Juli 1993 in Kraft getreten.

2.3 Handwerke

Vor der Gewerberechtsnovelle 1992 war für Absolventen berufsbildender mittlerer und höherer Schulen der Zugang zum Befähigungsnachweis in einem einschlägigen Handwerk nur über den Ersatz oder das Bestehen einer einschlägigen Lehrabschlußprüfung²⁵ und die Ablegung der Meisterprüfung (gegebenenfalls unter Entfall bestimmter Prüfungsteile) möglich. Nunmehr wird die schulische berufliche Ausbildung ohne diesen Umweg durch Verordnung jeweils direkt dem einschlägigen Handwerk zugeordnet.²⁶ Diese derzeit (August 1993) noch ausstehende Verordnung des Bundesministers für wirtschaftliche Angelegenheiten ist nicht nur für die Schüler von Bedeutung, die im September 1993 den Besuch einer berufsbildenden Schule aufgenommen haben, sondern auch für all jene Schüler, die nach Lehrplänen unterrichtet werden, zu denen das Bundesministerium für wirtschaftliche Angelegenheiten bisher keine Verordnungen über den Ersatz von Lehrabschlußprüfungen erlassen hat.

Diese Verordnung sollte in der Form erlassen werden, daß nicht jede der (aufgrund der dynamischen Lehrplanentwicklung im Bereich des berufsbildenden-Schulwesens) zahlreichen Lehrplannovellen eine Novellierung der gegenständlichen Verordnung nach sich zieht. D.h., daß nicht eine Zuordnung der Handwerke zu den einzelnen Lehrplänen Ziel sein sollte, sondern eine Zuordnung zu Gruppen von Ausbildungsgängen, die jeweils einen Berufsbereich erfassen. Diese Innovation der Vorgangsweise wurde zu einer beträchtlichen Reduktion des Verwaltungsaufwandes führen.

Welche Auswirkungen hat dies auf den Zugang zum Handwerk?

1. Absolventen berufsbildender höherer Schulen erbringen nach drei Jahren fachlicher Tätigkeit den Befähigungsnachweis (einschließlich des Prüfungsteiles Unternehmerprüfung) für das einschlägige Handwerk. D.h. sie müssen keine Meisterprüfung ablegen.²⁷
2. Absolventen von Werkmeisterschulen wird nach vierjähriger fachlicher Tätigkeit die einschlägige Meisterprüfung teilweise ersetzt. Der Prüfungsteil Unternehmerprüfung ist jedoch abzulegen.²⁸
3. Absolventen mindestens dreijähriger berufsbildender mittlerer Schulen wird nach zweijähriger fachlicher Tätigkeit die Ablegung der Meisterprüfung im betreffenden Handwerk ermöglicht.²⁹

Die Handwerke sind nach "Verwandten Handwerken" und "Gruppen von Gewerben" geordnet.³⁰ Wenn der Befähigungsnachweis für ein Handwerk erbracht wird, ist es durch die Ablegung einer Zusatzprüfung³¹ möglich, den Befähigungsnachweis auch für ein verwandtes Handwerk zu erbringen. Diese Zusatzprüfung (über die für das Handwerk charakteristischen Kenntnisse und Fähigkeiten) gilt als Meisterprüfung. Ebenso ist es möglich, mittels Teil- und Ergänzungsprüfungen oder durch eine fachliche Tätigkeit von zwei Jahren den Befähigungsnachweis für ein Handwerk aus der gleichen Gruppe von Gewerben zu erbringen.³²

Allerdings dürfen sich nur die Personen, die die Meisterprüfung tatsächlich abgelegt haben, als "Meister" bezeichnen bzw. für Gewerbebetriebe bei der Namensführung und bei der Bezeichnung der Betriebsstätte die Worte "Meister", "Meisterbetrieb" oder ähnliche Formulierungen mit Beziehung auf das betreffende Handwerk verwenden.³³

Über die gem. § 94 GewO bestehenden Verwandtschaften zwischen den Handwerken hinaus kann der Bundesminister für wirtschaftliche Angelegenheiten durch Verordnung festlegen, welche Handwerke mit anderen Handwerken und welche Handwerke mit gebundenen Gewerben (§ 126) verwandt sind.³⁴

2.4 Gebundene Gewerbe

Neben einer durch die Abschaffung der konzessionierten Gewerbe bedingten Neuordnung der gebundenen Gewerbe in eine Gruppe nichtbewilligungspflichtiger und in eine Gruppe bewilligungspflichtiger Gewerbe³⁵ hat sich an den Voraussetzungen zur Erbringung des Befähigungsnachweises³⁶ nichts geändert. Wie bisher wird der Befähigungsnachweis u.a. durch eine Lehrabschlußprüfung oder ein Schulzeugnis und - sofern vorgesehen - die Befähigungsnachweisprüfung erbracht.

Eine Änderung ergibt sich allerdings durch das Herausfallen der Lehrabschlußprüfungen aus dem § 28 BAG. Eine Anzahl von Befähigungsnachweisen sehen nämlich als Zugangsvoraussetzung zum Befähigungsnachweis die positive Absolvierung bzw. den gem. § 28 BAG erfolgten Ersatz der Lehrabschlußprüfung, nicht jedoch das Zeugnis einer berufsbildenden Schule vor.

Da es derzeit zahlreiche Lehrpläne gibt, für die das Bundesministerium für wirtschaftliche Angelegenheiten keine Verordnungen über die Ersätze von Lehrabschlußprüfungen aufgrund schulischer Ausbildung erlassen hat, der Berufszugang für die Absolventen dieser Ausbildungsgänge jedoch gesichert werden muß, sind jene Befähigungsnachweisverordnungen³⁷, die als Zugangsvoraussetzung zum Befähigungsnachweis bisher lediglich den Ersatz der Lehrabschlußprüfung vorsehen, in der Form zu ändern, daß zusätzlich die einzelnen Schulformen in die Verordnungen aufgenommen werden.

Als Beispiel sei die erste, diese Notwendigkeit berücksichtigende Befähigungsnachweisverordnung für Inkassoinstitute (BGBl.Nr. 490/1993) angeführt. Dieser Befähigungsnachweis wird durch die erfolgreiche Ablegung einer aus drei Teilen (schriftlich, mündlich, Unternehmerprüfung³⁸) bestehenden Prüfung erbracht.

Gem. § 9 der Befähigungsnachweisverordnung ist zur Prüfung zuzulassen, wer durch Zeugnisse nachweist

1. ...
- 2.a) den erfolgreichen Besuch der Handelsakademie oder einer Sonderform der Handelsakademie und
 - b) eine mindestens zweijährige fachliche Tätigkeit gemäß § 22 Abs. 2 GewO 1973 oder
- 3.a) ... den erfolgreichen Besuch einer Handelsschule oder einer nicht unter Z 2 fallenden berufsbildenden höheren Schule und
 - b) eine mindestens dreijährige fachliche Tätigkeit gemäß § 22 Abs. 2 GewO 1973 oder
- 4.a) den erfolgreichen Besuch einer ... nicht unter Z 3 a fallenden berufsbildenden mittleren Schule und
 - b) eine mindestens vierjährige fachliche Tätigkeit gemäß § 22 Abs. 2 GewO 1973 ..."

Einen Spezialfall im Rahmen der gebundenen Gewerbe stellt der Zugang von HTL-Absolventen zum Handelsgewerbe (und Handelsagenten) dar. Gem. § 160 Abs. 1 GewO wird zusätzlich zum Zeugnis eine ein- oder zweijährige kaufmännische Tätigkeit verlangt werden. Die jeweilige Dauer dieser kaufmännischen Tätigkeit wird von der Intensität der kaufmännischen Ausbildung an den verschiedenen Schulformen abhängen und durch eine Verwaltungsanweisung des Bundesministers für wirtschaftliche Angelegenheiten vorgeschrieben werden. Für die Schulen, die einem Handwerk zugeordnet sind (v.a. Höhere technische Lehranstalten), ist wesentlich, daß gem. § 160 Abs. 2 GewO die Ausübung jedes gebundenen Gewerbes oder Handwerks als kaufmännische Tätigkeit gilt.

3. Gesamtbeurteilung

Insgesamt ist das neue System aus der Sicht des berufsbildenden Schulwesens positiv zu bewerten.

Während die bisherigen Verordnungen über den Ersatz von Lehrabschlußprüfungen aufgrund schulischer Ausbildung erforderlich waren, um den Absolventen den Zugang zu Handwerken zu eröffnen, kommt es nun durch die Novellierung von Berufsausbildungsgesetz und Gewerbeordnung zu einer Entkoppelung der schulischen Berufsausbildung vom dualen System. Der Umweg über den Ersatz der Lehrabschlußprüfungen entfällt; stattdessen wird von Absolventen berufsbildender höherer Schulen der Befähigungsnachweis direkt durch Schulzeugnis und Praxis (Handwerke, einige gebundene Gewerbe) bzw. aufgrund Schulzeugnis und Befähigungsnachweisprüfung (gebundene Gewerbe) erbracht. Für die Absolventen berufsbildender mittlerer Schulen besteht nunmehr die Möglichkeit, nach nur zwei Jahren facheinschlägiger Tätigkeit zur Meisterprüfung antreten zu können.

Der Hauptvorteil der neuen Regelung besteht jedoch darin, daß sich durch die Zugangsmöglichkeit auch zu verwandten Handwerken und innerhalb der Gruppen von Handwerken ein weites Spektrum im Zugang zu reglementierten Berufen eröffnet. Dieser Aspekt gewinnt zusätzliche Bedeutung im Zusammenhang mit der europäischen Integration (Europäischer Wirtschaftsraum, allfälliger EG-Beitritt), weil nationale Befähigungsnachweise aufgrund der EWR-/EG-weiten Anerkennung die Niederlassung in den Mitgliedstaaten sicherstellen. Damit wird dem legitimen Wunsch der Schüler des berufsbildenden Schulwesens nach mehr Flexibilität auch im internationalen Kontext entsprochen. □

MATHCAD

Andreas Zandomeneghi

PCN-DSK-404\MCAD*.MCD



1. Einleitung

Der sehr interessante Artikel von Horst Schwarz in der letzten Ausgabe der **PC-NEWS** hat mich auf die Idee gebracht, das Programm Mathcad, gewissermaßen ein Gegenpart zu "Derive", heute etwas näher vorzustellen. Es ist natürlich im Rahmen eines derartigen Artikels nicht möglich, alle Details zu beleuchten, es liegt mir eher daran, Neulinge, speziell solche, die der Mathematik etwas skeptisch gegenüber stehen, für die Software zu interessieren.

Was Herr Schwarz in seinem Artikel über den Sinn und Zweck von Computer-Algebra Systemen sagt, gilt naturgemäß auch für Mathcad: Bei der Beschäftigung mit der Mathematik wird das unmittelbare Erfolgserlebnis des Schülers gestärkt, man sieht die Ergebnisse unmittelbar und kann Veränderungen an den Resultaten bei neu eingegebenen Eingangsdaten unmittelbar und direkt erleben.

Darüber hinaus eignet sich Mathcad jedoch auch dazu, professionellen Ansprüchen, wie sie z.B. in der Technik gefordert sind, zu genügen. Erfolgen doch Dimensionierung und Berechnung von Bauteilen im Bauingenieur- und Maschinenbauwesen auch heute noch teilweise auf mühsamen und fehlerträchtigen Weg per Hand und Taschenrechner. Speziell wenn es um die Erstellung von professionellen Dokumentationen von Berechnungsergebnissen geht, hat meiner Meinung nach Mathcad die Nase noch vor anderen Paketen wie "Derive" oder "Axiom".

2. Historisches

Das Software-Paket Mathcad wurde von der amerikanischen Firma Mathsoft in Massachusetts (USA) auf den Markt gebracht und existiert bereits seit mehreren Jahren. Die Version 2.5 war (bevor sich Windows in breitem Rahmen durchsetzte) die letzte Version, die direkt unter DOS lief. Schon damals, ich war gerade Studienassistent an der TU-Wien, (1987/88), war mir Mathcad als sehr leistungsfähiges Programm positiv aufgefallen. Die zur Zeit aktuelle Version für Windows trägt die Nummer 4.0, wurde weiter verbessert, ohne die guten Grundprinzipien über Bord zu werfen. Als eventuellen Nachteil kann man allerdings werten, daß speziell die neueste Programm-Version Windows-Ressourcen geradezu gierig an sich reißt und deshalb auch einen schnellen Rechner, viel RAM und etwa 15MB Festplattenplatz benötigt. Die Version 4.0 ist seit einigen Monaten auf dem europäischen Markt erhältlich. Inzwischen sind auch deutsche und andere europäischsprachige Versionen verfügbar.

3. Wie funktioniert die Software?

Mathcad ist im wesentlichen ein Gleichungslöser. Diese Gleichungen erscheinen im Mathcad-Editor (sowie auf dem Ausdruck) in gleicher Form, wie beim Lösen von Gleichungen auf einem Blatt Papier. Die "Bausteine", die ein Mathcad-File, genannt "Dokument", aufbauen, heißen "Regionen". Es gibt drei Typen solcher Regionen: Gleichungen, Grafiken und Text. Diese können mit der Maus problemlos verschoben und nebeneinander oder nacheinander angeordnet werden.

Wird mit der Maus kein anderer Button in der linken Befehlsspalte angeklickt, ist Mathcad defaultmäßig im Gleichungsmodus. Mathcad enthält eine sehr große Zahl von bereits definierten Funktionen, dazu gehören (ohne Anspruch auf Vollständigkeit) die vier Grundrechnungsarten, Quadrat, alle Arten von Wurzeln, trigonometrische Funktionen, Matrix-Operationen, logarithmische und e-Funktionen, Integrale/ Differentiale sowie transzendente Funktionen.

Das Laden und Speichern von Dokumenten erfolgt wie bei Programmen, die unter Windows laufen, üblich ist.

Die wichtigsten Operatoren werden in Mathcad unter den folgenden Tasten (oder durch Buttons auf der linken Spalte) erreicht:

Eingabe am Keyboard	es erscheint	Bedeutung
:	:=	Zuweisung
=	=	Berechnung
.	.	Multiplikation
^	hochgestellt	Quadrat
~	≡	Identität
;	..	Intervall
@	Grafik	

Um das Programm kennenzulernen, ist am besten, mit einem einfachen Beispiel zu beginnen, Eingabewerte zu verändern und "online" zu beobachten, wie sich die Ergebnisse ändern. Es ist wichtig, sich vor Augen zu führen, daß die Berechnung der Gleichungen stets von oben nach unten, und wenn zwei Gleichungen in der selben Zeile stehen, von links nach rechts erfolgt. Bei Irrtümern, Fehlern etc., erscheinen sofort entsprechende Fehlermeldungen direkt im Dokument.

Eine weitere leistungsfähige Eigenschaft von Mathcad ist, daß Bitmaps sowohl per Clipboard als per ".BMP" Datei in ein Dokument eingebunden werden können. Somit können Skizzen, Pläne etc. die Übersichtlichkeit und "Lesbarkeit" einer Berechnung erhöhen. (Ich erstelle die Bitmaps meist mit dem "Designer".)

Auch das Austauschen von Daten, z.B. mit einem Spreadsheet-Programm, ist kein Problem. Es existieren die Funktionen READPRN und WRITEPRN sowie einige weitere, die Daten in ASCII-Form in Dateien schreiben und auch wieder lesen können. Auch diese "Aktionen" erfolgen im Ablauf der Mathcad-Berechnung, als von oben nach unten im Dokument.

Erwähnenswert ist noch, daß Mathcad wesentliche Teile des auch als Einzelprogramm erhältlichen symbolischen Solvers "Maple" enthält. Darauf gehe ich jedoch aus Platzgründen in diesem Artikel nicht weiter ein.

4. Einheitendefinition

Eine besondere Stärke von Mathcad liegt in der Möglichkeit, physikalische, d.h. einheitenbehaftete Ausdrücke, auf ihre dimensionsgerechte Schreibweise online zu überprüfen. Dazu ist es nur erforderlich, am Beginn des Dokuments Ausgangsgrößen zu definieren. Für unsere Beispiele benötigen wir Länge, Zeit und Masse. Wir schreiben deshalb die "Gleichungen"

$$m \equiv 1L, \quad sec \equiv 1T, \quad kg \equiv 1M$$

(Das Zeichen "≡" bedeutet "Identität" und ist als einzige Mathcad-Gleichung im gesamten Dokument definiert, also schon am Beginn bekannt, auch wenn sie erst als letzte Gleichung im Dokument stünde.). Danach müssen wir nur noch die aus diesen drei Basisgrößen abgeleitete Größen definieren. z.B.

$$ms := \frac{1}{1000} \cdot sec \quad \text{Millisekunde} \quad \text{oder}$$

$$N := kg \cdot \frac{m}{sec^2} \quad \text{Newton} \quad \text{usw.}$$

Da die meisten Größen immer wieder benötigt werden, empfiehlt es sich, ein "Prototypendokument" mit Grund- und abgeleiteten Einheiten zu erstellen, das als Ausgangsdokument für jede neue Mathcad-Berechnung dienen kann.

Als Resultat einer einheitenbehafteten Berechnung entsteht als Einheit des Berechnungsergebnisses eine, auf den ersten Blick etwas verwirrend aussehende Kombination der Grundgrößen. Es erscheint jedoch auch ein Platzhalter, in dem wir schließlich die gewünschte, abgeleitete Größe (z.B. "cm" oder "N") manuell eingeben können. Die

Zahlenwerte werden sofort dimensionsrichtig umgerechnet! (Das Prinzip werden wir in 5.3 nochmals sehen.)

5. Beispiele (auf Diskette beim Club erhältlich)

Es soll nun das Prinzip, wie wir bei Berechnungsbeispielen mit Mathcad rationell zu einem Ergebnis kommen, anhand einiger Beispiele verständlich gemacht werden:

5.1 Parabelberechnung (PARABEL.MCD)

Als "klassisches" Schulbeispiel wollen wir uns die Parabel

$$y(x) = x^2 + 2x - 6$$

etwas näher ansehen. Sie ist im Intervall [-5,5] darzustellen. Es soll eine "Kurvendiskussion" ausgeführt werden. Dazu wollen wir Funktionswerte der Funktion berechnen, den Graph im Bild darstellen, sowie den Extremwert (Scheitel) berechnen. Außerdem soll die Ableitungsfunktion in einem zweiten Bild zusammen mit der Ausgangsfunktion dargestellt werden.

Die Darstellung dieses Intervalls in Mathcad lautet:

$x := -5, -4.9 \dots 5$. Mehr ist nicht notwendig. Mathcad "weiß", daß x zwischen -5 und +5 liegt, und daß uns für die Rechnung nur Werte in Schritten von 0.1 interessieren.

Mit der Eingabe von $y(x)=$ erscheint eine komplette Liste von x -Werten mit den zugehörigen Funktionswerten.

Als nächstes wird der Funktionsgraph (Plot) generiert. Beim Anklicken des entsprechenden Symbols oder der Eingabe von "@" erscheint ein leeres Rechteck, an dessen Rändern vier Platzhalter zu sehen sind. Zumindest die Funktionen der beiden Achsen müssen dort eingetragen werden, um die Fehlermeldung zum Verschwinden zu bringen, die gleich bei der Erstellung des leeren Graphs erscheint.

Auf der x -Achse (Abszisse) tragen wir als "Funktion" x ein, auf der Ordinate $y(x)$ (wir könnten auch die komplette rechte Seite der Funktionsgleichung dort hineinschreiben.). Die beiden Randwerte auf der x -Achse sind Mathcad wegen der Intervallgrenzen bereits bekannt. Auch die Grenzen auf der Ordinate berechnet das Programm selbständig, das Intervall der Funktionswerte ist natürlich $[f(-5), f(5)]$.

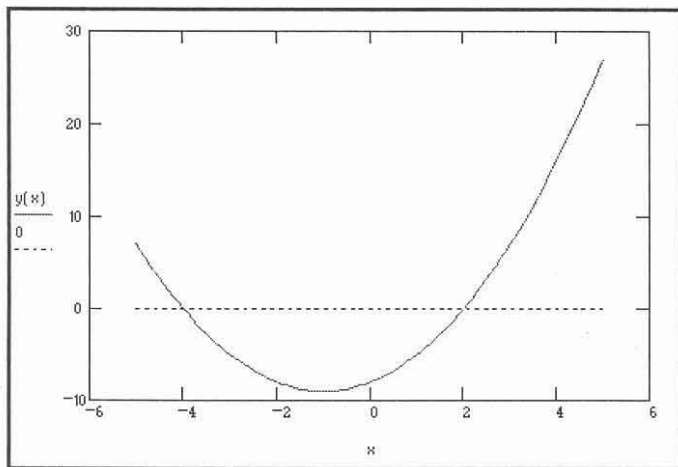
Schließlich stellen wir noch die x -Achse dar, in dem auf der Ordinate (Eingabe eines Kommas), die zweite "Funktion" "0" eingegeben wird.

Nun wollen wir die Nullstellen der Parabel berechnen. Dazu wird ein Startwert für x benötigt, der Mathcad-Befehl "Given", und die Funktionsgleichung $y(x)=0$. Je nach Startwert konvergiert die Lösung gegen die rechte oder die linke Nullstelle.

Die Ableitungsfunktion berechnen wir zu

$$y_s(x) = 2x + 2$$

In einem zweiten Bild wird diese zusammen mit der Ausgangsfunktion dargestellt.



5.2 Höhenschnittpunkt eines Dreiecks im Raum (DREIECK.MCD)

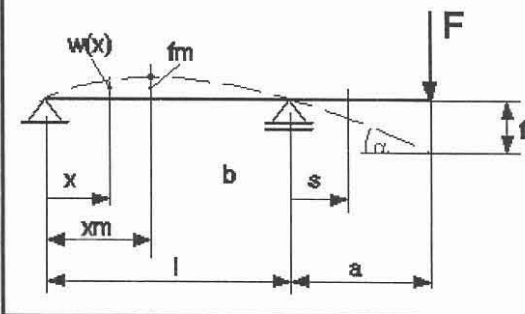
Hier wird das Beispiel aus den letzten PC-NEWS wiederum behandelt. Berechnet werden die selben Größen wie in der Derive-Berechnung. Die Resultate stimmen natürlich überein.

Interessant, die verschiedenen Konzepte der beiden Software-Pakete an diesem zweifach gerechneten Beispiel zu sehen!

5.3 Biegeträger (TRAEGER.MCD)

Sehen wir uns nun ein mehr technisches Problem an: Es soll die Durchbiegung eines zwei mal gelagerten Balkens, der am freien Ende mit einer Einzelkraft belastet ist, berechnet werden.

Es sollen die Durchbiegung $w(x)$, sowie die Größen f_m , f und α des folgenden Trägers berechnet werden:

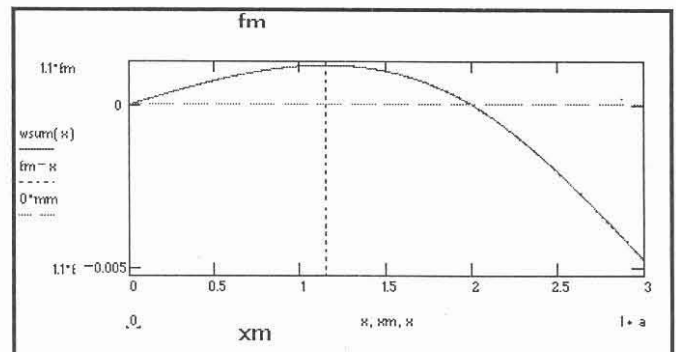


Hier kommen als Dimensionen "Meter" und "Newton" vor.

Die Eingangsgrößen des Problems sind (in [] steht die Dimension):

- Trägerlänge "l" und "a", [m]
- Belastung (Kraft) "F", [N]
- Flächenträgheitsmoment "J" des Querschnitts, [m⁴]
- E-Modul "E" (Werkstoffkenngröße) [N/m²]

Sind diese Werte bekannt, schlagen wir eine Formel (=Gleichung) für die Biegelinie dieses Balkens in einem entsprechenden Datenbuch (hier Dubbel, Taschenbuch des Maschinenbaus) nach und geben sie in Mathcad ein. Gleichung (1) gilt nur im linken, (2) nur im rechten Bereich. Die Mathcad "if-Bedingung" kombiniert die beiden Teile. Die Größe der Maximalabsenkung unter der Kraft und der maximalen Aufwölbung wurden direkt und alternativ durch Einsetzen in (1) und (2) berechnet.



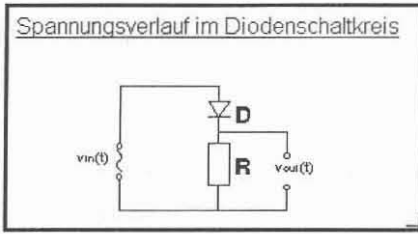
Während wir aber bei der Berechnung mit Papier, Bleistift und Taschenrechner darauf achten müssen, daß wir die richtigen Einheiten einsetzen oder richtig mit Zehnerpotenzen multiplizieren, nimmt uns Mathcad die Arbeit ab. Wir könnten z.B. l in cm und J in mm⁴ verwenden.

Während in diesem Beispiel der Aufwand für die Berechnung "zu Fuß" noch vertretbar erscheint, kommen bei Berechnungen, bei denen mehrere Trägerabschnitte mit komplizierteren Belastungsverläufen (Gleichlastanteile etc.) zu berechnen sind, die Vorteile von Mathcad erst voll zu tragen.

5.4 Schaltungen (DIODE.MCD, DIODE4.MCD)

Da der PC-Club erfahrungsgemäß viele Elektrotechniker zu seinen Mitgliedern zählt, darf auch ein Schaltungsbeispiel nicht fehlen, das ich mit Mathcad untersucht habe. (Nebenbei: Da ich selbst kein Elektrotechniker bin, verzeihe man mir die in dem Dokument evt. nicht ganz astreinen Bezeichnungen etc...)

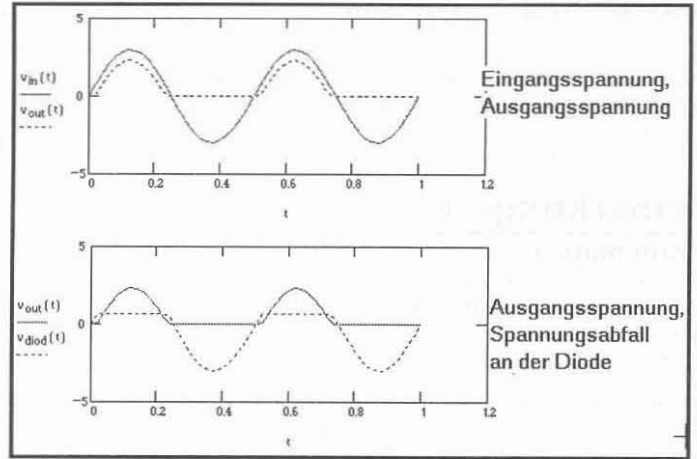
Wir betrachten einen Diodenschaltkreis, an dem eine sinusförmige Wechselspannung angelegt wird.



Und so stellen sich die Formeln im Mathcad-Dokument dar: 1)

$v_0 = 3 \cdot V$	Scheitelspannung
$u_F = 0.7 \cdot V$	Diodenknickspannung
$n = 50$	Anzahl der berechneten Werte
$f = 2 \cdot Hz$	Frequenz der Wechselspannung
$T = \frac{1}{f}$	Periodendauer der Wechselspannung
$dt = \frac{2 \cdot T}{n}$	Schritt-Zeitintervall
$t = 0 \cdot sec, \frac{2 \cdot T}{n}, 2 \cdot T$	Zeitpunkte für Berechnung
$v_{in}(t) = v_0 \cdot \sin(2 \cdot \pi \cdot f \cdot t)$	Eingangsspannung definieren
$v_{out}(t) = \text{if}(v_{in}(t) > u_F, v_{in}(t) - u_F, 0 \cdot V)$	Ausgangsspannung des Reizes (Diodenkennlinie)
$v_{diod}(t) = v_{in}(t) - v_{out}(t)$	Spannungsabfall an der Diode

Das entsprechende Mathcad Dokument findet auf einer einzigen A4-Seite Platz und zeigt Eingangs- und resultierende Ausgangsspannung im Bild.



Eine Variante dieses Beispiels berechnet einen Gleichrichterschaltkreis, an dem wiederum eine Wechselspannung anliegt. Im Ergebnis sehen wir die um die doppelte Diodenknickspannung "geschwächte" Ausgangsspannung des Kreises. (DIODE4.MCD).

>>> Schluß auf der nächsten Seite

PABLITOS Software

PABLITOS Software GesmbH

A-8063 Eggersdorf bei Graz

Tel. 0043-3117-3251 · Fax 3251-90

Software für den technisch- wissenschaftlichen Anwender:

Dies ist nur eine kleine Auswahl aus unserer Preisliste (Schulpreise inkl. Mwst.).

Statistik

STATISTICA /Windows	Statistik und Grafikprogramm von StatSoft/USA	e	14.990.-	!	Pr
QUICK STATISTICA /W	Statistik und Grafikprogramm von StatSoft/USA	e	10.128.-	!	Pr

Datenanalyse und automatisierte Datenerfassung

DADISP /DOS oder W	Analyse v. techn. Daten, Signalanalyse, Qualitätsprüfung	e	V 3.01	15.918.-	!	☐	Pr
Origin + 3D & Contour Module /W	Daten-Analysen u.v.m., Zusatzmodule zur Datenacquisition	e	V 3.0	9.120.-	!	☐	Pr

Mathematik

MathCad /W	löst anspruchsvolle mathemat. Probleme; neueste V.	e	V 4.0	4.440.-	!	
MathCad /W	löst anspruchsvolle mathemat. Probleme; anwenderspezifische Pakete auf Anfr.	d	V 3.1	3.024.-	!	
Mathematica /W Enhanced	weltweit führende Softw.f. numerische, symbol. u. grafische Mathem.	d		16.140.-	!	☐
MathType /W	perfekter mathemat. Formelsatz f. Textverarbeitung und DTP; Pi-Fonts erhältl.	e	V 3.0	3.588.-	!	

Rufen Sie uns an - wir senden Ihnen gerne Informationen und unsere aktuelle Preisliste zu!

Als Microsoft-Select-Partner informieren wir Sie gerne über das MS-Select-Angebot.

PABLITOS SOFTWARE A-8063 EGGERSDORF Tel. 03117-3251 Fax 03117-3251-90

Mathematica

Michael Kugler, N, TGM

DSK-404\MMC

Mathematica gesellt sich in die Reihe der Mathematik-Programme wie Derive oder Mathcad. Es muß nicht unbedingt als Konkurrent angesehen werden; jedes dieser Programme hat seine Anwender. Dieser Beitrag - weitere werden folgen - zeigt an Hand des Beispiels eines Serienschwingkreises, wozu Mathematica in der Lage ist.

Bemerkungen

Mathematica

Eine der Besonderheiten liegt in der Möglichkeit, Trickfilme zu erstellen. Da es in einer Zeitung nicht möglich ist, einen Film ablaufen zu lassen, habe ich aus den jeweiligen Sequenzen die beiden ersten und letzten Bilder dargestellt.

Die Syntax von Mathematica erscheint auf den ersten Blick ungewohnt. Ich habe versucht, dort, wo es mir wesentlich erschienen ist, einige Bemerkungen zu machen. Diese Bemerkungen lassen sich im Notebook, bedingt durch eine raffinierte Zellstruktur, ausblenden, so daß der jeweilige Benutzer nur die für ihn relevanten Bereiche sieht. Im vorliegenden Notebook sind alle Ebenen eingeleitet.

Mathematica und Winword

Mathematica arbeitet auf den verschiedensten Plattformen, Windows ist nur eine von ihnen. Damit die Kompatibilität gewahrt bleibt, verwendet Mathematica Postscript zur Formatierung von Text und Graphik. Daraus ergibt sich ein Problem in der Zusammenarbeit mit Winword. Ein direktes Einlesen des Notebooks in Winword ist nicht möglich. Es ist jedoch über einige Umwege möglich. Bei der Graphik gibt es jedoch ein Problem: Die Größenformatierung der in den Graphiken vorkommenden Zeichen werden von Winword schlichtweg ignoriert und mit einem relativ kleinen Wert belegt; sogar wenn in Mathematica mit Formatanweisungen die Graphiken mit größeren

Fonts ausgestattet werden. Daher ist die Beschriftung der Graphiken nicht optimal.

Das vorliegende Notebook

In jeder interaktiven Sitzung mit Mathematica werden die Eingabezeilen z.B. mit `In[3]:=` bezeichnet. Jeder Output erhält entsprechend `Out[3]=`. Damit Ein- und Ausgaben unterscheidbar sind, habe ich daher für die Eingaben **Lucida Sans Typewriter fett**, für die Ausgaben **Lucida Sans Typewriter** benutzt. Im Notebook sieht das ganze dann so aus:

```
In[23]:=1sg1=Solve[Im[zGes]==0,omega]
Out[23]={{omega -> -(-----)}
                Sqrt[c] Sqrt[1]}
```

Da in Mathematica alle eingebauten Funktionen mit einem Großbuchstaben beginnen, ist es sinnvoll alle selbstdefinierte Namen mit einem Kleinbuchstaben beginnen zu lassen, daher z. B. `omega` statt `Omega`.

Der Mathreader

Die Firma Wolfram-Research in Illinois, USA, als Urheber von Mathematica hat ein frei verfügbares Produkt herausgegeben, den Mathreader, mit dessen Hilfe Notebooks gelesen und ausgedruckt werden können. Dieser Mathreader ist ab sofort beim Klub erhältlich. Er enthält zusätzlich weitere informative Notebooks über Mathematica.



>>> *Schluß des Beitrags Mathcad*

6. Zusatzsoftware

Die bis jetzt beschriebenen Beispiele geben einen kleinen Einblick, wie Berechnungsbeispiele mit Mathcad gelöst werden können. Für komplexere Aufgaben zeigt es sich nun, daß das Auffinden der entsprechenden Definitions-Gleichungen des jeweiligen Problems etc. und das Formulieren des Problems in mathematischer Form nun der zeitraubendste Teil der Lösung ist. (Was für das Berechnen "zu Fuß" natürlich auch gilt.)

Um dem Anwender zeitraubende Eingaben zu ersparen, sind in der Zwischenzeit eine Reihe von Zusatzpaketen auf den Markt gekommen. Es gibt im wesentlichen zwei Arten von Zusatzsoftware:

- Elektronische Handbücher: Zu diesen zählen im wesentlichen Tabellen mit einer großen Anzahl von physikalischen Konstanten.
- Zusatzpakete: Hier werden Aufgaben aus Teilgebieten der Mathematik beispielhaft gelöst. Diese "Fertig-Dokumente" können für spezielle Fragestellungen leicht abgewandelt werden. Es sind beispielsweise folgende Pakete verfügbar: Statistik, Differentialrechnung, Bauingenieurwesen, Elektrotechnik, Mechanik usw.

Mit diesen Zusatzpaketen ist es sogar möglich, gewöhnliche, auch nichtlineare Differentialgleichungen (mechanische und elektr. Schwinger etc.) zu lösen, sei es, daß keine explizite, allgemeine Lösungsfunktion existiert oder daß es zu aufwendig wäre, diese ausfindig zu machen. Derartige Aufgaben erfordern allerdings einen entsprechend schnellen Rechner, um z.B. bei einer Runge-Kutta Näherungsrechnung mit 400 Stützstellen innerhalb vernünftiger Zeit zu einem Ergebnis zu kommen.

7. Ausblick

Abschließend kann ich feststellen, daß mit Mathcad die rechnerische Lösung einer Vielzahl von Problemen wiederum ein Stück einfacher geworden ist. Dennoch dürfen wir die Tatsache nicht aus den Augen verlieren, daß eine Rechnung nur so gute Ergebnisse liefern kann, wie

die Eingangsdaten "verlässlich" sind. Es macht also beispielsweise wenig Sinn, eine Trägerdurchbiegung auf 0.01mm zu berechnen, wenn die wirkende Kraft nur grob bekannt ist. Genauso muß die Gültigkeit von Gleichungen für entsprechende Werte beachtet werden usw. Zusammengefaßt, wir dürfen uns nicht dazu verleiten lassen, wegen leistungsfähiger "Rechen-Tools" den gesunden Menschenverstand außer acht zu lassen. Grundlegende Kenntnisse der Mathematik oder Mechanik kann auch noch so leistungsfähige Berechnungssoftware nicht ersetzen!

Sollten Sie weitere Fragen oder Anregungen zu meinen Ausführungen haben, können Sie mich über CompuServe (100276,1244) erreichen.

Literaturhinweise

- D. Donnelly, MathCAD for Introductory Physics, Addison-Wesley, 1992
- J. Rowell, Mathematical Modeling with MathCAD, Addison-Wesley, 1990
- Div. Autoren, Quarterly Electronic Magazine, Mathsoft Inc., 1993
- P. Voit, Mathcad 3.1 für Windows, Toolbox-Magazin, Red.DOS, 4/1992
- D. Reiermann, Mathcad für Windows, eine gute Lösung, PC-NEWS -31, S.38.
- H. Schwarz, Vektoralgebra mit Derive, PC-NEWS -35, S.18 □

Hinweise

Alle hier dargestellten Beispiele sind sowohl im Mathcad-Format der Version 3 als auch im Format der Version 4 auf der Diskette in den Verzeichnissen `MCAD\3` und `MCAD\4` enthalten.

¹⁾ Der Textfehler bei v_{out} entstand durch Konversion der in Version 4 erstellten Dokumente in die Version 3 mit einer nicht ganz vollständig installierten MathCad-Version in der Redaktion.

Resonanzkurve eines Serienschwingkreises

`Remove["Global`*"]; (*löscht alles*)`
`Needs["Algebra`ReIm`"] (*liest dieses Paket ein*)`

Was soll hier passieren?

Durch Berechnen des komplexen Widerstandes wird der Strom durch diesen Widerstand bestimmt. Die Abhängigkeit des Stromes von der Frequenz (die Resonanzkurve) und der Phase ist zu zeichnen.

Für beide Zeichnungen ist ein Film zu erstellen, in dem der Serienwiderstand der zu verändernde Parameter ist.

Das ganze Paket soll so allgemein wie möglich gehalten werden. Die speziellen Daten des Schwingkreises sollen erst bei der Auswertung der Zeichnungen eingesetzt werden.

Berechnung des Widerstandes

Im allgemeinen rechnet Mathematica mit komplexen Zahlen. Da üblicherweise R, C, L und die Frequenz reelle Werte sind, wird dies in der folgenden Regel Mathematica explizit mitgeteilt. Damit sind Vereinfachungen erst sinnvoll.

```
Im[r]^=0;
Im[l]^=0;
Im[c]^=0;
Im[omega]^=0;
```

Der Gesamtwiderstand `zGes` ist die Summe der einzelnen Widerstände.

```
zGes:=r + I omega l + 1/(c omega l)
```

Berechnung der Resonanzfrequenz

Wird der Imaginärteil des Gesamtwiderstandes Null, so herrscht Resonanz.

```
lsg1=Solve[Im[zGes]==0,omega]
```

```
{omega -> -(-----)},
          1
          Sqrt[c] Sqrt[l]

{omega -> -----}
          1
          Sqrt[c] Sqrt[l]
```

Da die erste Lösung einen negativen Wert hat, wird `omega0` der zweiten Lösung zugewiesen.

```
omega0=omega/.lsg1[[2]]
```

```
1
-----
Sqrt[c] Sqrt[l]
```

Eingabe der Daten

Alle für den Schwingkreis wichtigen Daten werden in einer Liste `daten` abgelegt. Wann immer es notwendig ist, Werte einzusetzen, kann diese Liste benutzt werden. Die Daten werden als Gleitkommazahlen eingegeben, damit wird jeder Ausdruck automatisch numerisch und nicht symbolisch ausgewertet.

```
daten={
  r->100.,
  l->40 10^-3,
  c->0.6 10^-6,
  uEin->1
};
```

Die Resonanzfrequenz ergibt sich dann durch Anwenden der Liste `daten` auf den Ausdruck `omega0`. (Genauer: Im Ausdruck `omega0` werden alle Namen die in der Liste `daten` vorkommen durch die Werte ersetzt, und anschließend wird der Ausdruck solange vereinfacht, bis sich keine Änderung mehr ergibt.)

```
omega0/.daten
6454.97
```

Zur Kontrolle wird der Gesamtwiderstand bei der Resonanzfrequenz berechnet.

```
zGes/.omega->omega0 /.daten
100.
```

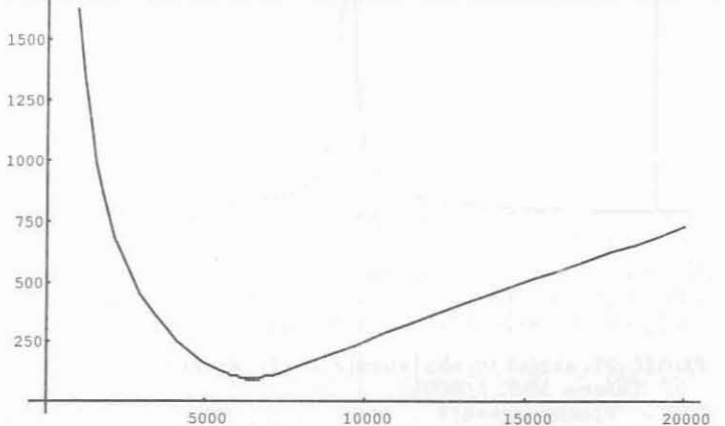
Bei einer kleinen Verstimmung muß der Widerstand wieder komplex sein:

```
zGes/.omega->(omega0 * 1.1) /.daten
100. + 49.2925 I
```

Zeichnen des Frequenzganges des Widerstandes

Mit diesen Daten soll nun der Frequenzgang des Gesamtwiderstandes (das ist der Betrag des komplexen Widerstandes) gezeichnet werden.

```
Plot[Abs[zGes/.daten],{omega,1000,20000}];
```



Aus FIDO/SCHULBRETT: Ein Vorschlag zur Rechtschreibreform:

ERSTER SCHRITT:

generelle kleinschreibung (werbeleute und graphiker benutzen sie bereits).

ZWEITER SCHRITT:

wegfall der dehnungen und schärfungen (dise masname eliminirt di meisten feler in den schulen).

DRITER SCHRIT:

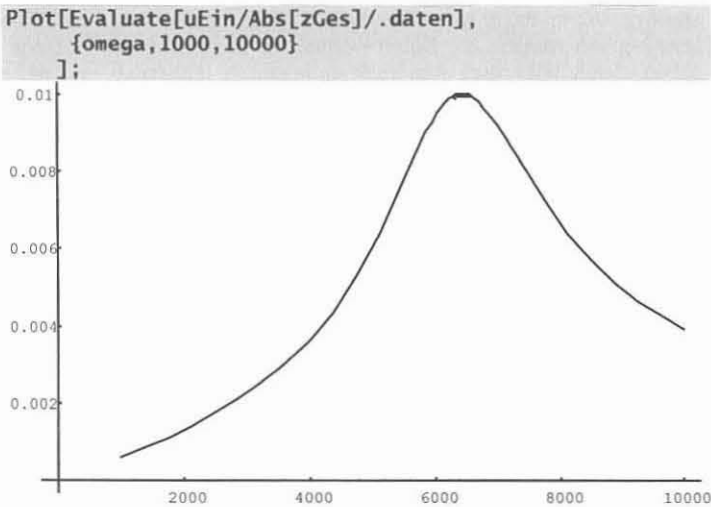
v, w und ph ersezt durch f; z ersezt durch s; sch ersezt durch s; x ersezt durch ks (das alfabet fird um fir buchstaben redusiert; sreib- und setsmaschinen fereinfachen sich; fertfole arbeitskrefte können der firtsaft sugefürt ferden).

FIRTER SRIT:

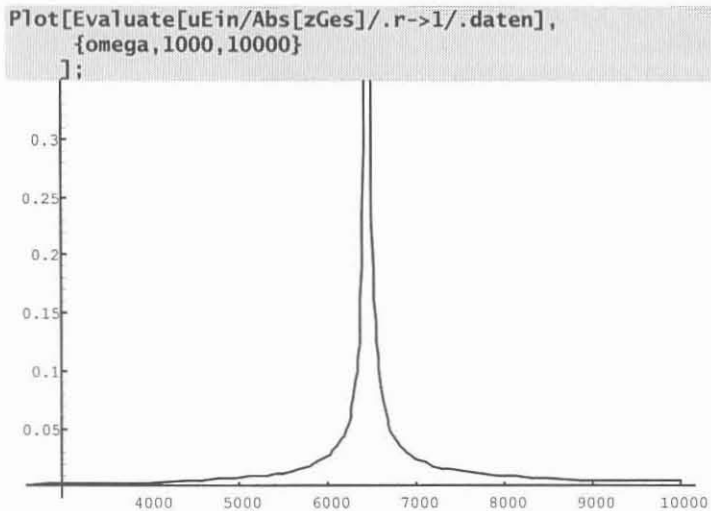
q, c und ch ersest durch k; j und y ersest durch i; pf ersest durch f; ferner k durch g; t durch d; p durch b (es sind son elf buksdaben ausgesalded, in den sulen können sdad aksig brosend rekdsreibung nüslikere feker fi fisik, kemi und mademadig mer geflegt ferden).

Der Stromverlauf

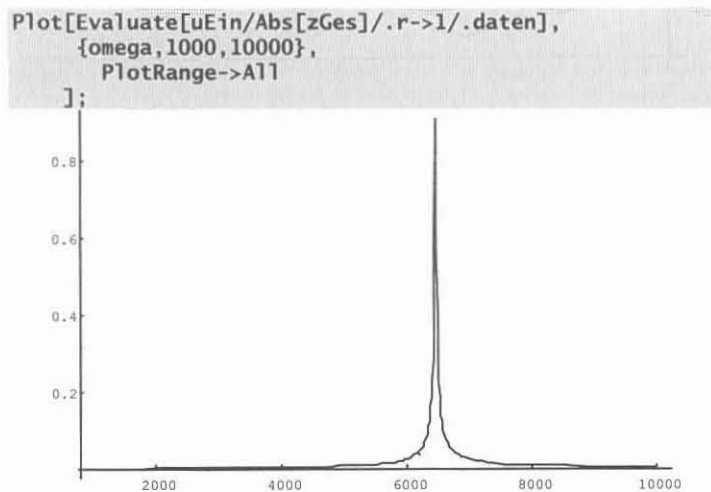
Spannung durch Widerstand ergibt den Strom, dieser wird nun mit den gegebenen Daten gezeichnet. (Der Befehl Evaluate in der Plot Anweisung dient nur zu der schnelleren Auswertung.)



Will man diese Kurve mit einem anderen Widerstand zeichnen, so muß die Variable r zuerst mit einem anderen [m1] Wert belegt werden. Auf diese Art kann die oben angelegte Liste daten verwendet werden, da r zu diesem Zeitpunkt der Substituierung bereits durch einen Wert ersetzt wurde.



Mathematica sucht einen vernünftigen Zeichenbereich; dabei kann es vorkommen, daß einzelne Punkte nicht mehr gezeichnet werden. Mit der Option PlotRange->All werden alle Punkte gezeichnet.



Und nun ein Movie

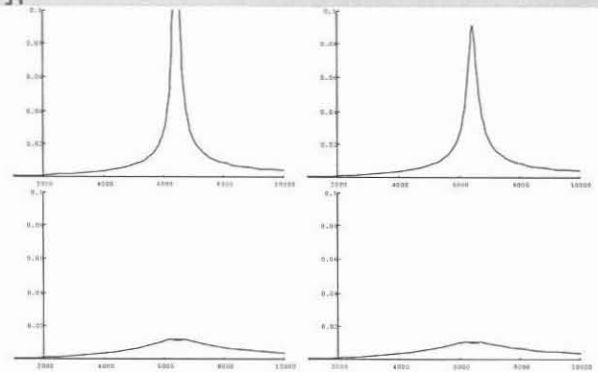
In dieser Bildfolge soll der Widerstand verändert werden. Dazu ist es sinnvoll, eine neue Funktion zu definieren, die zwei Parameter hat; die Frequenz und den Widerstand. Alle anderen Werte werden wieder von Parameterliste übernommen.

Dies ist überdies effektiver, da nur einmal die Substituierung vorgenommen wird, und nicht bei jeder Berechnung im Plot-Kommando.

```
strom[omega_,rw_]=uEin/Abs[zGes]/.r->rw/.daten
-----
1
6
-1.66667 10 I I
Abs[----- + --- omega + rw]
      omega      25
```

Nun können die Bilder gezeichnet werden. PlotRange ist notwendig, damit alle Bilder den gleichen Maßstab haben.

```
Table[
  Plot[strom[omegaBereich,rBereich],
        {omegaBereich,1000,10000},
        PlotRange->{{1000,10000},{0,0.1}}],
  {rBereich,1,100,10}];
```

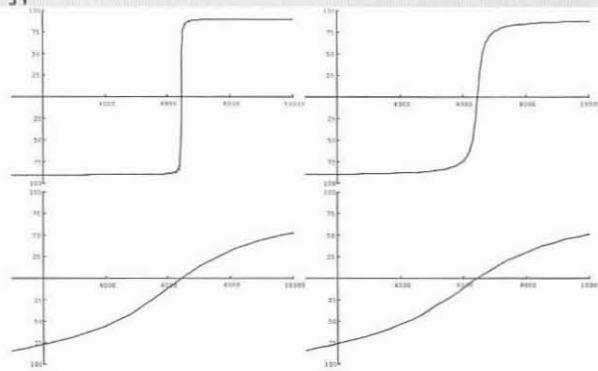


Der Phasengang

Für den Phasengang wird in ähnlicher Weise vorgegangen. Die Division durch Degree ist wegen der Umrechnung von Radianten notwendig.

```
phase[omega_,rw_]=Arg[zGes]/Degree/.r->rw/.daten
-----
6
-1.66667 10 I I
Arg[----- + --- omega + rw]
      omega      25
-----
Degree
```

```
Table[
  Plot[phase[omegaBereich,rBereich],
        {omegaBereich,1000,10000},
        PlotRange->{{1000,10000},{-100,100}}],
  {rBereich,1,200,10}];
```



Die normierte Darstellung

Zunächst normieren wir die Frequenz. Damit es übersichtlicher wird, ist es sinnvoll, die Belegung von ω_0 zu löschen.

```
Clear[omega0]
```

Die Verstimmung v ist folgendermaßen definiert:

$$v = \frac{\omega - \omega_0}{\omega_0}$$

dies nach ω aufgelöst ergibt:

```
lsg2= Solve[v==omega/omega0-omega0/omega, omega]
{{omega -> - (omega0 v - Sqrt[4 omega0^2 + omega0^2 v^2]) / 2},
 {omega -> (omega0 v + Sqrt[4 omega0^2 + omega0^2 v^2]) / 2}}
```

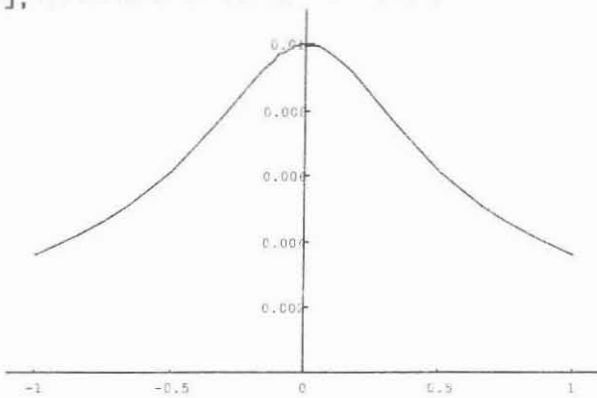
In unsere Formel für den Strom müssen wir nun ω durch diesen Ausdruck ersetzen, da nun v der Parameter zum Zeichnen ist. Die vorhin gelöschte Belegung für die Resonanzfrequenz ω_0 wird wieder installiert. (Dies war nur für die Übersichtlichkeit, nicht aber für die Rechnung an sich notwendig).

```
omega0=omega/.lsg1[[2]]
1
```

```
Sqrt[c] Sqrt[l]
```

Mit der Funktion `strom1` wird nun der auf die Frequenz normierte Strom bezeichnet, und ω durch die erste der beiden Lösungen ersetzt, anschließend werden wieder die Daten eingesetzt.

```
strom1[v_, rVariabel_] =
  strom[omega, rVariabel] /. lsg2[[1]] /. daten;
Plot[strom1[vBereich, 100],
 {vBereich, -1, 1},
 PlotRange -> {{-1.1, 1.1}, {0, 0.011}}
];
```



Nun zur Normierung des Stromes. Dazu ist es notwendig, den maximalen Strom zu wissen. Dieser fließt bei der Resonanzfrequenz ω_0 .

```
iMax[rVariabel_] = strom[omega0, rVariabel] /. daten
1
```

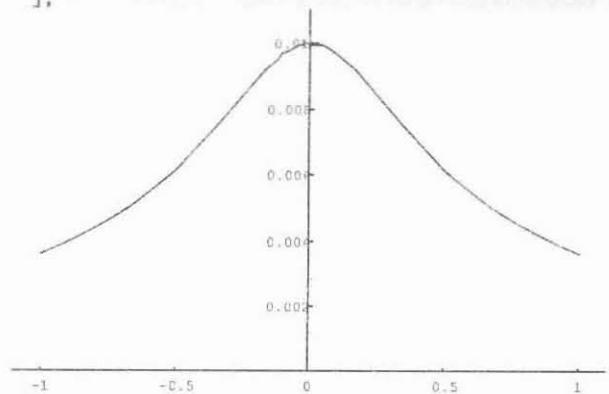
```
Abs[0. I + rVariabel]
```

Die Funktion `stromNormiert[v_, rVariabel_]` liefert die Werte des normierten Stromes bezüglich der Verstimmung v .

```
stromNormiert[v_, rVariabel_] =
  strom1[v, rVariabel] / iMax[rVariabel]
Abs[0. I + rVariabel] -
  Abs[rVariabel] -
  3.33333 10^6 I
----- +
  6454.97 v - Sqrt[1.66667 10^8 + 4.16667 10^7 v^2]
I
-- (6454.97 v - Sqrt[1.66667 10^8 + 4.16667 10^7 v^2])
50
]
```

Und nun die graphische Darstellung für einen Serienwiderstand von 100 Ohm.

```
Plot[stromNormiert[vBereich, 100],
 {vBereich, -2, 2},
 PlotRange -> {{-2.1, 2.1}, {0, 1}}
];
```

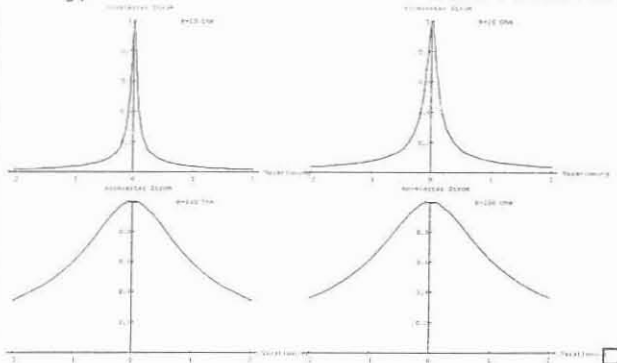


Der Übergang zu einem Movie ist nun nicht schwer. Zusätzlich werden in die Plotanweisung noch einige Verschönerungen eingebaut.

Info zum folgenden Plot

`AxesLabel` liefert die Achsenbeschriftung. `PlotStyle` wird benutzt um mit der Funktion `Text` einen String in der Graphik auszugeben. Mit `StringForm` lassen sich - ähnlich wie in C - Text und Werte von Variablen formatiert ausgeben.

```
Table[Plot[stromNormiert[vBereich, rVariabel],
 {vBereich, -2, 2},
 PlotRange -> {{-2.1, 2.1}, {0, 1}},
 AxesLabel ->
 {"Verstimmung", "normierter Strom"},
 PlotStyle ->
 Text[StringForm["R = ` Ohm", rVariabel],
 {1, 1}
 ],
 {rVariabel, 10, 200, 10}
];
```



Die neue Telebox: TELEBOX AUSTRIA

Gerhard Kmet, Radio-Austria

DSK-403, LIT-067

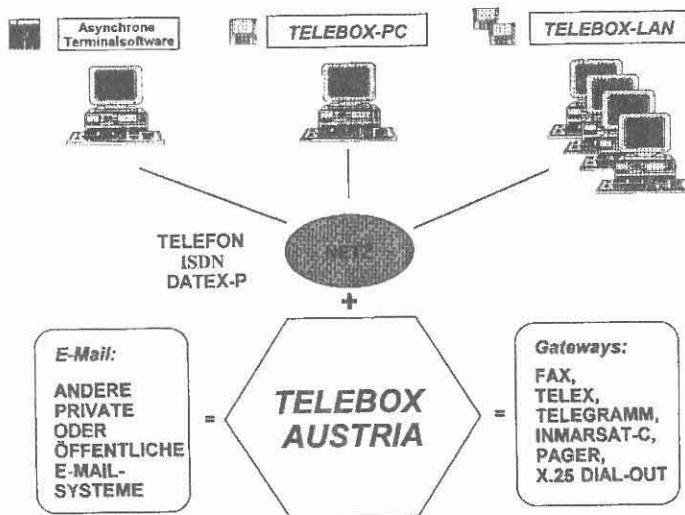
Die Dienstleistung **TELEBOX** wird schon seit 1984 als Gemeinschaftsdienst von Post und RAC angeboten; in Europa zählt RAC damit zu den Pionieren in Sachen electronic mail.

Dank dem Leistungsmerkmal "TELEX" entwickelte sich der Dienst anfangs sehr gut. Der Rückgang des Telexverkehrs wurde auch bei der **TELEBOX** stark spürbar, konnte jedoch durch den neugeschaffenen Fax-Uebergang und durch Anwendungen geschlossener Benutzergruppen einigermaßen kompensiert werden.

Heute ist für die weitere Entwicklung des e-mail-Verkehres das enorme Wachstum bei den PCs von besonderer Bedeutung. Zur Zeit werden weltweit etwa 150 Millionen PCs überwiegend für Büroaufgaben eingesetzt. Darunter befinden sich bereits 42 Millionen "e-mail Benutzer". (Etwa 65% davon in den USA, 30% in Westeuropa und 5% anderswo. Quelle: EMMS Newsletter.)

Ausgehend vom gewaltigen technischen Fortschritt bei PCs und Netzwerken (Schnelligkeit/Speicherleistung, hohe Übertragungsgeschwindigkeit) konnten "menschengerechtere" Programme (für mehr Benutzerfreundlichkeit und Komfort) geschaffen werden. Erst dadurch wurden die Voraussetzungen zur breiten Akzeptanz dieses Mediums geschaffen und in der Folge ging die Verwendung von e-mail über den Kreis der Computer-Freaks hinaus und hielt Einzug in die Büros.

Dieser Entwicklung Rechnung tragend, entschloß sich RAC, den **TELEBOX**-Dienst neu zu gestalten. Um sich von ausländischen Dienst Anbietern gleichen Namens zu unterscheiden, wird unsere dritte e-mail-Generation **TELEBOX AUSTRIA** (TBXA) genannt.



Die Kapazität des Systems ist für etwa 4000 Teilnehmer-Fächer ausgelegt. Als Hardware-Plattform wird eine ausfallgesicherte Cluster-Konfiguration (3 Rechner) von Digital Equipment verwendet. Es werden gedoppelte Plattenspeicher, X.25-Einheiten, und DS700-Terminalserver (geeignet zum lokalen Anschluss schneller Modems und ISDN-Terminaladaptoren) eingesetzt.

Angeschlossen sind: X.400-, Internet-, Inmarsat-C-, Fax-, Alert- und Telex-Gateways. Alle Einheiten sind über Ethernet verbunden und über eine DECbridge mit dem Backbone Ethernet von RAC verknüpft. Als Betriebssystem- und Betriebssystemnahe-Software werden Produkte von DIGITAL verwendet. Die e-mail-Anwendungssoftware (GEO-II) kommt von GeoNet in Deutschland. Das System ist als X.400 PRMD direkt an die ADMD ADA400 von RAC angeschlossen und über X.25/Intermail mit 14 anderen GeoMail-Systemen (deren Standorte von San Francisco bis Kemerovo, Sibirien, reichen) verbunden.

Wie bisher ist die **TELEBOX AUSTRIA** über Telefon und Datex-P erreichbar. Neu ist der Zugang via SprintNet im "One Stop Billing-Verfahren" aus 38 Ländern, darunter die GUS-Staaten.

Als wesentliche Neuerung ist die "Front-End"-Software anzusehen. Sie bewirkt, daß der kundeneigene PC zu einem "Remote-User-Agent" (RUA) wird, der, ohne mit der **TELEBOX** verbunden zu sein, viele Leistungsmerkmale unterstützt und die Arbeitsweise vereinfacht. Diese Software, die unter Windows läuft, heißt in der Einzelplatzversion **TELEBOX-PC** bzw. **TELEBOX-LAN** in der Mehrplatzversion. Sie besteht tatsächlich aus zwei Programmen: **MailStation** und **Mailer**.

Mit **MailStation** bearbeitet der Kunde seine gesamte Korrespondenz lokal in seinem PC (Nachrichten lesen, beantworten, weiterleiten, erstellen, verschlüsseln!, speichern, drucken usw., usw.), noch ohne mit der **TELEBOX** verbunden zu sein, was Kosten spart.



MailStation ist LAN-fähig; der Nachrichtenaustausch mit anderen an LAN angeschlossenen Benutzern bleibt lokal. Für die Kommunikation nach außen verfügt jeder einzelne Arbeitsplatz im LAN über ein e-mail-Fach in der TBXA.

Mailer ersetzt das Terminalprogramm, d.h. er ist für die Kommunikation mit der **TELEBOX AUSTRIA** verantwortlich. Er baut automatisch, zu den vom Kunden definierten Zeiten, die Verbindung zum Kundenfach in der TBXA auf, versendet die dafür bereitgestellten Nachrichten, bzw. nimmt die in der TBXA eingetroffenen entgegen. Die Übertragung findet abgesichert durch das ZMODEM-Protokoll statt; optional ist Komprimierung und/oder zusätzliche Verschlüsselung möglich. Natürlich können nicht nur Texte, sondern jegliche Arten von Binärdateien übertragen werden, also auch Anwendungs-Software, Bild- und Sprach-"Annotations", Bild- und Sprach-Dateiübertragungen benötigen allerdings schnelle Netzzugänge - z.B. ISDN.

Mit dieser "Front-End-Software" steht der **TELEBOX** ein vollkommen neues, grosses und zukunftsträchtiges Kundenpotential offen: das der Einzelplatz-PCs und der Mehrplatz-PCs im LAN-Verbund. Dadurch und dank der nunmehr möglichen transparenten Übertragung von Dateien erweitern sich die typischen Anwendungsgebiete für die **TELEBOX** beträchtlich.

Die e-mail lässt sich jetzt mühelos in bestehende EDV-Strukturen integrieren und erlaubt beispielsweise die Außendienststeuerung oder die Anbindung von im In- und Ausland gelegenen Filialen an eine Firmenzentrale. Bestellungen, Tagesberichte, Geschäftsabschlüsse, Börsenkurse, Laborberichte und viele andere Nachrichtentypen können problemlos (auch verschlüsselt) ausgetauscht, verteilt und elektronisch weiterverarbeitet werden. Kurz, die **TELEBOX AUSTRIA** erschließt neue Anwendungsbereiche bzw. ermöglicht professionelle Lösungen wo bisher nur Kompromisse erzielbar waren.

TELEBOX AUSTRIA wird gegenwärtig von ausgewählten Kunden getestet. Die Migration aller unserer heutigen **TELEBOX**-Kunden und die offizielle Betriebsaufnahme wird im 1. Quartal 1994 stattfinden. □

Hinweis: Auf der Diskette PCN-403 finden Sie eine Demoversion des Programms **TELEBOX-PC**. LIT-067 beschreibt die Installation des Programms. □

BTX-Service Guide: 24 Gründe für BTX-Pan



Etwa 10% der **PC-NEWS**-Leser besitzen eine BTX-Kennung. Da BTX österreichweit zum Ortstarif erreichbar ist, übertrifft es andere Datendienste an Brauchbarkeit für Benutzer. Benutzung setzt aber auch Wissen über die gespeicherte Information voraus oder das Wissen, wie man sie findet.

Manches vermisst man aber es gibt einige Datenbanken, die fast keinen Wunsch offen lassen. Nachfolgend finden Sie eine Zusammenfassung der über BTX erreichbaren Datenbanken. Wenn Sie mehr darüber erfahren wollen, bestellen Sie den BTX-Service-Guide über die BTX-Seite *648783# (S 24,-), (Informationen über *899096#).

DATENBANKNAME	Zugang K Kennung A Anonym B Berechtigung	Zugangsseite	Betriebszeit	Kosten
ABE MARKETING	K	*3141#	0-24	3,5/min+25/dok
ANZEIGENZENTRALE	A	*85703570#	0-24	-
BRANCHENVERZEICHNIS	K	*5544#	0-24	-
BTX-ANBIETERVERZEICHNIS	A	*192#	0-24	-
BTX-BÖRSENKURSE	K	*20700#	0-24	0,2/Seite
BTX-TEILNEHMERVERZEICHNIS	A	*191#	0-24	-
BUSINESS	B	*3141#	0-24	3/min+25/dok
DUDEN Bd.8 und 10	A	*85703570#	0-24	-
DUDEN INFORMATIK	A	*857035#	0-24	-
D&B	K	*3141#	0-24	5/min+70/dok
ELEKTRONISCHER INDEX	A	*3520#	0-24	-
ELEKTRONISCHES TELEFONBUCH	A	*8808#	0-24	-
FIRMENBUCH	K	*4028#	Mo-Fr 8-19	-
GRUNDSTÜCKSDATENBANK	B		Mo-Fr 8-20	einmalig 15000,-, 2/Seite
HANDELSREGISTER	K	*6212#	Mo-Fr 7-20	60/Firma, 120/Firma mit Beteiligung, 90/Person/Seite
HEROLD EINKAUF DATA	K	*3141#	0-24	5/min+15/dok
ISIS SOFTWARE	K	*3141#	0-24	5/min+25/dok
KSV-DATENBANK	K	*3141#	0-24	5/min+30..110/dok
MEYER'S LEXIKON	A	*85703570#	0-24	-
OFFICIAL AIRLINE GUIDE	K	*3141#	0-24	23/min+0,5/Seite
ÖBB ZUGAUSKUNFT	A	*2830#	0-24	-
POSTLEITZAHLEN UND ABGABESTELLEN	A	*85703570#	0-24	-
PRESSEDATENBANK	K	*21109#	0-24	99/Auftrag
REUTERS AUSTRIAN MERKETS SERVICE	K	*3141#	0-24	3/min
TELEXVERZEICHNIS	A	*81000002#	0-24	-
TIM	A	*65154#	0-24	-
UMWELT-DATENBANK	K	*3141#	0-24	3/min+25/dok
UMWELT-NETZWERK	K	*3141#	0-24	3/min+25/dok
WER LIEFERT WAS	K	*3141#	0-24	3,5/min+12/dok
WISSENSCHAFTSDATENBANK	A	*85703579#	0-24	-
TUB-GRAZ	K	*85703570#	0-24	-
ZOLLWERTKURSE	K	*3141#	0-24	20/Abfrage

Beispiel für eine Recherche mit den zusammengefaßten Datenbanken der IIG (DUDEN, MAYERs usw.). Gesucht wurde das Stichwort "thread" (siehe Beitrag über Windows-NT (**PC-NEWS** 29, S 35).

In diesen Lexika kann gesucht werden:

```

BTX Lexika
-----
BEWERTERLEXIKA
DEUTSCHES WISSENSLEXIKON
WISSENSCHAFTSDATENBANK EULENSPIEGEL
DUDEN Bd. 8 und 10
DUDEN Bd. 9: BEDEUTUNGSWORTERBUCH
DUDEN Bd. 10: BEDEUTUNGSWORTERBUCH
JARDON LEXIKON (ENGLISCH)
MAYER'S LEXIKON
D&B
GESAMTDATENBANK (Zusammenfassung)
In den ausgewählten Lexika wird die
Recherche durchgeführt. Ein Suchergebnis
wird nachfolgend angezeigt.
GESAMTINHALT LEXIKONER WÄHLEN
KONTAKT RECHNER VERLASSEN
  
```

Eingabe des Stichwortes:

```

BTX Lexika
-----
Beim BTX-Gesamtlexikon sind mehrere IIG-
Lexika zu einem Lexikon zusammengefaßt.
Der Umfang des Gesamtlexikons kann mit
Wahl verändert werden.
IIG: DUB, DUB, DUB, INF, AUS, DUB, PLC, JOB, DEL
Stichworte bestehen aus Buchstaben und
Ziffern. Sonderzeichen werden ignoriert.
Bei der Eingabe von Anfangsbuchstaben
von Stichworten müssen diese mit
[Blank] abgeschlossen werden (Standard-
Benutzerprofil).
Die Eingabe wird mit [ ] abgeschlossen.
Bitte eingeben oder mit
Gesamtlexikon wählen.
Bitte STICHWORT eingeben!
  
```

Gefunden (Querverweise angezeigt):

```

englischer Jargon
thread: n. (USNET) Genie, Compu-
Serve, Common abbreviation of local
thread, a more or less continuous
chain of postings on a single
topic
three-finger salute
ZURÜCK+ FORTSETZUNG+ LEXIKON AUSWAHL+
RECHNEN+ GEFUNDEN+ IN VERLASSEN
  
```

Opalis V 1.2 BTX-Decoder für OS/2

Susanne Rupprecht, 2:313/9.24, Georg Czedik-Eysenberg, 2:313/9.33

Opalis für OS/2 ist der erste BTX-Decoder, der speziell für dieses moderne Betriebssystem entwickelt wurde. So jung er ist, sind seine Leistungsmerkmale bereits bestechend und sein "Look and Feel" ist für OS/2-Eingeschwehrene erfreulichst ausgefallen.

Neu sind die Protokoll- und Film-Funktionen. Erstere ermöglichen das sequenzielle Aufzeichnen von BTX-Sitzungen, mit letzteren lassen sich Protokolle mit neuen Mitschnitten mischen, und es kann zeitgesteuert oder manuell vor- und rückwärtsgeblättert werden. Quicktasten für An- und Abwahl, Kurzwahlregister, bedienungsfreundlichste, editierbare Makros und Paßwortschutz sind nur einige der weiteren begeisternden Features von Opalis.

Die ISDN-Fähigkeit zeigt, daß Opalis am neuesten Stand der Technik ist.

Besonders hervorzuheben ist auch das REXX-Interface von Opalis. Die OS/2-eigene programmierbare Kommando-Sprache REXX wird dadurch

um mehr als 30 Befehle erweitert, sodaß maßgeschneiderten automatischen BTX-Abläufen kaum noch Grenzen gesetzt sind.

Einziger Wermutstropfen:

Opalis als deutscher Decoder funktioniert zwar ausgezeichnet im deutschen BTX, spielt jedoch nicht so gut mit dem österreichischen BTX zusammen. Erstens, da er nur die C0-Norm, nicht die C2-Norm unterstützt und zweitens, da bei den Protokoll- und Film-Funktionen das fehlende Seitenende-Zeichen im Österreichischen BTX Probleme macht.

Wenn eine entsprechende Anpassung an österreichische Verhältnisse erfolgt, so ist der Opalis-BTX-Decoder nur zu empfehlen.

Zum Testen gibt es eine Gratis-Schnupper-Version auf Diskette. Für weitere Informationen wenden Sie sich bitte an den **MCCA**, BTX *2550#. □

BTX-Diskussion im FIDO, immer aktuell

strafbar?

Georg Czedik, 2:313/9.33



In Zukunft wird vielleicht der Betrieb nicht zugelassener Modems auch strafbar sein. Ist da irgendetwas im Gange?

Alwin Pawlata, 2:313/1.10



Leider ja.

Es gibt eine EG-Richtlinie, die die Mitgliedsstaaten verpflichtet, dafür Sorge zu tragen, daß nicht zugelassene Endgeräte nicht in den Handel kommen. Und Österreichs vorausseilender EG-Gehorsam hat dazu geführt, eine diesbezügliche Strafbestimmung in den Entwurf eines neuen Fernmeldegesetzes einfließen zu lassen.

Zur Zeit wird die bereits dritte Fassung dieses Entwurfes beraten, ein Inkrafttreten per 1.1.94 wird angestrebt.

Im Gegensatz zum Handel ist der Betrieb von Endgeräten in der Vorlage nur soweit erwähnt, als die Verwendung zugelassener Endgeräte ohne weitere Bewilligung gestattet ist (Bemerkenswert dazu: Zur Zeit ist der Betrieb einer *Funkanlage* laut geltendem Recht bewilligungspflichtig, das trifft auch so alltägliche Geräte wie einen Fernsehempfänger. Ist die Funkanlage jedoch ein zugelassenes Endgerät, z.B. ein Schnurlostelefon, ist keine Betriebsbewilligung erforderlich).

Übrigens ist unter der *Liberalisierung*, die EG-weit von Politikern propagiert wird, nur die Möglichkeit zu verstehen, zugelassene Endgeräte aus dem Handel (und nicht nur von der Post) zu beziehen, und sie ohne weiters betreiben zu dürfen. Auch in der EG werden Zulassungen nur landesspezifisch erteilt werden. Zu glauben, in Zukunft jedes Gerät anstecken zu dürfen, zumindest wenn es irgendwo in der EG zugelassen ist, ist ein Trugschluß!

Es ist anzunehmen, das der Betrieb von nicht zugelassenen Endgeräten in der Folge durch die Geschäftsbedingungen der dann privatisierten Post untersagt werden wird. Strafbestimmungen kann ich mir da nicht vorstellen, jedoch die Kündigung der Anschlüsse im Falle einer Verletzung der Geschäftsbedingungen.

In dem Auch-Nicht-EG-Land Schweiz ist das Handelsverbot bereits seit heuer Gesetz. Ein Händler hat dort nach wiederholtem Verstoß mit einer Haftstrafe zu rechnen - dementsprechend sind in der Schweiz nicht zugelassene Geräte aus den Geschäften verschwunden. □

Modem

Bernhard Kümel



Kann mir vielleicht jemand erklären, warum's Modems gibt, die nur 2k4 schaffen, obwohl sie 9k6 faxen können? Warum läßt man die nicht auch gleich mit 9k6 Daten schicken?

Alexander Keck, 2:310/81.7



Ganz einfach: FAXe unterhalten sich beim Verbindungsaufbau mit einer sehr niedrigen Baudrate (300 Bd soviel ich weiß) um ihre Möglichkeiten aufeinander abzustimmen. Danach werden vom sendenden Fax die Daten mit 9600Bd übertragen. Das erfolgt aber nur noch in eine Richtung (HALF-DUPLEX). Das Problem dabei ist simpel: Bei FULL-DUPLEX Übertragungen muß man vom Modem her einiges berücksichtigen und mit verhältnismäßig großem Aufwand verhindern, daß es zu einer Überlagerung kommt, sprich, daß die gesendeten Daten nicht die empfangenen stören. Da diese Kanaltrennung aufwendig ist, sind schnelle Modems viel teurer. Bei einem FAX braucht man diese Kanaltrennung überhaupt nicht, da, nachdem sich die Geräte über die beste Übertragungsform geeinigt haben, alles nur noch in eine Richtung geht: keine Probleme mit Überlagerungen! □

Hardware-Voraussetzungen für OS/2 - Nebenfront !

Walter Riemer, EN/NA, TGM

Der Autor benützt OS/2 seit nunmehr 1 Jahr zur (soweit das in der EDV überhaupt möglich ist) vollsten Zufriedenheit. Die Version 2.0 war noch kein überzeugender "Hit", weil sie vieles recht behäbig anging und dementsprechend viel Zeit brauchte, vor allem aber, weil der "enhanced mode" im WINDOWS-Modus nicht unterstützt wurde und daher etliche neuere WINDOWS-Programme nicht liefen. Mit der Version 2.1 hat sich das geändert: Alle WINDOWS-Programme laufen unter WINOS2 eher verlässlicher als unter WINDOWS selbst; wenn der beliebte Schutzverletzungsfehler auftritt (übrigens seltener als unter WINDOWS selbst), besteht keine Gefahr, andere Programme in Mitleidenschaft zu ziehen, es sei denn, es handelt sich auch um WINDOWS-Programme in der selben WINOS2-Session (das kann man aber vermeiden, indem man mehrere WINDOWS-Programme in mehreren WINOS2-Sessions laufen läßt).

Die Tatsache, daß man für jedes DOS- oder WINDOWS-Programm eine individuelle Umgebung (zum Beispiel ein eigenes AUTOEXEC.BAT) einstellen kann, trägt sehr zur Optimierung der Programmläufe bei.

Im Rechenzentrum der Abteilung EN im TGM ist OS/2 auch auf einem 386SX - 25 MHz-Rechner mit 8 MB Hauptspeicher installiert und läuft auch auf diesem eher schwachen Rechner durchaus zufriedenstellend und jedenfalls nicht unerträglich langsam. Auf diesem Rechner gab es nur das Problem, daß er ursprünglich mit einer "Billigsdorfer"-VGA-Karte ausgestattet war, mit der zum Beispiel das DOS-Malprogramm DeLuxe Paint Enhanced überhaupt nicht lief (der Rechner stürzte einfach ab!) und OS/2 (auch die Version 2.1) nur 2 mal von 3 mal ordnungsgemäß hochlief und auch gelegentlich "unterwegs" seinen Geist aufgab. Dieser Fehler war nach Ersetzen durch eine bessere ET-4000-Karte behoben.

Konsequenz 1 daher: Volle IBM-Kompatibilität ist auch für OS/2 Vorbedingung!

Im Zuge des Wunsches, den im häuslichen Computer des Autors vorhandenen und unter DOS bzw. WINDOWS jederzeit nutzbaren LPT2-Anschluß auch unter OS/2 zu nutzen, stellte sich folgender überraschender Sachverhalt heraus:

OS/2 benützt für LPT1 den Interrupt 7 und für LPT2 den Interrupt 5. Alle Druckerkarten sind werksseitig auf LPT1 / IRQ 7 gejumpert; viele erlauben jedoch auch ein Umstellen auf LPT2 / IRQ 5.

Die für den AT-Bus geeigneten Druckerkarten haben jedoch (fast ?) alle den Fehler, daß sie zwar imstande sind, den Interrupt 7 zu generieren (was man auch mittels CheckIt oder dergleichen überprüfen kann), nicht aber den Interrupt 5; anders ausgedrückt, eine Fähigkeit, die sie laut Spezifikation haben sollten, haben sie de facto nicht. Laut CheckIt bleibt der Interrupt 5 frei ("available").

Unter DOS oder WINDOWS spielt das keine Rolle, da sich diese Betriebssysteme nicht um den Interrupt kümmern. OS/2 legt jedoch dezidiert Wert auf diesen Interrupt, wodurch man kaum in der Lage ist, mehr als ein bis höchstens zwei Zeichen zu drucken; dann meldet OS/2, daß der Drucker nicht ansprechbar ist.

Konsequenz 2 daher: Falls man unter OS/2 auch LPT2 nutzen möchte, sollte man ein hochwertigeres Mother-Board haben als eines mit AT-Bus und auf das Interrupt-Problem bei LPT2 speziell achten.

Etwas lästig erschien anfangs, daß unter OS/2 SyQuest-Wechselfestplatten anders formatiert sind als unter DOS, sodaß derselbe Datenträger nicht in beiden Betriebssystemen verwendbar ist. Da aber DOS vom Autor nur mehr in sehr seltenen Ausnahmefällen gebraucht wird (AutoCad 11 läuft wegen des eigenen Speichermanagements, das sich mit dem von OS/2 nicht verträgt, nicht unter OS/2), hat sich das Problem eigentlich selbst entschärft. Der OS/2-Treiber für die SyQuest wird in einer späteren Version von OS/2 sicher auch noch verbessert werden; momentan behandelt er die SyQuest nämlich noch wie eine große Diskette, wodurch die Geschwindigkeit nicht gerade mitreißend ist.

Wenn man diesen kleinen Mißlichkeiten aber die vielen Annehmlichkeiten gegenüberstellt, muß man mit OS/2 einfach zufrieden sein. Nur ein Beispiel: Wenn man einen Schiebeknopf am Rollbalken eines Windows-Fensters mit der Maus "packt", beim Verschieben aber mit dem Mauszeiger den Balken seitlich verläßt, bleibt der Verschiebeversuch wirkungslos. Nicht so unter OS/2: Hier wird offenbar nur die maßgebende (horizontale oder vertikale) Koordinate ausgewertet und auch ein "abirrender" Verschiebeversuch führt zum Ziel. So einfach geht das!

OS/2 Treiber 2.0 für Mitsumi CD-ROM Laufwerke

Schmid, PC-Gourmet



OS/2 unterstützt standardmäßig nur CD-ROM Laufwerke mit SCSI-Anschluß, oder?



Nicht richtig, meint **PC-Gourmet** in Wien und beweist das Gegenteil mit seinem neuen OS/2 Treiber für Mitsumi LU005S, FX-001 und FX-001 D Laufwerke.

Der ab sofort verfügbare Treiber für die OS/2 Versionen 2.0 und 2.1 unterstützt CD-ROM, CD-Audio sowie multisession Photo-CD. Die Installation von OS/2 ist

mit diesem Treiber auch von CD möglich und auch das bei OS/2 2.1 mitgelieferte Multimediaset MMPM/2 wird unterstützt. Die integrierte Stromsparfunktion ermöglicht ein Abschalten des Laufwerkmotors nach einer bestimmten Zeit der Nichtbenutzung. Sollte es Probleme mit eingebauten Zusatzkarten (Netzwerk etc.) geben, kann eine Version installiert werden, die den Datentransfer statt mit Interrupts über Polling erledigt.

Preis: 390,-,- ÖS (inkl. MWSt.) zgl. Versandkosten

Verfügbarkeit: sofort

Info bei: PC-Gourmet, Schmid & Witschel OEG
Ansprechpartner: Hr. Schmid

PC-Gourmet, Schmid&WitschelOEG,
Porzellangasse 60/7/Top33, 1091 Wien

TEL: (0222) 317 49 17

FAX: (0222) 317 49 18

Was ist neu bei DOS 6.2?

Der folgende Text stammt - der Erzeuger möge es nicht als Copyright-Verletzung sehen - aus dem Hilfeprogramm von DOS 6.2 und ist als ergänzende Information zu unserer Sammelbestellung für die Buchversion von DOS 6.2 gedacht.

Willkommen bei Microsoft-DOS, dem weit verbreitetsten Betriebssystem für Personalcomputer. Im Vergleich zu MS-DOS 6 wurde MS-DOS 6.2 hinsichtlich der Sicherheit und des Bedienungskomforts optimiert. Im folgenden werden die neuen Funktionen von MS-DOS 6.2 vorgestellt. Weitere Informationen über den Funktionsumfang von MS-DOS 6 und MS-DOS 6.2 finden Sie im Kapitel "Willkommen" im MS-DOS 6 Benutzerhandbuch.

Sicherheitsfunktionen und Verbesserungen

Ein neuer Teil von MS-DOS 6.2 ist das neue Dienstprogramm **ScanDisk**. Es dient zur Identifizierung, Diagnose und Korrektur von Datenträgerfehlern auf unkomprimierten Laufwerken sowie auf Laufwerken, die mit DoubleSpace komprimiert wurden. ScanDisk ermöglicht die Korrektur von Systemfehlern (z.B. von querverbundenen Dateien und verlorenen Zuordnungseinheiten) sowie von physischen Datenträgerfehlern. Anhand des von ScanDisk geführten Korrekturprotokolls können die vorgenommenen Änderungen rückgängig gemacht werden. DoubleSpace führt ScanDisk aus, um die Zuverlässigkeit eines Datenträgers vor dessen Komprimierung zu prüfen. Sie können ScanDisk auch selbst starten, indem Sie an der Eingabeaufforderung **SCANDISK** eingeben. Weitere Informationen finden Sie unter **Eine Einführung zu ScanDisk**.

Ebenso neu ist in **DoubleSpace** die **DoubleGuard**-Sicherheitsfunktion. Diese stellt im Rahmen einer Integritätsprüfung den Schutz der Daten vor einer möglichen Beschädigung sicher, die auf den Datenträger geschrieben werden. Sobald DoubleGuard feststellt, daß der von DoubleSpace belegte Speicher von einem anderen Programm beschädigt wurde, wird sofort der Systemabschluß Ihres Computers eingeleitet, damit möglichst wenige Daten Schaden nehmen.

Standardmäßig ist die DoubleGuard-Funktion aktiviert. Sie können sie jedoch deaktivieren, um den Arbeitsspeicher zu entlasten. Verwenden Sie dazu den Befehl **DBLSPACE /DOUBLEGUARD**, oder deaktivieren Sie im Dialogfeld 'Optionen' das Kontrollkästchen für die DoubleGuard-Funktion. (Starten Sie DoubleSpace, und wählen Sie dann aus dem Menü 'Hilfsmittel' den Befehl 'Optionen'.)

Der Manager von MS-DOS für den Erweiterungsspeicher, **HIMEM**, testet beim Starten automatisch den Speicher Ihres Computers. Dieser Test kann nicht mehr zuverlässige Speicher-Chips identifizieren. (Unzuverlässiger Speicher kann zu Systeminstabilitäten und Datenverlust führen.) Um die Speicherüberprüfung auszuschalten, fügen Sie dem Startbefehl für HIMEM die Option **/TESTMEM:OFF** hinzu. Weitere Informationen finden Sie bei **<HIMEM.SYS>**.

Das Installationsprogramm **Setup** konfiguriert **SMARTDrive** jetzt automatisch als schreibgeschützten Cache-Speicher. Sogar nach dem Aktivieren von Schreibzugriffen auf den Cache-Speicher zeigt MS-DOS erst dann die Eingabeaufforderung an, wenn SMARTDrive den Inhalt des Cache-Speichers auf die Festplatte geschrieben hat. Dadurch wird sichergestellt, daß die Daten vor dem Ausschalten des Computers in jedem Fall gespeichert werden.

Bei den Befehlen **MOVE**, **COPY** und **XCOPY** werden Sie jetzt aufgefordert, das Überschreiben von gleichnamigen Dateien beim Kopieren zu bestätigen. (Wenn der Aufruf dieser Befehle aus einer Stapelverarbeitungsdatei heraus erfolgt, werden gleichnamige Dateien allerdings weiterhin ohne Bestätigung überschrieben.)

Bedienungskomfort und weitere Optimierungen

Bei MS-DOS 6.2 können Sie ein **DoubleSpace-Laufwerk problemlos dekomprimieren** oder sogar DoubleSpace vollständig aus dem Speicher entfernen. Weitere Informationen finden Sie bei der Beschreibung des Befehls **<DBLSPACE /UNCOMPRESS>**.

SMARTDrive unterstützt jetzt auch die Cache-Speicherung von CD-ROM-Laufwerken. Um diese Unterstützung zu erhalten, müssen Sie das MSCDEX Programm laden, bevor Sie SMARTDrive laden. Geben Sie SMARTDRV an der Eingabeaufforderung ein, um zu prüfen, ob die Cache-Speicherung Ihres CD-ROM-Laufwerk aktiviert ist. Dies ist der Fall, wenn SMARTDRV den Laufwerkbuchstaben für Ihr CD-ROM-Laufwerk auflistet. Weitere Informationen finden Sie bei der Beschreibung des Befehls **<SMARTDRV>**

DoubleSpace lädt komprimierte Disketten und andere komprimierte Wechselplatten automatisch -- sogar wenn Windows ausgeführt wird. Standardmäßig ist die automatische Ladefunktion (Automount) aktiviert. Um den Arbeitsspeicher zu entlasten, können Sie diese Funktion jedoch deaktivieren. Verwenden Sie dazu den Befehl **<DBLSPACE /AUTOMOUNT>**, oder deaktivieren Sie im Dialogfeld 'Optionen' das Kontrollkästchen für die automatische Ladefunktion. (Starten Sie DoubleSpace, und wählen Sie dann aus dem Menü 'Hilfsmittel' den Befehl 'Optionen'.)

Ab sofort ist es möglich, die **Ausführung einzelner Befehle der Datei AUTOEXEC.BAT** und anderer Stapelverarbeitungsdateien zu **umgehen**. (Bei MS-DOS 6 war dies nur bei der Datei CONFIG.SYS möglich.) Diese Funktion ermöglicht eine vereinfachte Identifizierung von Problemen bei der Fehlersuche in Systemkonfigurationsdateien oder in Stapelverarbeitungsprogrammen.

Um die Ausführung der Befehle in der Datei AUTOEXEC.BAT einzeln auszuführen, drücken Sie beim Starten des Computers die **F8-TASTE**. Weitere Informationen finden Sie unter "Umgehen von Befehlen in den Dateien CONFIG.SYS und AUTOEXEC.BAT" unter dem Thema **<CONFIG.SYS-Befehle>**. Zur schrittweisen Ausführung anderer Stapelverarbeitungsdateien verwenden Sie den Befehl **<COMMAND /Y>**.

Bei Bedarf können Sie beim Starten des Computers **verhindern, daß DoubleSpace geladen wird**. Dieser Schritt erleichtert die Fehlersuche und macht zeitweilig zusätzlichen Arbeitsspeicher für andere Anwendungen verfügbar. Weitere Informationen finden Sie unter "Umgehen von DoubleSpace" unter dem Thema **<CONFIG.SYS-Befehle>**.

Beim Ausführen des Befehls **<DISKCOPY>** werden Daten jetzt auf der Festplatte zwischengespeichert. Das Kopieren von Disketten ist damit noch schneller und einfacher zu bewerkstelligen.

Durch eine verbesserte Belegung des Erweiterungsspeichers können mit dem Microsoft **Defragmentierer** viel größere Datenträger mit weit mehr Dateien und Verzeichnissen defragmentiert werden. Weitere Informationen über das Ausführen des Defragmentierers finden Sie bei der Beschreibung des Befehls **<DEFRAG>**.

Optimiert wurde auch die Lesbarkeit der Ausgabe der Befehle **DIR**, **MEM**, **CHKDSK** und **FORMAT**. Zahlen, die größer als 1000 sind, werden mit 1000er-Trennzeichen dargestellt. Statt der Meldung "1000000 Byte frei" wird jetzt zum Beispiel "1.000.000 Byte frei" angezeigt. □

Hilfethemen von DOS 6.2: ANLSYS · Append · Attrib · Batch-Befehle · Break · Buffers · Call · Cd · Chcp · Chdir · Chkdsk · CHKSTATE.SYS · Choice · Cls · Command · CONFIG.SYS-Befehle · Copy · Country · Ctty · Date · Dblspace · Dblspace-Tips · DBLSPACE.SYS · Debug · Defrag · Del · Deltree · Device · Devicehigh · Dir · Diskcomp · Diskcopy · DISPLAY.SYS · Dos · Doskey · Dosshell · DRIVER.SYS · Drivparm · Echo · Edit · EGA.SYS · Emm386 · EMM386.EXE · Erase · Exit · Expand · Fasthelp · Fastopen · Fc · Fcbs · Fdisk · Files · Find · For · Format · Gerätetreiber · Goto · Graphics · Help · HIMEM.SYS · If · Include · Install · Interlnk · INTERLNK.EXE · InternationaleBefehle · Intersvr · Keyb · Label · Lastdrive · Lh · Loadfix · Loadhigh · Md · Mehrf.-Konfig · Mem · Memmaker · MenuColor · MenuDefault · MenuItem · Mkdir · ModeCommands · More · Move · Msav · Msbackup · Mscdex · Msd · Nlsfunc · Numlock · Path · Pause · Power · POWER.EXE · Print · Prompt · Qbasic · RAMDRIVE.SYS · Rd · Rem · Ren · Rename · Replace · Restore · Rmdir · ScanDisk · Set · Setver · SETVER.EXE · Share · Shell · Shift · SIZER.EXE · Smartdrv · SMARTDRV.EXE · Sort · Stacks · Submenu · Subst · Switches · Sys · Time · Tree · Type · Undelete · Unformat · Ver · Verify · Vol · VSafe · Xcopy.

DOS/WINDOWS für Einsteiger

Teil 1. Hardware-Komponenten

Robert Syrovatka, N, TGM

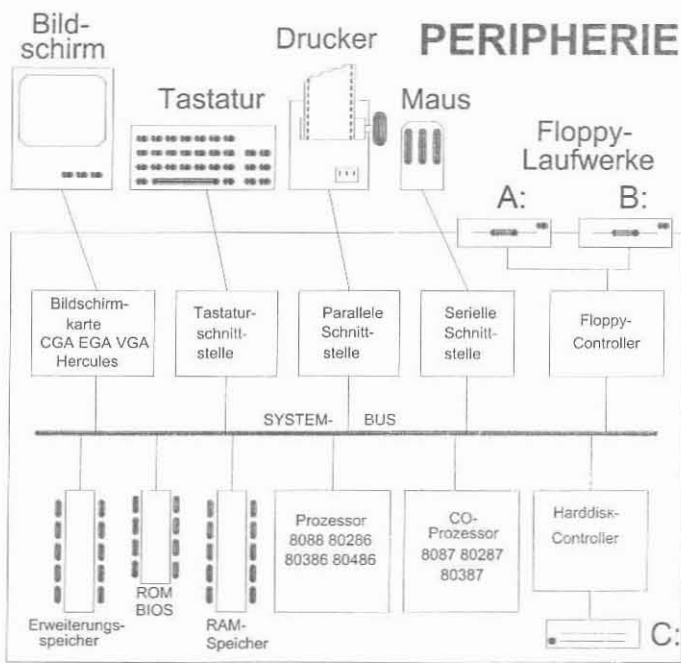
Ein Rechner besteht aus einigen in einem Gehäuse fix eingebauten Hardware-Komponenten und den Peripheriegeräten. Dabei kann die Ausstattung je nach Verwendungszweck unterschiedlich sein.

Mindestanforderungen sind: "Motherboard" (Prozessor, 640 kByte RAM-Speicher, Tastatur-Schnittstelle), Bildschirm- und Floppycontroller- Karte und mindestens ein Laufwerk, an Peripheriegeräten Tastatur und Bildschirm (die "CONSOLE").

Standard-Ausstattung für effizientes Arbeiten sind heute mindestens ein 386-DX Prozessor (eventuell mit Co-Prozessor), eine mind. 170 MB-Festplatte mit Controller, 4 MB Speicher insgesamt, VGA-Bildschirmkarte, ferner Schnittstellenkarte mit Drucker und Maus.

Empfehlenswert: ist heute, wegen des enormen Preisverfalls, ein **486-DLC-Rechner** oder, allerdings etwas teurer, ein **486-DX-Rechner** mit **VESA-LOCALBUS** und der entsprechenden VESA-LOCALBUS -

Controller-, sowie Bildschirmkarte.



Speicher: Programme und das Betriebssystem können nur im konventionellen Speicher, das ist der Bereich bis 640 kByte abgearbeitet werden. Erweiterungsspeicher (oberhalb 640 kB) werden jedoch von vielen Programmen (vor allem WINDOWS) genützt, um Programmteile und Daten auszulagern oder zwischenspeichern. Es kann auch ein VIRTUELLES Laufwerk (RAMDISK) in diesem Bereich eingerichtet werden. Der Zugriff auf Programme und Daten im Erweiterungsspeicher ist um ein Vielfaches schneller, als bei der schnellsten Festplatte. Heutige Boards werden fast ausnahmslos mit **SIMMs** bestückt, von denen mindestens **4 Stück** zu je 1 MBYTE für "eine Bank" erforderlich sind. Bei Bedarf können die restlichen 4 Steckplätze mit 1 MB- oder 4 MB- SIMMs bestückt werden, wobei dann insgesamt 8 MB bzw. 20 MB Speicher zur Verfügung stehen.

1.1 Tastatur

Für die Kommunikation mit dem Rechner dient (neben dem Bildschirm) in erster Linie die Tastatur. Das Tastenfeld ähnelt dem einer Schreibmaschinentastatur, allerdings mit einigen zusätzlichen Funktionstasten und einem "Ziffernblock". Auf der deutschen bzw. englischen Tastatur sowie in vielen Programmbeschreibungen und in den Meldungen des BIOS sind diese Tasten unterschiedlich bezeichnet:

engl	deutsch	Bezeichnungen	Funktion
ESC	UNT	Escape Unterbrechen	Programmsteuerung
BACKSPACE	RÜCK	Rücktaste Back	Zeichen links vom Cursor löschen
TAB	TAB	Tabulator	Cursor springt um einige Zeichen
CAPSLOCK	GROSS	Umschaltartretierung	Großbuchstaben (VORSICHT!!)
SHIFT	UM	Umschalten	Groß/Kleinbuchst./Rückstellg.
CTRL	STRG	Control Steuerung	gleichzeitig mit einer Taste
ALT	ALT	Alternate	gleichzeitig mit einer Taste
ALTGR	ALTGR		gleichzeitig mit einer Taste
SPACE	LEER	Leertaste Zwischenraum-Taste	Leerzeichen
PRINTSCREEN	DRUCK		Bildschirm drucken
SCROLLLOCK	ROLLEN		Bildschirm rollen
BREAK	PAUSE	Unterbrechung	
NUMLOCK	NUM	Num-Artretierung	Ziffernblock EIN/AUS
INS	EINF	Insert Einfügen	Umsch. Einfügen/Überschreiben
DEL	ENTF	Delete Entfernen	Zeichen unter Cursor löschen
HOME	POS1	Home Position1	Cursor an den Zeilenanfang
END	ENDE	Ende	Cursor an das Zeilenende
PG UP	Bild ↑	Page Up Bild auf	Bildschirm 1 Seite aufwärts
PG DN	Bild ↓	Page Down Bild ab	Bildschirm 1 Seite abwärts
← →	← →	Left/ Right Links/Rechts	Cursorsteuerung
↑ ↓	↑ ↓	Up/ Down Auf/Ab	Cursorsteuerung
ENTER	EING	Enter, Eingabe, Return	Ende der (Kommando-)Zeile

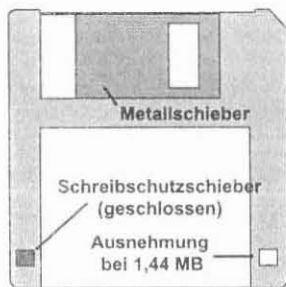
Folgende Zeichen können auch mit gedrückter und niedergehaltener Tastenkombination CTRL+ALT (XT) bzw. ALT GR (AT) und einer Ziffernfolge am Ziffernblock (Zehnertastatur) eingegeben werden (wenn z.B. der deutsche Zeichensatz "keyb gr" noch nicht geladen ist). Bei manchen Tastaturen bzw. Programmen können diese Zeichen nur über die ALT-ZIFFERN-Kombination am Ziffernblock eingegeben werden:

Zeich	Bezeichnungen	Funktion	ALT
*	asterix Stern	für Wildcards	42
/	slash Schrägstrich	für Parameterangaben	47
<	less than	Eingabeumleitung	60
>	greater than	Ausgabeumleitung	62
@	AT SIGN Klammeraffe	Steuerzeichen (in Batch-Dateien)	64
[open bracket eckige Klammer auf	für Programmiersprachen	91
\	Backslash	für Pfadangaben	92
]	closed bracket eckige Klammer zu	für Programmiersprachen	93
{	curved open bracket geschw.Kl. auf	für Programmiersprachen	123
	broken bar	Pipe-Symbol (Filter: more)	124
}	curved closed bracket geschw.Kl. zu	für Programmiersprachen	125

1.2 Diskettenlaufwerke

Jeder Rechner hat mindestens 1 Diskettenlaufwerk. Die Laufwerksbuchstaben A: B: sind immer Diskettenlaufwerke! Es gibt im wesentlichen 2 Diskettenformate - 3 1/2 Zoll und 5 1/4 Zoll mit unterschiedlicher Speicherkapazität:

3 1/2"



720 kB DS-DD, 2DD (Double Side, Double Density)
 1,44 MB DS-HD, 2HD (Double Side, High Density)
 2,88 MB DS-EHD, EHD (Double Side, Extra High Density)

Drehzahl 360 U/min 1,68 m/s 6,03 km/h

3 1/2"-Disketten sind schreibgeschützt, wenn der Schreibschuttschieber offen ist.

Auf 1,44 MB-Laufwerken können 720 kB Disketten beschrieben und gelesen werden, dies gilt natürlich auch für die noch wenig gebräuchlichen 2,88 MB EHD-Laufwerke.

3 1/2"-Disketten sind weniger empfindlich, weil sie in einem stabilen Kunststoffgehäuse eingebaut sind und die Magnetschicht durch einen Metallschieber abgedeckt ist. Dies ist besonders beim Transport (z.B. mit der Post) praktisch. Sie sind allerdings ebenfalls gegen Coca Cola u.a. sehr allergisch!

360 kB DS-DD, 2DD (Double Side, Double Density)
 1,2 MB DS-HD, 2HD (Double Side, High Density)

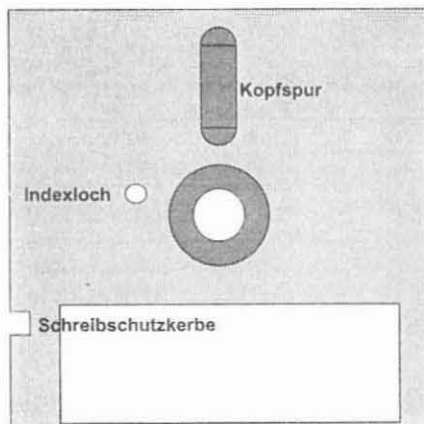
Drehzahl 300 U/min 2,09 m/s 7,54 km/h

5 1/4" Disketten sind schreibgeschützt, wenn die Schreibschutzkerbe überklebt ist.

Auf 1,2 MB-Laufwerken können auch 360 kB Disketten gelesen und beschrieben werden, bei manchen älteren 360 kB-Laufwerken gab es Probleme beim Lesen.

5 1/4"-Disketten immer in der Diskettenhülle aufbewahren. Bei der Arbeit die im Fenster freiliegende Magnetschicht nie berühren oder beschmutzen (Coca Cola etc.) - auf keinen Fall mit Reinigungsmitteln oder Chemikalien säubern. Nicht knicken, nicht mit Kugelschreiber beschriften. Nicht höheren Temperaturen (Sonne, Heizung) aussetzen.

5 1/4"



Trotz Schreibschutzes können Disketten die sich (vor allem bei älteren Laufwerken) im geschlossenen Laufwerk befinden beim Ein- oder Abschalten des Rechners stellenweise überschrieben (besonders kritisch ist Spur 0) und damit unleserlich werden! Daher:

Beim Ein- und Ausschalten des Rechners keine Disketten im geschlossenen Laufwerk lassen! Wenigstens Verriegelung öffnen!

1.3. Festplattenlaufwerke

Festplatten, auch Hard-Disks (früher Winchester) genannt, sind sehr schnelle "Disketten" mit hoher Speicherkapazität (heute sind mind. 150 MB Standard). Die Platten sind in einem staubdicht abgeschlossenen Gehäuse untergebracht, eine Verschmutzung ist daher nicht möglich. Sie sind allerdings sehr empfindlich gegen mechanische Erschütterungen, sowohl im Betrieb als auch im abgeschalteten Zustand.

Während bei Diskettenlaufwerken der Schreib-/Lesekopf auf der flexiblen Magnetscheibe aufliegt, schwebt bei der Festplatte der Kopf in ca. 1/1000 mm Abstand über der starren Magnetplatte. Bei einer Drehzahl von 3600 U/min ergibt dies eine Geschwindigkeit von ca

60 km/h bei 3 1/2" - Festplatten (und sogar 80 km/h bei älteren 5 1/4"-Festplatten). Ein Aufschlagen des Kopfes infolge Erschütterungen führt zu einem "Zerkratzen" der Magnetschicht. Das gleiche Problem tritt (vor allem bei älteren Festplatten) auch beim Ein- und Abschalten des Rechners auf, wobei der Kopf auf der Magnetplatte "startet" oder "landet". Nach einem Zerkratzen dieser Magnetschicht, dem sogenannten "Head-Crash" kann die Festplatte, wenn die Spur 0 zerstört ist, völlig unbrauchbar werden.

Vor dem Abschalten ältere Festplatten immer parken !

Beim Parken werden die Köpfe der Festplatte auf die Parkspur gefahren (meist die höchste Spur oder sogar eine eigene Landspur) und landen dort. Wenn die höchste Spur im Laufe der Zeit beschädigt werden sollte, ergibt dies im schlimmsten Fall eine Verringerung der Speicherkapazität um ca. 1 Promille.

Moderne Festplatten parken allerdings beim Abschalten automatisch!

Festplatten-Typen:

MFM- oder ST506-Festplatten

sind heute kaum mehr gebräuchlich. Charakteristisch hierfür sind zwei Flachbandkabel vom Controller zur Festplatte: Ein breiteres 34-poliges Steuerkabel und ein schmäleres 20-poliges Datenkabel. Es sind maximal 2 Festplatten möglich, sie werden im Setup (bei AT' s) als Hard disk C: und Hard disk D: eingetragen.

AT-BUS- oder IDE-Festplatten

Die heute am meisten gebräuchlichen Festplatten. Sie haben nur ein 40-poliges Flachbandkabel zwischen Controller und Festplatte. Es sind ebenfalls maximal 2 Festplatten möglich, sie werden ebenfalls im Setup (bei AT' s) als Hard disk C: und Hard disk D: eingetragen.

SCSI-Festplatten

Diese Festplatten sind durch ein 50-poliges Flachbandkabel mit einem eigenen SCSI-Controller verbunden. Es können bis zu 8 Festplatten-Streamer- oder sonstige -Geräte an einen SCSI-Controller angeschlossen werden. Die Festplatte wird nicht im SETUP eingetragen! Es sind Festplatten-Kapazitäten bis zu einigen Giga-Byte möglich!

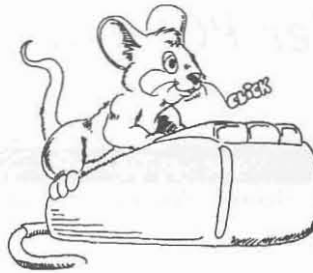
ESDI-Festplatten

sind für hohe Plattenkapazitäten und schnelle Datenübertragung ausgelegt. Sie sind, wie bei MFM-Platten, durch ein 34-poliges Steuerkabel und ein 20-poliges Datenkabel mit dem Controller verbunden. ESDI-Festplatten sind selten und teuer.

AT-BUS- und SCSI-Festplatten können auch gemeinsam nebeneinander in einem Rechner eingebaut sein. MFM- und AT-BUS-Festplatten können hingegen nicht gemeinsam installiert sein!

VESA-LOCAL-BUS-Controllerkarten gibt es sowohl für IDE- als auch für SCSI-Festplatten. Der VESA-LOCAL-BUS gestattet eine vielfach schnellere Übertragung der Daten von und zur Festplatte (aber auch zur Bildschirnkarte) !

ACHTUNG: UNBEDINGT FESTPLATTEN-TYP NOTIEREN und am besten beim Rechner aufheben. Wenn im **SETUP EINE FALSCHER TYPE** eingetragen ist, wird die **FESTPLATTE NICHT MEHR ERKANNT !!!**



1.4 Maus

Als **"Eingabegerät"** ist eine Maus heute unentbehrlich. Bei fast allen Programmen ist es zwar auch möglich, die erforderlichen Funktionen mit Hilfe der Tastatur auszuführen, aber manche Aktionen sind ohne Maus äußerst umständlich. Als **"Microsoft-Maus"** wird sie fast von allen Programmen unterstützt. Die

"Microsoft-Maus" ist eine **"Zwei-Tasten-Maus"**, die meisten Mäuse sind daher umschaltbar (2 oder 3 Tasten). Drei Tasten-Mäuse werden aber nur von sehr wenigen Programmen unterstützt.

Ein geeigneter Maustreiber gestattet allenfalls die Verwendung sowohl als 2- bzw. 3-Tasten-Maus.

1.5 Bildschirm und Bildschirnkarte

MONITOR und **GRAPHIK-KARTE** bilden **eine Einheit**. Heute kommt praktisch nur mehr ein **VGA-Farbschirm** mit **VGA-Graphikkarte** (VGA = Video Graphics Array), in Frage, da bei nahezu allen Programmen die Unterstützung durch die Farbe eine entscheidende Bedeutung hat. Monochrom-Monitore und Monochrome-Karten (Herculeskarte) finden allenfalls noch bei reiner Textverarbeitung Anwendung.

MONITOR: Für eine Bild-Auflösung von 640 x 480 Bildpunkten reicht ein 14-Zoll Monitor, für eine Auflösung von 800 x 600 sollte es besser ein 15-Zoll Monitor, und für 1.024 x 768 Bildpunkte ein 17-Zoll Monitor sein. Der Monitor baut das Bild wie beim Fernsehen **zeilenweise** auf und wiederholt diesen Aufbau mehrmals je Sekunde. Um ein ermüdungsfreies Arbeiten zu garantieren sollte diese Bildwiederholfrequenz mind. **70 Hertz non interlaced** erreichen, was bei den hohen Auflösungen auch eine entsprechend hohe **SCAN-Frequenz** (=Zeilenfrequenz) bedeutet. Viele Monitore schaffen die hohen Auflösungen nur im **interlaced**-Betrieb.

INTERLACED: Es werden bloß 70 **Halbbilder** je Sekunde geschrieben, also tatsächlich eigentlich nur 35 Bilder je Sekunde.

NON INTERLACED: Es werden tatsächlich 70 Bilder je Sekunde geschrieben.

Auflösung	Wiederhol-frequenz	SCAN-Frequenz INTERLACED	SCAN-Frequenz N.INTERLACED	MONITOR TYPE
pixel	Hz	kHz	kHz	Zoll
640 x 480	70	16,80	33,6	14
800 x 600	70	21	42	15
1024 x 768	70	26,88	53,76	17
1280 x 1024	70	35,84	71,68	20
833 x 625	50	15,625	-----	Fern-sehen

Beim **Fernsehen** werden 50 Halbbilder/Sekunde, also **25 Bilder/Sekunde** bei einer Zeilenfrequenz (= "SCAN"-Frequenz) von 15625 Hz und einer Bild- (Wiederhol-)frequenz von 50 Hz geschrieben. Ein Fernsehbild wird allerdings zumeist aus etwas größerem Abstand betrachtet, wodurch ein einigermaßen flimmerfreies Bild erweckt wird.

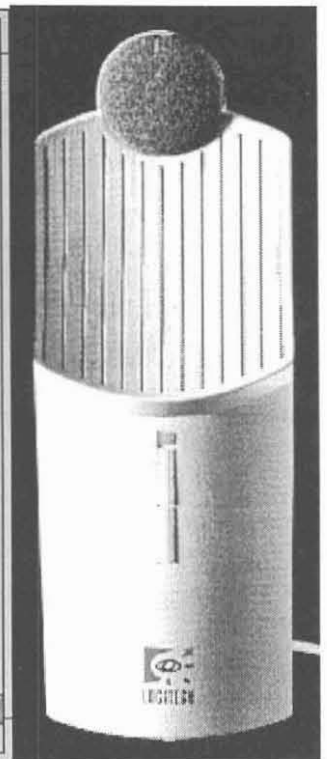
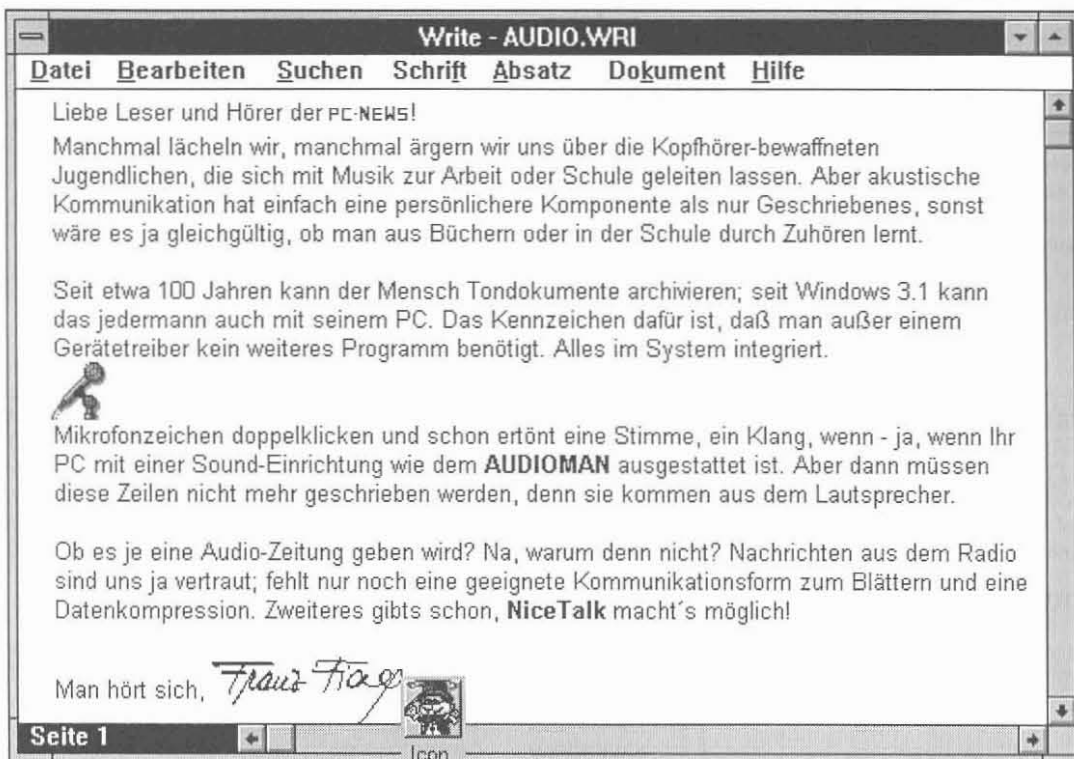
BILDSCHIRNKARTE: Sie muß ebenfalls obige Bedingungen erfüllen, also zum Schirm "passen". Für hohe Auflösung bzw. schnelleren Bildschirmaufbau ist auf der Bildschirnkarte ein eigener Speicher (256 kByte 1 MByte) vorhanden, der meist erweitert werden kann. Die Umschaltung der einzelnen Betriebsmodi erfolgt durch DIP-Schalter(Grundeinstellung) und per Software. Wenn ein VESA-Localbus vorhanden ist, macht eine VESA-LOCAL-BUS-Bildschirnkarte den Bildschirmaufbau rasant! □

Fortsetzung: PC-NEWS-37, Folge 2: MS-DOS-Betriebssystem

AUDIO-PC: der PC, den man hört!

Franz Fiala, N, TGM

PCN-DSK-402



Nur die als Multimedia-PC angepriesenen Geräte kommen mit Audioaufnahme und -wiedergabehardware zur Welt. Alle anderen brauchen Sound-Blaster-, Adlib-, oder andere Karten; oder einfach einen AUDOMAN.

AUDIOMAN

Dieser benötigt keinen Steckplatz und wird über die parallele Schnittstelle betrieben; er hat eine eigene Batterie und braucht bei portablem Betrieb kein Netzteil. 14-tägigen Dauerbetrieb hat der AUDIOMAN ohne Batteriewechsel problemlos überdauert. Es gibt auch keinen Schalter um ihn einzuschalten (oder um das Ausschalten zu vergessen). Der AUDIOMAN schaltet sich selbsttätig ein oder aus.

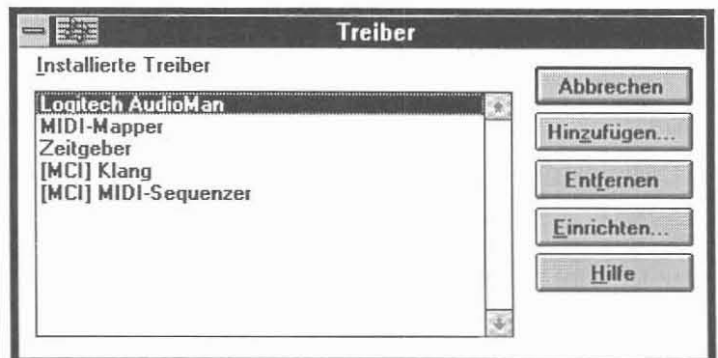
Will man in die Welt des Multimedia einsteigen, stellt man zunächst fest, daß eigentlich gar keiner der PC's, mit denen man es zu tun hat, mit multimedialen Ein- oder Ausgabegeräten ausgestattet ist. Für Versuche von 'Wanderarbeitern' wie Lehrer zu Hause und in der Schule ist der AUDIOMAN ein Hit. Anstecken, Windows hochfahren, fertig! Selbstverständlich muß an jedem Rechner die AUDIOMAN-Software einmal installiert worden sein.

Systemintegration

Windows 3.1 stellt uns über die Systemsteuerung ein selten benutztes Symbol zur Integration zusätzlicher Treiber zur Verfügung. Im PC gibt es nur einen bescheidenen 1-Bit-Lautsprecher, sodaß alle zusätzlichen Hardwareeinrichtungen über das Treiber-Symbol angebunden werden müssen. Andererseits gewährleistet diese späte Erweiterungsmöglichkeit, daß wir nicht auf einer zu frühen Stufe der technologischen Entwicklung stecken bleiben. Die erforderliche Software dazu stammt vom Hersteller der Hardware. Windows sorgt dafür, daß verschiedenste Konstruktionen betrieben werden können. Einige Dateien werden bei der Installation in das Subdirectory übertragen, genau gesagt sind es eine DLL und eine ausdrückbare Datei mit der Endung H11, die ein Programm enthält, das bei Hochfahren von Windows zum AUDIOMAN übertragen werden. Folgende Eintragungen nimmt die AUDIOMAN-Installation in SYSTEM.INI vor:

```

SYSTEM.INI
[drivers]
Wave=audioman.drv
...
[Logitech AudioMan]
Port=LPT1
Volume=5
  
```



Der Treiber für den AUDIOMAN fügt sich in die Liste der vorgegebenen Treiber ein. Man kann ihn auch durch Drücken der Taste EINRICHTEN einrichten. Es erscheint das einzige AUDIOMAN-spezifische Menü:

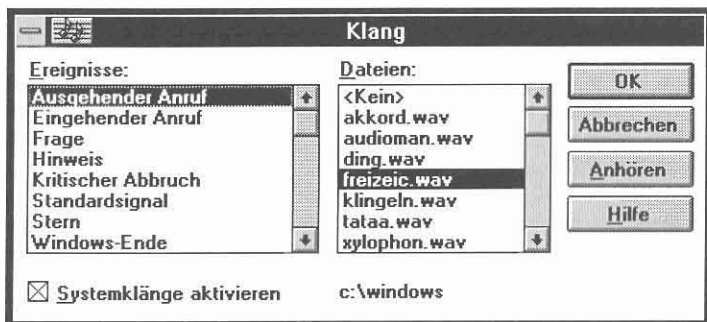
Man erfährt den Zustand der Batterie und kann die Lautstärke in Stufen bis 9 einstellen, was übrigens auch über zwei Taster im Gerätegehäuse möglich ist.



Ein zweites Symbol in der Systemsteuerung ist dafür verantwortlich, welches Geräusch, wann ertönt. In der Abbildung sind die bei Windows mitgelieferten Klänge zu sehen. Mehr an Untermauerung ist kein Problem, denn der AUDIOMAN hat ein integriertes Mikrofon, mit dem die Aufnahme weiterer Texte, Geräusche oder Klänge kein Problem ist. Wenn Umgebungsgeräusche zu vermeiden sind, kann man das eingebaute Mikrofon durch eine direkte elektrische Verbindung zu einer Tonquelle ersetzen.

Welche Klänge gibt es?

Eigentlich gibt es nur einen Begrüßungstext des Vorstandsvorsitzenden von LOGITECH mit ca. 200k, den man aus Platzgründen besser löscht. Warum nicht mehr? Einige Klänge und Geräusche werden bei Windows bereits mitgeliefert, wie Sie weiter im Klang-Fenster unten sehen können, und geräteunabhängig in WAV-Dateien abgelegt werden. Der Betriebssystemhersteller stellt sich vor, daß man bestimmten markanten Ereignissen ebenso markante Geräusche zuordnet. Die Zuordnung kann man im zweiten neuen Symbol "Klang" einstellen:



Diese Zuordnung wird in der Datei WIN.INI so dargestellt:

```
WIN.INI
[Sounds]
RingOut=FREIZEIC.wav,Ausgehender Anruf
RingIn=KLINGELN.wav,Eingehender Anruf
SystemDefault=ding.wav,Standardsignal
SystemExclamation=akkord.wav,Hinweis
SystemStart=tataa.wav,Windows-Start
SystemExit=xylophon.wav,Windows-Ende
SystemHand=akkord.wav,Kritischer Abbruch
SystemQuestion=akkord.wav,Frage
SystemAsterisk=akkord.wav,Stern
```

Wie generiert man Klänge?



Diesmal im Zubehör findet sich das Mikrofonsymbol. In einem einfachen Rekorder mit Vorlauf/Rücklauf/Wiedergabe/Stop und Aufnahme sowie einem Sichtfenster für den Klangverlauf können einfach eigene Texte oder Klänge eingegeben und editiert werden.



Der wichtigste Bearbeitungsvorgang: das Löschen leerer Dateiteile vor dem Beginn und nach dem Ende der Aufzeichnung, sind möglich. Einfache Effekte, wie Lautstärke und Geschwindigkeit verändern, Echo zufügen und verkehrt abspielen sind ebenso wie das Zusammenmischen von Tondateien bereichern das Angebot.

Die so aufgenommene und bearbeitete Datei bekommt einen Namen und kann genauso wie ihre Windows-Kollegen an markanten Stellen abgespielt werden.

Ton in Dokumenten

Dank OLE kann man die erzeugten Ton-Dokumente nicht nur bei Systemereignissen, sondern auch an beliebigen Stellen in Dokumenten einbetten. Das können Text-Dokumente genauso sein wie Access-Datenbanken oder Excel-Diagramme. Als Beispiel dient unser Write-Begrüßungstext auf der vorigen Seite. Die Bearbeitungsfolge, die diesen Bildschirm an der Stelle des Mikrofons erzeugt, ist etwa folgende:

Einfügen-Klang; der bereits gezeigte Klangrecorder-Bildschirm zur Tonaufnahmen erscheint, man kann jetzt eine Klangsequenz aufzeichnen oder eine vorhandene Datei mit Bearbeiten-Einfügen verwenden.

An dieser Stelle findet sich jetzt das Mikrophon-Symbol; klickt man es zwei mal an, kann man die Aufnahme hören.

Man kann sich in Zukunft also zwischen einer README oder einer HEARME-Datei entscheiden, derzeit aber nur, wenn der Kommunikationspartner entsprechend ausgerüstet ist. Dank AUDIOMAN kein wirkliches Problem mehr.

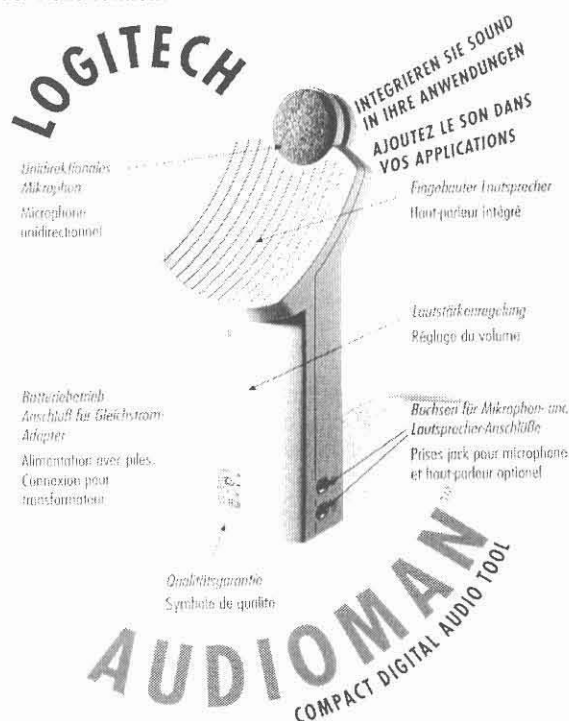
Wie verwaltet man Klänge?

Wie verwaltet man Texte? Texte haben wir bisher in verschiedenen Unterverzeichnissen themengerecht zusammengefaßt. Bei Klängen kann man es genauso tun. Es besteht aber auch die Möglichkeit, ACCESS dafür zu verwenden. Dabei werden nicht einfach nur die Klänge aneinandergerichtet, es können auch die Attribute Dateiname, Länge, Samplefrequenz usw. in der Datenbank mitaufgenommen werden. Die PC-NEWS werden darüber berichten.

Wenn man es auch nur als netten Aufputz betrachten kann; es ist ein bißchen weniger blechern, wenn statt des nervtötenden Piepsens ein Dreiklang ertönt, wobei man es dem Dur-Dreiklang überlassen kann eine OK-Meldung zu bedeuten und einem MOLL-Dreiklang oder eine Dissonanz das Gegenteil.

Aufbau

Im folgenden Bild, gescannt von der Verpackung, sieht man am besten die Art der Konstruktion.



Wer mehr will als nur diese einfachen Zuordnungen und Einbindungen, kann mehr haben. Mitgeliefert mit der AUDIOMAN-Software wird auch eine Light-Version einer ICON-Programmserie, die den Symbolen Leben einhaucht. Die Krönung aber ist es, beliebige ASCII-Texte vorgesprochen zu bekommen. NICE-TALK macht's möglich. Mehr darüber in den nächsten PC-NEWS.

Gekauft bei: PESACO, Triester-Straße 7, 1100 Wien, TEL: (0222) 606 71 76, FAX: (0222) 606 71 77, Preis: S 1.790,-. Für Schüler und Lehrer (Mitgliedskarte bzw. PC-NEWS mitnehmen) gibt es eine Schulversion mit -25% Rabatt.

Support: Fragen aller Art können Sie mit GO LOGITECH im CompuServe-Information System (CIS) stellen. Aktuelle Treiber, speziell Treiber für DOS, die im Originalpaket nicht enthalten waren, wurden von dort geladen und können als DSK-402 über den Diskettendienst bezogen werden.

Hinweis: In einigen der dargestellten Bildschirmbilder findet man in den oberen Rahmen ein kleines Notensymbol. Es bedeutet, daß sich das Öffnen und Schließen des Fensters akustisch mitteilt. Wie man das bewerkstelligt, wird in den nächsten PC-NEWS berichtet. □

Die HP-LaserJet Drucker

Friedrich Pöschko, ASCOM

Überblick

Die Fa. *Hewlett-Packard* (kurz: HP) hat in den letzten Jahren unter den Laser- und den Tintenstrahl-Druckern für "kleine" Computer (PCs, etc.) eine Quasi-Monopolstellung erreicht. HP ist nicht nur Marktführer, sondern setzt seit mehreren Jahren auch die Standards auf diesen beiden Druckersektoren. "HP-Kompatibilität" ist ein Schlagwort für Drucker-Käufer, Software-Produzenten, etc. und gleichzeitig oberstes Ziel für die anderen Druckerproduzenten, deren Produkte nicht selten als keine eigenständigen Entwicklungen, sondern vielmehr als Clones (Nachbauten) von HP-Druckern betrachtet werden müssen.

Der *HP LaserJet IV*, das Flaggschiff der Fa. Hewlett-Packard unter den Laserdruckern, setzt dabei gegenüber den bisherigen Geräten neue Maßstäbe, was Ausgabequalität, Bedienungsfreundlichkeit, Systemoffenheit, etc. anbelangt.

Das Erscheinen dieses Druckers war Anlaß für diesen Artikel.

Grundlagen und Begriffe

Raster, Vektor

Bei Computerausgabegeräten wird zwischen Vektor- und Raster-Geräten unterschieden.

Vektor-Geräte setzen das auszugebende Bild (Text, Grafik) aus Linien, Kurvenzügen, etc. zusammen. Ein Beispiel hierfür ist ein Linien-Plotter (Stiftplotter, Schneidplotter, etc.).

Raster-Geräte setzen das Bild hingegen aus einzelnen Rasterpunkten zusammen. Beispiele sind Monitore und die meisten Drucker, so auch Laserdrucker, die ein Bild aus einzelnen Tonerpartikeln aufbauen.

Text befindet sich im Computer in Form von Zeichenketten, z. B. "ABC". Bilder können in Vektor-Form vorliegen (so z. B. kann eine Linie durch Anfangs- und Endkoordinaten, Dicke, Auszeichnungsart wie etwa strichliert, etc. vollständig beschrieben werden), aber auch als Rasterbild (Anhäufung von Punkten).

Jegliche Vektor-Information, egal, ob sie nun aus Text oder Vektor-Grafik besteht, muß zwecks Drucker-Ausgabe als Rasterbild irgendwann auf dem Ausgabeweg durch eine Abbildungsvorschrift von der Vektor-Darstellungsform in ein Rasterbild übergeführt werden. Bereits in Rasterform vorliegende Information wird meist nicht umgerechnet, sondern direkt ausgegeben, weil das die höchste Qualität liefert.

Der angesprochene Umrechnungsprozeß kann auf verschiedenen Stationen des Weges vorgenommen werden: entweder im Computer (der Drucker erhält dann vom Computer ein fertiges Rasterbild) oder im Drucker (der Drucker erhält eine vektorisierte Beschreibung des auszugebenden Bildes und ein in ihm eingebauter eigener Computer rechnet diese auf Rasterdaten um). Im letzteren Fall müssen Drucker und Computer dieselbe Bild-Beschreibungssprache verstehen, der bekannteste Standard dafür heißt *Postscript*TM.

Bemerkenswert ist, daß das vom Auge als einzelne Punkte wahrgenommene Bild vom Gehirn wieder in die Vektor-Form übergeführt und anschließend interpretiert wird. Die Vektorisierung gelingt dabei dem Gehirn umso einfacher, je höher die Auflösung des Rasterbildes ist. Dabei tritt bei einer gewissen Auflösungshöhe eine Sättigung ein, weil für das Auge zwei nebeneinanderliegende Punkte nicht mehr unterscheidbar sind. Höhere Auflösungen als dieser Grenzwert haben nur dann einen Sinn, wenn Vergrößerungen von der Druckausgabe angefertigt werden sollen. Für einen als normal anzusehenden Leseabstand von 40 cm liegt dieser Grenzwert bei ca. 450 dpi.

Auflösung

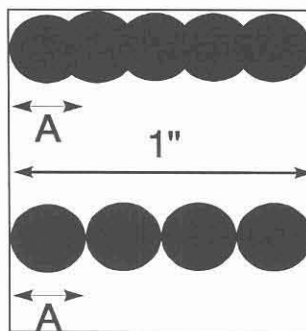
Die Auflösung eines Rasterbildes besagt, aus wievielen Punkten pro Maßeinheit (cm, Zoll, etc.) ein Bild aufgebaut ist. 300 dpi (dots pro inch) etwa bedeutet, daß entlang einer Strecke von einem Zoll maximal 300 (nicht überlappende, siehe unten) Rasterpunkte ausgegeben werden

können. Ab und zu wird auch die Kehrwert-Angabe für die Auflösung verwendet (z. B. 1/100 mm). Streng genommen beschreibt dieser Kehrwert nicht die Auflösung, sondern die minimale Punktgröße eines Rasterpunktes.

Adressierung

Die beiden Begriffe Auflösung und Adressierung werden oft unzulässig vermengt. Ein anschauliches Beispiel ist etwa bei Monitoren gegeben, deren Lochmaske z. B. 600 Zeilen an Leuchtpunkten auf dem Schirm besitzt. Natürlich können oft auch höher adressierte (!) Bilder dargestellt werden, z. B. solche mit 768 Zeilen (dazu muß bei gleich vielen Bildern pro Sekunde nur der Monitor mit der höheren von der Grafikkarte kommenden Zeilenfrequenz fertig werden), aber die Auflösung steigt dennoch nicht über 600 Zeilen. Der optische Effekt durch das nicht-ganzzahlige Abbildungsverhältnis (ideal ist ein Verhältnis von 1:1) resultiert in einer Zunahme der "Unschärfe" des Bildes.

Auch bei Druckern ist zwischen Auflösung und Adressierung zu unterscheiden. Die "Auflösung" endet definitionsgemäß dort, wo sich zwei maximal dicht liegende Rasterpunkte zu überlappen beginnen. Vgl. hierzu die folgenden Grafiken:



Im unteren Bild überlappen sich die Punkte nicht. Adressierung und Auflösung stimmen überein. Im oberen Bild hingegen überlappen sich die Rasterpunkte. Hier werden mehr Punkte adressiert als aufgelöst. Beachte, daß die Anzahl an "Strecken A", die pro Zoll möglich sind, in beiden Bildern die Auflösung in dpi angibt (1/A dpi, wobei hier $A=0.25$ " gilt und somit die Auflösung 4 dpi beträgt), währenddessen die Adressierung in beiden Bildern einen

unterschiedlichen Wert aufweist (im oberen Bild 5 dpi, im unteren 4 dpi).

Sehr oft werden technische Daten von Druckern in Prospekten durch Angabe der Adressierung als "Auflösung" schöngefärbt. Z. B. kann es sein, daß ein Matrix-Drucker etwa 300 dpi adressieren kann, daß aber der Nadeldurchmesser so hoch ist, daß tatsächlich nur etwa 240 dpi aufgelöst werden.

Im folgenden Absatz ist eigentlich anstelle des Wortes "Auflösung" das Wort "Adressierung" zu verwenden. Nachdem sich die Vermengung beider Begriffe im täglichen Sprachgebrauch derart eingebürgert hat, wollen wir die strenge Unterscheidung bleiben lassen und nur noch von "Auflösung" sprechen:

Nachdem immer Flächen bedruckt, Auflösungen aber eindimensional angegeben werden, ist zu beachten, daß die Datenmenge von Rasterbildern mit dem Quadrat ihrer Auflösung wächst, so etwa steigt sie bei einer Verdoppelung der Auflösung auf das Vierfache. Eine oft vorliegende Zahl für Raster-Datenmengen sollte man sich in diesem Zusammenhang für Überschlagsrechnungen merken: Eine A4-Seite, welche mit einer Auflösung von 300dpi auf einer Fläche von 8" mal 11" mit Schwarzweiß-Information bedruckt wird, benötigt einen Platz von 990.000 kB, also rund 1 MB. Bei 600dpi steigt diese Menge auf grob 4 MB.

Auszeichnung

Auszeichnungsarten eines Texts sind: fett, kursiv, unterstrichen, etc.

Schriftstil

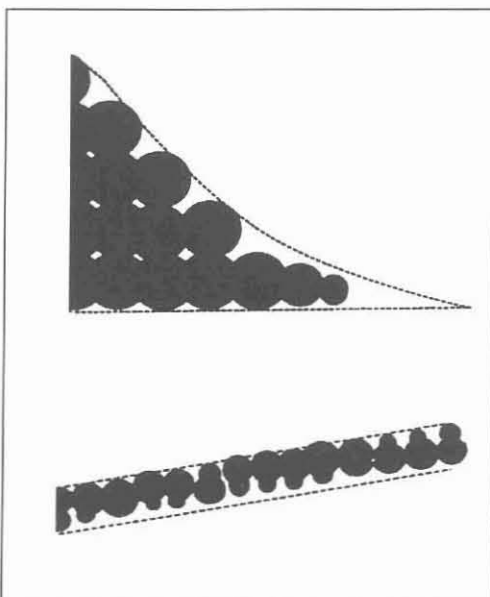
z. B. Times, Helvetica, etc. Schriftstile werden nach verschiedensten Kriterien zu "Schriftfamilien" zusammengefaßt.

Spezielle Features der LaserJet III- und IV-Drucker

RET

RET steht für *Resolution Enhancement Technology*. Es handelt sich dabei um ein Verfahren, das die Auflösung von Rasterbildern bei gleicher Adressierung verbessern (steigern) kann. RET arbeitet mit unterschiedlichen Größen von Rasterpunkten. Diese unterschiedlichen Größen werden hauptsächlich durch die Steuerung der Zeitdauer, in welcher der Laser auf der Walze Ladung aufbringt, erzeugt, wodurch größere und kleinere geladene Kreisscheiben auf der Walze entstehen, in denen dann Tonerpartikel haften bleiben. Dadurch ergeben sich unterschiedliche Punktgrößen auf dem Papier. Für dieses Verfahren wurde von HP ein eigener Toner mit geringerer Korngröße als vorher üblich entwickelt ("HP Micro Toner").

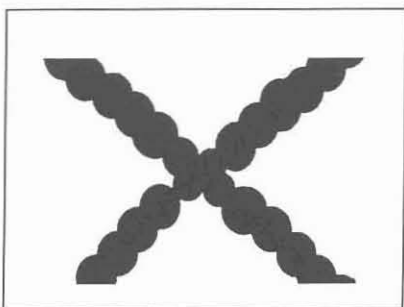
Die Möglichkeit, unterschiedlich große Punkte auf dem Papier zu erzeugen, wird nun folgendermaßen zur Erzielung einer höheren Auflösung verwendet: Ein Algorithmus zur Umrißbildung (engl. *Image Processing*) berechnet alle im Druckbild enthaltenen Umrißlinien. Die schwarzen Flächen innerhalb der Umrisse werden dann mit unterschiedlich großen Rasterpunkten dergestalt gefüllt, daß sich schärfere Kanten oder weniger stufige Linien ergeben (siehe die folgenden Bilder):



Das erste Bild zeigt das Einfügen kleinerer Punkte zwischen großen an kurvigen Umrisen, damit Kurvenverläufe glatter (weniger ausgefranst) wirken; ferner das Verwenden kleinerer Punkte an Kanten und Ecken, die dadurch nicht so zackig wirken. Das zweite Bild zeigt den Stufeneffekt, der bei geringen Winkel zur Horizontale oder zur Vertikale geneigt sind, am stärksten

zutage tritt, und seine optische Verringerung durch RET.

RET behebt auch durch die Drucktechnologie bedingte Darstellungsfehler; so etwa wird bei sich überschneidenden Linien die Kreuzungsstelle auf der Walze doppelt vom Laserstrahl getroffen und dadurch stärker geladen als jeder andere Punkt der Linie. Daher bleibt an dieser Stelle mehr Toner haften, wodurch sich eine unerwünschte Ansammlung von Tonerpulver und optisch eine Verdickung der Linien an der Kreuzungsstelle auf dem Papier ergibt. RET erkennt dies und kompensiert diesen Effekt durch Verwendung kleinerer Punkte an dieser Stelle, wie in der nachfolgenden Grafik sichtbar ist.



Der große Vorteil von RET besteht darin: Dokumente können mit RET nun mit einer höheren Auflösung gedruckt werden (die minimal verwendete Punktgröße sinkt ja), wobei allerdings computerseitig die Adressierung gleich bleibt (der RET-Algorithmus erhöht sie für den

Computer unsichtbar durch das Einfügen kleiner Punkte im Druckbild). Als Quintessenz können also daher z. B. frühere mit 300 dpi erstellte Dokumente nun etwa mit "echten" 450 dpi gedruckt werden, wobei alle Bilder gleich groß, alle Zeilenumbrüche gleich etc. bleiben! Es ist also keine Umformatierung von bestehenden Dokumenten nötig.

Intellifont™

Intellifont™ ist ein Typographiesystem, das aus der professionellen Satz- und Drucktechnik kommt. Text wird dabei nicht mehr durch Zeichensätze in Rasterinformation dargestellt (wodurch für jeden Schriftstil, jede Größe, jede Auszeichnungsart etc. eine eigene Zeichensatz-Datei erforderlich ist), sondern durch eine vektorisierte Beschreibung (Umriß, Füllung). Damit können sowohl im Drucker integrierte (*Hardfonts*) als auch in Dateiform in den Drucker ladbare Schriften (*Softfonts*) in beliebigen Größen und Formen (z. B. mit einem Punktraster gefüllt, mit einem Grauton hinterlegt, ungefüllt, etc.) ausgegeben werden, wobei nur mehr eine einzige Zeichensatz-Datei pro Auszeichnungsart und Schriftstil benötigt wird. So kann für einfache Anwendungen grob schon mit etwa 6 Font-Dateien (z. B. Times und Helvetica, jeweils normal, fett, kursiv) das Auslangen gefunden werden.

Die unterschiedlichen Punktgrößen werden nun nicht einfach durch Multiplikation mit einem konstanten Faktor erzeugt; Versuche haben ergeben, daß besonders kleine Schriften dann schlecht lesbar wären. Diese erfahren vielmehr eine Sonderbehandlung durch die sogenannten *hints* (engl.: Hinweise). So z. B. muß bei einer kleineren Schrift der Größenunterschied von Klein- zu Großbuchstaben geringer oder die Strichstärke dicker gewählt werden, um dieselbe Lesbarkeit bzw. denselben optischen Effekt zu erzeugen.

Postscript™

Postscript™ ist eine vektororientierte Beschreibungssprache für Bilder (die Texte und Grafiken enthalten können). Die Beschreibung eines Bildes durch Vektoren hat mehrere Vorteile: immer ist die Auflösungsunabhängigkeit gegeben; das Bild kann, je nach Auflösung des Ausgabegerätes, in maximaler Qualität dargestellt werden. Oft (aber nicht zwingenderweise) benötigt ein Bild in Vektorform auch weniger Platz als in Rasterform mit akzeptabler Auflösung; das ist sowohl für die Speicherung als auch für die Übertragung günstig.

Ein Beispiel: Ein Quadrat mit Kantenlänge 5 cm kann etwa durch vier Linien (Angabe der Koordinaten, Dicke der Linien etc.) mit wenigen Zeichen (Zahlenwerten, Bytes etc.) beschrieben werden; dasselbe Rechteck mit 600 dpi auf einer A4-Seite ausgegeben, benötigt incl. der weißen Ränder 4MB Platz.

Postscript™ wird zur Ausgabe auf verschiedensten Ausgabegeräten verwendet (Satzmaschinen etc., sogar schon Monitore, siehe NEXT™) und hat zahlreiche Befürworter gefunden. So etwa kann die Flut der für eine Softwareapplikation nötigen Treiber durch Postscript™ wirksam eingedämmt werden; von einem einzigen Postscript™-Treiber werden zahlreiche Druckermodelle erfaßt.

Unterschied zwischen PCL und Postscript™

Postscript™ ist eine wortähnliche Beschreibungssprache; so etwa wird eine Linie durch ein Befehlswort wie "LINE", gefolgt von Koordinatenwerten, ebenfalls im Klartext, etc. bestimmt. Hingegen wird bei PCL5 codiert; anstatt eines langen Postscript™-Befehlswortes genügen bei PCL meist 3 Byte, etwa, um ein bestimmtes Füllmuster zu bezeichnen. Raster-Grafiken werden zudem bei PCL in einem binären datenkomprimierten Format zeitsparend übertragen. Bei Text hat hingegen keines der beiden Verfahren entscheidende Vorteile; seit der Version PCL5 enthält Intellifont™ und ist damit wie Postscript™ eine Umriß-Beschreibungssprache.

Beim PCL-Druck wird die Umrechnung von Vektor auf Raster vom druckenden Computer vorgenommen, beim Postscript™-Druck vom Drucker. Je nach Art des auszugebenden Bildes und den Ressourcen von Computer bzw. Drucker kann die eine oder die andere Art der Ausgabe schneller sein.

Geschichte

HP-Laserdrucker werden üblicherweise von Befehlen in der sogenannten "Printer Control Language" (PCL) gesteuert. Der Sprachumfang wird dauernd erweitert; neue Laserdrucker "verstehen" PCLs höherer Versionsnummer als alte. Diese enthalten den Sprachumfang einer PCL niedrigerer Versionsnummer als Untermenge, sodaß eine bleibende Aufwärtskompatibilität gegeben ist. Zur Zeit der Abfassung dieses Artikels (LaserJet IV) hält HP bei der Version 5e+ (das "e" steht darin für "Extended").

Der Übergang von PCL4 auf PCL5, der sich mit der Einführung des LaserJet III ergab, stellt einen Quantensprung in der Drucktechnologie dar. PCL5 enthält nämlich Intellifont™ (siehe oben). Mit dem LaserJet IV wurde mit RET ein weiteres qualitätssteigerndes Feature eingeführt.

Auch die Basisausstattung mit Ressourcen (Speicher, Zeichensätzen etc.) stieg ständig. So wird etwa der LaserJet 4M mit einem minimalen Speicher von 6MB RAM und standardmäßig eingebautem Postscript™-Interpreter angeboten, der Drucker "verstehet" somit die Sprachen PCL5e, HP/GL2 und Postscript™ 2.0. Die Anzahl eingebauter Zeichensätze beträgt 115 (80 Intellifont™ und 35 Postscript™ - Zeichensätze).

Sowohl die erwähnte Politik der Kompatibilität als auch diese technischen Merkmale haben HP zum Trendsetter und Marktführer bei PC-Laserdruckern werden lassen.

Wertung

HP setzt seit mehreren Jahren die Standards auf dem Laserdruckersektor, "HP-Kompatibilität" ist nicht nur ein Schlagwort, sondern ein wichtiges Kriterium auf dem Laserdrucker-Markt. Diese Quasi-Monopolstellung von HP hat ihre Gründe.

Zum Ersten ist sie einer intelligenten Firmenpolitik zu verdanken: neue HP-Drucker sind immer aufwärtskompatibel, d. h. man kauft den vorhergehenden Drucker sozusagen mit dem neuen mit und verliert bei

der Umstellung keine Daten (alle Dokumente sehen, auf dem neuen Drucker gedruckt, genauso aus wie auf dem alten, so etwa ändert sich der Zeilenumbruch nicht oder Grafiken bleiben gleich groß, etc.). Viele alte Software-Applikationen z. B. haben nur einen Druckertreiber für einen HP LaserJet II - das ist für spätere HP-Laserdrucker kein Problem, ein "II" ist im III und im IV sozusagen mit eingebaut. Dieses Argument wird auch in Zukunft stichhaltig bleiben, denn selbst bei der Verwendung von Postscript™ sind Kompatibilitätsprobleme nicht restlos auszuschließen.

Zweitens kommt ein Preisargument dazu: Seit das Unternehmen HP seine Drucker in Österreich auf mehreren Vertriebswegen an den Kunden bringt (sowohl direkt als auch über Fachhändler genauso wie über Elektronik-Märkte), sind HP-Drucker trotz des Markennamens bei gleicher Leistung nicht wesentlich teurer als "Kompatible".

Drittens ruht sich HP nicht etwa auf derzeit geltenden Standards aus, sondern entwickelt bestehende ständig weiter und versteht es, die eigenen Weiterentwicklungen zu Standards zu machen.

Die von HP mit den Modellen LaserJet III und IV in den PC-Laserdruckermarkt eingeführten Verfahren zur Steigerung der Druckqualität (Intellifont™, RET, Erhöhung der Auflösung auf 600 dpi) setzen gegenwärtig einen neuen Standard, an dem sich die Konkurrenz bereits zu orientieren beginnt, wovon zahlreiche Clones zeugen. Der LaserJet IV (ohne Zusatz) ist für den Heimbereich heute wohl noch etwas zu teuer, abgespeckte Versionen wie den IV L kann man guten Gewissens heute bereits für den durchschnittlichen Heimanwender empfehlen.

Ausblick

Die Überlegenheit von Marktführern gegenüber der Konkurrenz tendiert zur Zementierung, so auch die von HP auf dem PC-Laser- und Tintenstrahl-Druckersektor. In den nächsten Jahren wird, sollte nicht ein revolutionär anderes Druckverfahren von einem Konkurrenten eingeführt werden, "HP-Kompatibilität" auf diesem Sektor ein wichtiges Argument bleiben oder sogar ein noch wichtigeres werden.

Testbericht HP LaserJet 4

Friedrich Pöschko, ASCOM

Dieser Artikel befaßt sich mit dem "Grundgerät" HP LaserJet IV (ohne Zusatz in der Typenbezeichnung). Alle anderen Geräte der Baureihe IV stellen Weiterentwicklungen dar, die entweder Zusätze aufweisen oder im Gegenteil abgespeckte Versionen des IV sind (siehe Typenübersicht der LaserJet-Drucker im Anschluß an diesen Artikel).

Druckqualität

Der LaserJet IV bietet echte 600 dpi, kann aber auch mit 300 dpi gesendete Seiten optisch verbessern (durch den eingebauten RET-Algorithmus). Dieses Feature ist in 4 Stufen einstellbar (RET aus; RET mit leichter/mittlerer/kraftiger Umriß-Korrektur).

Die Ausdrücke des Geräts auf verschiedensten Ausgabemedien (Papier, Folie, etc.) sind als hochqualitativ zu bezeichnen und genügen nicht nur Heim- sondern auch Korrespondenz-, Büro- und kleinemengigen Verlagsansprüchen. Einzig für hochqualitative Vergrößerungen von Druckausgaben reichen die 600 dpi nicht aus.

Bei 300 dpi-Druckern - auch den besten - kann wohl von jedem mit freiem Auge eine Zackigkeit im Schriftbild wahrgenommen werden (was auch physikalisch herleitbar ist; bei als normal anzusehendem Leseabstand von 40 cm ist das bei Normalsichtigen noch bis zu ca. 450 dpi der Fall). 300 dpi-Ausgaben (z. B. von älteren Dokumenten) werden durch RET im Schnitt auf 450-500 dpi verbessert, sodaß man über diesem Grenzwert liegt; neue Dokumente wird man wohl ohnehin mit 600 dpi drucken. Es sei angemerkt, daß es einen 1200 dpi-Enhancement-Kit zum LaserJet IV gibt, der Gegenstand eines eigenen Artikels im Anschluß ist.

Papierqualität

Die verwendeten Papiersorten spielen wenig Rolle, es kann auch handelsübliches Fotokopierpapier minderer Qualität eingesetzt werden; Konkurrenzgeräte neigen oft dazu, dünnes Papier zu wellen (entweder

durch die Verwindung des Papiers auf dem Druckweg oder durch zu starke Durchnässung mit Toner). Einzig bei sehr rauhem Papier und kleinen Schriftgraden (< 8 Punkt) zeigen sich geringfügige Ausrisse im Druckbild.

Das Druckgut (Papier, Karton, Etiketten) darf max. 135 g/m² schwer sein (das ist in etwa schwerer Zeichenkarton) und max. 0,18 mm dick (das reicht für praktisch alle Selbstklebe-Etiketten). Die Minimalgröße (damit ein Druckgut-Transport möglich ist) beträgt 9 cm x 16 cm, die Maximalgröße 21,6 cm x 35,6 cm. Die tatsächlich bedruckbare Fläche ist nur wenig geringer, sie liegt bei 20,7 cm x 34,5 cm. Damit werden alle wichtigen Papierformate, auch ausländische (US Letter, US Legal) erfaßt.

Handling

Auch dieses ist sehr angenehm und ausgesprochen unproblematisch. Ausschub-Seiten entstanden immer nur durch menschliche Fehlbedienung, nie etwa gab es Papierstaus, Papier-Zerknittern, schiefes Einziehen, Wellen des Papiers bei schwarzen Flächen, etc.. Ein Umstand, der nicht nur Zeit und Papier (sprich: Geld) spart, sondern sicher auch vom umweltschützerischen Standpunkt aus zu begrüßen ist.

Für Umweltschutz-Freaks sei weiters gesagt: Der Drucker produziert extrem wenig Ozon; weiters wird der LaserJet IV wie alle anderen HP-Laserdrucker vom HP-Tonerkassetten-Recycling-Programm erfaßt. Toner und Walze sind eine Einheit (Cartridge); als Resultat ergibt sich für den Anwender ein sehr einfacher Tonerwechsel garantiert ohne schwarze Finger. Die Walze, die etwa die zehnfache Standzeit eines Tonerzyklus bringt, wird dabei von HP immer wieder verwendet, bis die vorgesehene Abnutzung erreicht ist; die Cartridge wird während ihrer Lebensdauer nur mit Tonerpulver wiederbefüllt.

Das Druckgut wird entweder aus einem Multifunktions-Einzug (der Druckgut verschiedener Formate aufnehmen kann, z. B. A4-Bögen und Folien genauso wie DIN-C6-Kuverts), auch Mehrzweck-Kassette genannt, oder aus einer eingebauten Kassette für 250 Blatt A4-Papier entnommen, zusätzlich kann eine 500 Blatt fassende, gegen Aufpreis erhältliche, Kassette montiert werden. Die Programmierung, aus welcher Kassette Druckgut entnommen wird, ist voll softwareunterstützt.

Möglichkeiten

Diese sind vielfältig. Man kann etwa einen Serienbrief schreiben, bei dem das Gerät vollautomatisch abwechselnd ein Kuvert aus der Mehrzweckkassette entnimmt (und sich das Papierformat auf DIN lang und die Ausdrucksrichtung auf quer stellt) und danach 4 Bögen A4 für den eigentlichen Brief aus der Papierkassette (Papierformat A4, Ausdrucksrichtung Hochformat).

Die meisten Funktionen können sowohl am Bedienpanel (das übrigens austauschbar ist und zusätzlich zur englischen Standardausgabe in der Landessprache mitgeliefert wird), als auch per Software (ESC-Sequenzen) als auch über ein Fernbedienfeld (Simulation des Druckerpanels auf dem Schirm des Computers) eingestellt werden. Die Fernbedienfeld-Software wird mit dem Drucker geliefert.

Es besteht die Möglichkeit, vollkommen frei programmierbare Makros in den Drucker zu laden (diese können nicht nur Befehlsanweisungen und Kombinationen davon (z. B. fett, Schriftstil Helvetica, 12 Punkt,) umfassen, sondern ganze Zeichensätze oder sogar Grafiken. Dieses Feature empfiehlt sich für immer wiederkehrende Aufgaben bei Vorliegen größerer Datenmengen (wie etwa Briefkopf-Firmenlogos bei Serienbriefen, etc.).

Technisches

Der Speicher ist bis zu 34 MB RAM ausbaubar. Das Gerät kommt mit einer Standardausstattung von 2 MB, in 4 SIMM-Sockel können entweder 1, 2, 4, oder 8-MB-SIMMs eingesetzt werden, sodaß maximal eben 34 MB möglich sind (2 MB on board und 4 Stk. 8MB-SIMMs). Es finden keine "normalen" SIMMs (30-polige Bauform) Verwendung, sondern sogenannte "Big SIMMs" (auch PS/2-SIMMs genannt, 72-polig).

Ein Postscript™-Modul kostet einen SIMM-Platz und daher sind bei Einsatz eines solchen nur noch 26 MB möglich, womit für normale Anwendungszwecke wohl auch das Auslangen gefunden werden kann.

Bei Einsatz von Postscript™ braucht man mindestens 6MB, d. h. es sind dann mindestens 4 MB nachzurüsten.

An sich ergibt sich mit dem Standard-Speicherausbau von 2 MB ein Widerspruch - wie kann ein Drucker mit 600 dpi eine komplette A4-Seite in seinem Speicher aufbauen, wo die doch einen Speicher von 4MByte braucht ?

Durch ein intelligentes Feature (Kompressionsalgorithmus) kommt man trotzdem für die meisten Seiten mit den standardmäßig eingebauten 2MB durch, die Seite wird "komprimiert" im Druckerspeicher aufgebaut, wodurch bei hohen Kompressionsfaktoren sogar mehrere Seiten im Druckerspeicher Platz haben.

Mit einem Speicherausbau von 6MB ist alles machbar, mehr Speicher dient dann nur noch allfälligem Zeitgewinn.

Bei der Angabe der Druckleistung hat sich HP wohl auf die sichere Seite begeben: angegeben werden 8 Seiten/min., der LJ4 des Autors schafft bis zu 8,7 (das klingt vielleicht haarspaltiger, macht jedoch beim Ausdruck von 1100 Ausfertigungen eines 4seitigen Serienbriefs einen Unterschied von immerhin 45 Minuten aus).

Das Gerät hat 45 Schriften (35 Intellifont™- und 10 TrueType™-Fonts), bei Nachrüsten des Postscript™-Moduls erhält man weitere 35 Postscript™-Fonts, weitere Fonts können entweder in Form von Schriftkassetten (Frontschlitz) oder Steckmodulen (kostet einen SIMM-Steckplatz) eingesetzt werden. Darüberhinaus besteht noch die Möglichkeit, so viele Softfonts in den Drucker zu laden, wie es der Speicherausbau zuläßt.

Das Postscript™-Modul unterstützt den Level 2 dieser Sprache, der abwärtskompatibel zu allen niedrigeren Postscript™-Leveln ist.

Die Lebensdauer einer Tonerfüllung wird mit 6000 Seiten bei einem mittleren Schwärzungsgrad einer Seite von 5% (dieser Wert ist relevant für reinen Textdruck) angegeben. Bei einem Preis von ca. öS 1500,- für

eine Toner cartridge kommt man somit auf einen Seitenpreis (nur die Tonerkosten berücksichtigt) von 0,25 öS.

Kompatibilität

Im Unterschied zu den zahllosen auf dem Markt befindlichen Konkurrenzgeräten (die laut Verkäufer natürlich 100% HP-kompatibel sind, was sich in praxi später meist als nicht ganz wahr herausstellt) erwirbt man mit dem LJ4 einen Original-HP-Drucker und vermeidet Kompatibilitätsprobleme von vornherein.

Einen Treiber für einen HP LaserJet (sei es II, III oder IV) gibt es wohl in jedem Programm auf jeder Maschine unter jedem Betriebssystem, dann kommt PCL zum Einsatz; wenn nicht, kann auf HP/GL bzw. HP/GL2 ausgewichen werden; wenn auch dieser Standard von der Software nicht unterstützt wird, dann gibt es zumindest einen Postscript™-Treiber.

Der Umgang mit verschiedenen Datenformaten und Druckersprachen, der beim Einsatz in einer Mehr-Betriebssystem-Umgebung notwendig ist (PCL, Postscript™, HP/GL, ...) gestaltet sich in der Praxis sehr unkompliziert: der Drucker besitzt einen "AUTO"-Modus, in dem er die gesendeten Daten zu interpretieren und auf die Sprache rückzuschließen versucht, was praktisch immer gelingt.

Zahlreiche Anschlußmöglichkeiten stehen zur Verfügung: Parallel (Bi-Tronics), Seriell (RS-232, RS-422), optional Netzwerk-Anbindungen (Thin-Ethernet und 10BaseT-Ethernet, unter Novell™ NetWare™, EtherTalk™, Local Talk™, TCP/IP™, etc.). Mit "Bi-Tronics" bezeichnet HP eine Abart der bekannten Centronics-Schnittstelle, die im Unterschied zu dieser nur bidirektional ausgelegt ist. Über die ansonsten als Statusleitungen fungierenden und zu Datenleitungen umfunktionierten Leitungen wie PaperEmpty, etc. kann der Drucker Nachrichten an den Computer senden. Dadurch kann der Computer Statusinformationen des Druckers abrufen und nicht nur genauere Fehlermeldungen geben (Papierstau, etc.) sondern z. B. auch die Drucker settings wie gegenwärtige Schriftgröße etc. am Bildschirm anzeigen.

Auch bei der Programmierung der Druckersoftware hat HP an einen Einsatz im Netzwerk gedacht: Es ist möglich, für jeden Netzwerkbenutzer ein eigenes Drucker-Setting (Standardzeichensatz, Papierformat, etc.) im Drucker zu speichern. Welchem Netzwerkbenutzer ist es noch nicht passiert, daß ein Job an den Drucker aufgegeben und wertvolle Zeit und wertvolles Papier verschwendet wurde, weil mittlerweile ein anderer Netzwerkbenutzer den Drucker umkonfiguriert hat ? Bis jetzt mußte jeder Netzwerkbenutzer, wollte er darin sichergehen, eine komplette Konfiguration seiner Druckerumgebung in den Drucker laden und den Zugriff anderer Benutzer bis zur Beendigung seines Druckjobs sperren. Mit einem LaserJet IV kann das vermieden werden - jeder User sendet, sobald seine gewünschte Umgebung dem Drucker bekannt ist, seine Kennung zum Drucker, und schon ist der Drucker "sein gewohnter".

Nach Meinung des Autors macht sich der Mehrpreis von Original-HP-Druckern gegenüber Clones im Laufe des Druckerlebens jedenfalls bezahlt - vor allem dann, wenn der Drucker Postscript™ nicht unterstützt und/oder Softwareapplikationen unter verschiedenen Betriebssystemen zum Einsatz kommen. Der Autor spricht aus leidvoller Erfahrung.

Abschließend ist zu sagen, daß der LaserJet IV der erste von insgesamt 5 Laserdruckern in der Karriere des Autors ist, mit dem die Arbeit als reines Vergnügen bezeichnet werden kann.

Quellen

- PC Professionell, Ausg. 9/1991 S. 166 f.
- Werbebroschüren der Fa. Hewlett-Packard zu LaserJet III und IV
- Katalog der Fa. MIDAS-Computer, Wien (HP-Vertragshändler)
- Handbuch zum LaserJet IV
- Skriptum zur Vorlesung "Graphische Datenverarbeitung", Dipl.-Ing. Johann Weiss (Autor), Fa. Sysgraph/Wien, Institut für Computertechnik der TU-Wien (Hrsg.)

Danksagungen

Der Autor dankt den folgenden Firmen bzw. deren Exponenten für ihre freundliche Unterstützung beim Zustandekommen dieses Artikels:

- ASCOM Computerzubehör, Wien
- Hewlett-Packard Austria
- MIDAS Computer, Wien
- PC-Gourmet, Wien

Warenzeichen

- Adobe™ und Postscript™ sind ein Warenzeichen von Adobe Systems, Inc.
- Intellifont™ ist ein Warenzeichen der Fa. Agfa Compugraphic
- TrueType™, LocalTalk, EtherTalk und Macintosh sind Warenzeichen der Fa. Apple Computer Inc.
- NetWare™ und Novell™ sind Warenzeichen der Fa. Novell Inc.

Alle anderen, im Text nicht gekennzeichneten, Warenzeichen sind solche der Fa. Hewlett-Packard. □

PCL versus Postscript -Ein Test

Friedrich Pöschko, ASCOM

DSK-404\HPLJ-4\BILDSCH.CDR, DRUCKER.WRI

Wie bereits erwähnt, wird unter PCL vornehmlich Rasterinformation vom Computer zum Drucker geschrieben, d. h. das Bitmap-Bild vom Computer berechnet und an den Drucker geschickt, der davon einfach ein 1:1-Abbild auf Papier herstellt.

Seit der Version 5 von PCL bilden in den Drucker ladbare Umriß-Schriften (Intellifont™-Schriften) eine Ausnahme davon. Dieses Feature muß jedoch nicht unbedingt benutzt werden, für gewöhnlich wird auch Textinformation als Bitmap-Grafik gesendet.

Postscript™ hingegen ist eine vektororientierte Sprache; das Druckbild wird in der vektoriellen Beschreibungsform zum Drucker gesendet und dieser - genauer: ein Postscript™-Interpreter darin - nimmt die Abbildung auf Raster vor.

Mit beiden Sprachen können grob dieselben Ergebnisse erzielt werden, wengleich gesagt werden muß, daß eingebaute Postscript™-Schriften speziell für ein Druckermodell optimiert werden und daher die weitaus beste Ausgabequalität liefern. Bei hoher Auflösung (600 dpi) und normal großen Schriftgraden (> 9 Punkt) ist der Qualitätsunterschied jedoch kaum zu bemerken.

Abhängig von der Bildinformation können sowohl Datenmenge als auch Druckzeit zwischen den beiden Verfahren stark schwanken, wobei paradoxerweise niedrige Datenmenge nicht zwangsläufig auch geringe Druckzeit bedeutet.

Meist ist die Datenmenge egal; der Benutzer ist vielmehr an einer Optimierung der Druckzeit interessiert. Es empfiehlt sich dazu, genauer zu überlegen, welche Komponenten beteiligt sind und wo denn genau Druckzeit "verbraten" wird.

Eine volle A4-Seite mit 300dpi-Auflösung ist als Rastergrafik ca. 1MB groß (siehe vorne). Dieselbe Seite in 600 dpi-Auflösung belegt schon 4MB. Unabhängig vom Inhalt! Enthält die Seite etwa einen einzigen Strich diagonal über die ganze Seite, so belegt die entsprechende Postscript™-Datei hingegen nur einige hundert Byte (!). Nur bei bereits in Rasterform vorliegenden Grafiken wird auch bei Postscript™ Rasterinformation übertragen.

Nur bei sehr komplexen Bildern kann ein Bild in Rasterform weniger Platz belegen als in Vektorform. Solche Bilder treten - außer auf speziellen Gebieten - i. Allg. mit einer Wahrscheinlichkeit von weniger als 1-5% auf. Rein vom Platzverbrauch (Speicherung, Übertragung) ist daher meist Postscript™ der Vorzug zu geben.

Mit der geringeren Datenmenge geht jedoch nicht zwangsläufig eine Druckzeitverminderung einher.

Sendet man eine Rastergrafik an den Drucker, so muß der Computer die Seite in seinem Speicher mit der Druckerauflösung aufbauen (und evtl. dabei aufgrund der Datenmenge in einem zeitraubenden Vorgang auf Festplatte zwischenspeichern) und dann über das Nadelöhr Centronics-Schnittstelle schicken. Der Drucker muß nur die Daten übernehmen (in seinen internen Speicher kopieren), sie zum Druck aus seinem Speicher holen und auf die Walze schreiben. Der Drucker hat dabei kaum Rechenarbeit zu erledigen, die Geschwindigkeit des Druckvorgangs wird wesentlich von der Geschwindigkeit des Computers und der Daetübertragung über die Schnittstelle beeinflusst.

Im Unterschied dazu ist der Umsetzungsprozeß bei Postscript™ ein zweifacher: zunächst setzt der Computer das i. Allg. in Vektorform vorliegende Bild in Vektor-Sprache (Postscript™) um, danach wird diese Anweisungsdatei über die Schnittstelle übertragen. Erst der Drucker setzt die Daten in Rasterinformation um.

Von entscheidender Bedeutung für die Gesamtdruckzeit sind dabei folgende Parameter:

- die Rechengeschwindigkeit des Computers;
- die Geschwindigkeit der Datenübertragung über die Schnittstelle;
- die Menge der übertragenen Daten;
- die Rechengeschwindigkeit des Druckers.

Bei PCL ist die Menge an übertragenen Daten (nicht an Information !) in zahlreichen Fällen wesentlich höher als bei Postscript™. Daher kommt bei PCL dem letzten Punkt so gut wie keine Bedeutung zu, der Drucker muß die gesendete Information nur noch einfach auf Papier abbilden. Computer und Schnittstelle bestimmen wesentlich die Druckzeit.

Bei Postscript™-Ausgaben ist die Menge an übertragenen Daten meist niedriger als bei PCL; der wesentliche Zeitfaktor ist hier der letzte Punkt, die Berechnung des Rasterbildes aus den gesendeten Postscript™-Informationen durch den Drucker.

Nachdem die Geschwindigkeit der Datenübertragung über die meist verwendete CENTRONICS-Schnittstelle nur selten signifikant gesteigert werden kann, ist, kurz gesagt, bei Raster-Ausgabe ein schneller Computer anzuraten, bei Postscript™ Ausgabe hingegen ein schneller Postscript™-Interpreter im Drucker.

Ein Beispiel: Hat man etwa eine Schreibmaschinen-Seite Text zu drucken, so sind das ca. 5kB ASCII-Text plus etwas Header und evtl. etwas Soft-Font-Daten; alles zusammen höchstens einige zig kB. Das ist wesentlich schneller übertragbar als dieselbe Seite mit 300 dpi in Rastergrafik (1MB).

In Postscript™-Druckern der niedrigen Preisklassen kommen nicht selten langsame 8 und 16 Bit-Prozessoren (meist irgendwelche 8051-Derivate) zum Einsatz. Der LaserJet IV verwendet hingegen einen schnellen RISC-Prozessor und ist daher bei Postscript™-Ausgabe den meisten anderen Druckern sogar dort überlegen, wo diese anderen Drucker bei Rasterausgabe schneller sind.

Der am schwersten einzuschätzende Faktor in der Zeitrechnung ist der Computer. So kann etwa ein 386DX mit 16MB RAM unter Windows 3.1 ein ziemlich schneller Rechner sein (wenn wenig Tasks laufen und das Druckbild zur Gänze im RAM aufgebaut werden kann), sogar schneller als ein 486DX mit weniger RAM. Aber auch sehr langsam, wenn z. B. viele Tasks laufen und somit die Druckdaten auf Platte ausgelagert werden müssen.

Der folgende Absatz zeigt Meßergebnisse für 2 Files, die einmal unter Raster (PCL) und einmal unter Postscript™ ausgegeben wurden. Das erste File (BILDSCH.CDR) ist eine mit dem Programm CORELDRAW! 3.0 unter Windows 3.1 erstellte Vektor-Grafik, das zweite File ein 14-seitiger Text ohne Grafik im MS-Write-Format (DRUCKER.WRI). Das erste File liegt dem deutschen CORELDRAW! 3.0 bei, das andere File liegt dem deutschen MS-Windows 3.1.

Der Testrechner war ein 386DX40 mit folgender Ausstattung: 16 MB RAM, davon 2MB SmartDrive; der Drucker ein LaserJet IV mit Postscript™-Modul und 6 MB RAM.

Es ergaben sich folgende Ausdruckzeiten (handgestoppt, Angaben in min:sek):

Datei BILDSCH.CDR

Druckmethode	Rechenzeit PC	Rechenzeit LJ IV	Rechenzeit Gesamt
Postscript™	0:24	0:51	1:15
Raster 600 dpi	0:50	0:40	1:30

Bemerkung: 128 Graustufen

Datei DRUCKER.WRI

Druckmethode	Datenmenge (kB)	Rechenzeit Gesamt	Bemerkung
Postscript™	75	2:14	
Raster 600 dpi	44	2:16	Print TrueType™ as TrueType™
Raster 600 dpi	2.250	4:12	Print TrueType™ as Graphics

Interpretation:

Aufgrund des hohen zur Verfügung stehenden Speichers mußte bei keinem einzigen Druckvorgang auf Platte ausgelagert werden.

Bei der Vektorgrafik ergeben sich kaum Unterschiede in der Totalzeit. Bei Postscript™ ist zwar der Rechner - wie erwartet - schneller fertig, der Drucker rechnet aber dafür auch länger.

Interessant ist die Textdatei. Damit sich signifikante Zahlen ergaben, wurde absichtlich ein so langes Dokument (14 Seiten) gewählt. Außerdem eines, das 2 im LaserJet IV eingebaute TrueType-Fonts enthält.

Wie man sieht, gibt es kaum Unterschiede zwischen Postscript™ und dem Intellifont™ benutzenden PCL5e des LaserJet IV. Das eine benutzt die eingebauten Postscript-Fonts™ und das andere die eingebauten TrueType™-Font des Druckers. Wird dagegen auf die im Drucker eingebauten TrueType™-Font verzichtet und wird der Text als Rastergrafik übertragen (wie es bei nicht eingebauten Font oder bei einer PCL-Version kleiner als PCL5 zwangsweise gemacht werden müßte), dann ist Postscript™ dem Rasterdruck eindeutig überlegen.

Typenübersicht, technische Spezifikationen der LaserJet-Drucker III und IV

LaserJet Modell	Seiten/min	Ausgabe format	dpi	Sprache(n)	Interface	Features	Speicher Standard bis max. (MB)	Bemerkung
3P	4	A4	300	PCL5	Centronics, RS-232		1-5	
3Si	16	A4 doppelseitig	300	PCL5	Centronics, RS-232		2-17	"Double"
4L	4	A4	300	PCL5, HP/GL2	Centronics, RS-232	RET	1-?	"Light"
4LM	4	A4	300	PCL5, HP/GL2	Centronics, RS-232, Apple Local Talk	RET	4-?	"Light Macintosh"
4P	4	A4	600	PCL5, HP/GL2	Centronics, RS-232	RET	2-?	
4MP	4	A4	600	PCL5, HP/GL2	Centronics, RS-232	RET	6-?	"Light"
4	8	A4	600	PCL5e, HP/GL2	Bi-Tronics, RS-232, RS- 422	RET	2-34	"Standard"
4M	8	A4	600	PCL5e+, HPGL/2, PS	Bi-Tronics, RS-232, RS- 422, Apple Local Talk	RET	6-22	"Macintosh"
4Si	16	A4 doppelseitig	600	PCL5e+, HPGL/2	Bi-Tronics, RS-232, RS- 422	RET	4-34	"Double"
4Si-MX	16	A4 doppelseitig	600	PCL5e+, HPGL/2, PS	Bi-Tronics, RS-232, RS- 422, Apple Local Talk	RET	6-22	"Double Macintosh"

□

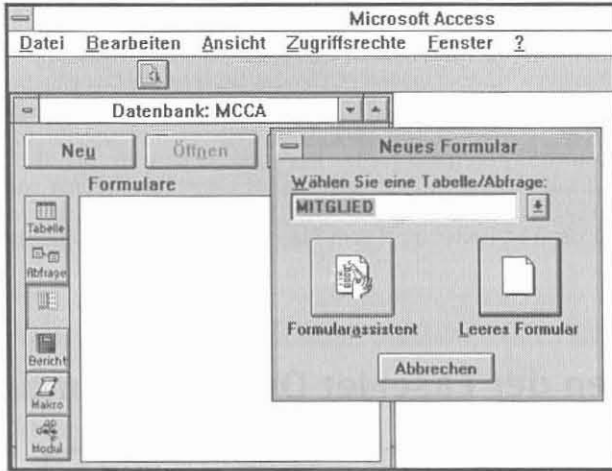
ACCESS, Eine Datenbank unter Windows

Folge 2

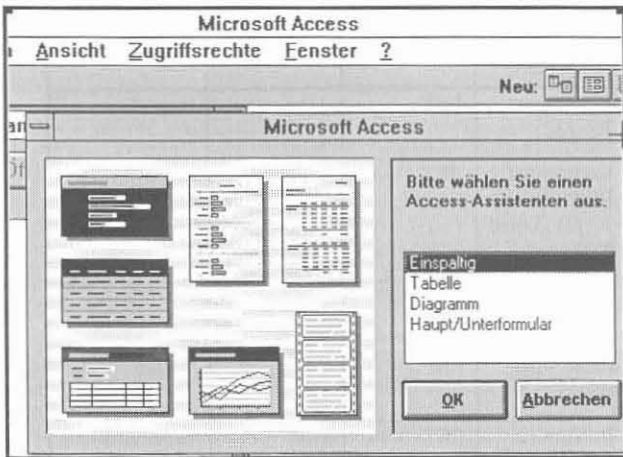
Helmuth Schlögl, MCCA

DSK-395, 396, 397

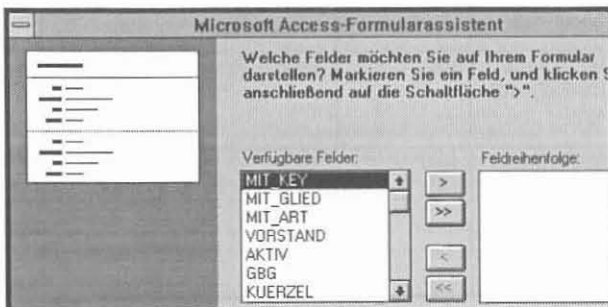
In der ersten Folge haben wir die Tabelle "Mitglied" erstellt. Diese enthält nur die Struktur. Um sie komfortabel mit Daten füllen zu können, brauchen wir eine Eingabemaske (Formular).



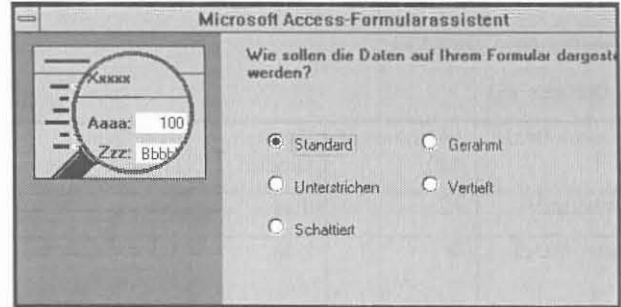
Wir drücken die Schaltfläche "Formular" und dann "Neu". Es bieten sich nun zwei Möglichkeiten an. Ein leeres Formular, wo man die einzelnen Felder selbst arrangieren kann, oder ein Formularassistent, der viel Arbeit abnimmt. Wir verwenden den Assistenten und wählen als Tabelle "Mitglied" und starten ihn.



In einer Auswahlbox können wir uns anschaulich für ein Layout unseres Formulars entscheiden. Einspaltig wollen wir es gestalten, was nicht besagt, daß wir unsere Felder nicht nach unseren Wünschen nachher verschieben können.



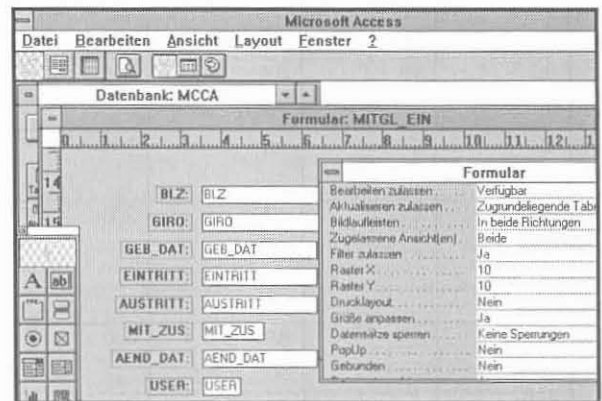
Jedes einzelne Feld kann mit ">" übernommen bzw. mit "<" rückgesetzt werden. Analog dazu werden alle Felder mit ">>" übernommen und mit "<<" zurückgesetzt.



Die letzte Gestaltungsmöglichkeit, die uns noch bleibt, ist reine Geschmacksache. Wir können aber nachträglich noch einiges nachholen.

So schaut nun ein Teil unseres Formulars in der Rohfassung aus, in der schon einige Felder durch einfaches Verschieben (Feld mit der linken Maustaste anklicken, halten und verschieben) auf die vorgesehene Position gebracht worden sind.

Um die Anpassung der Felder beim Verschieben im Formular besser vornehmen zu können, verwenden wir ein größeres Rastermaß. Der höchste Feinheitsgrad ist 64 Einheiten in der X- und 64 in der Y-Achse. Durch Klicken auf das weiße Formularkästchen und auf das, mit Pfeil markierte Symbol, werden die Formulareigenschaften sichtbar gemacht.



Ein Feld im Formular besteht immer aus einem Textfeld und dem Inhalt des Feldes. Beide sind fest miteinander verbunden. Beim Verschieben werden durch Mausklick und Halten der linken Maustaste immer beide verschoben. Will man sie von einander trennen muß man z.B. das Textfeld in der linken oberen Ecke fassen (Handsymbol) und positionieren. Ist es notwendig mehrere Felder gemeinsam zu verschieben, zieht man mit gedrückter linker Maustaste ein Rechteck auf. Damit sind alle gewünschten Felder markiert und lassen sich gemeinsam verschieben. Wie in Windowsanwendungen können auch Felder mit gedrückter Shifttaste einzeln markiert werden. Zum Ausrichten und Anpassen der Größe sind unter dem Menüpunkt "Layout" Werkzeuge vorgesehen.

Wie Sie sicherlich bemerkt haben, besteht ein Formular aus mehreren Bereichen.

Formularkopf: Hier ist der Titel plaziert. In diesem Teil kann man Felder positionieren, die der Auswahl eines Datensatzes dienen. (Suchkriterien, Datumsfelder etc.)

Detailbereich: Die veränderbaren Daten (Datensätze) befinden sich in diesem Bereich.

Formularfuß: In diesem Bereich werden wir Schaltflächen für die Steuerung unseres Formulars installieren.



Mit der "Toolleiste" ganz links werden eine Reihe von Zusatzmöglichkeiten wie:

Text, Rechenfeld, Schaltflächen, Listboxen, Kombifelder, gebundene und ungebundene Felder, Statistikfelder, Schalter etc. für die Gestaltung des Formulars verwendbar. Eine Palette ermöglicht das Färben des Hintergrundes, der Texte, der Feld-Rahmen, den 3-D Effekt von Feldern etc.

So könnte das Formular in "fast" fertigem Zustand aussehen. Hätte man Fotos der einzelnen Clubmitglieder könnte man diese einscannen und in die Datenbank einbinden.



Sicher ist es für den Anfang ein zu grober Überblick über die Gestaltungsmöglichkeiten eines Formulars. In der nächsten Folge möchte ich daher noch einiges mehr darüber berichten. Außerdem werden wir "Abfrage" von Daten, speziell für die Suche von Datensätzen kennen lernen. □

Anstandsregeln

Heinrich Wenzel, 2:313/7

Allgemeine Anstandsregeln

§1 Gescheitheit

Ein guter Lehrer darf, muß jedoch nicht gescheit sein. Sollte er gescheit sein, was allerdings in den seltensten Fällen der Fall ist, so darf er auf keinen Fall diese ihm von Gott gegebene Gescheitheit vor den Schülern zeigen, um ihnen das Gefühl der Überlegenheit zu geben. Entdeckt also ein Schüler einen Fehler in einer Äußerung des Lehrers, so muß dieser (falls es ihm sein schauspielerisches Talent ermöglicht) eine möglichst dumme Miene aufsetzen und möglichst verdattert die Fragen "Wos? Wie? Won? Wo?" stellen. Tut er dies möglichst oft - er kann dies desto öfter tun, je öfter er einen Fehler macht, also sehr oft - so werden ihn die Schüler bald für eine absolute Null halten, womit er sein Ziel erreicht hätte.

§2 Prüfungsrecht

Ein guter Lehrer darf prüfen sooft und soviel er will, solange er nur solche Schüler prüft, die seine Fragen auch mit hundertprozentiger Sicherheit beantworten können. Sein Blick muß durch die Menge der Schüler schweifen und jene Schüler übergehen, die seinem Blick nicht standhalten können oder rot werden, wenn er sie ansieht, also jene Schüler, die mit totaler Unwissenheit die Klasse betreten haben. Er muß mehrere schwere Fragen an die Klasse richten und nach jeder Frage seinen Blick umherschweifen lassen. Sollte es wider Erwarten einen Schüler geben, der sich bei keiner der sehr schweren Fragen hinter seinem Nachbar versteckt hat, so darf der Lehrer diesen prüfen, womit er sein Ziel erreicht hat: Notendurchschnitt 1!

§3 Freizeit

Ein guter Lehrer muß immer den Eindruck erwecken, als hätte er sehr viel zu tun, was aber nicht der Fall ist. Es ist jedoch nötig, diesen Eindruck zu hinterlassen, um von seinen Schülern ernstgenommen zu werden. Hierbei muß der Lehrer höchst mathematisch vorgehen: Er muß behaupten, in einer Woche so wenig Freizeit zu haben, wie er tatsächlich an einem einzigen, höchst beschäftigten Tag hat, also in der Regel 7-8 Stunden. Auf eine Woche umgelegt hat also ein Lehrer lediglich eine freie Stunde am Tag, obwohl er in Wirklichkeit 7-8 Stunden am Tag frei hat, das allerdings nur, wenn es an diesem Tag wirklich heiß hergeht. Weiters muß

der Lehrer immer über die vielen Maturaarbeiten klagen, die er vorzubereiten und zu verbessern hat, auch wenn er nur an einer Volksschule unterrichtet. Tut er das, hat er sein Ziel erreicht: Er muntert damit die Schüler auf, soviel für die Schule zu tun, wie auch er für die Schule tut, also sehr viel.

Anstandsregeln für das weibliche Geschlecht

§4a Aussehen

Eine gute weibliche Lehrerin, und die meisten Lehrerinnen sind weiblich, hat jung zu sein und gut auszusehen. Ist sie nicht jung und/oder sieht sie nicht gut aus, muß sie wenigstens den Eindruck erwecken, jung zu sein und gut auszusehen. Sie muß weiters die Schüler mit lieblichen Blicken verwöhnen, nicht die weiblichen, aber sehr wohl die männlichen, aber die meisten Schüler sind männlich, im Gegensatz zu den Schülerinnen, von denen ein erstaunlich großer Prozentsatz dem weiblichen Geschlecht angehört. Sie muß auch sehr nett sein, vor allem zu den männlichen Schülern, darf ihnen also keine schweren oder besser gar keine Fragen stellen. Tut sie das, dann hat sie ihr Ziel erreicht: Die Schüler werden sich fortan auf die wahren Vorzüge des Lebens konzentrieren: Auf die holde Weiblichkeit! Vor allem die männlichen, die weiblichen weniger.

§5a Parfümerie

Eine gute weibliche Lehrerin darf Parfüm verwenden, aber nicht in einem übertriebenen Maße, wie es jedoch manche Lehrerinnen sehr wohl praktizieren. Es ist also nicht nötig, daß eine Lehrerin nach einem verunglückten Parfümtanker riecht. Dieses Ziel kann die Lehrerin sehr leicht dadurch erreichen, indem sie ihre tägliche Parfümdusche einstellt und fortan einfaches, ordinäres Wasser zu ihrer Körperpflege verwendet. Damit hat sie ihr Ziel erreicht: Die von ihr verbesserten Hausübungen riechen nicht mehr nach einem Liebesbrief, den eine geistig minderbemittelte 13-Jährige ihrem 15-jährigen Angebeteten schickt.

Anstandsregeln für das männliche Geschlecht

§4b Schreierei

Ein guter männlicher Lehrer darf schreien. Er darf sogar sehr laut schreien. Es wird ihm auch gewährt, während der Schreierei ausfallend zu werden oder mit der Faust auf den Tisch zu schlagen. Er darf dies alles, aber nur, wenn er mit seiner Frau schreit. Ein guter Lehrer darf aber nicht mit seinen Schülern schreien, nicht einmal seine Stimme heben darf er. Statt dessen muß er psychologisch viel wirksamere Methoden einsetzen, wie zum Beispiel das fünfmalige Schreibenlassen der sechzehner-Lektion aus dem Imperium Romanum. Tut er das, dann wird er sehr viel Respekt von der Klasse ernten und sicher nicht als Schreihals, es sei denn, von seiner Frau, bezeichnet werden. Möchte er von seiner Frau nicht als Schreihals bezeichnet werden, dann kann er ja auch sie das sechzehner-Kapitel aus dem Imperium Romanum fünfmal, oder besser zehnmal, schreiben lassen.

§5b Bevorteiligung

Ein guter männlicher Lehrer darf keine seiner Schüler bevorteiligen, vor allem nicht die weiblichen, also die Schülerinnen. Bei den männlichen Schülern macht das nicht so viel aus, denn die männlichen Schüler haben in der Regel eine stärkere Psyche als die weiblichen Schülerinnen, werden also diese Bevorteiligung viel leichter verkraften. Wohl aber darf ein Lehrer eine weibliche Schülerin benachteiligen, weil sich das nicht so stark auf die Psyche auswirkt. Der Lehrer hat also sehr wohl die Wahl der Bevor- bzw. Benachteiligung. Er kann es sich tatsächlich aussuchen, ob er die männlichen Schüler bevorteiligt, oder die weiblichen Schülerinnen benachteiligen möchte. Weder die männlichen noch die weiblichen Schüler noch die Schülerinnen ziehen eine der beiden Methoden vor. Der Lehrer hat aber sein Ziel erreicht: Die vollkommene Überlegenheit des männlichen Geschlechts, sowohl in körperlicher als auch in geistiger Hinsicht.

Struktogrammgenerator

Franz Fiala, N, TGM

DSK-378, 379

Liebe EDV-geplagte Schüler- und LehrerInnen!

Ein übersichtliches Struktogramm ist ein Ausdrucksmittel mit dem der EDV-Fachmann auch mit Laien über Programmläufe diskutieren kann. Dennoch sind sie bei den Schülern unbeliebt, weil das händische Zeichnen oft unflexibler ist als das Editieren des Codes selbst.

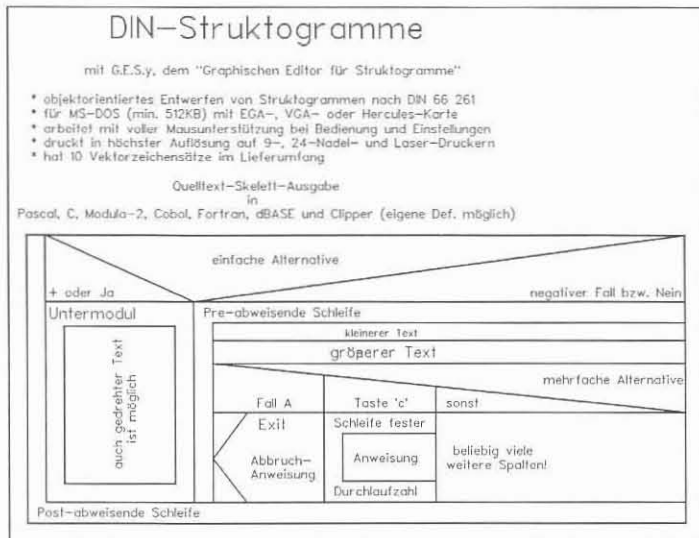
Auf der Suche nach geeigneten Hilfen stößt man zunächst auf den ABC-Flowcharter von Micrografix, der aber Flußdiagramme und nicht Struktogramme zeichnen kann; und Flußdiagramme sind ein Ausdrucksmittel, das man beim Programmwurf vermeiden will.

Ich habe eingangs ein Struktogramm als Kommunikationsmittel zwischen Programmierer und Anwender erwähnt.

Struktogramme sind aber für den Lernenden auch ein Hilfsmittel zum Verständnis der Abläufe in Rechnerprogrammen. Wenn Struktogramme auch bei Programmierern weniger beliebt sind (wie mir einige kritische Leser dieses Beitrags berichteten), als didaktisches Hilfsmittel hat es den Charakter eines Bildschirmspiels bei dem unbemerkt Lerninhalte transportiert werden.

Neulich fand ich ein preiswertes DOS-Programm, über das ich hier berichten möchte und dessen Kauf sich für Schulen, Schüler und Lehrer, die mit Programmierung zu tun haben, lohnt:

G.E.S.y. V2.1 ist ein DOS-Programm mit Windows-ähnlicher Oberfläche, mit dem alle Strukturelemente mit drag-drop-Technik in den Entwurf mit entsprechenden Kommentaren eingebunden werden können. Dabei stellen sich die einzelnen Blöcke selbsttätig auf die Textlänge ein; das Einfügen einer Bedingung im letzten Moment wird zum Kinderspiel. Folgende Elemente kennt der Struktogrammgenerator (nicht abgebildet: Parallelverarbeitung):



Nur in DOS?

Wenn man es als Nachteil empfindet, daß das Programm nicht in einer Windows-Version vorliegt: man kann den Output in PCX, TIFF, PostScript oder HPGL exportieren, sodaß die Weiterverarbeitung der Grafik auch in Windows problemlos möglich ist.

Bezugsquelle:

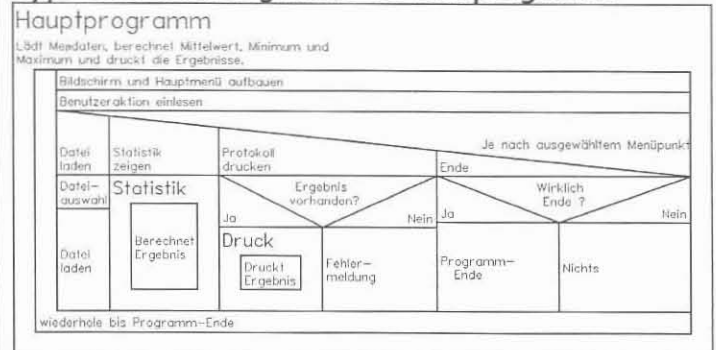
SIP-Software-Lösungen
Griesäckerstrasse 15
D-96117 Memmelsdorf
TEL:0951/43489, FAX:0951/420514
Preis: DM 200,- (VISA/EUROCARD)

Schutz

Das Programmpaket kann jeder benutzen, es installiert sich als DEMO-Version mit folgenden Einschränkungen: Maximal zwei-zeiliger Kommentar und nicht mehr als vier gleichartige Blöcke hintereinander. [5 Anweisungsblöcke hintereinander geht nicht, dagegen können ineinander verschachtelte Strukturen, wie Schleifen, Entscheidungen usw. gut geprobt werden]. Zum Lernen und für einfache Dinge ausreichend, zum Arbeiten zu wenig. Erst mit einer persönlichen Kennung, die man einfach als ASCII-Text in eine Datei schreibt, die aber ein daraus berechnetes Schutzwort enthält, wird es eine lizenzierte Version ohne Einschränkungen. *Anmerkung: Die folgenden Struktogramme und alle Struktogramme des Beitrags "Hardwarenahe Programmierung mit C" wurden mit der Demo-Version erstellt.*

Daher: Wer das Programm kennenlernen möchte, der kann die Diskette über die Mailbox **His Master's Voice** bestellen oder downloaden. Die Kosten des Programm sind gering im Vergleich zum didaktischen Wert.

Typisches Struktogramm für Meßprogramm



Templates

Besonders hübsch ist, daß man aus dem Struktogramm ein Programmgerüst ableiten kann (Templates für C, PASCAL, COBOL, BASIC, REXX oder dBASE sind vorhanden). Die jeweiligen Schlüsselwörter sind in reinen ASCII-Dateien abgelegt, man kann die entstehenden Programmsequenzen auch an andere Sprachen selbst anpassen. Ein komplettes Beispiel sehen Sie auf der folgenden Seite.

Fortsetzung

Und was ist mit den vielen, bis jetzt undokumentierten Programmen? Auch dafür gibt eine Lösung, allerdings kostet die DM 600,-: Dieses Programm generiert aus einem vorhandenen Listing ein Struktogramm. Einige Schüler planen schon einen Kauf!

Real programmers don't write specs. Users should consider themselves lucky to get any programs at all and take what they get.

Real programmers don't comment their code. If it was hard to write, it should be hard to read.

Real programmers don't write application programs, they program right down on the bare metal. Application programming is for feebs who can't do systems programming.

Real programmers don't eat quiche. **Real programmers** don't even know how to spell quiche. They eat Twinkies, Coke and palate-scorching Szechwan food.

Real programmers don't draw flowcharts. Flowcharts are, after all, the illiterate's form of documentation. Cavemen drew flowcharts; look how much it did for them.

Real programmers don't read manuals. Reliance on a reference is a hallmark of the novice and the coward.

Templates

Ein Template ist ein Programmgerüst, das man aus einem Struktogramm für viele Sprachen ableiten kann.

Zunächst muß eine Anweisungsdatei (Beispiel für ANSI-C) vorhanden sein, die die Anpassung an jede Sprache und an jeden persönlichen Schreibstil erlaubt:

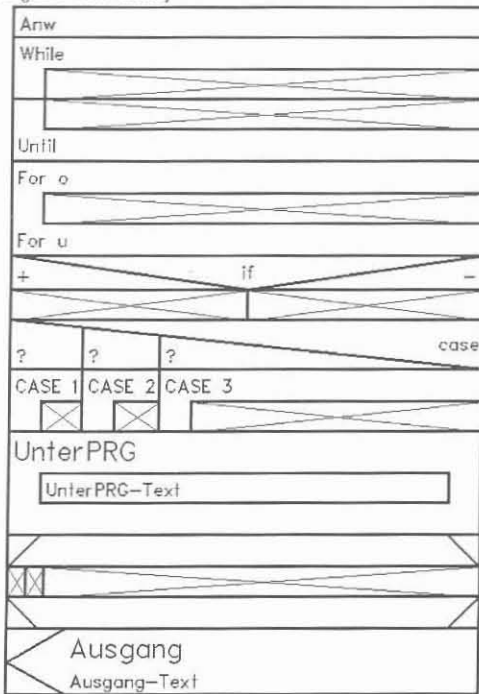
```

; Quelltextskelett-Übersetzungsdatei für ANSI-C
; Parameter :
; \ $      Objektnamen einfügen
; \ #      Objekttext einfügen
; \ +      oberen Text eines FOR-Objektes einfügen
; \ @      Sohnliste hier einfügen
; \ %      Sohnliste unter einem
;         Fallunterscheidungsobjekt einfügen
; \ iNAME  Datei NAME einfügen (nur Großbuchstaben!)
; \ \      das \-Zeichen
; \ ?      gibt eine Meldungbox aus (für in einer Sprache
;         undefinierte Objekte
;         neue Zeile
; \ &xxx   in der Spalte weitermachen (Kommentarbeginn)
; \ >     eine Tabulator-Position einrücken
; \ <     eine Tabulator-Position ausrücken
; \ 0..9   Label-Nummern - pro Objekt 10 Stück
; \ p     Parameter-Liste
;         (nur bei Prozedur- und Root-Sinnbild)
*.C
\ i FILEANSI.HDR \n
\ i FUNCANS.HDR \n/* \# *\nvoid\n$(\p)\n\n{\>\n@\<\n} /*
end of function \ $ *\n\n
/***** Unknown Statement: \# */\n
/* \# */\n
\n/* \# *\nwhile (0) {\>\n@\<} /* end while */\n
\n/* \# *\ndo {\>\n@\<} while (0); /* end do */\n
\n/* \# *\nfor (;) {\>\n@\<} /* end for */\n
\n/* \# *\nif (0) {\>\n@\<} else {\>\n@\<} /* end if */\n
\n/* \# *\nswitch (0) {\>\n
\n\<} /* end switch */\n
\n
\ncase 0: /* \# */\>\n%\nbreak;\<\n
\n/* \# */\n$(\p);\n
\n/* \# */\n\ $;\n
{ /* begin PARALLEL */\>\n
\< /* end PARALLEL */\n
{\>\n/* \# */\<\n}\n
    
```

Danach formuliert man das Struktogramm und erhält daraus das rechtsstehende Programmgerüst (Template).

SSD-Tester

Struktogramm zum Test der Quelltext-Skelett-Ausgabe von G.E.S.y



```

/*
#####
NAME      :
SYNOPSIS  :
DESCRIPTION:
RETURN VALUE:
EXAMPLE   :
SEE ALSO  :
#####
*/
/* Struktogramm zum Test der Quelltext-Skelett-
   Ausgabe von G.E.S.y */

void
SSD-Tester(PARAMETER)
{
    /* Anw */

    /* While */
    while (0) {
        /***** Unknown Statement: * */
    } /* end while */

    /* Until */
    do {
        /***** Unknown Statement: * */
    } while (0); /* end do */

    /* For u */
    for (;;) {
        /***** Unknown Statement: * */
    } /* end for */

    /* if */
    if (0) {
        /***** Unknown Statement: * */
    } else {
        /***** Unknown Statement: * */
    } /* end if */

    /* case */
    switch (0) {

        case 0: /* ? */

            /* CASE 1 */
            while (0) {
                /***** Unknown Statement: * */
            } /* end while */
            break;

        case 0: /* ? */

            /* CASE 2 */
            while (0) {
                /***** Unknown Statement: * */
            } /* end while */
            break;

        case 0: /* ? */

            /* CASE 3 */
            while (0) {
                /***** Unknown Statement: * */
            } /* end while */
            break;

    } /* end switch */

    /* UnterPRG-Text */
    UnterPRG(PARAMETER);
    { /* begin PARALLEL */
        { /* * */
        }
        { /* * */
        }
        { /* * */
        }
    } /* end PARALLEL */

    /* Ausgang-Text */
    Ausgang;

    /* end of function SSD-Tester */
}
    
```

VISUAL-BASIC 3.0 für Windows

Wolfgang Nigischer

DSK-405,406,407,(399)

Vorstellung von Visual Basic 3.0

Die Arbeit mit Windows ist einfach - die Erstellung von Windows Programmen ist eher schwierig. Die Entwickler mußten nicht nur die Sprache C beherrschen und anwenden, sondern auch das teure SDK von Microsoft erwerben und sich durch eine gewaltige Menge an Dokumentation durcharbeiten. Der Trend zu einfacheren Programmiersystemen kommt nicht von ungefähr, da die Komplexität moderner APIs zunimmt und zudem ständig Neuerungen stattfinden (OLE 1.0 → OLE 2.0; Windows 3.0 auf 3.1; zusätzliche API-Funktionen...).

Mit Visual-Basic hat sich diese Erkenntnis grundlegend geändert. Die Windows-Programmierung hat ihren Schrecken verloren. Man findet ein Entwicklungssystem vor, das eine schnelle und leichte Entwicklung von Windows-Programmen ermöglicht. Es wird dabei die grafische Oberfläche von Windows mit der Programmiersprache von Basic verbunden. *Basic ist die ideale Sprache für den Einsteiger, die aber auch professionelle Lösungen unter Windows liefert.* Allerdings nicht mit dem GWBASIC, das vielleicht viele noch kennen. Mit Visual-Basic stehen alle Eigenschaften von Windows 3.1, wie Multimedia, Netzwerkfähigkeit, Drag&Drop und OLE 2.0(!) zur Verfügung.

Wenn man von VB spricht, sollte man zwei Dinge unterscheiden:

1.) VB, die Sprache.

Visual-Basic ist ein moderner BASIC-Dialekt und ist wie C oder Pascal eine strukturierte Programmiersprache. Der Grundwortschatz von Basic wurde zu diesem Zweck um einige Kontrollstrukturen erweitert (Do Loop, While-Wend,...). Hinsichtlich der Funktionen wurde VB speziell um solche erweitert, die ein Zusammenarbeiten mit Windows erst ermöglichen. Auf Grund der Windows-Fähigkeit von VB mußte auch eine gravierende Änderung der Programmabarbeitung durchgeführt werden. Bei Standard-Basic wird das Programm Zeile für Zeile von Anfang bis zum Ende abgearbeitet. Generell gilt für die Windows-Programmierung, daß die Programme ereignisgesteuert ausgeführt werden (dazu mehr in den folgenden Seiten). Irgendwie wird auch eine gewisse Ordnung des eigenen Codes erzwungen (durch die verschiedenen Ebenen des Codefensters) (siehe auch **Bild 3** und **zweites Beispiel**). Während man in herkömmlichen Programmiersprachen jede Aktion des Anwenders selbst abfangen mußte, wird dem Programmierer mit VB diese Arbeit komplett abgenommen.

2.) Visual-Basic, die Entwicklungsumgebung.

VB ist ein modernes Entwicklungssystem, welches fast keine Wünsche offen läßt. So wird z.B. ein Werkzeugkasten (Toolbox) zur Erstellung von Button's, Textfeldern,... zur Verfügung gestellt, wobei für diese Elemente anschließend der Code automatisch generiert wird. Durch einfaches Anklicken der Elemente ist es möglich, den für dieses Element notwendigen Code (Programmsequenz) zu erstellen, wobei auch hier wiederum die Grundstrukturen des Unterprogramms Sub vordefiniert sind.

Mit Visual Basic erhalten Sie ein vollständiges Entwicklungssystem, bei dem Programmiersprache und Oberflächengenerator eine Einheit bilden und man zur Umsetzung eines VB Programms keine weiteren Hilfsmittel benötigt.

Die Grenzen von Visual-Basic

Ja, leider, es gibt sie. Denn die einfache Bedienung des Visual-Basic geht gleichzeitig auf Kosten der Geschwindigkeit. Eines sollte von vornherein geklärt sein: Aufwendige Programme, die mit sehr vielen Formularen¹ gleichzeitig arbeiten und auch die grafische Ausgabe des Visual-Basic recht intensiv nutzen, sollten extrem geschwindigkeitsoptimiert sein, am besten sollten sie direkt auf die Windows Systemebene zurückgreifen und die C-Routinen der sogenannten DLL (Dynamic Link Library) nutzen. Auch ist es mit VB alleine nicht möglich hardwarenahe zu

programmieren. Hierzu sind zusätzliche Produkte notwendig (siehe: Erweiterung von VB weiter unten).

Doch eines ist ganz klar: So schnell und ohne Probleme wie in Visual-Basic lassen sich sicherlich mit keiner anderen Windows-Programmiersprache Applikationen entwickeln. Trotzdem stehen alle Windows-Funktionen zur Verfügung und im Vergleich zu C kommt man schneller zu einem Ergebnis.²

Und wenn Sie nicht gerade beabsichtigen, ein Konkurrenzprodukt zu "CorelDraw" oder zur Textverarbeitung "Word für Windows" zu entwickeln.... -ja, dann liegen Sie mit dem VB richtig. Mit Visual-Basic steht nun die jüngste Entwicklungsstufe von Basic zur Verfügung. Diese Programmiersprache ermöglicht es auf sehr einfache und vor allem bequeme Weise, Programme zu erstellen, die alle Möglichkeiten nutzen, welche Windows dem Anwender zur Verfügung stellt.

Erweiterbarkeit von Visual-Basic

Will man die zuvor erwähnten Beschränkungen umgehen, besteht die Möglichkeit VB zu erweitern (siehe **Bild 1**), zusätzliche Elemente (Custom Controls³) meistens in Form von *.VBX,⁴ oder *.DLL-Dateien wie beispielsweise: Cmdialog⁵, Gauge-Controll (analoge Meßwertanzeige, wie Tacho oder Thermometer), Grid-Controll (Erstellt eine Tabelle, die mit Werten gefüllt werden kann), oder MSComm (Regelt den Austausch von Daten über die serielle Schnittstelle), Graph-Controll (bietet die Möglichkeit der Darstellung von numerischen Daten, wie Meßwerten, oder Geschäftsgrafiken), 3D Button Animated Button (erlaubt Buttons mit Bitmaps, Icons oder Bilddateien im Metafile - Format zu belegen und diese zu animieren), um nur einige zu nennen.

Mit vielen dieser Zusatz-Controls erhält man auch gleichzeitig die Erlaubnis, diese in kompilierter Form, zusammen mit dem eigenen Anwendungs-Programm, lizenzfrei weitergeben zu dürfen.

Mit anderen Worten: Wenn etwas mit VB alleine nicht durchführbar ist, kann man sich unter den mittlerweile sehr zahlreichen Dritt-Anbietern nach einer geeigneten Erweiterung umsehen.

6

²Ich kenne Firmen, die oftmals Prototypen in VB anfertigen, damit sich der Kunde von dem, was ihm bevorsteht, schneller ein Bild machen kann. Diese "Hüllen" aus VB werden anschließend perfektioniert; sowohl in Hinsicht auf den Basic-Code, als auch durch eine Verknüpfung mit in C geschriebenen Programmen. Die Schnittstelle zum Anwender ist somit von Anfang an in VB!

³Diese werden mit Hilfe des CDK erstellt.

⁴Diese Erweiterungen werden in der Toolbox sichtbar (siehe Bild 1). Die hier Aufgezählten werden in der professional Edition mit geliefert.

⁵Common dialog. Eine kurze Erklärung finden Sie unter: Die neuen Features von VB am Ende des Artikels. Ab VB 3.0 auch in der Standardversion enthalten.

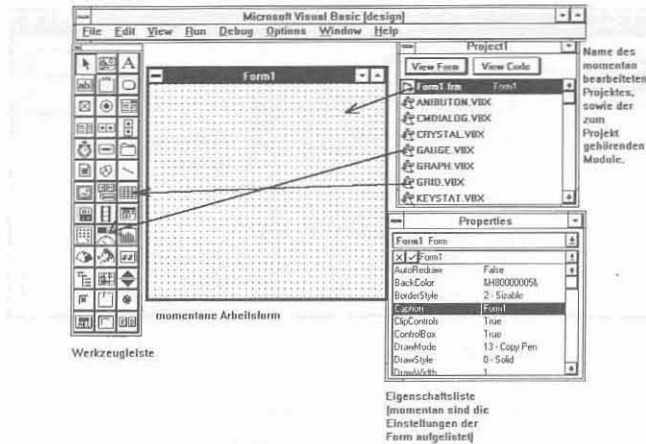
⁶Um die Verwirrung der Erweiterbarkeit zu steigern:

Es ist mittlerweile möglich durch ein entsprechendes Zusatzprogramm den Windows - Nachrichtenstrom eines beliebigen VB Forms oder Controls abzufangen. Diese als Subclassing bekannte Technik ermöglicht einer Anwendung die Ermittlung von Ereignissen, die von der verwendeten Sprache normalerweise nicht unterstützt werden. Und das ohne daß der Anwendungs - Entwickler auf C oder sonstige DLL - Programmierung zurückgreifen muß.

Auch läßt sich bereits auf solche Windows - API Funktionen zugreifen, die unter VB normalerweise nicht verwendet werden können. (SpyWorks für VB, von Desaware).

¹Die Anzahl der Formen ist bei der aktuellen Version 3.0 auf 230 limitiert, max 80 davon dürfen geladen sein (bei Verwendung von Windows 3.1)

Beim Start von Visual-Basic erscheint folgendes Bild:⁷



Wahlweise mit Symbolleiste:

Hierbei werden die am häufigsten benötigten Befehle des Menüs grafisch dargestellt. Rechts werden zusätzlich noch die X/Y Koordinaten des jeweils selektierten Controls in Bezug zur linken oberen Ecke der "Arbeitsform" angegeben.



Bild 1:

Startbildschirm von Visual-Basic mit Symbolleiste

Programmierung⁸

Schritt 1

Erstellen einer Form (also des Windows-Fensters, das der Anwender vom Programm zu sehen bekommt), zeichnen der benötigten Schaltflächen, Rahmen, Listboxen und dgl.⁹. Das heißt, schon beim Programmieren hat man unmittelbar mit jenen Elementen zu tun, die der Anwender später auf dem Bildschirm vorfindet. Es erinnert fast an "malen nach Zahlen" (Programmieren wie gemalt). Man wählt das entsprechende Objekt mit der Maus aus der Toolbox (Bild 2 links) aus. Anschließend erscheint über der "Arbeitsform" ein Fadenkreuz, ähnlich wie in Paintbrush, mit dem dann das jeweilige Objekt an der gewünschten Stelle in der Form gezeichnet werden kann. Man kann bereits jetzt in der Entwicklungszeit wichtige Voreinstellungen der Eigenschaften im "Properties"-Fenster¹⁰ vornehmen¹¹.

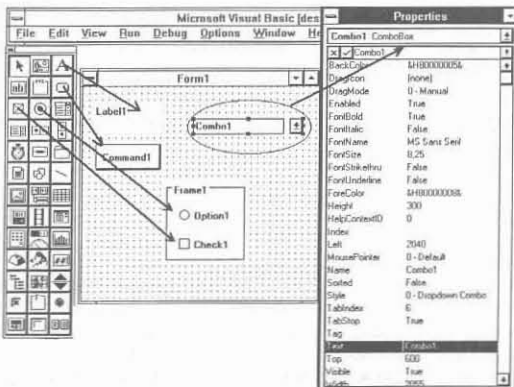


Bild 2: Erstellen einer "Arbeits" - Form

⁷Visual Basic Professional Version 3.0

⁸Hierbei gehe ich von einem kleinen Beispiel-Projekt aus. Bei größeren Projekten ist es ratsam, vorher einige Überlegungen im Bezug auf Aussehen, Benutzerfreundlichkeit u.s.w. anzustellen. Entsprechende "Richtlinien" und Vorschläge gibt es ja mittlerweile zu genüge...

⁹Visual Development

¹⁰In der deutschen Standard-Version ist das "Eigenschafts-Fenster"

¹¹Zu den Eigenschaften eines Objekts gehören beispielsweise sein Name, Inhalt, Größe und Farbe,... aber auch die ab der Vers. 2.0 eingeführte HelpContextID Nummer, die beim Hilfeaufruf des eigenen Programms die richtige Seite des Help-Files aufruft. (Bild 2)

Der nächste Schritt:

Schreiben des Programmcodes¹². Im Bild 3 wurde durch Doppel-Klicken der Command-Schaltfläche das Codefenster für dieses Objekt geöffnet. Oben links erscheinen die momentan bereits gezeichneten Objekte, rechts davon die möglichen Ereignisse. So steht Click beispielsweise für ein Ereignis, mit dem der "End-User" mit der linken Maustaste auf dieses Objekt klickt. Als Reaktion auf ein solches Ereignis ruft das betroffene Objekt jene Prozedur auf, die der Programmierer für dieses Ereignis geschrieben und vorgesehen hat. Die bisher bekannte lineare Programmierung weicht der ereignisorientierten¹³.

Visual-Basic = Quick-Basic + Ereignisprozeduren.

Diese Ereignisse werden in Ereignisprozeduren verarbeitet. Die Ereignisprozeduren sind SUB/END SUB-Prozeduren die von Visual-Basic aufgerufen werden¹⁴.

Nach deren Abarbeitung wird der Prozeß wieder an Windows zurückgegeben und so eine gewisse "Multi tasking Fähigkeit" erreicht.

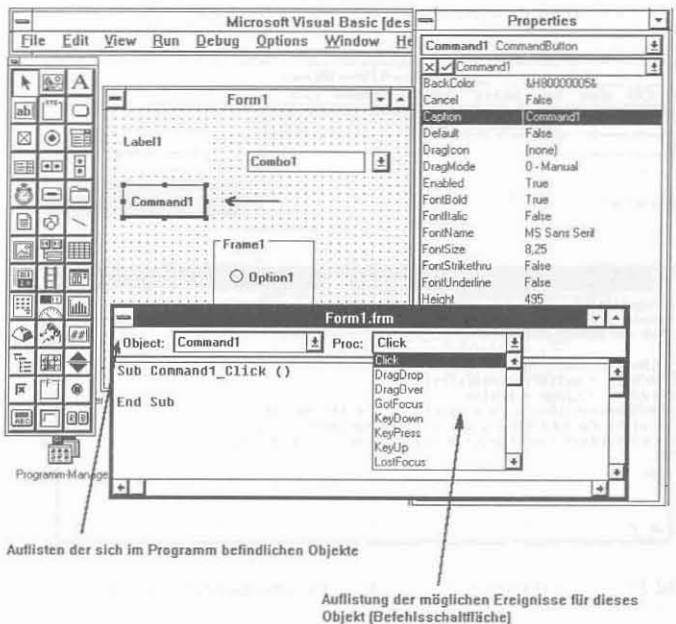


Bild 3: Codefenster für Command1-Schaltfläche

Genug der grauen Theorie:

Erstes Beispiel

Da das "Hallo World" meines Erachtens ziemlich abgedroschen ist, habe ich mir eine andere Aufgabenstellung ausgedacht: Es soll ein Programm erstellt werden, das die Fläche eines Rechtecks berechnet, und das Ergebnis in einer formatierten Ausgabe ausgibt. Damit das ganze ohne VB-Umgebung läuft, wird anschließend eine EXE-Datei erstellt.

¹²Sinnvollerweise sollte man vorher den gezeichneten Objekten aussagekräftigere Namen geben. Die Bezeichnung (Name) Eigenschaft (im Properties - Fenster angegeben) eines Objektes muß mit einem Buchstaben beginnen und darf max. 40 Zeichen enthalten, einschließlich Zahlen und Unterstrichungen.

¹³Das sind unter anderem: Bewegungen der Maus, Klicken mit der Maus, eine Taste drücken, oder Systemereignisse, wie beispielsweise DEE-Nachrichten oder Nachrichten des Timers. Es gibt keine aktive Programmzeile, so lange kein Ereignis auftritt.

¹⁴Eine klassische Basic-Unterroutine, die durch GoSub aufgerufen und durch Return beendet wird, gibt es in Visual Basic zwar, ist jedoch völlig überflüssig. Statt dessen ist es viel übersichtlicher, einfacher und flexibler die sog. "Subs" des VB zu verwenden. Eine Sub ist nichts anderes als ein BASIC-Untersprogramm, wird im Programmtext jedoch gesondert platziert und läßt sich wie jeder normaler BASIC-Befehl aufrufen. Wobei VB ein weiteres Codefenster öffnet und Ihnen das Gerüst der Unterroutine (oder Funktion) vorgibt (Bild 3)

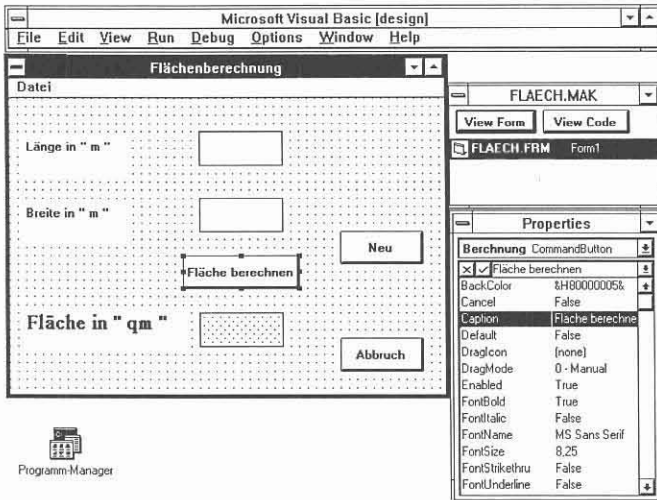


Bild 4: Erstellung der Form mit den Objekten

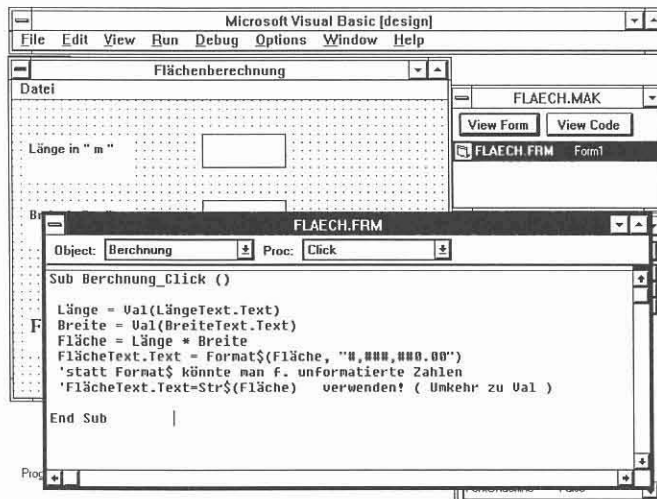


Bild 5: Den Code in der Schaltfläche Flächenberechnung unterbringen

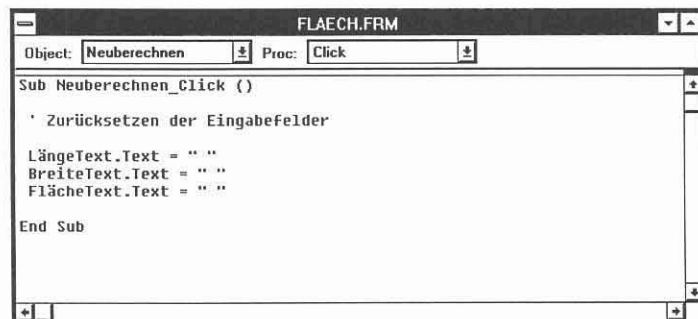


Bild 6: Die Schaltfläche Neu mit Leben erfüllen

Dasselbe gilt für die Abbruch Schaltfläche:

```
Sub Befeh12_Click ()
    End
    ' Programm-Ende
End Sub
```

Nach dem Austesten des Programms¹⁵ geht es an die Erstellung eines EXE - Files:

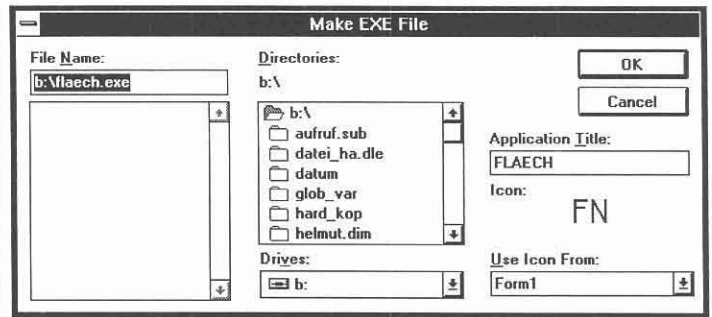


Bild 7: Erstellen eines EXE - Files

Ein Bild sagt mehr als tausend Worte; für diejenigen, die VB nicht kennen, eine kurze Erklärung zu Icon und zum Application Title, der Rest des obigen Bildes ist glaube ich klar: Unter Applications-Title wird die Überschrift des Programms eingetragen (das, was im Windows Programm-Manager unter dem Icon steht). Das gewünschte Icon kriert man zuerst selbst¹⁶ (ein Icon-Editor ist im Lieferumfang von VB enthalten - mit Source-Code!) und weist dann anschließend der Startform in der Eigenschaftsliste das gewünschte Icon zu. Zu guter letzt erhalten Sie eine EXE-Datei. Um das Programm weitergeben zu können, ist es noch notwendig, eine sog. Laufzeitbibliothek mit auf die Diskette zu kopieren (VBRUN300.DLL¹⁷). Es sind also mindestens zwei Dateien notwendig um ein mit VB erstelltes Programm zum laufen zu bringen. Nämlich die eigentliche EXE-Datei und diese Laufzeitbibliothek.

Visual Basic ist also kein reiner Compiler. Trotzdem findet bei dem Erstellen eines *.EXE-Files eine Aufbereitung der Programmzeilen statt, somit kann man aber auch eigentlich nicht mehr von einem Interpreter (wie GWBASIC) sprechen.

Wo werden diese Dateien bei der Installation hin kopiert? Nun ich würde sagen die eigene Programmdatei gehört in ein eigenes Verzeichnis. Die *.DLL und *.VBX-Dateien gehören in das Windows\System-Verzeichnis. Diese Dateien kommen für alle mit VB erstellten Programme zum Einsatz, das heißt, sie befinden sich nur einmal im Speicher, auch wenn mehrere mit VB erstellten EXE-Files aktiv sind¹⁸. Das bringt zwar eine Platzersparnis auf der Festplatte, birgt aber die Gefahr in sich, daß es bei Verwendung von mehreren VB-Programmen, die mit verschiedenen Versionen erstellt wurden, zu Problemen kommen kann. Ein entsprechendes Installationsprogramm ist ab VB 3.0 in der Standardversion und vorher in der Professional-Version enthalten¹⁹. Dieses Installationsprogramm eruiert unter anderem das richtige Windows-Verzeichnis, in das diese VBX- und DLL-Dateien hingehören, kriert eine eigene Programm-Gruppe im Programm-Manager und ein eigenes Verzeichnis für Ihre Programm- bzw. Datendateien.

¹⁵Es kann bereits während der Entwicklungszeit das Programm getestet werden, ohne, das man eine EXE - Datei erstellt.

¹⁶oder sucht sich eines von den vielen mit gelieferten Icons aus.

¹⁷Oder je nach Version mit dem das Programm compiliert wurde: VBRUN100.DLL, VBRUN200.DLL. Werden third-Party-Objekte in das eigene Programm mit eingebunden, so sind diese ebenfalls mitzuliefern (meistens in Form von DLL- oder VBX-Dateien, z.B.: GRID.VBX, wenn eine Tabelle programmiert wurde.

¹⁸Bei fast allen Zusatzcontrols sind auch im VB Buch die entsprechenden Benützungshinweise für Visual C++ angeführt. Das heißt, das alle diese Controls ohne Änderung von beiden Programmiersprachen verwendet werden können!

¹⁹In der Professional - Version sind auch zusätzlich Informations - Files über die Programmierung (unter anderem API's, so z.B. eine komplette VB API Referenz und API Deklarationen in Form von Help - Files) und zur Programmgestaltung enthalten.

Weiters: Programm - Beispiele, sowie der Windows Help - Compiler Version 3.1 enthalten.

Allerdings ist die Professional - Edition in englisch. Nur die Standart - Ausgabe ist in Deutsch erhältlich.

Bitte besorgen Sie sich einen Prospekt von Microsoft über VB, da stehen mehr Details drinnen, ich habe den Prospekt nur von einer alten Version

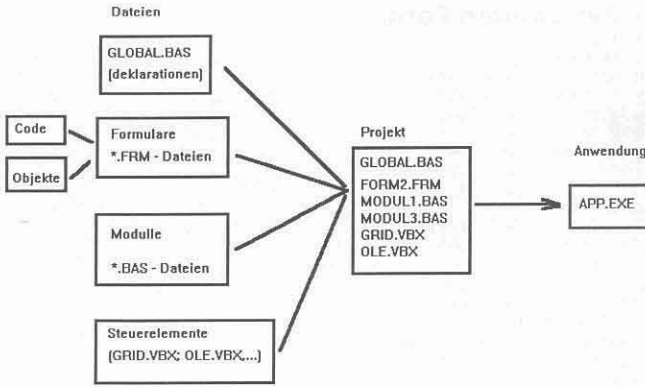


Bild 8: Dateien eines Projekts

Zum Bild 8: Eine Form besteht also aus verschiedenen Objekten, die mit entsprechenden Code versehen sind. Zusätzlich kann es noch eine Datei mit dem Namen GLOBAL.BAS, die generelle Deklarationen, sowie Makros enthält, und durch diese Definitionen auch die Lesbarkeit der Programme erhöhen, geben. Der Inhalt dieser Datei setzt sich häufig aus einem Teil des Inhaltes der KONSTANT.TXT und WINAPI.TXT zusammen.

Beispiele aus Constant.TXT

```
Global Const KEY_LEFT = &H25
Global Const KEY_UP = &H26
Global Const KEY_RIGHT = &H27
Global Const KEY_DOWN = &H28
Global Const KEY_SELECT = &H29
Global Const KEY_PRINT = &H2A
Global Const KEY_EXECUTE = &H2B
Global Const KEY_SNAPSHOT = &H2C
Global Const KEY_INSERT = &H2D
Global Const KEY_DELETE = &H2E
Global Const KEY_HELP = &H2F
```

Beispiele aus WINAPI.TXT

```
' Hilfe-Bereich.
' Befehle für WinHelp()
Global Const HELP_CONTEXT = &H1 'Zeige Thema in uITopic an
Global Const HELP_QUIT = &H2 'Beende Hilfe
Global Const HELP_INDEX = &H3 'Zeige Index an
Global Const HELP_HELPONHELP = &H4 'Hilfetexte für Hilfe anzeigen
Global Const HELP_SETINDEX = &H5 'Setze den aktuellen Index für Mehrfachindex-Hilfen
Global Const HELP_KEY = &H101 'Zeige Thema für Schlüsselwort in offabData

Global Const HELP_MULTIKEY = &H201
Declare Function WinHelp Lib "User" (ByVal hwnd As Integer,
ByVal lpHelpFile As String,
ByVal wCommand As Integer, dwData As Any) As Integer
Type MULTIKEYHELP
mkSize As Integer
mkKeylist As String * 1
szKeyphrase As String * 253 'Array-Länge ist beliebig, kann geändert werden
End Type
```

Module sind reine Codeteile eines Programms, ohne dazugehöriger Form. Das könnten beispielsweise immer wiederkehrende Eingabeüberprüfungen - Funktionen sein, die man sich der Übersicht halber in so einer BAS-Datei abspeichert. Man kann sich also mit dieser Methode eine eigene Bibliothek häufig benötigter Codes zurecht legen und diese dann von einer Form aus aufrufen.

Visual Basic unterstützt also einen modularen Programmaufbau.

Modulare Programme bieten eine Reihe von Vorteilen: Sie sind leicht zu verstehen, leicht zu erweitern, und vor allem, die Fehlersuche wird erheblich erleichtert, da sich die Suche nach einem Fehler auf ein einzelnes Modul beschränken kann.

Zweites Beispiel

Hierbei möchte ich hauptsächlich zeigen, wie der Inhalt einer Variablen weiter gereicht wird, da Variablen je nach Deklaration verschiedene Geltungsbereiche haben.

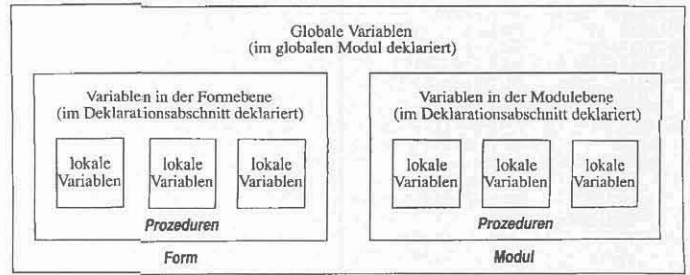


Bild 9: Geltungsbereich der Variablen

Wiederum wird die Fläche eines Rechtecks berechnet. Diesmal wird aber das Ergebnis in einer anderen Form dargestellt. Die Berechnung wird über die Befehlsschaltfläche der ersten Form aufgerufen. Man erkennt im Bild 10 deutlich in der fokussierten Form die Dimensionierung der beiden Variablen, sowie ein eigenes Unterprogramm, das der Deklaration-Ebene untergeordnet ist.

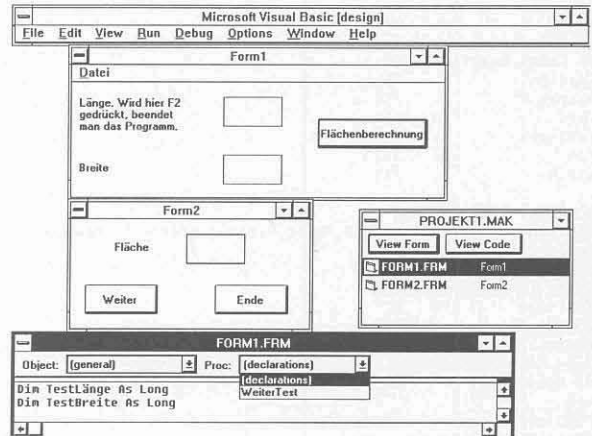


Bild 10: Alternative Form für eine Rechteck - Flächenberechnung

Als weitere Facette kommt in diesem Beispiel auch noch eine Funktion KeyDown vor, die bei Betätigen der F2-Taste das Programm vorzeitig beendet und ein kleines Menü dazu (siehe Bild 11).

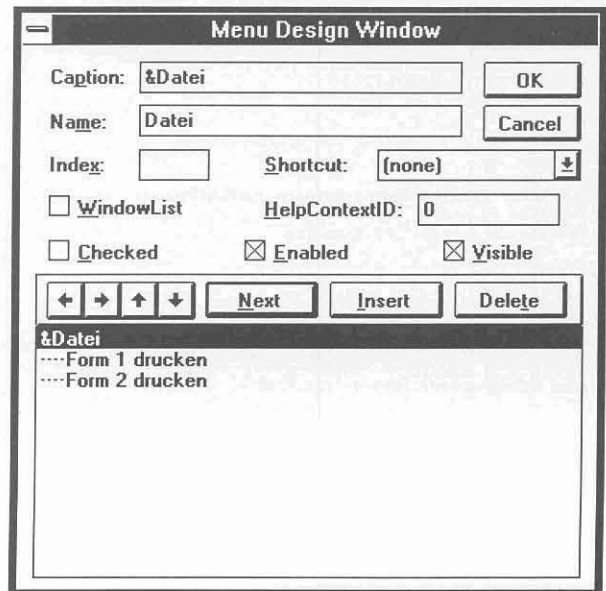


Bild 11: Menüentwurfswindow

Code der ersten Form²⁰

```

VERSION 2.00
Begin Form Form1
Caption       = "Form1"
ClientHeight = 1905
ClientLeft   = 1035
ClientTop    = 1305
ClientWidth  = 6165
Height       = 2595
Left         = 975
LinkMode     = 1 'Source
LinkTopic    = "Form1"
ScaleHeight  = 1905
ScaleWidth   = 6165
Top          = 675
Width        = 6285
Begin TextBox Breite
Height       = 495
Left         = 2520
TabIndex    = 3
Top          = 1200
Width        = 975
End
Begin CommandButton Berechnung
Caption      = "Flächenberechnung"
Default     = -1 'True
Height      = 495
Left        = 4080
TabIndex    = 4
Top         = 600
Width       = 1815
End
Begin TextBox Länge
Height      = 495
Left        = 2520
TabIndex    = 2
Top         = 240
Width       = 975
End
Begin Label Bezeichnung2
Caption     = "Breite"
Height     = 255
Left       = 120
TabIndex   = 1
Top        = 1320
Width      = 735
End
Begin Label Bezeichnung1
Caption     = "Länge. Wird hier F2 gedrückt, beendet man das
Programm."
Height     = 615
Left       = 120
TabIndex   = 0
Top        = 240
Width      = 1815
End
Begin Menu Datei
Dateimenu
Caption    = "&Datei"
Begin Menu PrintForm1
Caption   = "Form 1 drucken"
End
Begin Menu PrintForm2
Caption   = "Form 2 drucken"
End
End
End
Option Explicit 'Deklarationsebene der Form 1 !!!!
Dim TestLänge As Long
Dim TestBreite As Long
Dim Fläche As Long
Sub Berechnung_Click ()
TestLänge = Val(Länge.Text)
TestBreite = Val(Breite.Text)
Form2.Show
WeiterTest ' Weiter zur Unterroutine
End Sub
Sub Länge_KeyDown (Tastencode As Integer, Umschalten As Integer)
Const FTASTE_F2 = &H71
If Tastencode = FTASTE_F2 Then End
End Sub
Sub PrintForm1_Click ()
Form1.PrintForm
End Sub
Sub WeiterTest ()
Fläche = TestLänge * TestBreite
Form2.FlächenAusgabe.Text = Str$(Fläche)
End Sub

```

Code der zweiten Form

```

VERSION 2.00
Begin Form Form2
Caption       = "Form2"
ClientHeight = 1815
ClientLeft   = 1035
ClientTop    = 3585
ClientWidth  = 3960
Height       = 2220
Left         = 975
LinkMode     = 1 'Source
LinkTopic    = "Form2"
ScaleHeight  = 1815
ScaleWidth   = 3960
Top          = 3240
Width        = 4080
Begin CommandButton Ende
Caption      = "Ende"
Height      = 495
Left        = 2400
TabIndex    = 2
Top         = 1080
Width       = 1215
End
Begin CommandButton Weiter
Caption     = "Weiter"
Height     = 495
Left       = 240
TabIndex   = 3
Top        = 1080
Width      = 1215
End
Begin TextBox FlächenAusgabe
Height     = 495
Left       = 1920
TabIndex   = 1
Top        = 240
Width      = 975
End
Begin Label Bezeichnung1
Caption    = "Fläche"
Height    = 255
Left      = 720
TabIndex  = 0
Top       = 360
Width     = 735
End
End
Sub Ende_Click ()
Unload Form1
End
Sub Weiter_Click ()
Form1.Show
Form1.Länge.Text = ""
Form1.Breite.Text = ""
Form1.Länge.SetFocus ' setzt den Corsor inner auf Länge!
Form2.Hide
End Sub

```

Es ist auch möglich, gleichzeitig zwei verschiedene Prozeduren zu betrachten:²¹

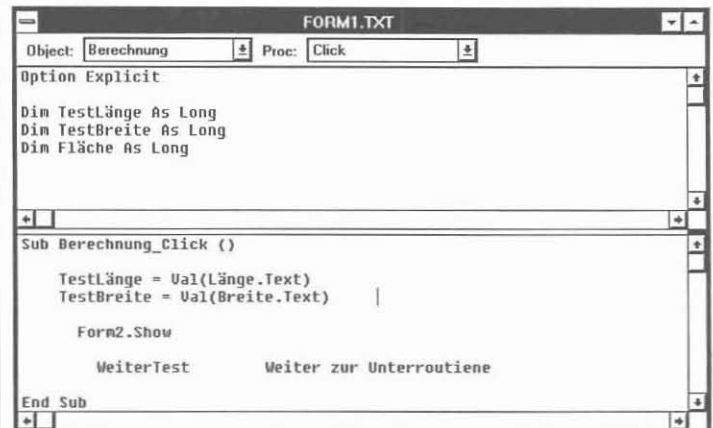


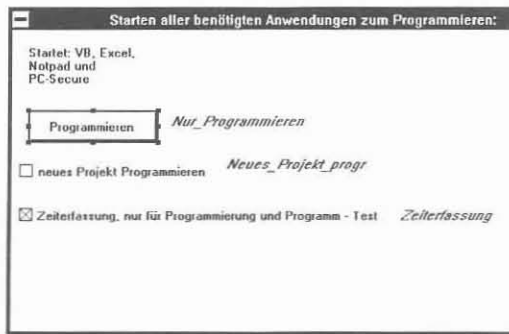
Bild 12: Gleichzeitige Darstellung verschiedener Codesequenzen

²⁰Mit der kleineren Schrift am Anfang möchte ich nur den grafischen Teil gegenüber dem eigentlichen Code differenzieren. Die hier angegebenen, hauptsächlich grafischen Angaben erledigen sich zum Teil mit Schritt 1 der Programmierung, nämlich dem Zeichnen der Steuerelemente und dem "Ausfüllen" des Eigenschaftsfensters.

²¹Ähnlich, wie man z.B. in Win Word der Text teilen kann, ist das auch in VB möglich.

Drittes Beispiel

Abschließend möchte ich noch kurz zeigen, wie in VB ein anderes Programm aufgerufen wird und wie CheckBoxes als Entscheidungsträger verwendet werden.



Mit *Kursiv*-Schrift habe ich die im Projekt verwendeten Namen angegeben.

Bild 13: Checkboxes im Programm

Der Code (auszugsweise) dazu:

```
Sub Nur_Programmieren_Click ()
    Startprogramm = Shell("C:\EXCEL\EXCEL.EXE", 2)
    Debug.Print "Startprogramm: "; Startprogramm
    ' Angabe des Programmes und der Startoption (2 ist auf Icon verkleinert,
    1 ist Vollbilddarstellung).
    Startprogramm = Shell("C:\WINDOWS\notepad.exe", 2)
    Debug.Print "Startprogramm: "; Startprogramm
    If Neues_Projekt_prog.Value = True Then
        Startprogramm = Shell("C:\WINDOWS\winfile.exe", 2)
    End If
    If Zeiterfassung.Value Then
        Startprogramm = Shell("C:\XTIMELOG\XTIMELOG.EXE", 1)
    End If
    Debug.Print "Startprogramm: "; Startprogramm
End If

Startprogramm = Shell("C:\VB\VB.EXE", 1)
Startprogramm = Shell("C:\PCTOOLS\PCSECURE.EXE")
' DOS-Programm aus PC-Tools 8.0
End
End Sub
```

Nobody is perfekt...

Nicht unerwähnt lassen möchte ich die umfangreichen Debug-Möglichkeiten, die VB anbietet. Bereits während der Codeeingabe wird nach Drücken der Return-Taste die zuletzt eingegebene Codezeile überprüft. Während der Testphase können über ein Direktfenster (watch window) sämtliche Variablen sichtbar gemacht werden (Debug.Print). Zusätzlich kann im Direktfenster durch die Eingabe Print "Variablenname"; der momentane Wert der gewünschten Variablen ermittelt werden. Wem das "Debug.Print"-Verfahren während der Entwicklungsarbeit zuviel Schreibarbeit bereitet, kann ab Version 2.0 bestimmte Variablen gezielt überwachen lassen (Watch Expressions) und zwar innerhalb der Prozedur oder im ganzen Programm. Auch werden auf Wunsch alle zu einem bestimmten Zeitpunkt (Möglichkeit des Haltepunkt setzen) aktiven Prozeduren aufgezzeigt. (Calls Dialog)

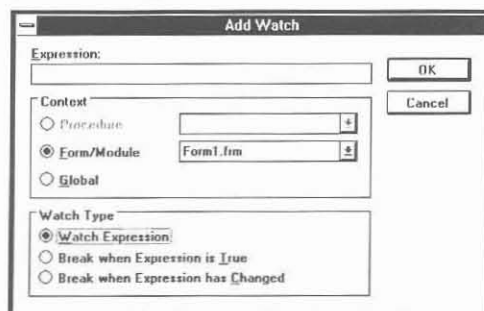


Bild 15: Beobachtung von Variablen

Um sich die Sache mit dem Direktfenster besser vorstellen zu können, hier wieder 2 Screen-Shoots aus meinem zweiten Beispiel - Programm: Das obere Fenster ist das Debug-Fenster, das sich wiederum in 2 Teile aufgliedert: Im oberen Teil wurden laut vorhergehenden Bild zwei Beobachtungswerte definiert. Der untere Teil kommt durch das

Debug.Print (im Source-Code des darunter liegenden Fensters) zustande.

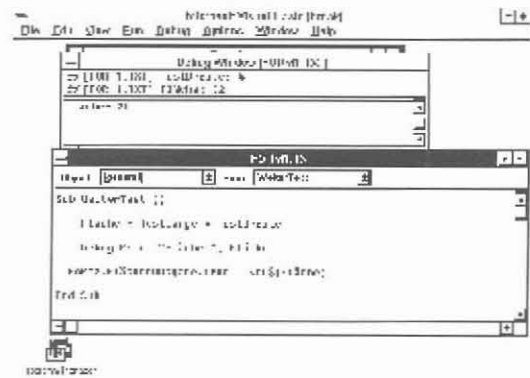


Bild 16: Debuggen in Visual-Basic

Wie vorher bereits erwähnt ist es dann noch möglich vor Erstellung einer EXE-Datei einen Probelauf des zu erstellenden Programms durchzuführen. Nicht zuletzt steht noch eine umfangreiche kontextsensitive On-Line-Hilfe mit Beispielen zur Verfügung. (Über 3 MB !).

Die neuen Features VB 3.0

- Access-Datenbank Zugriff und deren Daten-Editierung, ebenso auf Fox Pro Vers 2.0 und 2.5, dBASE III und IV, Paradox, Btrieve, sowie ODBC, wie etwa der SQL-Server. Bei dem ganzen Vorgang bleibt die eigentliche Datenbank-Engine dem Programmierer verborgen. Der bekommt nur ein neues Custom Control (Data Control) zu sehen.
- PopupMenu-Befehl
- Common dialog (für häufig benötigte Dialog-Boxen, wie "Datei öffnen", "Datei speichern", ...)
- OLE 2.0 Unterstützung
- Crystal Reports (Reportgenerator zur Bildschirm- oder Drucker-Ausgabe von Datenbanken)
- Outline-Control (zum Darstellen hierarchischer Strukturen, wie dem Verzeichnisbaum im Windows Dateimanager); nur Prof. Edition.
- Setup-Wizard, wie bereits im Artikel erwähnt).

Für alle, die jetzt Lust auf's Programmieren bekommen haben, hier die Systemvoraussetzungen, des Rechners, um mit VB arbeiten zu können:

- IBM PC oder Kompatibler mit 80286er Prozessor oder höher²³
- MS Windows 3.0 oder Höher, MS-DOS 3.1 oder höher.
- 1 MB, besser 2 MB Hauptspeicher
- Festplatte (VB 3.0 Prof. ca. 30 MB!), je nach Anzahl der zusätzlich installierten Beispiele).
- Diskettenlaufwerk
- Maus
- für Pen-Computing natürlich einen Pen-Computer und Windows für PEN-Computing
- für Multimedia-Programmierung ebenfalls entsprechende Erweiterungen
- für ODBC-Programmierung entsprechende Datenbanktreiber²⁴, direkte Datenbankzugriffe können ab Vers. 3.0 getätigt werden (MS Access, Fox Pro Vers 2.0 und 2.5, dBASE III und IV, Paradox, Btrieve)
- MAPI-Programmierung²⁵: MS Mail 3.0 oder höher

Ach übrigens: VB gibt es auch unter DOS. Im Programmpaket, das extra erworben werden muß, sind Umwandlungs-Programme mitgeliefert, die VB Windows-Source-Code in DOS-kompatiblen umwandeln. So werden spezielle Windows-Teile aufgegliedert (wie z.B. das Windows API) □

²²Das ist jener Standarddialog, den Sie unter den mit Windows mitgelieferten Zubehörprogrammen unter dem Menüpunkt "Datei" aufrufen (sieht bei allen Programmen gleich aus).

²³Besser ist ein 386er, mit ausreichend RAM (6 MB).

²⁴Ist ebenfalls in den VB Sprachumfang integriert.

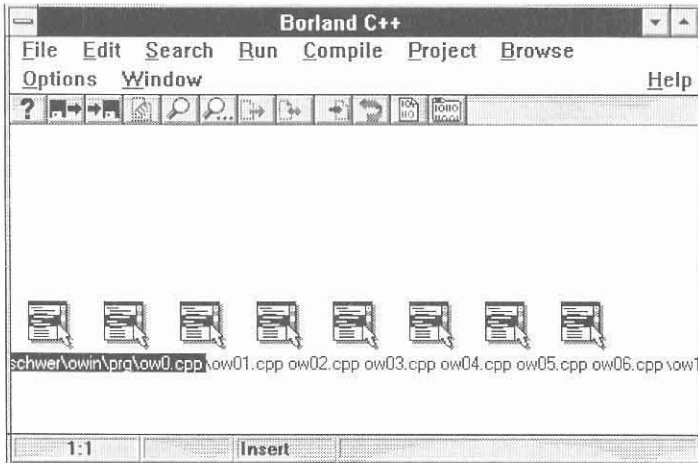
²⁵Kommunikation zwischen Applikationen und Electronic-Mail über beliebige Netzwerke.

Windows-Programmierung, BORLAND

Franz Fiala, N, TGM

DSK-404:\BD

BORLAND verwendet zwei verschiedene Benutzeroberflächen (IDE=Integrated Development Environment) für DOS und Windows. Während man mit der DOS-IDE alle Formen von Programmdateien entwickeln kann (also auch EXE-Files für Windows), kann man mit der Windows-IDE ausschließlich Windows-Programme generieren. Die Windows-IDE präsentiert sich so:



EasyWin

Das einfachste unter Windows ablaufende Programm wird mit **EasyWin** erzeugt. Man kann die gewohnten Funktionen aus `conio.h` und `stdio.h` verwenden, allerdings mit Einschränkungen:

- Verzicht auf Farbe und Grafik
- Beschränkung auf:
`gotoxy()`, `wherex()`, `wherey()`, `clrscr()`, `clrcol()`

Und so schaut ein Programmgerüst mit **EasyWin** aus:

```
#include <windows.h>
#pragma argsused

int PASCAL WinMain
(
    HANDLE hInstance,
    HANDLE hPrevInstance,
    LPSTR lpszCmdLine,
    int cmdShow
)
{
    _InitEasyWin();
    ...
    alles weitere ist beliebig
}
```

Wie man sieht, ersetzt `WinMain()` das bekannte `main()` aus C und C++. Der Startup-Kode ist auch unterschiedlich. Etwa benötigt man mehr Parameter als noch unter DOS:

hInstance Jede Windows-Applikation unterscheidet sich von einer anderen durch ein `HANDLE`, eine ganze Zahl.
hPrevInstance Gibt es mehr als nur eine Instanz, dann enthält dieser Wert das `HANDLE` auf die vorige Instanz. Gibt es keinen Vorläufer, ist dieser Wert 0.
lpszCmdLine Pointer auf die Kommandozeile.
cmdShow Zeigt auf das erste darzustellende Fenster
_InitEasyWin() Initialisiert das Fenster und die verwendbaren `stdio`- und `conio`-Funktionen.

Erstes Windows-Programm, EASY1.CPP

Dieses Programm zeigt die Verwendung der `stdio`-Funktionen. Die Zeile

```
#pragma argsused
```

verhindert die Fehlermeldung, daß die übergebenen Parameter ungenutzt bleiben.

```
// EASY1.CPP
#include <windows.h>
#include <stdio.h>
#include <conio.h>
#pragma argsused

int PASCAL WinMain
(
    HANDLE hInstance,
    HANDLE hPrevInstance,
    LPSTR lpszCmdLine,
    int cmdShow
)
{
    _InitEasyWin();
    printf("Hello world\n\x80");
    gotoxy(10,10);
    printf("Hallo Seminar\n");
    return 0;
}
```

Übergabeparameter, EASY2.CPP

Dieses Programm zeigt die Werte der vier Übergabeparameter an. Man sieht, daß die Kommandozeile mit Ausschluß des Programmnamens gezählt wird.

Am besten verwendet man **EasyWin** zum Debuggen, wenn der eingebaute Debugger nicht komfortabel genug arbeitet.

```
// EASY2.CPP
#include <windows.h>
#include <stdio.h>
#include <conio.h>
#pragma argsused

int PASCAL WinMain
(
    HANDLE hInstance,
    HANDLE hPrevInstance,
    LPSTR lpszCmdLine,
    int cmdShow
)
{
    _InitEasyWin();
    printf("HANDLE hInstance : %d\n", hInstance);
    printf("HANDLE hPrevInstance: %d\n", hPrevInstance);
    printf("LPSTR lpszCmdLine : %s\n", lpszCmdLine);
    printf("int cmdShow : %d\n", cmdShow);
    return 0;
}
```


Object Windows

Unser einfaches Programm soll lediglich einen Begrüßungstext kreieren, eine Abfrage, ob man denn wirklich aufhören will (Verabschiedung) und ein Window-gemäßes "Hello-World", in dem der Text nicht als ASCII-Text sondern mit der Maus als Grafik eingegeben wird.

Es wäre kein richtiges Programm, hätte es nicht auch einen Bug. Hier werden zwar die Linien richtig gezeichnet, beim nochmaligen Zeichnen aber gibt es unerwünschte Verbindungslinien, die noch zu beseitigen wären. Der Fehler entsteht durch die vereinfachte Aufzeichnung der Bildpunkte in einer Liste. Eigentlich müßte man eine Liste von Listen konstruieren und jede dieser Einzellisten wäre ein durchgehender Linienzug, daher sind Unterbrechungen möglich. So aber werden beim erstmaligen Neuzeichnen des Bildschirms auch jene Verbindungen gezogen, bei denen die Maus nicht gedrückt war. Als Übung könnten Sie daher dieses Beispiel entsprechend korrigieren und in Anlehnung an das Handbuch, in dem dieser Fehler natürlich nicht vorkommt, erweitern.

Diese Bildschirm steht sinngemäß stellvertretend für alle entstehenden Programmversionen OW0..OW06.CPP:



Windows-Programme werden mit der Windows-IDE hergestellt. Folgende Voraussetzungen sind für das Arbeiten mit Object Windows erforderlich:

```
Options
Compiler - LARGE Model
Linker Libraries
Container(Static)
Object Windows(Static)
StandardRunTime (Static)
Directories
Pfade auf INCLUDE- und LIB-Subdirectories
von RTL,CLASSLIB und OWL
```

Ein Fenster, OW0.CPP

Eine Object-Windows-Anwendung wird von der Klasse `TApplication` abgeleitet. Über die Größe des Fensters muß man sich in Windows zunächst nicht kümmern. Einer `TApplication` übergibt man den Titel (hier "Hello") die entsprechenden `HANDLE` und auch die Kommandozeile.

Die einfachste Anwendung (ein einzelnes Fenster) nimmt uns einmal die Aufgabe ab, sich um die Übergabeparameter kümmern zu müssen, indem die Klasse `TApplication` diese als Parameter übernimmt.

Das Objekt `a` der Klasse `TApplication` wird mit `a.Run()` ausgeführt.

```
//OW0.CPP
#define WIN31
#include <owl.h>

int PASCAL WinMain
(
HANDLE hInstance,
HANDLE hPrevInstance,
LPSTR lpCmdLine,
int nCmdShow)
{
TApplication a // Konstruktor rufen
(
"Hello",
hInstance,
hPrevInstance,
lpCmdLine,
nCmdShow
);
a.Run(); // Anwendung ausführen
return a.Status; // Beenden
}
```

Box zur Verabschiedung, OW01.CPP

Wie kann man nun etwas an diesem Basisvorgang verändern? Zunächst könnte man versuchen, eine bestehende Funktion von `TApplication` zu überladen; z.B. `CanClose()`.

Wie kann man den Benutzer fragen, ob er aufhören will? Man öffnet eine `MessageBox` und teilt dieser Box mit, welchen Titel ("Verabschiedung"), welche Botschaft ("Wollen Sie wirklich aufhören?") sie haben soll und welche Schalter (`MB_YESNO | MB_ICONQUESTION`) sie enthalten soll.

Die möglichen anderen Werte der Schalter und der Antworten der `MessageBox` erfährt man am besten über die eingebaute on-line-Hilfe.

Die `MessageBox` muß aber, wie viele andere Fenster auch, wissen, von welchem Fenster sie abstammt. `hInstance` ist ein Mitglied von `TModule` und enthält das aktuelle `HANDLE` der Applikation.

In diese neuen Klasse können die Funktionen zum Zeichnen des Fensters `InitMainWindow()` und `CanClose()` zum löschen des Fensters überladen werden.

Das Überladen des Fensters selbst generiert einen anderen als den Anfangstext durch Öffnen eines Fensters `TWindow`. Zum Verlassen des Programms einer `MessageBox` mit einem JA/NEIN-Knopf. `IDYES` bedeutet, daß der OK-Knopf selektiert wurde.

Anmerkung: Dieses und die folgenden Programme benutzen inline-Elementfunktionen, das sind solche, die innerhalb der Klassendeklaration definiert werden oder solche, die das zusätzliche Schlüsselwort `inline` vorangestellt bekommen. Die zweite Art mit explizitem `inline` erfordert mehr Platz und ist für Anfänger nicht so übersichtlich; daher wurden die Funktionen in die Klassendefinition mitaufgenommen. Es wird aber darauf hingewiesen, daß diese Schreibweise nur bei sehr kurzen Funktionen verwendet wird, hier aber der Einfachheit halber für alle Funktionen verwendet wurde. >>>

```
//OW01.CPP
#define WIN31
#include <owl.h>

class MyApp : public TApplication
{
public:
    MyApp
    (
        LPSTR AName,
        HANDLE hInstance,
        HANDLE hPrevInstance,
        LPSTR lpCmdLine,
        int nCmdShow
    ) : TApplication
    (
        AName,
        hInstance,
        hPrevInstance,
        lpCmdLine,
        nCmdShow
    )
    {
    }

    virtual void InitMainWindow()
    {
        MainWindow = new TWindow (NULL, "Hallo");
    }
    BOOL CanClose()
    {
        return MessageBox
        (
            MainWindow->HWindow,
            "Wollen Sie wirklich aufhören?",
            "Verabschiedung",
            MB_YESNO | MB_ICONQUESTION
        ) == IDYES;
    }
};

int PASCAL WinMain
// ... wie in OW0.CPP
}
```

Text im Fenster, OW02.CPP

Jede Anwendung `TApplication` hat so etwas wie einen Konstruktor, nämlich eine Funktion, die aufgerufen wird, wenn eine Instanz dieser Anwendung kreiert wird: `InitInstance()`;

```
void InitInstance()
{
    TApplication::InitInstance();
    HDC DC;
    char S[100];
    sprintf(S, "Begrüßungstext");
    DC = GetDC(MainWindow->HWindow);
    TextOut(DC, 100, 100, S, strlen(S));
    ReleaseDC(MainWindow->HWindow, DC);
}
```

aber auch eine Funktion, die beim allerersten Aufruf dieser Anwendung aufgerufen wird: `InitApplication()`.

Dieses `InitInstance()` druckt einen Begrüßungstext aus. Ganz so einfach geht das aber nicht:

Man schreibt nicht direkt in den Bildschirm, sondern in einen Display-Kontext `HDC`, den man von der Anwendung mit `GetDC()` erhält. Alle Ausgaben werden auf diesen Display-Kontext bezogen, indem dieser als Argument in der Ausgabefunktion `TextOut()` übergeben wird. Das Wichtigste aber ist, den Display-Kontext durch `ReleaseDC()` wieder freizugeben, sonst reagiert Windows ähnlich wie bei einem Stapelüberlauf auf Grund fehlender `POP`-Befehle.

Das Hauptfenster in `TApplication` ist einfach ein leeres Fenster. Will man etwas in das Fenster schreiben, muß man die Funktion `InitMainWindow()` überladen und ein eigenes Fenster einsetzen, dem man den Namen der Anwendung gibt.

Wie kann ein Fenster aktiv werden?

In der Klasse des neuen Fensters `TMyWindow` wird die virtuelle Funktion `WMLButtonDown()` überladen. Diese Funktion erhält als eine BORLAND-C-Besonderheit einen Dispatcher-Index in eckigen Klammern nachgestellt der die Nummer der Botschaft enthält, die diese Funktion aktivieren soll, hier `WM_LBUTTONDOWN`. Der zusätzliche Wert `WM_FIRST` ist lediglich ein Offset für alle Konstanten, der praktisch in jedem Dispatcher-Index vorkommt [`WM_LBUTTONDOWN`, `+ WM_FIRST`].

ACHTUNG: Diese Syntax ist ungenormt, vereinfacht zwar die Sache ungemein, ist aber überhaupt nicht portabel. Bei Microsoft-C müssen die Botschaften im Event-Handler mit `switch`-Anweisungen an die richtige Funktion verteilt werden, ähnlich, wie es auch bei der `TURBO-VISION`-Bibliothek (Fenster-Bibliothek für DOS-Programme) der Fall ist.

In diesem Beispiel wird die aktuelle Cursorposition beim Drücken der linken Maustaste ausgegeben.

```
HDC DC;
char S[100];
sprintf(S, "(%d,%d)", Msg.LP.Lo, Msg.LP.Hi);
DC = GetDC(HWindow);
TextOut(DC, Msg.LP.Lo, Msg.LP.Hi, S, strlen(S));
ReleaseDC(HWindow, DC);
```

```
//OW02.CPP
#include <stdio.h>
#include <string.h>
#define WIN31
#include <owl.h>

class TMyWindow : public TWindow
{
public:
    TMyWindow
    (
        PWindowsObject AParent,
        LPSTR ATitle
    ) : TWindow (AParent, ATitle)
    {
    }

    virtual void WMLButtonDown(RTMessage Msg)
    = [WM_FIRST + WM_LBUTTONDOWN]
    {
        HDC DC;
        char S[100];
        sprintf(S, "(%d,%d)", Msg.LP.Lo, Msg.LP.Hi);
        DC = GetDC(HWindow);
        TextOut(DC, Msg.LP.Lo, Msg.LP.Hi, S, strlen(S));
        ReleaseDC(HWindow, DC);
    }
};

class MyApp : public TApplication
// wie in OW01.CPP
```

Schreiben in das Fenster, OW03.CPP

Wie kann man in ein Fenster etwas schreiben? Dazu gibts die Funktion `Paint()` der Klasse `TWindow`, die für unseren Zweck in der Klasse `TMyWindow` überladen wird und einen neuen Begrüßungstext ausgibt.

Versuchen Sie einmal das Fenster in seiner Größe zu verändern!

Die Koordinatenangaben verschwinden alle wieder, der Begrüßungstext bleibt! Warum? Die Funktion `Paint()` wird bei jeder Veränderung des Bildschirms gerufen, die Funktion `WMLButtonDown()` dagegen nur bei jedem Mausklick.

```
// OW03.CPP
#include <stdio.h>
#include <string.h>
#define WIN31
#include <owl.h>

class TMyWindow : public TWindow
{
public:
    TMyWindow
    (
        PTWindowsObject AParent,
        LPSTR ATitle
    ) : TWindow (AParent, ATitle)
    {
    }

    virtual void WMLButtonDown(RTMessage Msg)
    = [WM_FIRST + WM_LBUTTONDOWN]
    {
        HDC DC;
        char S[100];
        sprintf(S, "(%d,%d)", Msg.LP.Lo, Msg.LP.Hi);
        DC = GetDC(HWindow);
        TextOut(DC, Msg.LP.Lo, Msg.LP.Hi, S, strlen(S));
        ReleaseDC(HWindow, DC);
    }

    virtual void Paint(HDC DC, PAINTSTRUCT& Paintinfo)
    {
        char S[100];
        sprintf(S, "Begrüßungstext");
        DC = GetDC(HWindow);
        TextOut(DC, 50, 50, S, strlen(S));
        ReleaseDC(HWindow, DC);
    }
};

class MyApp : public TApplication
// ... wie in OW02.CPP
```

Das Fenster weiß, was am Bildschirm steht, OW04.CPP

Was kann man also tun, daß sich der Fensterinhalt beim Verschieben mitändert? Man muß sich merken, was der Benutzer auf den Bildschirm gemalt hat! Wie? Nichts einfacher als das! Mit den Hilfsmitteln der Container-Bibliothek schafft man das mit ein paar Zeilen, wir wollen es besonders schön machen und verwenden für einen Punkt eine eigene Struktur mit hübsch überladenen Operatoren. Die Punkte selbst speichern wir in einer Liste genau in der Reihenfolge der Eingabe:

```
struct Pt
{
    int x;
    int y;
    Pt(int x1, int y1) {x=x1; y=y1;}
    Pt() {x=0; y=0;}
    int operator!=(Pt p) { return !(p.x==x && p.y==y); }
    int operator==(Pt p) { return (p.x==x && p.y==y); }
};
```

Nicht nur, daß die Struktur Pt ein Koordinatenpaar x,y enthält, es gibt auch einen Default-Konstruktor Pt(), der die Punktkordinaten auf Null setzt und einen Konstruktor mit Anfangswerten. Die beiden Operatoren == und != stellen fest, ob zwei Punkte gleich sind. Sie werden im restlichen Code vergeblich nach der Anwendung von == suchen. Wer braucht ==?

Wir werden die erforderliche Liste natürlich nicht selbst schreiben, sondern eine fertige Liste aus der Container-Bibliothek, am besten eine solche mit Templates verwenden. Man muß bedenken, daß eine vordefinierte Liste immer etwas mehr kann als man gerade benötigt. Sie enthält also auch Vergleichsmöglichkeiten, die wir eigentlich gar nicht brauchen. Für einfache Variablentypen, für die der Vergleichsoperator implizit definiert ist, also für ganze Zahlen und Gleitkommazahlen ist der Vergleich auf Gleichheit oder Ungleichheit kein Problem. Ist es aber schon für Strukturen. Wann sind zwei Strukturen gleich? Genau diese Frage stellt der Compiler, wenn man versucht, die Punktstruktur der vordefinierten Template-Klasse aufzudrücken. Nun, jetzt hat er sie, und in der Zeile

```
BI_ListImp<Pt> Points;
```

wird die Liste definiert. Der erste Punkt wird mit

```
Points.add(Pt(0,0));
```

alle weiteren Punkte mit

```
Points.add(Pt(Msg.LP.Lo,Msg.LP.Hi));
```

in die Liste eingefügt. Der erste Punkt (0,0) dient als Markierung für das Ende der Liste. Mit diesem Behelf merken wir uns alle Benutzereingaben (beachten Sie: 3 Zeilen) und geben die so gespeicherte Liste in der Funktion Paint() wieder aus. Für das Auslesen der Liste wird ein zerstörungsfreier Iterator verwendet:

```
BI_ListIteratorImp<Pt> i(Points);
Pt p, p0;
while ((p=i++) != p0)
{
    sprintf(S, "(%d,%d)", p.x, p.y);
    TextOut(DC, p.x, p.y, S, strlen(S));
}
```

Hier findet man auch die Abfrage auf das Ende der Liste mit dem Operator !=. Dieses Beispiel wurde dem in der Beschreibung von Object-Windows nachempfunden. Der Unterschied zur Dokumentation ist, daß dort zur Speicherung der Bildschirmpunkte ein Array verwendet wird und hier, bei der einfachen Liste der erwähnte Fehler beim Neuzeichnen auftritt.

```
//OW04.CPP
// ... wie in OW03.CPP
class TMyWindow : public TWindow
{
    struct Pt
    {
        int x;
        int y;
        Pt(int x1, int y1) {x=x1; y=y1;}
        Pt() {x=0; y=0;}
        int operator!=(Pt p) { return !(p.x==x && p.y==y); }
        int operator==(Pt p) { return (p.x==x && p.y==y); }
    };

    BI_ListImp<Pt> Points;

public:
    TMyWindow
    (
        PTWindowsObject AParent,
        LPSTR ATitle
    ) : TWindow (AParent, ATitle)
    {
        Points.add(Pt(0,0));
    }

    virtual void WMLButtonDown(RTMessage Msg)
    = [WM_FIRST + WM_LBUTTONDOWN]
    {
        HDC DC;
        char S[10];
        sprintf(S, "(%d,%d)", Msg.LP.Lo, Msg.LP.Hi);
        DC = GetDC(HWindow);
        TextOut(DC, Msg.LP.Lo, Msg.LP.Hi, S, strlen(S));
        Points.add(Pt(Msg.LP.Lo, Msg.LP.Hi));
        ReleaseDC(HWindow, DC);
    }

    virtual void Paint(HDC DC, PAINTSTRUCT& Paintinfo)
    {
        char S[20];
        sprintf(S, "Begrüßungstext");
        DC = GetDC(HWindow);
        TextOut(DC, 50, 50, S, strlen(S));
        BI_ListIteratorImp<Pt> i(Points);
        Pt p, p0;
        while ((p=i++) != p0)
        {
            sprintf(S, "(%d,%d)", p.x, p.y);
            TextOut(DC, p.x, p.y, S, strlen(S));
        }
        ReleaseDC(HWindow, DC);
    }
};

class MyApp : public TApplication
// ... wie in OW03.CPP
```


Linien zeichnen, OW05.CPP

Windows ist ja aber eine grafische Benutzeroberfläche, die wir bisher nur für Texte mißbraucht haben. Dieses Beispiel ist aber gleich gut auch für Grafik verwendbar, indem wir statt der Koordinatenangabe eine Linie ziehen. Dazu müssen wir aber nicht nur den Mausklick, sondern auch die Mausbewegung berücksichtigen. Die Maus soll nur zeichnen, wenn die Maustaste gedrückt ist.

Innerhalb der Fensterklasse TMyWindow muß man sich daher zwei Dinge merken

1. Den aktuellen Display-Kontext DragDC, der beim Drücken der linken Maustaste gesetzt wird. und
 2. den aktuellen Zustand der linken Maustaste in ButtonDown, da die Maus auch ohne Taste bewegt wird.
- Die beiden neuen Funktionen WMouseMove() und WMLButtonDown() und WMLButtonUp() sind neu gestaltete Funktionen.

Das Zeichnen gelingt ja ganz gut aber beim Erneuern mit Paint() werden noch die alten Punktkoordinaten gesetzt.

Das Programm OW05.CPP ist hier nicht abgebildet sondern nur die endgültige Version OW06.CPP:

```
//OW06.CPP
#include <stdio.h>
#include <string.h>
#define WIN31
#include <owl.h>
#include <listimp.h>

class TMyWindow : public TWindow
{
    HDC DragDC;
    BOOL ButtonDown;
    struct Pt
    {
        int x;
        int y;
        Pt(int x1, int y1) {x=x1; y=y1;}
        Pt() {x=0; y=0;}
        int operator!=(Pt p) { return !(p.x==x && p.y==y); }
        int operator==(Pt p) { return (p.x==x && p.y==y); }
    };
    BI_ListImp<Pt> Points;

public:
    TMyWindow
    (
        PTWindowsObject AParent,
        LPSTR ATitle
    )
    : TWindow (AParent, ATitle)
    {
        Points.add(Pt(0,0));
        ButtonDown = FALSE;
    }

    virtual void WMouseMove(RTMessage Msg)
    = [WM_FIRST + WM_MOUSEMOVE]
    {
        if (ButtonDown)
        {
            LineTo(DragDC,Msg.LP.Lo,Msg.LP.Hi);
            Points.add(Pt(Msg.LP.Lo,Msg.LP.Hi));
        }
    }

    virtual void WMLButtonDown(RTMessage Msg)
    = [WM_FIRST + WM_LBUTTONDOWN]
    {
        if (!ButtonDown)
        {
            ButtonDown = TRUE;
            DragDC = GetDC(HWindow);
            MoveTo(DragDC,Msg.LP.Lo,Msg.LP.Hi);
            Points.add(Pt(Msg.LP.Lo,Msg.LP.Hi));
        }
    }

    virtual void WMLButtonUp(RTMessage Msg)
```

```
    = [WM_FIRST + WM_LBUTTONUP]
    {
        if (ButtonDown)
        {
            ButtonDown = FALSE;
            ReleaseDC(HWindow,DragDC);
        }
    }

    virtual void Paint(HDC DC, PAINTSTRUCT& Paintinfo)
    {
        char S[20];
        sprintf(S,"Begrüßungstext");
        DC = GetDC(HWindow);
        TextOut(DC,50,50,S,strlen(S));
        BI_ListIteratorImp<Pt> i(Points);
        Pt p, p0;
        while ((p=i++) != p0)
        {
            LineTo(DC,p.x,p.y);
        }
        ReleaseDC(HWindow,DC);
    }
};

class MyApp : public TApplication
{
public:
    MyApp
    (
        LPSTR AName,
        HANDLE hInstance,
        HANDLE hPrevInstance,
        LPSTR lpCmdLine,
        int nCmdShow
    ) : TApplication
    (
        AName,
        hInstance,
        hPrevInstance,
        lpCmdLine,
        nCmdShow
    )
    {
    }

    virtual void InitMainWindow()
    {
        MainWindow = new TMyWindow (NULL,Name);
    }
    BOOL CanClose()
    {
        return MessageBox
        (
            MainWindow->HWindow,
            "Wollen Sie wirklich aufhören?",
            "Verabschiedung",
            MB_YESNO | MB_ICONQUESTION
        ) == IDYES;
    }
};

int PASCAL WinMain
(
    HANDLE hInstance,
    HANDLE hPrevInstance,
    LPSTR lpCmdLine,
    int nCmdShow)
{
    MyApp *a= new MyApp
    (
        "Hello",
        hInstance,
        hPrevInstance,
        lpCmdLine,
        nCmdShow
    );
    a->Run();
    return a->Status;
}
□
```

In WINWORD zweiseitig drucken, *Eine Frage, drei Antworten*

Wolfgang Dreising, Thomas Rösken



Ich möchte aus WINWORD heraus beidseitig drucken. Gibt es eine Möglichkeit, daß zuerst die ungerade Seiten ausgedruckt werden, damit man dann, den Papierstapel noch einmal einlegen kann, um dann auf die Rückseite die entsprechenden geraden Seiten zu drucken?

Dirk Gräfe, 2:243/4301.4 Bochum, voll anne Ruhr dran



Das geht mit einem WordBasic-Makro (-Programm). Darauf aufbauend kannst Du Dir wohl das gewünschte Programm zusammenstellen. Was Umfang=3 bedeutet und die anderen Parameter, findest Du in der Onlinehilfe (mit dem Cursor auf das Wort "DateiDruck" gehen und F1 drücken).

```
Sub Main
  for x = 1 to 30 step 2
    y$=str$(x)
    DateiDruck .Art=0, .Exemplare=1, .Umfang=3, .von=y$, \
      .bis=y$
  next
End Sub
```

Manfred Murer, 2:301/246.1



Hier wie ich's mache:

```
Sub MAIN
  Dim dlg As DokumentStatistik
  GetCurValues dlg
  TotPages = Val(dlg.Seiten)

  For i = 1 To TotPages Step 2
    DateiDruck .Umfang = 3, .Von = Str$(i), .Bis = Str$(i)
  Next i
  MsgBox "Papierstapel einlegen..."

  For i = 2 To TotPages Step 2
    DateiDruck .Umfang = 3, .Von = Str$(i), .Bis = Str$(i)
  Next i
End Sub
```

Jens Heidemann, 2:243/6121.16 Windows? ...nur unter OS/2



Ok, auf vielfachen Wunsch nochmals (sorry für die lange Mail) das Makro für Winword das doppelsetig druckt (klappt bei mir mit dem Deskje):

```
Sub Main
  Drucker = 1 ' 1 für Laserjet II, III und ähnliche
              ' 2 für Laserjet IIp, IIp und ähnliche

  titel$ = "Beidseitiger Ausdruck"
  If drucker = 1 Then
    dr1$ = "letzte"
    dr2$ = "unten"
  Else
    dr1$ = "erste"
    dr2$ = "oben"
  EndIf

  Dim dlg As DokumentStatistik
  GetCurValues dlg
  Anz = Val(dlg.Seiten)
  Redim dlg As UserDialog
  Art = 0
```

```
Start:
Begin Dialog UserDialog 290, 75, titel$
  Text 11, 50, 40, 18, "&von:"
  Text 96, 50, 40, 18, "&bis:"
  TextBox 50, 47, 40, 18, .Von$
  TextBox 126, 47, 40, 18, .Bis$
  OptionGroup .Art
    OptionButton 10, 7, 115, 18, "&Alle Seiten"
    OptionButton 10, 28, 85, 18, "&Seiten:"
  OKButton 180, 10, 95, 25
  CancelButton 180, 40, 95, 25
End Dialog

dlg.Art = Art

On Error Goto ende

Dialog dlg

If dlg.Art = 0 Then
  von = 1
  Bis = Anz
Else
  Von = Val(dlg.Von$)
  Bis = Val(dlg.Bis$)
  Art = 1
End If

Von1 = Von
If drucker = 1 Then
  Bis1 = Bis
Else
  Von2 = Von
EndIf

If Von = 0 Or Von > Bis Or Bis > Anz Then
  MsgBox
  "Das Dokument hat nicht diese Anzahl Seiten oder die " + \
  "Reihenfolge ist falsch herum angegeben.", titel$, 16
  Goto Start
ElseIf Von = Bis Then
  x = MsgBox("Nur eine Seite zu drucken!", titel$, 65)
  If x = 0 Then Goto Ende
  DateiDruck .Umfang = 3, .Von = Str$(Von), .Bis = Str$(Bis)
Else
  'Drucken der geraden Seiten
  If drucker = 1 Then
    If InStr(Str$(Bis / 2), ".") <> 0 Then Bis1 = Bis - 1
    For i = Bis1 To Von Step - 2
      DateiDruck .Umfang = 3, .Von = Str$(i), .Bis = Str$(i)
    Next i
  Else
    If InStr(Str$(Von / 2), ".") <> 0 Then Von2 = Von + 1
    For i = Von2 To Bis Step 2
      DateiDruck .Umfang = 3, .Von = Str$(i), .Bis = Str$(i)
    Next i
  EndIf

  'Info
  If InStr(Str$(Von / 2), ".") = 0 Then
    x = MsgBox
    ("Wenn der Ausdruck fertig ist, die " + dr1$ + \
    " Seite vom Stapel entfernen und Stapel mit " + \
    "der Aufschrift nach " + \
    dr2$ + " wieder in das Zusatzfach einlegen.",
    titel$, 65)
  Else
    x = MsgBox
    ("Wenn der Ausdruck fertig ist, " + \
    "Stapel mit der Schrift " + \
    "nach " + dr2$ + \
    " wieder in das Zusatzfach einlegen", titel$, 65)
  EndIf
  If x = 0 Then Goto Ende
  'Drucken der ungeraden Seiten
  If InStr(Str$(Von / 2), ".") = 0 Then Von1 = Von + 1
  For i = Von1 To bis Step 2
    DateiDruck .Umfang = 3, .Von = Str$(i), .Bis = Str$(i)
  Next i
End If
Ende:
End Sub
□
```

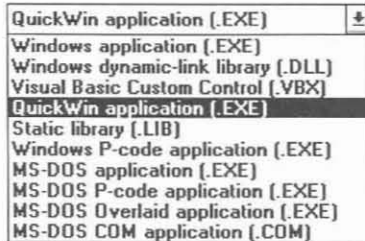
Windows-Programmierung, MICROSOFT

Franz FIALA, N, TGM

DSK-404\MS

QuickWin

Für einen sanften Übergang von der DOS- zur Windows-Programmierung sorgt QuickWin. Was dabei allein zu tun bleibt, ist die Auswahl des richtigen Ausgangsformats im Optionen-Menü:



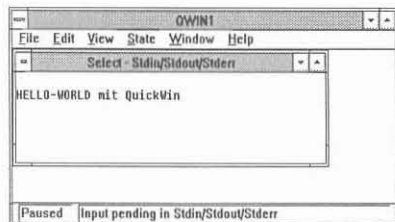
Hier sieht man auch gleich, welche weiteren Formate existieren.

Das ist alles! D.h. das "HELLO WORLD"-Programm mit QuickWin ist identisch mit dem "HELLO-WORLD"-Programm in DOS und schaut so aus:

```
// QWIN1.CPP
```

```
#include <iostream.h>

void main()
{
    cout << "\nHELLO-WORLD mit QuickWin\n";
    cin.get();
}
```



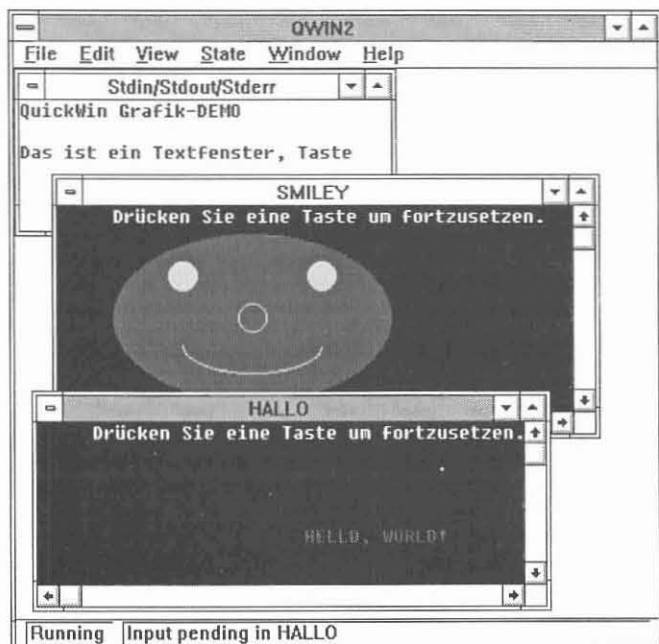
Es gibt bereits ein Elternfenster, das gleichzeitig mehrere QuickWin-Fenster verwaltet, hier nur das eine. Jede QuickWin-Applikation kann auch abgeleitete Fenster haben.

Das Schöne an QuickWin ist, daß keinerlei Änderungen der

üblichen Gewohnheiten bei der DOS-Programmierung nötig sind und, daß neben einfachen Ein- und Ausgaben auch Grafik und Farbe kein Hindernis sind, wie das folgende Beispiel zeigt:

Mehrere Fenster, Grafikfenster

Dieses Programm öffnet 3 Fenster und wartet auf einen Tastendruck. Bildlaufleisten können verwendet werden. Die Bildschirme können auf volle Bildschirmgröße erweitert werden. Die darin befindlichen Texte können mit Hilfe des Menüs im Elternfenster QWIN2 kopiert werden.



```
// QWIN2.CPP öffnet mehrere Grafik-Fenster
```

```
#include <iostream.h>
#include <graph.h>

void main()
{
    int handle;

    // Nachricht an ein Textfenster (stdin/stderr/stdout)
    cout << "QuickWin Grafik-DEMO\n";

    // Warten
    cout << "\nDas ist ein Textfenster, Taste\n";
    cin.get();

    // mit _setvideomode() wird ein Grafikfenster eröffnet
    // es hat den Anfangsnamen GRAFIC1.
    _setvideomode( _MAXRESMODE );
    handle = _wggetactive( ); // jedes Fenster hat ein handle
    _wgclose( handle ); // welches die Bearbeitung eben
                        // dieses Fensters ermöglicht
                        // in diesem Fall wird das Fenster
                        // geschlossen, damit ein neues
                        // mit einem eigenen Namen vergeben
                        // werden kann

    // Ein neues Fenster erstellen
    handle = _wgopen( "SMILEY" );
    _wgsetactive( handle );
    _setvideomode( _MAXRESMODE );

    // Kopf zeichnen
    _setcolor( 13 );
    _ellipse( _GFILLINTERIOR, 40, 20, 240, 140 );

    // Augen, Nase und Mund
    _setcolor( 14 );
    _ellipse( _GFILLINTERIOR, 80, 40, 100, 60 );
    _ellipse( _GFILLINTERIOR, 180, 40, 200, 60 );
    _ellipse( _GBORDER, 130, 70, 150, 90 );
    _arc( 90, 80, 190, 120, 115, 100, 165, 100 );

    // Warten auf Taste
    _settextposition( 1, 6 );
    _outtext("Drücken Sie eine Taste um fortzusetzen.");
    _inchar( );

    // Eröffnen eines Fensters im Textmodus
    handle = _wgopen( "HALLO" );
    _wgsetactive( handle );
    _setvideomode( _TEXTC80 );

    // Hintergrundfarbe und Textfarbe setzen
    _setbkcolor( 1 );
    _settextcolor( 2 );

    // Textausgeben
    _settextposition( 6, 25 );
    _outtext( "HELLO, WORLD!" );

    // Farben einstellen
    _setbkcolor( 1 );
    _settextcolor( 15 );

    // Warten auf Taste
    _settextposition( 1, 6 );
    _outtext("Drücken Sie eine Taste um fortzusetzen.");
    _inchar( );

    // Cursor ist auf der Zeile unterhalb der Nachricht
    _settextposition( 7, 1 );
}
```

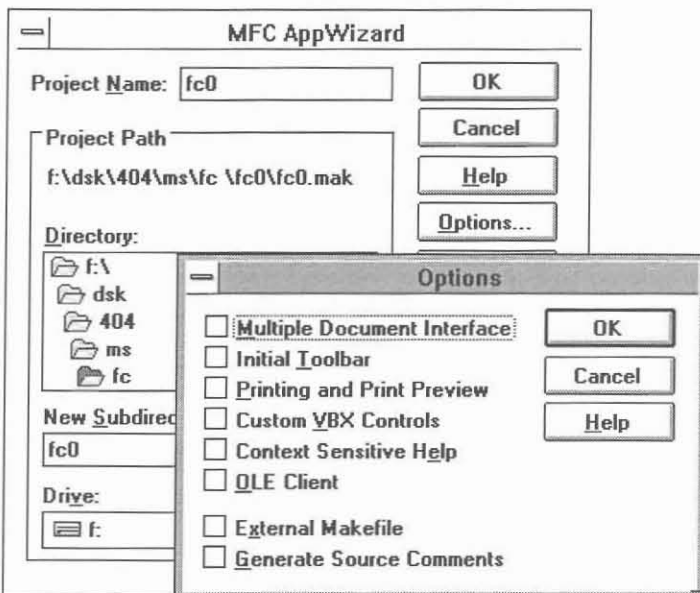
Das Programm QWIN2 (es heißt im Original SMILEY) ergibt das nebenstehende Bildschirmbild, wobei die Fenster zwecks gleichzeitiger Darstellbarkeit frei arrangiert wurden. Die üblichen DOS-Grafikbefehle können mit geringen Abweichungen verwendet werden.

Foundation Classes

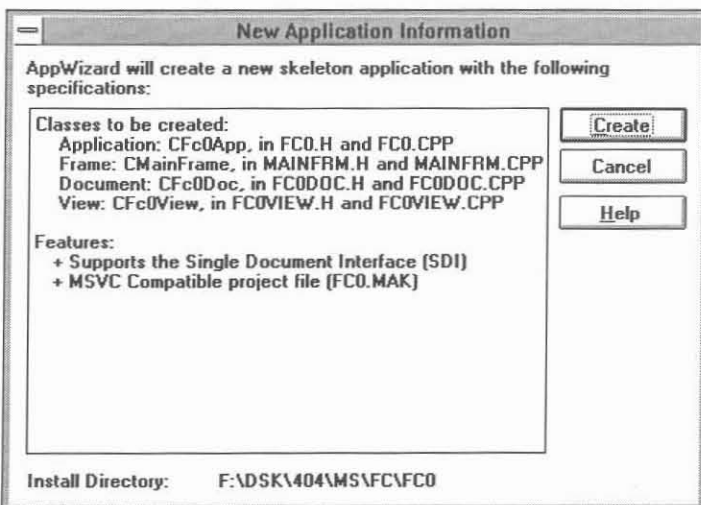
So einfach wie mit Quick-Windows geht es in C++ nicht, dafür kann man die ganze Bandbreite der Windows-Möglichkeiten nutzen. Ein Beispiel für die Leistungsfähigkeit dieser Bibliothek: Ein immerwiederkehrendes, lästiges Problem für Programmierer: Speichern und Laden von Daten. In der Windows-Oberfläche im Dateien-Menü mit den Menüpunkten Speichern, Speichern unter und Laden zu finden. Eine einzige Funktion muß für die eigenen spezifischen Daten geschrieben werden `Serialize()` in der Klasse `CDocument` und schon kann diese Funktion benutzergesteuert wie mit jedem anderen Windows-Programm verwendet werden.

Aber zurück zum Anfang: Um den Einstieg zu erleichtern und um den administrativen Overhead für die jedenfalls erforderlichen Basisklassen zu erzeugen, wird gleich ein Hexenmeister, der Application Wizard bemüht, dem - je nach Bedarf - der Class Wizard folgt.

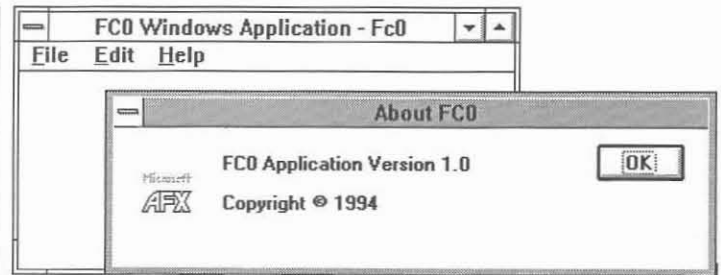
Ein Programm ist nicht mehr eine einzelne Datei sondern es sind gleich mehrere, die in einem Projekt zusammengefaßt sind. Ein Projekt wird eröffnet, indem ein Projektname vergeben wird, der gleichzeitig der Name des gleichnamigen Subdirectory ist. In den Optionen bestimmt man, welche Eigenschaften dieses Projekt haben soll. Je nach dem wird der automatisch generierte Code länger oder kürzer ausfallen. Zum Ausprobieren wählen wir nichts von den schönen Dingen, denn wir wollen die einfachste mögliche Anwendung zeigen:



Diese Auswahl wird mit einem Zwischenfenster bestätigt:



Nachdem man mit **Project-Build** die Dateien kompiliert und linkt und danach mit **Project-Execute** ausführt, meldet sich unsere Application und hat auch schon ein "About-Fenster".



Festhalten: Im Subdirectory FC0 sind 31 Dateien mit 2,78MB entstanden sowie ein weiteres Subdirectory RES, das das verwendete Symbol für



die Anwendung enthält:

Text im Fenster

Eine Anwendung besteht aus mindestens einer `CDocument`- und einer `CView`-Klasse beziehungsweise davon abgeleiteten Klassen. Diese Ableitung besorgt der Application-Wizard für uns, denn wir erhalten zwei Dateien `FC0DOC.CPP` und `FC0VIEW.CPP` und auch die zugehörigen Header-Dateien und darin die abgeleiteten Klassen `CFc0Doc` und `CFc0View`. Der für diese Anwendung gewählte Name `FC0` wird in die Bezeichnungen der Klassen unverwechselbar mitaufgenommen.

Was sind nun CDocument und CView?

Ganz grob kann man sagen, daß `CDocument` die Daten verwaltet und `CView` die Darstellung und die Kommunikation mit dem Benutzer. Ein Programm, das nur rechnet und mit Dateien kommuniziert, benötigt im Prinzip keine `CView`-Klasse (das nur zur Illustration).

Immer wenn der Bildschirm neu gezeichnet wird, wird die Funktion `OnDraw()` der Klasse `CView` aufgerufen. Will man daher etwas darstellen, dann muß diese Funktion das besorgen. Übergibt man das zu Zeichnende dem Konstruktor von `CView`, dann wird es zwar einmal gezeichnet aber kaum bewegt man das Fenster, verschwindet der Text oder die Zeichnung, da die Klasse `CView` nur einmal gezeichnet wird und dann nur mehr die Funktion `OnDraw()`.

Für das Schreiben eines Textes in den Bildschirm benötigen wir einen sogenannten **Display-Context**, der mit der `OnDraw()`-Funktion gleich mitgeliefert wird. Alle Ausgaben an den Bildschirm erfolgen über einen solchen Display-Context:

```
void CFc0View::OnDraw(CDC* pDC)
{
    pDC->TextOut(100,100,"HALLO");
    pDC->MoveTo(100,100);
    pDC->LineTo(50,50);
}
```

Der zwischen den Klammern stehende Code ist alles, das man selbst schreiben muß, um den HALLO-Text und eine Linie von 100,100 nach 50,50 zu ziehen.

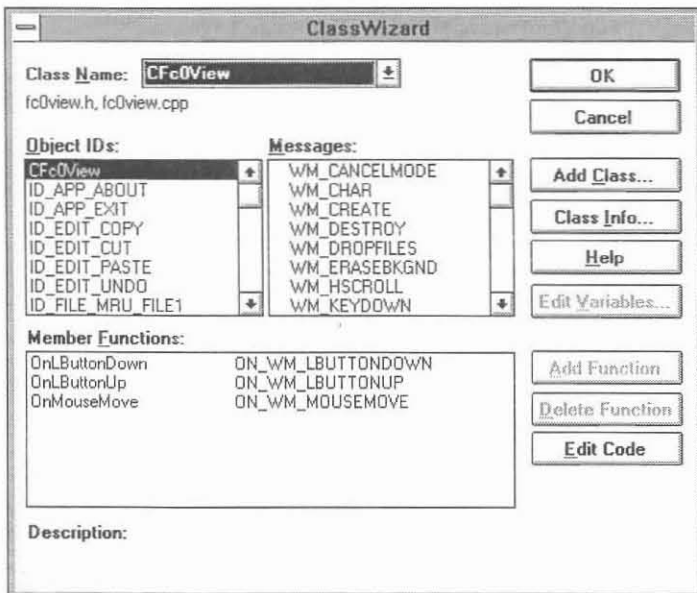
Unser Ziel ist es, mit der Maus einen schönen Begrüßungstext zu zeichnen, wie es sich bei einer grafischen Benutzeroberfläche gehört.

Der Cursor soll beim Klicken der linken Taste einen Punkt zeichnen, und danach beim Ziehen der Maus eine Linie. Das Loslassen der Maus unterbricht die Linie. Ein wichtiges Problem ist daher die Kommunikation mit dem Benutzer.

Kommunikation mit dem Benutzer

Gemäß dieser Aufgabenstellung müßte jeweils beim Betätigen der Maus etwas in den Datenstrukturen verändert werden. Daher benötigen wir je eine Funktion für jede der Mausektivitäten.

Wir haben uns gemerkt, daß alle Kommunikation mit dem Benutzer über eine `CView`-Klasse abläuft. Normalerweise müßte man jetzt in der Header-Datei den entsprechenden Prototyp und in der CPP-Datei den dazugehörigen Code einfügen. Nicht bei den Foundation-Classes: Man bemüht einfach einen neuen Hexenmeister, den Class-Wizard im Menü Browse und erhält folgendes Bild.



Im linken Rahmen sieht man einzelnen Objekte und im Kasten daneben, die Windows-Nachrichten (WM...), die von dem betreffenden Objekt bearbeitet werden können. Aus dieser Nachrichtenliste wählt man jene Nachrichten aus, die durch das Programm bearbeitet werden sollen; in unserem Fall sind das WM_LBUTTONDOWN, WM_LBUTTONUP und WM_MOUSEMOVE. Wenn wir sie anklicken, generiert der Wizard zugehörige Member-Funktionen, die man im unteren Rahmen aufgezählt sieht, der Wizard erledigt das aber auch in den zugehörigen Header-Dateien und auch in der Code-Datei. Nichts mehr zu tun. Es entstehen die Funktionen OnLButtonDown(), OnLButtonUp() und OnMouseMove().

Als erste Versuch wollen wir lediglich die aktuelle Cursorposition bei jedem Mausklick anzeigen lassen:

```
void CFc0View::OnLButtonDown(UINT nFlags, CPoint point)
{
    CClientDC dc(this);

    char S[100];
    sprintf(S, "(%d,%d)", point.x, point.y);
    dc.TextOut(point.x, point.y, S);
}
```

Die Klasse CClientDC ist ein solcher Display-Context, der diesen auf unser Fenster beschränkt. Alle Ausgaben (hier TextOut) erfolgen über diesen Display-Context.

Gezeichnetes Hello

Jetzt wollen wir zeichnen, ohne das Gezeichnete zu merken (Das hat die wichtige Konsequenz, daß die Zeichnung beim Verändern des Fensters verschwindet).

Wir merken uns in einer Variablen ButtonDown, ob die Maustaste gerade gedrückt ist, außerdem merken wir uns die aktuelle Position des Cursors in oldpoint. Diese Variablen werden in der Headerdatei unseres View, FC0VIEW.H als private-Variable eingetragen:

```
BOOL ButtonDown;
CPoint oldpoint;
```

Der nun folgende Kodeteil ist das, was man einfügen muß:

```
void CFc0View::OnLButtonDown(UINT nFlags, CPoint point)
{
    CClientDC dc(this);

    char S[100];
    sprintf(S, "(%d,%d)", point.x, point.y);
    dc.TextOut(point.x, point.y, S);
    if (!ButtonDown)
    {
        ButtonDown = TRUE;
        oldpoint=point;
    }
}

void CFc0View::OnLButtonUp(UINT nFlags, CPoint point)
{
}
```

```
if (ButtonDown)
{
    ButtonDown = FALSE;
}
}

void CFc0View::OnMouseMove(UINT nFlags, CPoint point)
{
    if (ButtonDown)
    {
        CClientDC dc(this);
        dc.MoveTo(oldpoint);
        dc.LineTo(point);
        oldpoint=point;
    }
}
```

Immer wenn die linke Maustaste gedrückt wird, erfolgt die Ausgabe der Mausposition; Außerdem wird die Position in oldpoint festgehalten. Bei Mausbewegung wird die Funktion OnMouseMove() aktiviert und eine Linie von oldpoint zum aktuellen Punkt gezogen aber nur, wenn eben die Taste gedrückt ist: if (ButtonDown).

Man kann den gewünschten Begrüßungstext schon zeichnen! Versuchen Sie aber jetzt während des zeichnens das Fenster zu verändern, etwa indem Sie es am Rand ergreifen und vergrößern. Die Zeichnung verschwindet, da das Fenster neu gezeichnet wird und die Folge der Punkte nirgendwo festgehalten wurde. Lediglich der Begrüßungstext "HALLO" und die Probelinie der Funktion OnDraw() bleiben erhalten.

Daher ist es unser abschließendes Ziel, die Punktfolge in geeigneter Form zu speichern. Wir benutzen dazu eine Liste der Klasse CObList. [Auf die Problematik typenreiner Datenstrukturen am Beispiel einer Liste wird im Sonderdruck SON-3 besonders eingegangen. Hier sei nur soviel wiederholt, daß mit der Version 2.0 der Foundation-Classes entweder typenrichtige Datenstrukturen gebildet werden können, wenn sie von der Klasse CObject abgeleitet wurden, wie das auf unsere Punkte vom Typ Cpoint zutrifft. Handelt es sich um andere Datentypen, muß statt CObList CPtrList verwendet werden, die aber lediglich void-Pointer verwaltet und beim Rücklesen aus der Liste einen cast-Operator benötigt. Ja, und void-Pointer sind gerade das, was man mit den sogenannten template-s vermeiden kann, etwas, was auch in den Foundation-Classes, d.h. besser gesagt in Visual-C++ wünschenswert wäre.]

Speichern eine Folge von Ereignissen in einer Liste

Wir wollen Punkte des Typs CPoint speichern. Die Klasse CPoint ist aber nicht von CObject abgeleitet, daher kann eine typenrichtige Liste mit CPoint-Objekten nicht unmittelbar gebildet werden. Dazu gibt es zwei Wege: 1. Erzeugen eines neuen Typs, z.B. myPoint, der von CObject abgeleitet ist und einen CPoint enthält und Benutzung der Liste CObList oder 2. Benutzung der Liste CPtrList und Verwendung des cast-Operators beim Lesen der Elemente.

Der Einfachheit halber benutzen wir die zweite Möglichkeit. Zunächst deklarieren wir eine Liste des Typs CPtrList in der Klassendefinition von CFc0View:

```
CPtrList *liste;
```

und generieren die Liste im Konstruktor

```
CFc0View::CFc0View()
{
    ButtonDown = FALSE;
    liste = new CPtrList(sizeof(CPoint));
}
```

und vernichten sie im Destruktor

```
CFc0View::~CFc0View()
{
    delete liste;
}
```

Der Code für die Funktionen für die Mausbetätigung ändern wir wie folgt. Das Wiederholen der Zeichnung besorgt wieder die Funktion OnDraw(), in der die Liste ausgelesen wird.

```
void CFc0View::OnDraw(CDC* pDC)
{
    pDC->TextOut(100,100, "HALLO");
    pDC->MoveTo(100,100);
}
```

```

pDC->LineTo(50,50);

if (!liste->IsEmpty())
{
    POSITION pos;
    for( pos = liste->GetHeadPosition(); pos != NULL; )
    {
        CPoint *p=(CPoint*)liste->GetNext( pos );
        pDC->LineTo(*p);
    }
}

void CFc0View::OnLButtonDown(UINT nFlags, CPoint point)
{
    CClientDC dc(this);

    // char S[100];
    // sprintf(S,"%d,%d",point.x,point.y);
    // dc.TextOut(point.x,point.y,S);

    if (!ButtonDown)
    {
        ButtonDown = TRUE;
        oldpoint=point;
        liste->AddHead(new CPoint(point));
    }
}

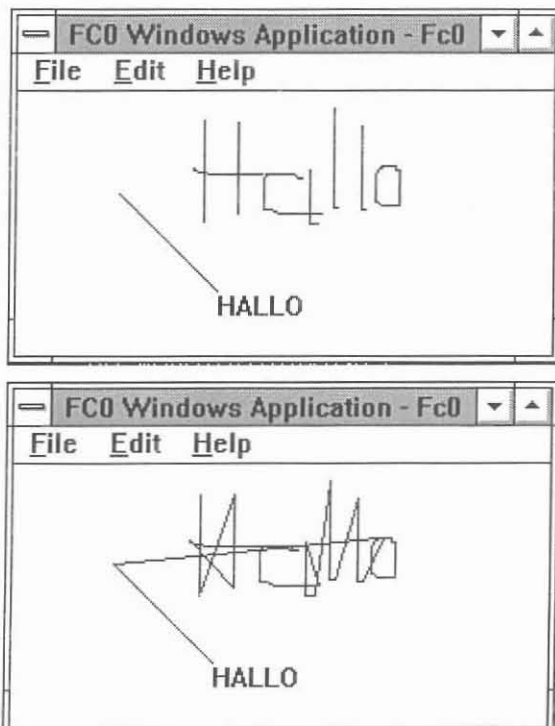
void CFc0View::OnLButtonUp(UINT nFlags, CPoint point)
{
    if (ButtonDown)
    {
        ButtonDown = FALSE;
    }
}

void CFc0View::OnMouseMove(UINT nFlags, CPoint point)
{
    if (ButtonDown)
    {
        CClientDC dc(this);
        dc.MoveTo(oldpoint);
        dc.LineTo(point);
        oldpoint=point;
        liste->AddHead(new CPoint(point));
    }
}

```

Zusammenfassung

Es ist nicht allzuviel des Kode, den man für ein einfaches Fenster schreiben muß; man ist aber auch noch weit davon entfernt, die Zusammenhänge wirklich zu verstehen.



C++-Beiträge in den PC-NEWS

PC-NEWS 20, S.22..25: Zeigt den Übergang von Strukturen zu Klassen an Hand eines Beispiels mit einer Struktur eines Gehaltsempfängers, bestehend aus einem Namen (String) und einem Gehalt.

PC-NEWS 21, S.21..23: Ableitung, Vererbung, virtuelle Funktionen, abstrakte Basisklassen

PC-NEWS 22, S.41: Zur Schreibweise in C, S.41..42: Friends, S 42..44: Überladen, S.44: Referenzvariable, S.45: Neues in C++, S.45..48: OOP mit C++, S.49: Wie lernt man am besten C++? S.49..51: C++ im Unterricht, Ein Beitrag zur Sprachendiskussion.

PC-NEWS 23, S.26..30: Materialien für den C-Unterricht, S.59..62: Entwicklung einer Klasse von Parametersubstitution über Parameterübergabe, Polymorphie, S.63..64: Erweiterungen von C in C++.

PC-NEWS 25, S.33..36: Eine kleine Grafikbibliothek, zeigt insbesondere zum Abschluß den Begriff Polymorphie.

PC-NEWS 30: S.47..64: Eine IO-Klasse für hardwarenahes Programmieren.

PC-NEWS 35: S.40..44. **ADIM**-Skripten im Vergleich, darunter auch C++-Skriptum, sowie Fehlerkorrektur dazu.

PC-NEWS 35: S.52..53: const und static in C und C++.

PC-NEWS 35: S.54..68: Hauptspeicherzugriffe mit eigenen Klassen.

Weitere Hinweise für C-Programmierung

ADIM-Bände 40 (C), 50 (C++)

PC-NEWS-31, Von PASCAL zu C

Der schnelle Weg zum hohen C, PCN-SON-003

Real programmers programs never work right the first time. But if you throw them on the machine they can be patched into working in only a few 30-hours debugging sessions.

Real programmers don't use Fortran. Fortran is for wimpy engineers who wear white socks, pipe stress freaks, and crystallography weenies. They get excited over finite state analysis and nuclear reactor simulation.

Real programmers don't use COBOL. COBOL is for wimpy application programmers.

Real programmers never work 9 to 5. If any **real programmers** are around at 9 am, it's because they were up all night.

Real programmers don't write in BASIC. Actually, no programmers write in BASIC, after the age of 12.

Real programmers don't document. Documentation is for simps who can't read the listings or the object deck.

Real programmers don't write in Pascal, or Bliss, or Ada, or any of those pinko computer science languages. Strong typing is for people with weak memories.

Real programmers know better than the users what they need.

Real programmers think structured programming is a communist plot.

Real programmers don't use schedules. Schedules are for manager's toadies. Real programmers like to keep their manager in suspense.

Real programmers think better when playing adventure.

Real programmers don't use PL/I. PL/I is for insecure momma's boys who can't choose between COBOL and Fortran.

Real programmers don't use APL, unless the whole program can be written on one line.

Hardwarenahes Programmieren in C und C++

Franz Fiala, N, TGM

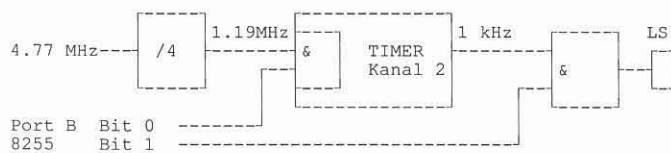
DSK-401

Teil 6a: Der Lautsprecher im PC

Nachdem wir in den vergangenen Folgen den PC insgesamt besprochen haben, können wir nun ins Detail gehen - und beginnen mit dem Lautsprecher. Ein Bit nur, das wir steuern müssen, und eine Menge Probleme.

Der Lautsprecher ist im PC am gefahrlosesten programmierbar, da seine Einstellung keine weiteren lebenswichtigen Funktionen berührt, wenn man nur ein paar Grundsätze beherzigt. Außerdem wird der Lautsprecher vom BIOS und vom MSDOS nicht weiter unterstützt: wir haben also gleichzeitig die Gelegenheit zu zeigen, wie man eine neue Hardwareeinheit ins System integrieren kann (eine der nächsten Folgen). Bevor wir den Lautsprecher tatsächlich ein- und ausschalten, benötigen wir einige Schaltungsdetails:

Lautsprecherbeschaltung



Der Lautsprecher ist also über Port B des Bausteins 8255 programmierbar. Das BIOS sorgt beim Booten dafür, daß der Port B auch wirklich als Ausgang geschaltet ist. Die Adressen von Port-B entnehmen wir der Tabelle in den **PC-NEWS-27**, S.32, bzw. **PC-NEWS-30**, S.52. Bit 0 schaltet mit 1 den Timer Kanal 2 ein, Bit 1 verbindet den Timer mit dem Lautsprecher. Auch der Timer wurde durch das BIOS bereits voreingestellt, wir müssen uns zunächst um seine Programmierung nicht sorgen.

Timer steuert Lautsprecher

Aus der Skizze sehen wir, daß unabhängig von der tatsächlichen Prozessorgeschwindigkeit Kanal 2 des Timers immer mit einem Viertel von 4,77 MHz getaktet wird. In der Grundeinstellung erzeugt der Kanal 2 des Timers immer das übliche Piepsen mit 1kHz.

Wie erfolgt nun das Einschalten des Lautsprechers in C? Da Port-B die Adresse 0x61 hat, könnte man einfach schreiben:

```
outportb(0x61,3);
```

Das wäre zwar - was die erforderlichen Verbindungen in der obigen Skizze betrifft - goldrichtig, aber wir müssen beim Schalten beachten, daß die Bits 2-7 von Port B andere, für den PC lebenswichtige Funktionen haben und daher nicht verändert werden dürfen. Zuerst muß man den Zustand von Port B feststellen und nur die Bits 0 und 1 setzen und dann das Resultat wieder hinausschreiben. Etwa so:

```
c = inportb(0x61); c |=3; outportb(0x61,c);
```

Wie in Folge 4 (**PC-NEWS-30**, S.48) beschrieben wurde, sind die Ein- und Ausgabebefehle oft von Compiler zu Compiler unterschiedlich. Um einigermaßen compilerunabhängig arbeiten zu können, wurden diese Funktionen durch Makros `IN_PORT()` und `OUT_PORT()` ersetzt (erkennbar an der Großschreibweise), die in der Headerdatei `portable.h` ausgeführt sind. Außerdem wurde darauf geachtet, daß alle Typen in Großschrift und teilweise abgekürzt geschrieben werden können. Das wird in der Datei `mytypes.h` sichergestellt. Also kann `VOID` statt `void` und `UCHAR` statt `unsigned char` geschrieben werden. Durch diese strukturierte Schreibweise wird auf die Bedeutung einzelner Bezeichner hingewiesen. Da Typen grundsätzlich an bestimmten Stellen des Programms zu liegen kommen (links von Variablenamen), erhält das Listing eine gut lesbare Struktur. Diese Technik wird vor allem bei Windows-Programmen angewendet.

Mit diesen Vorbemerkungen können wir ein Programm zum Ein- und Ausschalten des Lautsprechers schon schreiben:

HC0601

- Auf Taste warten
- Timer mit LS verbinden
- Auf Taste warten
- Timer von LS trennen

```
/* HC06LS0.C */
/*
 * Tonerzeugung durch voreingestellten Timer
 *
 */
```

```
#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <mytypes.h>
#define P_IO
#include <portable.h>

VOID main(VOID)
{
    UCHAR c;

    clrscr();
    printf("Ein- und Ausschalten des Timers "
           "und des Lautsprechers\n\n");
    printf("Taste verbindet den Lautsprecher "
           "dem Timer \n");
    getch();

    c = IN_PORT(0x61); /* 8255, Port-B */
    c |= 2;           /* LS ein */
    OUT_PORT(0x61,c);

    c = IN_PORT(0x61); /* 8255, Port-B */
    c |= 1;           /* Timer-Kanal-2 ein */
    OUT_PORT(0x61,c);

    printf("Taste trennt den Lautsprecher "
           "und den Timer\n");
    getch();

    c = IN_PORT(0x61);
    c &= ~1;         /* Timer-Kanal-2 aus */
    OUT_PORT(0x61,c);

    c = IN_PORT(0x61);
    c &= ~2;         /* LS aus */
    OUT_PORT(0x61,c);

    clrscr();
}
```

Die im Programm vorkommenden Sequenzen zum Ein- und Ausschalten des Tones bieten sich für eine Formulierung in einem Unterprogramm an. Erinnern wir uns, daß wir für die einzelnen Hardwareadressen in der Header-Datei `IODEF.H` eigene Bezeichnungen eingeführt haben, die uns auch bei späterer Lektüre des Programms erklären, was denn eigentlich 0x61 bedeutet. Die gewählte Abkürzung hieß `XT_PPI_B`, was soviel heißt wie: seit dem XT bekannt, PPI ist die Kurzbezeichnung für den 8255 und B ist unser Port.

```

VOID spk_on(VOID)
{
    UCHAR c = IN_PORT(XT_PPI_B); /* Lautsprecher EIN */
    c |= 2;
    OUT_PORT(XT_PPI_B,c);
}

VOID spk_off(VOID)
{
    UCHAR c = IN_PORT(XT_PPI_B); /* Lautsprecher EIN */
    c &= ~2;
    OUT_PORT(XT_PPI_B,c);
}

VOID tim2_on(VOID)
{
    UCHAR c = IN_PORT(XT_PPI_B); /* Timer EIN */
    c |= 1;
    OUT_PORT(XT_PPI_B,c);
}

VOID tim2_off(VOID)
{
    UCHAR c = IN_PORT(XT_PPI_B); /* Timer AUS */
    c &= ~1;
    OUT_PORT(XT_PPI_B,c);
}

```

```

/* HC06LS1.C */
/*
 * Tonerzeugung durch voreingestellten Timer
 * =====
 * Mit eigenen Schaltfunktionen
 */

```

```

#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <mytypes.h>
#define P_IO
#include <portable.h>
#include <iodef.h>
#include <mylib.h>
#ifdef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
    clrscr();
    printf("Ein- und Ausschalten des Timers "
           "und des Lautsprechers\n\n");
    printf("Taste schaltet den Lautsprecher "
           "und den Timer ein\n");
    getch();

    spk_on(); tim2_on();

    printf("Taste schaltet den Lautsprecher "
           "und den Timer aus\n");
    getch();

    spk_off(); tim2_off();
    clrscr();
}

```

Nachdem wir die nützlichen Unterprogramme aus dem Hauptprogramm herausgelöst haben, fügen wir sie in die Bibliothek `mylib.h` ein. Damit das Hauptprogramm sowohl zum Testen der einzelnen Bibliotheksroutinen als auch zum Testen der fertigen Bibliothek ohne Änderung des Codes möglich wird, wird ein kleiner Trick angewendet.

Zuerst verlagert man die Routinen `spk_on()`, `spk_off()`, `tim2_on()` und `tim2_off()` in eine Datei `spk.c` im Verzeichnis `SOURCE`. In diese Datei werden auch noch weitere Funktionen kommen. Dann ergänzt man in der Headerdatei `mylib.h` um die beiden neuen Prototypen und merkt an, in welcher Quelldatei sie sich befinden. Um die Bibliothek zu bilden, müßte man jetzt die gesamte Bibliothek neu kompilieren. Das sollte man aber erst tun, wenn man über die Funktionsfähigkeit der neuen Routinen durch eingehende Tests Bescheid weiß. Dieses und die folgenden Programme sind jetzt unsere Testprogramme.

Während dieser Testphase sollen die neuen Funktionen mitkompiliert werden (sie können sich ja bei kleinen Änderungen noch inhaltlich verändern) und später, wenn die Tests abgeschlossen sind, sollen die Testprogramme genauso ausgeführt werden können.

Der kleine Trick besteht darin, daß in der Headerdatei `mylib.h` ein Makro `MYLIB` existiert, das - wenn es definiert ist - die Inklusion der Datei `spk.c` verhindert. Während des Testens wird also `spk.c` inkludiert, dann nicht mehr, die Funktionen werden während des Linkens aus der fertigen Bibliothek `MYLIB.LIB` geholt. Die folgenden Zeilen kommen im Programm `HC06LS1.C` dazu; die beiden Funktionen entfallen, siehe das hier nicht abgedruckte Programm `HC06LS1A.C`.

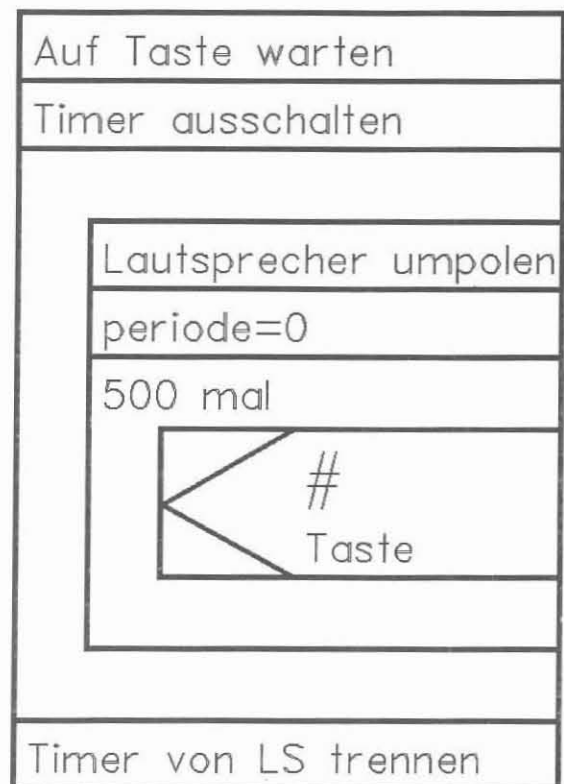
Diesen Vorgang des Auslagerns von Funktionen in eine wachsende Bibliothek wird in der Folge wiederholt in gleicher Weise durchgeführt.

Tonerzeugung durch Softwareverzögerung, HC06LS2.C

Selbstverständlich kann der Ton im Lautsprecher auch durch programmgesteuertes Umpolen am letzten UND-Gatter erzeugt werden. Die Frequenz ergibt sich aus der Verzögerung, die das Programm bewirkt. Es ist das keine empfehlenswerte Methode zur Erzeugung definierter Zeitabstände, denn dazu eignet sich natürlich der Timer wesentlich besser; aber das Ergebnis gibt uns wichtige Aufschlüsse über die Arbeitsweise des Rechners.

Die Verzögerung entsteht durch die `for`-Schleife, die 500 mal durchlaufen wird. Da das bei verschiedenen Rechner verschieden lang dauern wird, muß man hier je nach CPU und Takt eine andere Größe einsetzen, will man dasselbe Ergebnis erhalten. Ein wesentlicher Zeitfaktor der Schleife ist der Aufruf von `bioskey()`, damit man den Ton auch unterbrechen kann.

hc06ls02



Anmerkung: Im obigen Struktogramm ist leider ein Fehler passiert, der Abbruch durch Tastendruck muß **nach** der 500mal durchlaufenen Schleife und **nicht innerhalb** erfolgen.

```
VOID spk_toggle(VOID) /* Lautsprecher umpolen */
{
    UCHAR c = IN_PORT(XT_PPI_B);
    c ^= 2;
    OUT_PORT(XT_PPI_B,c);
}
```

```
/* HC06LS2.C */
/*
 * Ton durch Verzögerungsschleife
 * =====
 */
#include <dos.h>
#include <conio.h>
#include <bios.h>
#include <stdio.h>
#include <mytypes.h>
#define P_IO
#include <portable.h>
#include <iodef.h>
#include <mylib.h>
#ifdef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
    UINT t;          /* Zeitzähler */

    clrscr();
    printf("Erzeugen eines Tones "
           "durch Umpolung des Lautsprechers\n\n");
    printf("Periodendauer durch "
           "Verzögerungsschleife verlängert\n");
    printf("Start mit Taste, Abbruch mit Taste\n\n");
    getch();

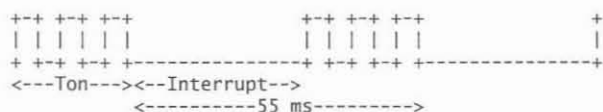
    tim2_off();      /* Timer ausschalten */

    for(;;)
    {
        spk_toggle(); /* Lautsprecher umpolen */
        for (t=0; t<500; t++); /* kleine Verzögerung */

        if (bioskey(1)) /* Abbruch mit Taste */
        {
            getch();
            break;
        }

        spk_off();
        tim2_off(); /* LS und Timer trennen */
        printf("Warum 'gurgelt' der Ton?\n");
        getch();
        clrscr();
    }
}
```

Leider ist das Ergebnis nicht ganz 'sauber', der Ton ist keinesfalls rein, es hört sich an, als würde sich auch noch ein anderes Programm bei der Steuerung des Lautsprechers mitmischen. Und genauso ist es! Man ist leider nicht allein auf der Welt, nicht einmal im PC. Interrupts können laufend stattfinden und der Timer-Kanal 1 tut das auch etwa 18mal pro Sekunde.



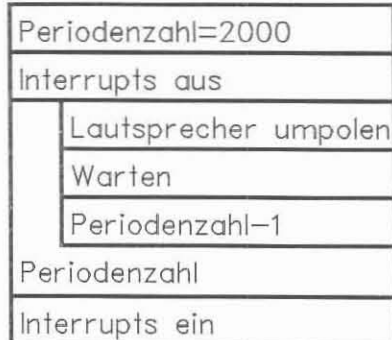
Ausschalten der Interrupts, HC06LS3.C

Was ist zu tun? Glücklicherweise kann man die Interrupts ausschalten. Der C-Compiler stellt uns zwei Funktionen dafür zur Verfügung: enable() und disable().

Leider ergibt das eine weitere Schwierigkeit, daß mit dem einfachen Ausschalten aller Interrupts auch derjenige für die Tastatur mitausschaltet wird und daher der Abbruch per Tastendruck nicht mehr möglich ist.

Als vorläufige Lösung wird eine Softwarewarteschleife eingesetzt, die den Ton vorzeitig beendet und danach die Interrupts wieder einschaltet. Das Struktogramm ist ein Auszug, der die Handhabung der Interrupts zeigt:

hc06ls03



```
/* HC06LS3.C */
/*
 * Ton mit Verzögerungsschleife, ohne Interrupts
 * =====
 */
#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <mytypes.h>
#define P_IO
#include <portable.h>
#include <mylib.h>
#ifdef MYLIB
#include "..\SOURCE\SPK.C"
#endif

VOID main(VOID)
{
    ULONG periodenzahl;
    UINT t;

    clrscr();
    printf("Erzeugen eines Tones "
           "durch Umpolung des Lautsprechers\n\n");
    printf("Periodendauer durch "
           "Verzögerungsschleife verlängert\n");
    printf("Start mit Taste, "
           "Konstante Anzahl von Perioden\n\n");
    getch();

    spk_on();
    tim2_off();

    periodenzahl = 2000; /* erster Versuch */
    do
    {
        spk_toggle();
        for (t=0; t<1000; t++);
        periodenzahl--;
    }
    while (periodenzahl);

    printf("Ton gurgelt noch immer, "
           "Achtung, jetzt Interrupts aus!\n\n");
    getch();

    periodenzahl = 2000;
    disable();
    do
    {
        spk_toggle();
        for (t=0; t<1000; t++);
        periodenzahl--;
    }
    while (periodenzahl);
    enable();

    printf("Ton sollte jetzt sauber, "
           "wie beim Timer gewesen sein\n");
    getch();

    /* Lautsprecher und Timer AUS */
    spk_off(); tim2_off();
}
}
```


Kombiton, HC06LS4.C

Wir haben also zwei unabhängige Möglichkeiten kennengelernt, Töne zu erzeugen: Zuerst die Verbindung des Timers mit dem Lautsprecher, dann die direkte Umpolung des Lautsprechers. Als Gag kann man natürlich beide Möglichkeiten kombinieren, was im Programm HC06LS4.C gezeigt wird.

```

/* HC06LS4.C */
/*
 * Kombiniertes Ton
 * =====
 */
#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <mytypes.h>
#define P_IO
#include <portable.h>
#include <mylib.h>
#ifndef MYLIB
#include "..\SOURCE\SPK.C"
#endif

VOID main(VOID)
{
    ULONG zyklen;
    UINT t;

    clrscr();
    printf("Kombiniertes Ton aus Timer+Umpolung\n");
    printf("Periodendauer verlängert\n");
    printf("Start mit Taste, "
        "Konstante Anzahl von Perioden\n");
    getch();

    printf("Zuerst nur der Timer-Ton (Taste)\n");

    spk_on(); tim2_on();

    printf("Nach Taste kombiniert mit Umpolung, "
        "endet selbst\n");
    getch();
    zyklen = 3000;
    disable();
    do
    {
        spk_toggle();
        for (t=0; t<1000; t++);
        zyklen--;
    }
    while (zyklen);
    enable();

    printf("Jetzt wieder nur der Timer (Taste)\n");
    getch();
    /* Lautsprecher und Timer AUS */
    spk_off(); tim2_off();
}

```

6b: Programmierung des Timers

Eine weit bessere Möglichkeit zur Tonerzeugung stellt uns der Timer zur Verfügung. Wir müssen aber wissen, wie man den Timer programmiert.

Zunächst sollten wir uns darüber klar sein, daß der Timer ein alter Herr ist und man einen solchen wichtigen Baustein heute anders konzipieren würde. Der Timer besteht aus drei voneinander unabhängigen Kanälen, die in unserem PC fest verdrahtet sind. Kanal 0 ist für die Zeitgebung eingestellt löst in Abständen von 53 ms einen Interrupt aus; dieser Kanal wird uns noch später beschäftigen. Kanal 1 ist für den RAM-Refresh verantwortlich und normalerweise nicht verwendbar. Kanal 2 ist auf 1 kHz eingestellt und kann über Gatter mit dem Lautsprecher verbunden werden; die zugehörige Logik haben wir im ersten Teil kennengelernt.

Alle Kanäle sind über die Adressen 0x40, 0x41 und 0x42 anzusprechen und werden über einen gemeinsamen Steuerkanal 0x43 programmiert. Welcher Timer-Kanal gerade programmiert wird, bestimmen zwei Bits im Steuerwort. Wie die Programmierung erfolgt und was die einzelnen Bits bedeuten zeigt das folgende Programm:

```

/* HC06TII.C */
/*
 * Timer-Demonstration
 * =====
 */
#include <stdio.h>
#include <conio.h>
#define P_IO
#include <portable.h>
#include <mylib.h>
#ifndef MYLIB
#include "..\SOURCE\SPK.C"
#endif

VOID main(VOID)
{
    clrscr();
    printf("Der Timer arbeitet völlig unabhängig "
        "von gleichzeitig laufender Software\n");
    printf("Taste\n");
    getch();
    spk_on(); tim2_on();
    printf("Während der Timer für uns arbeitet, "
        "können wir machen was wir wollen\n"
        "Das ist der normale Timer-Ton mit 1 kHz "
        "(Taste)\n");
    getch();
    printf("Der Timer kann auch umprogrammiert werden!\n"
        "Steuerwort auf 0x43\n"
        "7 6 5 4 3 2 1 0 Bit\n"
        "0 0 Kanal 0 Uhr: 53 ms, "
        "0x36, 0x00, 0x00 ??\n"
        "0 1 Kanal 1 DRAM-Refresh, 15 us\n"
        "1 0 Kanal 2 LS: 1 kHz, "
        "0xb6, 0x33, 0x05\n"
        "0 0 Latch\n"
        "0 1 LSB folgt\n"
        "1 0 MSB folgt\n"
        "1 1 LSB-MSB folgt\n"
        "0 0 0 Interrupt on count\n"
        "0 0 1 One-Shot\n"
        "0 1 0 Generator\n"
        "0 1 1 Rechteck-Generator\n"
        "1 0 0 Triggered-Strobe-Soft\n"
        "1 0 1 Triggered-Strobe-Hard\n"
        "0 Binary\n"
        "1 BCD\n"
        "Versuchen wir die Zeitkonstante 0x1000 (Taste)\n");
    getch();
    OUT_PORT(0x43, 0xb6);
    OUT_PORT(0x42, 0x00); /* Zeitkonstante LSB */
    OUT_PORT(0x42, 0x10); /* Zeitkonstante MSB */
    printf("Jetzt wieder Originalprogrammierung 0x0533 "
        "(Taste)\n");
    getch();
    OUT_PORT(0x43, 0xb6);
    OUT_PORT(0x42, 0xa8); /* Zeitkonstante LSB */
    OUT_PORT(0x42, 0x04); /* Zeitkonstante MSB */

    printf("Taste\n");
    getch();

    spk_off(); tim2_off();
}

```

Die Timer-Kanäle werden beim Einschalten des Rechners durch das BIOS programmiert. Die Zeitkonstante von Kanal 0 wird auf 0 eingestellt. Dabei entsteht die längste Periodendauer von 65536 Taktschritten.

Die Programmierung erfolgt - wie Sie am obigen Beispiel sehen können - so, daß zuerst das Steuerwort in die Adresse 0x43 geschrieben wird und danach die Zeitkonstante mit dem niederwertigen Byte zuerst in das entsprechende Timerregister geschrieben wird. Diese Abfolge ist aber durch Veränderung des Steuerwortes änderbar.

Damit so sensible Vorgänge wie das Steuern des Timers nicht immer wieder neu erfunden werden müssen, werden wir die Kommunikation eigenen Funktion übertragen, die wir in der Datei `tim.c` zusammenfassen:

`tim2_init()` versetzt den Timer-Kanal-2 in den Anfangszustand:

```

VOID tim2_init(VOID)
{
    OUT_PORT(PC_TIM|TIM_CTL, 0xb6);
    /* Zeitkonstante niederwertiger Teil */
    OUT_PORT(PC_TIM|TIM_CH2, 0x33);
    /* Zeitkonstante höherwertiger Teil */
    OUT_PORT(PC_TIM|TIM_CH2, 0x05);
}

```

tim2_set() initialisiert den Timer-Kanal-2 auf einen beliebigen Anfangswert; damit können wir die Tonhöhe des ausgegebenen Tones steuern.

```
VOID tim2_set(UINT count)
{
    OUT_PORT(PC_TIM|TIM_CTL,0xb6);
    /* Zeitkonstante niederwertiger Teil */
    OUT_PORT(PC_TIM|TIM_CH2,(UCHAR)count);
    /* Zeitkonstante höherwertiger Teil */
    OUT_PORT(PC_TIM|TIM_CH2,(UCHAR)(count>>8));
}
```

tim2_read() liefert den aktuellen Zählerstand des Timers zurück.

```
UINT tim2_read(VOID)
{
    UINT h,l;
    l=IN_PORT(PC_TIM|TIM_CH2);
    h=IN_PORT(PC_TIM|TIM_CH2);
    return ((h<<8)+l);
}
```

tim2_wait() wartet time Perioden

```
VOID tim2_wait(UINT time)
{
    UCHAR h;

    tim_on();
    tim2_set(0xff00);
    do
    {
        do
        {
            /* niederwertigen Teil lesen und verwerfen */
            h=IN_PORT(PC_TIM|TIM_CH2);
            h=IN_PORT(PC_TIM|TIM_CH2);
        }
        while (h);
    }
    while (time--);
    tim_off(); tim2_init();
}
```

Timer auslesen, HC06TI2.C

Obwohl die Timer vorteilhafterweise interruptgesteuert arbeiten sollten, da der Zählzeitpunkt dann exakt eingehalten wird, kann man auch den Zählerstand der Timer ablesen. Dabei muß man allerdings beachten, daß man nicht einen bestimmten Zählerstand erwarten darf, wie das folgende Programm zeigt.

```
/* HC06TI2.C */
/*
 * Timer auslesen
 * =====
 */
#include <bios.h>
#include <stdio.h>
#include <conio.h>
#define P_IO
#include <portable.h>
#include <mylib.h>
#ifndef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
    UINT timer;

    clrscr();
    printf("Der Timer kann auch ausgelesen werden\n");
    printf("Timer auslesen: (Taste)\n");
    getch();
    tim2_on();
    tim2_set(0xffff);
    do
    {
        timer = tim2_read();
        printf("%4x ",timer);
    }
    while (bioskey(1)==0);
    getch();
    tim2_off();
    tim2_init();
    getch();
}
```

Dieses Programm initialisiert den Timer-Kanal-2 auf die längste mögliche Zeitkonstante 0xffff und versucht in einer do..while-Schleife den Zählerstand am Bildschirm darzustellen. Wie oft das während eines Zählzyklus gelingt, sehen Sie aus der folgenden Darstellung:

```
a9e2 a146 992a 910c 88e6 80c6 7644 6dfa 65dc 5dc0 55a4 4d88 4566 3d46 3526 2d08
1fb2 171e e00f f3c8 eb34 e336 da0e dice c9aa c18e b96e b152 a930 a112 98f4 90d6
813e 78a0 707e 6862 6042 581e 4ffe 47dc 3f8e 379c 2f7c 275c d4e8 c968 c114 b8f8
aba6 a30c 9aea 92ca 8aa8 8288 7a62 7242 6a26 5b04 538e 4b76 4350 3b36 331a 2afe
1daa 1516 cfe 4d8 fc8c f498 ea16 eid4 d9ae d190 c96e c152 b934 bl0e 9ed6 956e
```

Man sieht daß ein Timer von der Anfangszahl beginnt und nach unten zählt. Erreicht er die Null, löst er je nach Betriebsart ein Signal an seinem Ausgang aus. Im Falle des Kanals 2 polt er dabei den Lautsprecher um. Im Falle des Kanals 0 löst er gemäß Verdrahtung einen Interrupt aus.

Dabei ist aber zu bedenken, welche Konfiguration beim Start des Programms verwendet wird (Hier: 486SL-25, Borland-C in einem Widows-Fenster mit VGA-Display). Andere Konfigurationen können ein ganz anderes Bild ergeben. Im Extremfall (XT, 4MHz, langsames Display) kann man per Programm dem Zählzyklus gar nicht mehr folgen, wenn man zur Darstellung eines bestimmten Zählerstandes länger braucht als der Zähler für einen Durchlauf.

Feststellung des Null-Durchgangs, HC06TI3.C

Wie stellt man auf Grund des Auslesens des Timers fest, wann der Timer 'bei 0 vorbeikommt'? Kanal-2 kann ja wegen der festen Verbindung mit dem Lautsprecher keinen Interrupt auslösen, der uns das präzise melden könnte. Im folgenden Programm wird gezeigt, daß man sich daran orientieren kann, daß der neue Zählerstand grundsätzlich kleiner ist als der alte, da der Timer ein Abwärtszähler ist, nur, wenn die Zählung wieder von vorn beginnt, also der Start-Zählerstand eingelesen wird, ist das nicht der Fall.

```
/* HC06TI3.C */
/*
 * Timer Nulldurchgang feststellen
 * =====
 */
#include <bios.h>
#include <stdio.h>
#include <conio.h>
#include <mylib.h>
#include <mylib.h>
#ifndef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
    UINT timer2;
    UINT timer2alt;

    clrscr();
    printf("Timer nach Nulldurhgang auslesen: (taste)\n");
    getch();
    tim2_on();
    tim2_set(0xffff);

    timer2alt=tim2_read();
    do
    {
        timer2=tim2_read();
        if ((timer2alt>>8)<(timer2>>8))
        {
            printf("%4x ",timer2);
        }
        timer2alt=timer2;
    }
    while (bioskey(1)==0);
    getch();

    tim2_off();
    tim2_init();
    getch();
}
```

Als Ergebnis dieses Programms erhält man (je nach Rechnergeschwindigkeit) folgendes Bild:

```
Timer nach Nulldurhgang auslesen: (taste)
ffda ffda ffce fffc ffd2 fffa 7be ffc2 ff02 ffc4 fff0 ffcc ec58 ffe2 ffe2 ffd4
ff02 fffa ffc4 ffc0 ffd8 ffd6 ffce ffe4 ffec fff8 fff8 ffd8 ff06 ffc2 ffec f356
ffe4 ffca ffd8 ffc2 ffd2 ffea ffe8 ffbe ffce ffd6 ffe6 ffce ff06 ffe4 fffe fff6
ffe8 ffe8 ffbe ffd2 ffc2 ffce fffa fff6 ff02 ff00 ffe2 ffd2 ffc4 ffce ffe6 fff4
```

Im Prinzip liegen die Erkennungszeiten des Nulldurchgangs bei maximal 256 Taktschritten (0xff02); dabei gibt es aber immer wieder einige Ausreißer (0xec58 und 0x07be), die durch das periodische Auftreten des Interrupts von Kanal 0 hervorgerufen werden. Während dieser Interruptzeiten hat unser Programm eine Zwangspause und kann nicht im normalen Tempo weiterarbeiten. Abhilfe: Interrupts abdrehen, wenn man die Zeit genau wissen will.

Zeit messen, HC06TI4.C

Wir nutzen also die Fähigkeit des Timers, um kleine Funktionen zu formulieren, die uns eine der Funktion `delay()` ähnliche (aber feiner abgestufte) Zeitverzögerung ermöglicht.

```
VOID tim2_waitus(UINT time)
{
    UINT ende=0xffff-time;

    disable();
    tim2_on();
    tim2_set(0);

    do
    {
    }
    while (tim2_read()>ende);

    tim2_off(); tim2_init();
    enable();
}
```

```
VOID tim2_waitms(UINT time)
{
    disable();

    do
    {
        tim2_waitus(1193);
    }
    while (time--);

    enable();
}
```

```
VOID tim2_waits(UINT time)
{
    disable();

    do
    {
        tim2_waitms(1000);
    }
    while (time--);

    enable();
}
```

Beachten Sie, daß jeweils beim Eintreten in die jeweiligen Funktionen die Interrupts mit `disable()` ausgeschaltet werden und beim Verlassen mit `enable()` wieder eingeschaltet werden.

Die Grundverzögerung wird in der innersten Routine `tim2_waitus()` erreicht. Die Zählleinheit ist nicht genau eine Mikrosekunde sondern $0,84 \mu\text{s} = 1/1200000 \text{ s}$. Daher muß der Multiplikator der Funktion `tim2_waitms()` auch 1193 betragen.

Diese Routinen sollten nur sparsam verwendet werden, halten sie doch die interne Uhrzeit wegen der Abschaltung der Interrupts an. Allzugroße Genauigkeit im Mikrosekundenbereich darf auch nicht erwartet werden.

```
/* HC06TI4.C */
/*
 * Zeitverzögerung
 * =====
 */

#include <stdio.h>
#include <conio.h>
#include <mylib.h>
#ifndef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
    UINT count = 0;

    clrscr();
    printf ("Warten mit tim2_waitus (Taste)\n");
    getch();

    do
    {
        tim2_waitus(1000);
        count++;
    }
    while (count<5000);

    printf ("5 Sekunden gewartet\n");

    printf ("Warten mit tim2_waitms() (Taste)\n");
    getch();

    tim2_waitms(10000);

    printf ("10 Sekunden gewartet\n\n");

    printf ("Warten mit tim2_waits() (Taste)\n");
    getch();

    tim2_waits(10);

    printf ("10 Sekunden gewartet\n");
    tim2_init();
    getch();
}
```

Wir kennen den Timer jetzt ausreichend gut, um ihn erfolgreich zur gezielten Steuerung des Lautsprechers verwenden zu können.

6b: Töne macht der Timer

Programmiert man den Timer wie angegeben und verbindet man gleichzeitig den Timer mit dem Lautsprecher (`spk_on()`) und schaltet den Timer ein (`tim2_on()`), dann übernimmt der Timer die Tonerzeugung und das laufende Programm muß sich nicht mehr darum kümmern.

Gleitender Ton, HC06LS5.C

Das erste Beispiel ist ein Programm zur Erzeugung eines in der Frequenz ansteigenden Tones.


```

/* HC06LS5.C */
/*
 * Gleitende Töne
 * =====
 */
#include <dos.h>
#include <stdio.h>
#include <conio.h>
#include <mylib.h>
#ifndef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
    UINT count_start = 0x0100;
    UINT count_stop = 0x1000;
    UINT count = count_start;

    clrscr();
    spk_on();
    tim2_on();
    tim2_init();
    printf("Gleitende Töne, pro Ton 50ms \n");
    printf("Zunächst der normale Timer-Ton mit 1 kHz. "
        "Gleitton beginnt mit Taste\n");
    getch();

    do
    {
        tim2_set(count);
        delay(50);
        count +=20;
    }
    while(count<count_stop);

    printf("\nNormaler Timer-Ton mit 1 kHz (Taste)\n");
    getch();
    tim2_init();
    printf("Ende mit (Taste)\n");
    getch();
    tim2_off();
    spk_off();
    tim2_init();
}
    
```

Nach 50 ms wird die Periodendauer des Tones um 20 erhöht.

Rauschen, HC06LS6.C

Eine zufällig gewählte Periodendauer wird 1 ms lang ausgegeben.

```

/* HC06LS6.C */
/*
 * Rauschen
 * =====
 */
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <mylib.h>
#ifndef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
    UINT rand_start = 0x0050;
    UINT rand_stop = 0x1000;
    UINT count;

    clrscr();
    printf("Rauschen\n");
    printf("Taste\n");
    getch();
    spk_on(); tim2_on();
    do
    {
        count = random(rand_stop-rand_start) + rand_start;
        tim2_set(count);
        delay(1);
    }
    while(!kbhit());

    printf("\nNormaler Timer-Ton mit 1 kHz (Taste)\n");
    getch();
    tim2_init();

    printf("Ende mit Taste\n");
    getch();
    tim2_off();
    spk_off();
    tim2_init();
}
    
```

Töne als Frequenz angeben, HC06LS7.C

Der Timer wird in Vielfachen der Timer-Taktfrequenz programmiert. Für viele Anwendung ist aber das Programmieren mit einer Frequenz nahe-liegender. In Anlehnung an die `sound()`-Funktion der BORLAND-Bibliothek wird eine Funktion `soundf()` formuliert, die ebenfalls eine Frequenz als Argument annimmt aber - ein Unterschied muß sein - mit einer Gleitkommazahl als Argument. Als Unterscheidung dient das nachgestellte `f`. Beachten Sie, daß in einer reinen C++-Bibliothek kein Unterschied im Namen bestehen müßte, da der unterschiedliche Typ im Argument zur Unterscheidung ausreicht. Die Ausschaltfunktion `nosoundf()` enthält neben den eigentlichen Schaltern für Timer und Lautsprecher auch noch die Initialisierung des Timers auf den Anfangswert mit `tim2_init()`, damit sich eine programmgesteuerte Veränderung nicht auf Systemebene auswirkt.

```

VOID soundf(FLOAT frequenz)
{
    UINT periode = (UINT)(1192500.0/frequenz);
    tim2_set(periode);
    spk_on();
    tim2_on();
}

VOID nosoundf(VOID)
{
    spk_off();
    tim2_off();
    tim2_init();
}
    
```

```

/* HC06LS7.C */
/*
 * Frequenzeingabe
 * =====
 */
#include <dos.h>
#include <mylib.h>
#ifndef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
    soundf(1000.0);
    delay(500);
    soundf(2000.0);
    delay(500);
    soundf(3000.0);
    delay(500);
    soundf(4000.0);
    delay(500);
    nosoundf();
}
    
```

Frequenz und Dauer kombiniert, HC06LS8.C

Verwendet man Töne häufiger, ist das abwechselnde Aufrufen von `soundf()` und `delay()` lästig, `soundf1()` kombiniert beides und beendet den Ton auch regulär.

```

VOID soundf1(FLOAT frequenz, UINT dauer)
{
    soundf(frequenz);
    delay(dauer);
    nosoundf();
}
    
```

```

/* HC06LS8.C */
/*
 * Variable Dauer
 * =====
 */
#include <mylib.h>
#ifndef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
    soundf1(1000.0,500);
    soundf1(2000.0,500);
    soundf1(3000.0,500);
    soundf1(4000.0,500);
}
    
```

PC als Instrument, HC06LS9.C

Um jetzt noch aufeinander abgestimmte Töne unserer üblichen Instrumentenstimmung verwenden zu können, genügt eine Frequenztafel. (Natürlich könnte man jede Frequenz dieser Tabelle bei Bedarf jeweils neu berechnen, eine Tabelle spart aber Rechenzeit.)

```

FLOAT freq_tab[] =
{
  130.8, /* C 0 */
  138.6, /* Db C# 1 */
  146.8, /* D 2 */
  155.6, /* Eb D# 3 */
  164.8, /* E 4 */
  174.6, /* F 5 */
  185.0, /* Gb F# 6 */
  196.0, /* G 7 */
  207.7, /* Ab G# 8 */
  220.0, /* A 9 */
  233.1, /* Hb A# 10 */
  246.9, /* H 11 */
  261.7, /* c 0 */
  277.2, /* db c# */
  293.7, /* d */
  311.1, /* eb d# */
  329.6, /* e */
  349.2, /* f */
  370.0, /* gb f# */
  392.0, /* g */
  415.3, /* ab g# */
  440.0, /* a */
  466.2, /* hb a# */
  493.9, /* h */
  523.3, /* c1 */
  0.0,
  0.0
};

```

Das folgende Programm zeigt wie man diese Tabelle dazu benutzt, eine chromatische, eine Dur- und eine Moll-Tonleiter zu spielen.

```

/* HC06LS9.C */
/*
 * Verschiedene Tonleitern im PC
 * =====
 */

```

```

#include <stdio.h>
#include <conio.h>
#include <mylib.h>
#ifdef MYLIB
#include "..\source\spk.c"
#endif

VOID main(VOID)
{
  FLOAT *frequenz;
  INT tonnummer;
  UINT tondauer=200;

  clrscr();

  printf("Chromatische Tonleiter C1..C..c, "
        "Start mit Taste\n\n");
  getch();

```

```

  frequenz = &freq_tab[0];
  do
  {
    soundf1(*frequenz,tondauer);
    frequenz++;
  }
  while (*frequenz>1.0);
  printf("\n\n");

  printf("\nDUR-Tonleiter C1..C..c, "
        "Start mit Taste\n\n");
  getch();
  frequenz = &freq_tab[0];
  tonnummer = 0;
  do
  {
    soundf1(*frequenz,tondauer);
    switch (tonnummer)
    {
      case 0: case 2: case 5: case 7: case 9:
        frequenz++; tonnummer++;
      default:
        frequenz++; tonnummer++;
    }
    tonnummer %= 12;
  }
  while (*frequenz>1.0);
  printf("\n\n");

  printf("\nMOLL-Tonleiter C1..C..c, "
        "Start mit Taste\n\n");
  getch();
  frequenz = &freq_tab[0];
  tonnummer = 0;
  do
  {
    soundf1(*frequenz,tondauer);

    switch (tonnummer)
    {
      case 0: case 3: case 5: case 8: case 10:
        frequenz++; tonnummer++;
      default:
        frequenz++; tonnummer++;
    }
    tonnummer %= 12;
  }
  while (*frequenz>1.0);
  printf("\n\n");

  getch();
}

```

Das wärs zunächst über den Lautsprecher; wir werden ihn aber noch mehrmals antreffen, etwa beim Schreiben einer BIOS oder DOS-Erweiterung. Auch für den Timer werden wir eine eigene C++-Klasse entwerfen, die genau weiß, welche Einstellung der Timer hat, denn derzeit kann man die Register des Timers nicht zurücklesen.

In der nächsten Folge werden Hardware-Interrupts und einfache Interrupt-Service-Routinen geschrieben. □

Real programmers don't use LISP. Only effeminate programmers use more parentheses than actual code.

Real programmers disdain structured programming. Structured programming is for compulsive, prematurely toilet-trained neurotics who wear neckties and carefully line up sharpened pencils on an otherwise uncluttered desk.

Real programmers have no use for managers. Managers are a necessary evil. Managers are for dealing with personnel bozos, bean counters, senior planners and other mental defectives.

Real programmers scorn floating point arithmetic. The decimal point was invented for pansy bedwetters who are unable to "think big."

Real programmers like vending machine popcorn. Coders pop it in the microwave oven. **Real programmers** use the heat given off by the cpu. They can tell what job is running just by listening to the rate of popping.

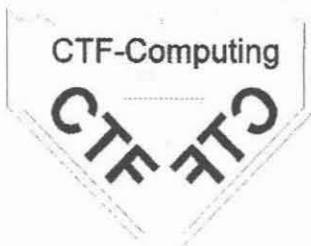
Real programmers don't drive clapped-out Mavericks. They prefer BMWs, Lincolns or pick-up trucks with floor shifts. Fast motorcycles are highly regarded.

Real programmers don't believe in schedules. Planners make up schedules. Managers "firm up" schedules. Frightened coders strive to meet schedules. **Real programmers** ignore schedules.

Real programmers don't like the team programming concept. Unless, of course, they are the Chief Programmer.

Real programmers know every nuance of every instruction and use them all in every real program. Puppy architects won't allow execute instructions to address another execute as the target instruction. **Real programmers** despise such petty restrictions.

Real programmers don't bring brown bag lunches to work. If the vending machine sells it, they eat it. If the vending machine doesn't sell it, they don't eat it. Vending machines don't sell quiche.

**HARDWARE****SOFTWARE****BERATUNG****KUNDENDIENST****A-1150 Wien, Mariahilferstraße 142**

Telefon: 0222/892 35 90 Telefax: 0222/892 35 90

A-1180 Wien, Schulgasse 63

Telefon: 0222/408 52 56 Telefax: 0222/408 99 78

Mo.-Fr. 9.⁰⁰-18.⁰⁰ u. Sa. 9.⁰⁰-13.⁰⁰

Preisblatt 1/94 - Seite 1

Personal Computer SETANGEBOTE

PC 386dx 40 Mhz 13.990,-

80386dx 40 Mhz, 4 Mb Ram, Multi I/O(2 x ser./1 x par/1 x game), IDE (AT-Bus) FDHD-C, 3 1/2" 1.44 Mb Diskettenlaufwerk, 170 Mb Festplatte, 16-Bit 1MB SuperVga Grafikkarte, MiniTower-Gehäuse, Tastatur, SuperVGA Monitor Strahlungsarm, Maus

PC 486dlc 40 Mhz 14.990,-

wie PC 386dx 40 Mhz jedoch mit ti 80486dlc 40 Mhz, 128Kb Processorcache und 170 Mb Festplatte.

PC 486dlc 40 Mhz Multimedia 17.990,-

wie PC 486dlc 40 Mhz jedoch mit Soundkarte, Multimedia Gehäuse mit Lautsprecher, Microfoneingang, und Kopfhörerausgang.

PC 486dlc 40 Mhz Multimedia 19.990,-

wie PC 486dlc 40 Mhz jedoch mit Soundkarte, CD ROM (Multisession, Multimedia Gehäuse mit Lautsprecher, Microfoneingang, und Kopfhörerausgang.

PC AMD 486dx 40 Mhz LocalBus 17.990,-

AMD 80486dx 40 Mhz, 32-Bit LocalBus, 4 Mb Ram, Multi I/O(2 x ser./1 x par/1 x game), IDE (AT-Bus) FDHD-C, 3 1/2" 1.44 Mb Diskettenlaufwerk, 170 Mb Festplatte, 16-Bit 1 MB SuperVga Grafikkarte, MiniTower-Gehäuse, Tastatur, SuperVGA Monitor Strahlungsarm, Maus

PC INTEL - 486dx2 66 Mhz LocalBus.. 20.990,-

wie AMD-486dx 40 jedoch mit 80486dx2 66 Mhz.

PENTIUM - 60Mhz Localbus 59.990,-

Intel Pentium 60Mhz, echtes 64 Bit MAINBOARD, 8 MB Ram, Vesa LB SCSI Controller, Multi I/O(2 x ser./1 x par/1 x Game), FDHC-C, 3 1/2" 1.44 Mb Diskettenlaufwerk, 340 Mb SCSI Festplatte, 1 MB Vesa LB VGA Karte, Profitower, Tastatur, SuperVGA Monitor 1024 x 768NI/LR, Maus, Windows 3.1 dt., Dos 6.0dt.

Personal Computer SETAUFPREISE

von 170 Mb auf 240 Mb 890,-

von 170 Mb auf 340 Mb 1.290,-

von 170 Mb auf 425 Mb 4.290,-

von 170 Mb auf 525 Mb 5.490,-

2tes Laufwerk 790,-

Hauptspeicher-Aufrüstungen pro 1MB 690,-

Aufpreis für Desktop-Gehäuse 190,-

Aufpreis für ProfiTower-Gehäuse 890,-

Aufpreis auf LB VGA Cirrus 790,-

Aufpreis auf ET4000/W32 1.090,-

Aufpreis auf LB IDE I/O 490,-

Aufpreis DOS und Windows 1.490,-

Aufpreis 14" 1024x768NI/LR 590,-

Aufpreis 15" 1280x1024NI/LR 2.590,-

Aufpreis 17" 1280x1024NI/LR 8.590,-

M A I N B O A R D S

80386 dx 40 Mhz, 128 Kb Cache 1.790,-

80486 dlc 40 Mhz, inkl. Coprozessor .. 3.490,-

80486 dx/33/40/50/66 MB VLB o. CPU . 1.990,-
mit 256 KB Processorcache bestückt

PCI 80486dx33/66 ohne CPU 8.990,-

mit 256 Kb Processorcache bestückt, 1 x SCSI PCI Controller, bis 128 MB on Board, 3 x PCI Slot, 5 x ISA 16Bit Slot, ZIF Socket für Overdrive Pentium, inkl. PCI S3 VGA Karte

AMD 80486 dx 40 Mhz LocalBus 5.990,-

mit 256 Kb Processorcache bestückt.

Intel 80486 dx2 66 Mhz LocalBus 9.490,-

mit 256 Kb Processorcache bestückt.

Intel 80486 dx2 66 Mhz /EISA 12.990,-

mit 256 Kb Processorcache bestückt T-Bus)

IDE (AT-BUS) - FESTPLATTEN

170 Mb - 1 Jahr Garantie 2.990,-

210 Mb - 1 Jahr Garantie 3.290,-

250 Mb - 1 Jahr Garantie 3.690,-

340 Mb - 1 Jahr Garantie 4.490,-

425 Mb - 1 Jahr Garantie 6.990,-

525 Mb - 1 Jahr Garantie 8.990,-

SCSI FESTPLATTEN

240 MB - 1 Jahr Garantie 4.490,-

360 MB - 1 Jahr Garantie 6.990,-

525 MB - 5 Jahre Garantie 10.490,-

1GB - 1 Jahr Garantie 14.990,-

1.2GB - 5 Jahre Garantie 17.990,-

1.8GB - 5 Jahre Garantie 22.990,-

FDHD I/O - KONTROLLER

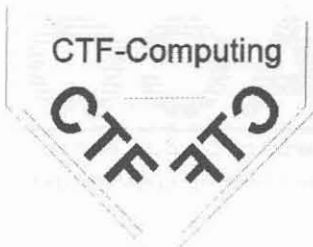
FDHD-C mit Multi I/O 390,-

IDE FDHD-C mit Multi I/O LocalBus 890,-

IDE FDHD-C CACHE LocalBus 2.990,-

SCSI FDHD-C LocalBus 2.990,-

Multi I/O (2 x ser./1 x par./1 x game).... 290,-

**HARDWARE****SOFTWARE****BERATUNG****KUNDENDIENST****A-1150 Wien, Mariahilferstraße 142**

Telefon: 0222/892 35 90 Telefax: 0222/892 35 90

A-1180 Wien, Schulgasse 63

Telefon: 0222/408 52 56 Telefax: 0222/408 99 78

Mo.-Fr. 9.⁰⁰-18.⁰⁰ u. Sa. 9.⁰⁰-13.⁰⁰

Preisblatt 1/94 - Seite 2

Multi I/O HIGHSPEED (16550) 390,-

SuperVga - GRAFIKKARTEN

16-Bit/256Kb Super VGA 490,-

16-Bit/512 Kb Super VGA 690,-

16-Bit/1 Mb Super VGA 890,-

ET-4000 TrueColor 16-Bit/1 Mb 72 Hz . 1.290,-

Cirrus 1 Mb(> 2 Mb erw.) LocalBus 1.790,-

ET-4000 W32 TrueColor LocalBus 1.990,-

S3 1Mb(> 2 Mb erw.) LocalBus od. ISA 2.490,-

WEITEK P9000 LocalBus 2MB VRAM . 6.490,-

SuperVga - MONITORE

14" Graustufen VGA 1.990,-

14" SuperVga 1024x768i 3.490,-

14" SuperVga 1024x768i LR 3.890,-

14" SuperVga 1024x768 NI/LR 4.490,-

15" MultiScan 1280x1024 NI/LR 6.490,-

17" MultiScan 1280x1024 NI/LR 12.990,-

17" MultiScan NI/LR Sony Trinitron 14.990,-

17" Multiscan 1280x1024 NI/LR NOKIA 17.990,-

PersonalComputer - GEHÄUSE

MiniTower 790,-

MiniTower mit Card Controll 1.190,-

Desktop 890,-

MiniTower Mercury 1.490,-

ProfiTower 1.690,-

Server Tower 2.990,-

EINGABE - GERÄTE

102-Tastatur dt. 490,-

CHERRY Tastatur 990,-

Maus 2-Tasten MS-Kompatibel 190,-

Maus 3-Tasten MS-Kompatibel 290,-

STREAMER - LAUFWERKE

Conner 250 Mb Streamer 2.790,-

inkl. Dos und Windowssoftware inkl. 1 Cartridge

Conner 250 Mb Centronics Streamer . 5.990,-

inkl. Dos und Windowssoftware inkl. 1 Cartridge

PREISE für TGM Mitglieder !

FLACHBETT - SCANNER

Genius Farbscanner 600 DPI 15.990,-
inkl. SCSI Schnittstelle, mit Software 1200 DPI

TINTENSTRAHL - DRUCKER

Canon BJ-10sx (speziell f. Notebook) 3.490,-

Canon BJ-10sx inkl. ASF 3.990,-

Canon BJ-10sx ASF 1.290,-

Canon BJ-200 (300 Zeichen/sec.) 4.490,-

Canon BJ-300 mit Endlos-Traktor ! 7.990,-

Canon BJ-330 mit Endlos-Traktor ! 10.990,-

Canon BJ-600 Farbdrucker 9.990,-

HP-DeskJet 550C Farbdrucker 9.490,-

MULTIMEDIA - AUDIO

Adlib-/Sounblaster V2.0 komp. Karte . 990,-

SoundBlaster Pro Deluxe Edition 1.890,-

SoundBlaster PRO 16-Bit Basic 2.590,-

SoundBlaster PRO 16 ASP Multi-CD .. 3.590,-

MULTIMEDIA - CDROM

MitsumiCD-Rom Double Speed 3.590,-

Toshiba 3401SCSI DoubleSpeed 6.990,-

NEC CDR-84/1 SCSI DoubleSpeed 5.990,-

Soundblaster CD-ROM Double Sp. 4.490,-

MULTIMEDIA - VIDEO

VideoBlaster + TV Decoder 8.490,-
ermöglicht Bilder auf Recorder aufzunehmen !!!

FAX - MODEMS

Prolink 2400 intern 1.490,-

Prolink 2400 extern 1.990,-

Prolink 2400 Pocket 2.490,-

Prolink 14.400 intern 2.990,-

Prolink 14.400 extern 3.490,-

Prolink 14.400 Pocket 3.990,-

Alle Modems mit FAX-, Modemsoftware für DOS, sowie Win FAX
Lite Software für Windows.

AKTION MODEM 14.400 Baud

Externes Datenmodem mit 14.400 Baud Fax und
Datenmodus nur 2.990,-

PREISLISTE - JÄN / FEB 1994

Komplettsysteme

PC 386 DX 40 MHz	9.900,-
CPU AMD 386DX40, AMI-Bios, 128kB Cache, 4MB Ram, 2ser, 1par, 1Game, 1,44MB FDD Teac 250MB HDD, SVGA-Card 1MB, wahlweise Slimcase oder Minitower Design, LED, Tastatur	
PC 486 DX 33 MHz VesaLB	15.000,-
VesaLB Mainboard mit INTEL 486DX33, AMI Bios 256kB Cache, 4MB Ram, 2ser, 1par, 1Game, VesaLB IDE Contr., 1,44MB FDD, 250MB HDD, Vesa LB Bildschirmkarte S3-Chip 1MB(auf 2MB auf- rüstbar) Minitower, Tastatur	
PENTIUM P-60	ab 44.990,-
lieferbar; Testgerät im Shop	

Mehrpreise

Mehrpreise ermitteln Sie selbst lt. Preisliste:

z.B.:	PC 486 DX 33 Mhz	15.990,-
	- CPU 486DX33	3.990,-
	+ CPU 486 DX 2-66	6.990,-
	ergibt	18.990,-

Mainboards

Mainboard 386 DX 40	1.690,-
AMD 386DX40CPU, 128kB Cache, AMI BIOS	
Mainboard 486 VesaLB	1.790,-
2Vesa Slots, 256kB Cache ohne CPU	
GENOA Turbo Express 486 VLB	2.090,-
3Vesa Slots, ZIF-Sockel, 256kB Cache ohne CPU	
Mainboard 486 VesaLB/EISA	2.990,-
2Vesa, 3Eisa, 3Isa Slots, 256kB Cache ohne CPU	

INTEL - 486DX CPU

INTEL 486 DX 33 MHz CPU	3.990,-
INTEL 486 DX 2-50 MHz CPU	4.490,-
INTEL 486 DX 50 MHz CPU	5.990,-
INTEL 486 DX 2-66 MHz CPU	6.990,-

Speichererweiterungen

Simm Modul 1MB 70nS	590,-
Simm Modul 4MB 70nS	2.190,-

AT-BUS Festplatten

Western Digital 2170 14ms 170MB	2.890,-
Western Digital 2250 12ms 250MB	3.290,-
Western Digital 2340 12ms 340MB	4.190,-
Western Digital 2420 10ms 420MB	5.290,-

AT-BUS Festplatten

Quantum LPS 540 AT 9mS 540MB	6.490,-
Größere Festplatten und SCSI Festplatten auf Anfrage je nach Verfügbarkeit liefern wir WD, Conner oder Quantum	

Diskettenlaufwerke

TEAC Diskettenlaufw. 3 1/2" 1,44MB	590,-
TEAC Diskettenlaufw. 5 1/4" 1,2MB	690,-
Einbaurahmen für 3 1/2" LW	48,-

Kontroller und Bildschirnkarten

IDE Kontroller 16 Bit-ISA	199,-
2FDD, 2HDD, 2ser, 1par, 1Game	
IDE Kontroller VesaLB	490,-
DC 600 IDE Cache Kontr. ISA	1.390,-
DC 680 IDE Cache Kontr. VesaLB	2.290,-
DC 820 SCSI Cache Kontr. EISA	5.590,-
ADAPTEC 1542CF SCSI-2 Controller	2.990,-
TRIDENT 8900CL VGACard 1MB	890,-
DIAMOND Speedstar Pro 1MB ISA	1.790,-
DIAMOND Stealth 24 1MB S3 ISA	2.580,-
andere DIAMOND Bildschirnkarten auf Anfrage	
Chicony VLB S3 Chipset 1MB 50Mhz	2.190,-
Chicony VLB S3 Chipset 2MB 50Mhz	2.590,-

CD-Rom und Streamer Laufwerke

MITSUMI CD-Rom multisess.	2.490,-
MITSUMI CD-Rom multisess. db-sp.	3.290,-
TEAC TCD50 SCSI Db.Sp. multis.	5.490,-
Conner Tape 250MB 3 1/2"+3M Cart. form	2.690,-
COLORADO Trakker 250MB ext. par.	5.490,-

Streamerkassetten

VERBATIM Datacartridge DC 2120	220,-
VERBATIM Datacartr. DC 2120 form.	250,-
VERBATIM Datacartridge DC 600	250,-
VERBATIM Datacartridge DC 6150	270,-
VERBATIM Datacartridge DC 6250	290,-
VERBATIM Datacartr. DAT 4MM	199,-
VERBATIM Datacartr. DAT 8MM	199,-

LOGITECH-Eingabegeräte

MouseMan cord. Combo	790,-
MouseMan cordless	1.390,-
Mouse Man Bus	790,-
MouseMan large	790,-
Pilot Mouse seriell oder PS/2	449,-

PREISLISTE - JÄN / FEB 1994

LOGITECH-Eingabegeräte

ScanMan 32 Win inkl. OMNI-PAGE	2.190,-
ScanMan 256 für Windows	2.690,-
ScanMan Color für Windows	6.590,-
Scan Man Easy Touch par. Interf.	4.990,-
Audio Man für par. Interface	1.790,-

siehe Testbericht in diesem Heft.

Sound Man 16	1.990,-
--------------	---------

16Bit-, 44KHz Stereo, Yamaha OPL-3, Midi-Schnittst.

Cyber Man	1.190,-
-----------	---------

Foto Man Plus Digitalkamera	9.990,-
-----------------------------	---------

für PCC-TGM Mitglieder, Schüler und Lehrer Schulversionen

mit 25% Rabatt erhältlich.

Gehäuse und Tastaturen

Slimline Case, TÜV PS, LED	1.490,-
MiniTower Case, TÜV PS, LED	1.490,-
Midi Towercase, TÜV PS, LED	1.790,-
Big Tower Case, TÜV PS, LED	2.190,-
CHICONY 102 Tasten Cherry Switch	390,-
CHERRY G81-3000HAD	890,-

Computerzubehör

GENIUS Mouse One	240,-
GENIUS Mouse Too	390,-
GENIUS Cordless Mouse	490,-
Lüfter für 486er CPU	190,-
Monitorfilter 14" Glas mit Erdklemme	290,-
Monitorschwenkarm bis 20kg.	790,-
Computertisch Grundmodell	690,-
Computertisch Byboard	190,-
Computertisch Monitorständer	190,-
Computertisch Druckerablage	290,-
Computertisch komplett	1.190,-
Druckerumschalter manuell 1:2	250,-
Druckerumschalter manuell 1:4	290,-
Druckerkabel parallel	49,-
Tastaturverlängerung 5pol DIN	96,-
Monitorverlängerung 15pol Sub-D	120,-
Diskettenbox f. 50 Stk. 3 1/2"	69,-
Diskettenbox f. 80 Stk. 3 1/2"	79,-

Monitore

14" TVM 1024x768 b/w LR	2.190,-
14" Goldstar 1024x768 color MPRII	3.890,-
14" Samsung 1024x768 nl LR MPRII	4.890,-
14" Philips 1024x768 nl LR MPRII	4.790,-
15" Tystar 1280x1024 nl LR MPRII	5.990,-
15" Philips nl,LR,Autoscan,Stereo	6.990,-
17" Targa 1280x1024 nl,LR,Digital	10.990,-

20" u. 21" Monitore auf Anfrage

CANON-Drucker

CANON BJ10SX 360dpi A4	3.490,-
CANON Einzelblatteinzug BJ10SX	990,-
CANON BJ10SX + Einzelblatteinzug	3.990,-

CANON-Zubehör

Tintenpatrone für BJ10SX/BJ230	390,-
Akku für BJ10SX	450,-
Tonercartridge für LPB4	1.290,-

Pelikan Recycling Cartridges & Inkjet Refills auf Anfrage im Shop

TEXAS INSTRUMENTS Laserprinter

microWriter Basic, 1,5MB, 300DPI	7.990,-
microWriter PS 23, 2MB, 300DPI	12.990,-
microWriter PS 65, 2MB, 300DPI	16.990,-

TEXAS INSTRUMENTS Zubehör

1MB Upgradekit	1.490,-
RS232 Schnittstelle	790,-
microwriter Tonerkassette	690,-
microWriter Bildkassette	2.290,-

Software

Lotus Smart Suite für Windows	5.490,-
Lotus Ami Standard für Windows	890,-
Lotus 1-2-3 portable f. Dos	790,-
Lotus 1-2-3 V4.0 für Windows	2.190,-
Lotus Improv V2 für Windows	2.490,-
Lotus Approach V2.1 für Windows	2.290,-
Lotus Organizer V1 für Windows	990,-
BusinessTeamWin (Text, Kalk, Db.)	2.290,-
Microsoft Money	990,-

Microsoft Schulversionen/Updates

Microsoft Excel 4.0	2.990,-
Word für Windows 6.0	2.490,-
Access 1.1	2.990,-
Visual C++	1.490,-

Schulversionen nur für Schüler, Lehrer und Institute. Nachweis erforderlich. Andere Softwareprodukte auf Anfrage

Disketten

No Name Disketten 3 1/2" 1,44MB form.	99,-
VERBATIM Datalife 3 1/2" 1,44MB	149,-
VERBATIM Datalife 3 1/2" 1,44MB form.	159,-

Auch wir müssen einmal Urlaub machen.

Vom 23.1.1994 - 7.2.1994 ist unser Geschäft geschlossen. Wir bitten um Ihr Verständnis.

Preise sind Kassaabholpreise inkl. MWSt.

Druckfehler, Irrtümer und Änderungen vorbehalten.

Stand per 14.1.1994

excon Warenvertriebsges.m.b.H.

Canon
hp HEWLETT
PACKARD

intel
inside

ZENITH DATA SYSTEMS
A Bull Company

Seagate
Quantum

LOGICMBH

Microsoft
NOVELL

SONDERANGEBOT

(Gültig bis max.28.02.94, solange der Vorrat reicht!)

EXCON 486DX40VL /250

ÖS 17.970,-

486DX40 (AMD) cpu, 256k Cache, 4MB RAM, 250MB/12ms Festplatte (WD-Caviar), Cirrus Logic VL-LOCAL Bus VGA Karte, True Color, IDE-VL-Bus Contoller, 1.44MB 3½" Floppy, Desktop Gehäuse mit 200W Netzteil, 2 Seriell, 1 Parallel Port, MS-DOS und Windows 3.1 vorinstalliert, Mouse, Tastatur (Datacomp)

EXCON 486DX40VL /250 - SET15"

ÖS 22.980,-

wie oben, inkl. 15" SVGA Monitor AOC CMLB-536, max.1280x1024 strahlungsarm nach MPR II



EXCON 486DX2-66VL /340

ÖS 24.990,-

486DX2-66 Intel cpu, 256k Cache, 8MB RAM, 340MB/12ms Festplatte (Seagate), Cirrus Logic VL-LOCAL Bus VGA Karte, True Color, IDE-VL-Bus Contoller, 1.44MB 3½" Floppy, Desktop Gehäuse mit 200W Netzteil, 2 Seriell, 1 Parallel Port, MS-DOS und Windows 3.1 vorinstalliert, Mouse, Tastatur (Datacomp)



EXCON 486DX2-66VL /340-SET15"

ÖS 29.970,-

wie oben, inkl. 15" SVGA Monitor AOC CMLB-536, max.1280x1024 strahlungsarm nach MPR II

✂-----

Firma
Excon Warenvertriebsges.m.b.H.
Rögergasse 6-8
1090 Wien
incl.MWSt.

Ich bestelle lt. Sammelbestellung 02/94 wie
wie oben beschrieben:

Type: _____ ÖS _____

Name: _____ Plz: _____ Ort: _____
Straße/Nr.: _____ Telefon: _____

Datum: _____ Unterschrift: _____

Impressum, Offenlegung

Grundlegende Richtung: Auf Anwendungen im Unterricht bezogene Informationen über Personal-Computer-Systeme. Berichte über Veranstaltungen von **ADIM, HMV, MCCA, PCC-S, PCC-TGM**.

Medieninhaber: **PC-NEWS-Eigenverlag**

Herausgeber: **ADIM, His Master's Voice, MCCA, PCC-S, PCC-TGM**

Druck: Zlínské tiskárny a.s., POBOX 79, CZ-76097 Zlín-Kudlov, TEL: 0042-67-27239, FAX: 0042-67-28066

VERSAND: CONCEPT, Baumgasse 52, 1030 Wien, TEL: 713-59-41, FAX: 713-87-72

ADIM Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik

ADIM-Wien: Martin Weissenböck, Postfach 23, 1191 Wien
TEL: 369-88-59-8, FAX: 369-88-59-7, TX: 75210388=weim a *56458#, 912-218-106

BTX: *56458#, 912-218-106

ADIM-Graz: Klaus Scheibler, Postfach 37, 8028 Graz.
TX: 75210859=sber a

FIDO: His Master's Voice 2.310/1, Dateien ADIM

BTX: *56458#, 913-110-525

EMAIL: CompuServe: 100016.172, FIDO: 2.310/1.35@fidonet.org

Internet: 100016.172@compuserve.com

Konto: PSK, Blz. 60000, Kto. 7.254.969, ADIM Postgiro München, BLZ: 70010080, Kto: 120914-800

Postscheckamt Chur, Konto: 70-40051-3
Volksbank Brixen, Konto 37283

DVR-Nr.: 0547328

Telefonische Sprechstunde: 369-88-59-8 Montag ab 20:00

Themen: Technik, Modems, Skripten, Unterricht.

HMV, His Master's Voice, Mailbox
unterstützt vom **PCC-TGM**

Anschrift: Werner Illsinger, Flurschützstraße 36/12/5, 1120 Wien
815-48-71 (8-N-1), 14400..300 Bit/s

BBS: 2.310/1.0@fidonet.org

FIDO: PSK, Blz. 60000, Kto. 7.918.896,

Konto: Werner Illsinger

MCCA BTX und Micro Computer Club Austria

Anschrift: Traungasse 12, Postfach 143, 1033 Wien

Clublokal: Am Heumarkt 4, 1030 Wien
TEL: 818 68 58, FAX: 818 68 58, TX: 75210079=mcca a

FIDO: His Master's Voice 2.310/1 und vam. Mitteilungen BTX

BTX: *2550#, 912-222-064

DVR-Nr.: 0536229

Jahresbeitrag: Firma: 1200,- Einzel: 600,- Ermäßig: 300,-

PCC-S Personal-Computer-Club-Salzburg

Anschrift: PCC-S, Itzlinger Hauptstraße 30, 5022 Salzburg
TEL: 0662-536 10, FAX: 0662-536 10-52

Bürozeiten: Mo - Fr: 8.00 - 12.00 (über Direktion der HTBLA-Salzburg)

Konto: Salzburger Sparkasse Blz.: 2300 Kto.: 330720, PCC-S

DVR-Nr.: 0559610

Jahresbeitrag: Schüler 100,-, Lehrer 250,-, förderndes Mitglied 1000,-

PCC-TGM Personal-Computer-Club-Technologisches Gewerbemuseum

Anschrift: Wexstraße 21, Postfach 59, 1202 Wien.
TEL: 332-23-98, FAX: 332-23-98-2

FIDO: His Master's Voice 2.310/1, 2.316/3, 2.318/1, Mitteilungen PCCTGM

BTX: *5645# 912-222-584

Bürozeiten: Mi: 19.00-20.30 (Frau Jelinek)

Konto: EÖSPC, Blz:20111, Kto. 053-32338, PCC-TGM

DVR-Nr.: 0596299

Jahresbeitrag: Schüler: 200,- Student: 300,- sonst: 400,-

Die Autoren

Czedik-Eysenberg, Georg, Dr., Informatiker, seit 1990 im erweiterten Vorstand des **MCCA**, BTX: 912213983, FIDO: 2.313/9.33, EMAIL: czedik@siemens.co.at.

Fiala, Franz, Dipl.-Ing., Lehrer an der höheren Abteilung für Nachrichtentechnik und Elektronik im TGM, Redakteur der **PC-NEWS**. BTX: 912-218-242; FIDO 2.310/1.36, EMAIL: 100024.1325@compuserve.com.

Illsinger, Werner, Ing, EDV-GesmbH, Absolvent des TGM, Sysop der Mailbox **His Master's Voice**. 815-48-71 (USR HST DS). FIDO 2.310/1.0.

Jaburek, Walter J., DDr, Berater bei Diebold und gerichtlich beeideter Sachverständiger. FIDO: 2.316/602.16.

Kmet, Gerhard, Radio Austria, Leiter des Produktmarketings für die TELEBOX und für VSAT. TELEBOX: RAC-KMET, INTERNET: RAC-KMET@telebox.ada.at

Krause, Werner, Mag., GRG Wien-XXIII.

Kugler, Michael, Mag., Mathematiker, Lehrer an der höheren Abteilung für Nachrichtentechnik und Elektronik im TGM. kuglerm@email.tgm.ac.at.

Nigischer, Wolfgang, Glasbau, Mitglied beim PCC-TGM, Absolvent der Fachschule Karlstein, Waidhofen/Thaya, FAX:028 42/53 942, CIS: 100121,230.

Pöschko, Friedrich, Student der Nachrichtentechnik an der TU-Wien, Schwerpunkte Mikroelektronik und Computerperipherie. Mitarbeiter der Fa. ASCOM, freier Mitarbeiter der Firmen PC-Gourmet und Ultrasonic. Fax: 52-666-95; Fido: 2.310/22.9.

Riemer, Walter, Dipl.-Ing., Lehrer an der höheren Abteilung für Nachrichtentechnik und Elektronik im TGM, Leiter des Rechenzentrums der Abteilung, Autor mehrerer Lehrbücher für den Unterricht, Ingenieurkonsulent für Elektrotechnik. BTX: 912-216-422, EMAIL: rriemer@email.tgm.ac.at.

Rupprecht, Susanne, Dr., seit 1993 im ordentlichen Vorstand des **MCCA**, BTX: 912214758, FIDO: 2.313/9.24.

Schlick, K., Abt.II/7, BMUK, Kl.: 4491.

Schlögl, Helmuth, Jahrgang 1940, Obmann des MCCA seit 1983, beschäftigt sich seit 1981 mit BTX (Pilotversuch 300 Teilnehmer), Beruf Bankbeamter, seit 1960 in der GiroCredit Bank (vormals Girozentrale), davon 25 Jahre EDV, bis 1991 mit BTX.

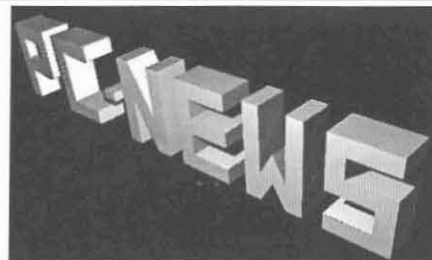
Syrovatka, Robert, Ing., Lehrer an der höheren Abteilung für Nachrichtentechnik und Elektronik am TGM; Obmannstellvertreter des **PCC-TGM**, BTX: 912211867, Internet: syro@mail.tgm.ac.at.

Zandomeneghi, Andreas, Dipl.Ing., geb.1963, Entwicklungsingenieur bei der Fa. Schelling AG in Cham/Schweiz. Außerdem freiberuflich als Consultant für Berechnungen, Maschinenbau und PC-Software tätig. CIS: 100276,1244.

Zetik, H.Fritz, Schüler der 4AN, Höhere Abteilung für Nachrichtentechnik und Elektronik am TGM.

Druckfehler und Irrtümer vorbehalten.

Alle Fernsprechnummern ab Ortsnetz Wien.
Vorwahl Wien aus Österreich: 0222
(aus dem Ausland: ++431)



PC-NEWS-Eigenverlag, PC-NEWS-Redaktion

Anschrift: Franz FIALA, Siccardsburgg. 4/1/22, 1100 Wien
TEL: 604-50-70, FAX: 604-50-70-2

FIDO: His Master's Voice 2.310/1, Dateien PCNDISK

BTX: *56452#, 912-218-242

EMAIL: TELEBOX: FRANZ-FIALA, BIX: ffiala, CompuServe: 100024.1325, FIDO: 2.310/1.35@fidonet.org, Internet: 100024.1325@compuserve.com

Konto: PSK, Blz. 60000, Kto. 7486.555, **PC-NEWS-Eigenverlag**

DVR-Nr.: 0735485

Bezugsbedingungen

Mitglieder des PCC-S, PCC-TGM:	kostenlos
Mitglieder des MCCA Jahresabo (5 Hefte)	S 90,-
Einzelbezugspreis	S 50,-
Jahresabo Inland, inkl.Versand (5 Hefte)	S 150,-
Jahresabo Ausland, inkl.Versand (5 Hefte)	S 220,-

PC-NEWS-36

ISSN: ISSN 1022-1611

Bezahlte S 2000,-/A4-Seite, +10%+20%, Beilage

Werbung: S1,-/Stück+Gewichtszuschlag (>40g)

Auflage: 2500 Stück.

Kopien: Mit Quellenangabe gestattet. Zwei Belegexemplare erbeten.

Herstellung: WinWord 2.0B, FOLEX

Erscheint: Wien, Februar 1993

Zusatzinformation: PCN-DSK 374..407, PCN-LIT 66,67, PCN-SON 3

Beitragskennzeichnung: Name, [Firma], [Zusatzinformation/Programm]

zeichnung: Nicht gekennzeichnete Beiträge stammen von der Redaktion.

Kopien: Mit Quellenangabe gerne gestattet. Zwei Belegexemplare erbeten.

PC-NEWS-Leser Gesamt: 1910

PCC-TGM	1650
PCCS	50
MCCA	50
Abo	160 (davon ADIM ca. 40)

Statistik

Quelle	Seitenzahl	Firma	Seite
Zu dieser Ausgabe	6	CompDelphin	3
Clubs	2+2	CTF	70,71
Redaktion	2	excon	74
Autoren	35,5	König	5
Eigen	21	Microsoft	76
Inserate	9,5	Pablitos	21
Gesamt	76+2	PESACO	72,73
Beilagen	1	Softwareschunel	Beilage

PCC-TGM-Preise, Stand Februar 1994

Literatur, Sonderdruck pro Seite	öS -,80
Tabellen A5/A4, verschweißt	öS 15,-/25,-
Kop.Disketten 360k/1.2M/720k/1.44M	öS 40,-/50,-/50,-/60,-
Leerdisketten 360k/1.2M/720k/1.44M	öS 10,-/20,-/20,-/30,-
360k/1.2M/720k/1.44M PCSIG-Disketten 360k/720k	öS 15,-/50,-
3-fach Verteiler für PC-Netzteil	öS 100,-
Bausatz µPROFI-51, incl. Handbuch	öS 950,-
EPROM für µPROFI-51 (PC-NEWS-2/92)	öS 55,-
Scannerdienst erste Seite (+Disk)	öS 10,-
Scannerdienst Folgeseite, pro Format	öS 2,-
Verpackungskostenanteil	öS 23,-/29,-

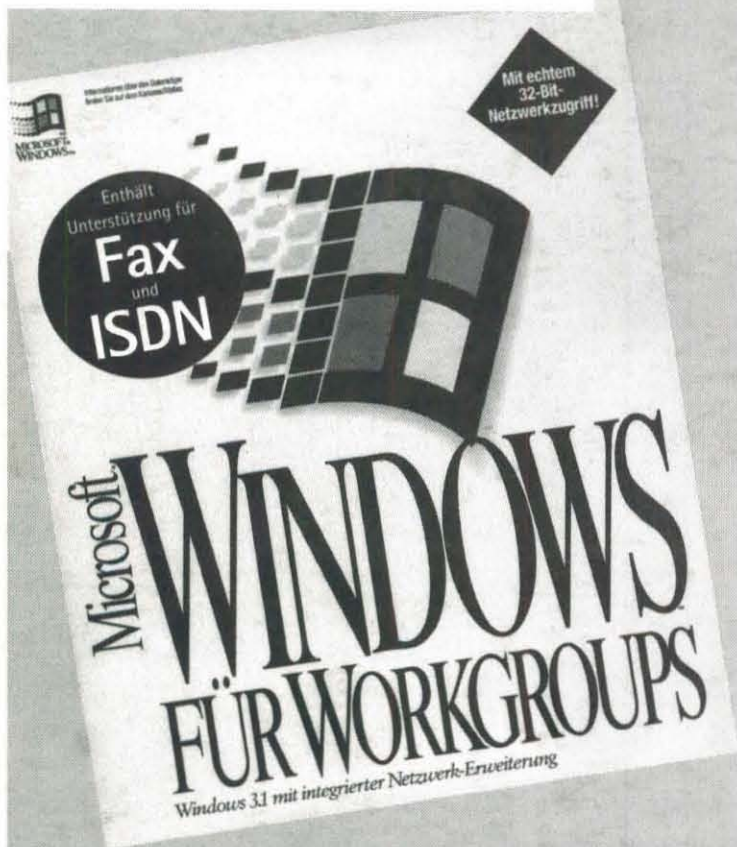
PC-NEWS-Vorschau

Termine	Redaktions schluß für Beiträge	Redaktions schluß für Inserate	Druck- beginn	Versand- termin	Beim Leser	Thema	Werbung
PC-NEWS-36	10. 1.	12. 1.	17.1.	31.1.	7.2.	Windowss-Programmierung	Institute TU-Wien
PC-NEWS-37	7.3.	9.3.	14.3.	28.3.	5.4.	LAN/ISDN/INTERNET/FIDO	Elektronik-HTLs
PC-NEWS-37a	11.4.	13.4.	18.4.	2.5.	9.5.	LAN/ISDN/INTERNET/FIDO	IFABO
PC-NEWS-38	9.5.	11.5.	16.5.	30.5.	6.5.	Mikrokontroller	
PC-NEWS-39	8.8.	10.8.	16.8.	29.8.	5.9.	Windows-Programme	
PC-NEWS-40	17.10	20.10.	24.10.	7.11.	14.11.	Makro-Programmierung	
PC-NEWS-40a	17.10	20.10.	24.10.	7.11.	24.11.	Makro-Programmierung	PC-EXPO

PC-NEWS-37

- Internet im TGM
- FIDO-Point und FIDO-Message-Areas
- ISDN - Compuserve
- ACCESS-Kurs Teil 3
- Hardware-Interrupts in C/C++
- Test: Photoman
- Test: WinWord 6.0, Excel 5.0, VBA
- Test: DAT-Drives

Das neue Windows



Überlegen Sie nicht lange. Windows für Workgroups 3.11 ist das schnellste und beste Windows, das es je gab. Jetzt mit 32-Bit-Technologie für beschleunigten Zugriff auf Daten und Netzwerk.

Und neu mit Fax- und ISDN-Unterstützung für Ihren PC. Das integrierte Netzwerk ermöglicht Peer to Peer Betrieb und arbeiten im Novell und Windows NT Netz. Läuft optimal mit MS-DOS 6.2. Mehr Info unter 0660/6520.

Microsoft®