

Mitteilungen aus den Höheren Technischen Lehranstalten für aktive PC-Anwender

PC-NEWS Eigenverlag Siccardsburg- gasse 4/1/22 1100 Wien	ADIM FIDO MCCA PCC-TGM	Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik His Master's Voice-Mailbox BTX und Micro Computer Club Austria Personal Computer Club - Technologisches Gewerbe-Museum	ISSN 1022-1611 P.h.b. Verlagspostamt 1100 Wien
---	---	--	---

Schwer-● Programmierung

Von der Idee...



...zum Entwurf...

Eingabe einlesen

Abbruch, wenn 0

umwandeln

Eingabe ausgeben

```

for (;;)
{
    printf("Zeit in SEK (0...Abbruch)");
    scanf("%ld",&eingabe);
    if (!eingabe) break;
    umwandeln(eingabe,&std,&min,&sek);
    printf
        ("%7ld S =%4d h%3d m%3d s\n",
         eingabe,std,min,sek);
}

```

```

00110011
01001001
10010010
11001010

```

...zum Kode...

...zum Programm.

Inhalt

Liebe Leser.....	2
Leserbriefe, Bazar	4
PC-NEWS-Disketten, Autoren	5
Clubteil	6
Urheberrecht, Datenschutz	10
Berichte	11
P-CAD Version 6	12
High-Tech-Unterricht.....	13
Ingenieurwesens im 21. Jhdt	13
PAN - Public Access Network.....	14
AMMU, Arbeitsgruppe Mathematik..	16
Vektoralgebra mit DERIVE	18
Bretter-Recycling	24
ACCESS-Lehrgang; Dokumentation..	26

angeTESTet

Mein Kampf mit OS/2	28
SCSI-Controller - ISA oder VLB?	30
VLB-Universal-Adapter	31
Public-Domain	32
FAX-Pocket-Modem 14000 bit/s	36
TF555 - Fax-Umschalter	37
FaxPak	38
Olivetti QUADERNO PC-XT-20	39
Waldgeister, Spiel	39

Schwer-●

An alle Freunde der ADIM!	40
Korrekturen zu ADIM-Band-50	43
ADIM-Bestellschein	45
TURBO-PASCAL-TOOLBOX	47
PASCAL-Einleseschleifen	48
Hard-Disk-Management	50
Querschnitte durch C	52
Hardwarenahe Programmierung	54
C Users Group Library	69
Source Code CDROM, März 1992	71

Liebe LeserInnen!

Die letzte Ausgabe dieses Jahres ist umfangreich und präsentiert sich auch in neuer Qualität. Diese Erscheinung will betonen, daß es dem Betreiber des Verlags und der Redaktion ernst ist, weiterhin solide Information zu bieten.

Das neue Gewand

Druckauflösung	1200 dpi (HP-LJ-4 mit WinJet 1200)
Fließtext	SquareSerif
Überschriften	Lucida Sans
Programme	Lucida Sans Typewriter
Clubheader	Lucida Handwriting
Herausgeber	VAGround

Der **PC-NEWS-Eigenverlag** erzeugt die **PC-NEWS** als eine PC-orientierte Zeitschrift für den Unterrichtssektor, zwei Clubs (**PCC-TGM** und **MCCA**) und unabhängige Abonnenten sind seine Abnehmer. Die **PC-NEWS** erscheinen, da sie eine periodische Zeitschrift sind, unabhängig davon, ob die Clubs die **PC-NEWS** bestellen oder nicht. Wenn sie es nicht tun, erreicht eine Ausgabe nur die Abonnenten; die Auflage ist bescheiden und kostenintensiv und nicht kostendeckend; wenn die Clubs mitmachen, wie sie es bisher taten, kann Kostendeckung erzielt werden.

Bei einem ausgewogeneren Verhältnis zwischen Abonnenten und Clubmitgliedern (derzeit 1:15) wäre die Situation weniger angespannt. Der **PC-NEWS-Eigenverlag** bemüht sich, durch Werbung neue Abonnenten zu gewinnen und lädt auch Sie ein, mit der **Ausgabe für Freunde** am Bestellschein für unsere gemeinsamen Ziele zu werben. Wir alle: Leser, Autoren, Redaktion, Verlag, Clubs, Inserenten, ja sogar die Druckerei bilden eine Gemeinschaft mit dem Ziel der verstärkten Informationsverbreitung über das Arbeiten mit dem PC.

Zwei Kleinigkeiten am Titelblatt sind erwähnenswert: eine kleine Zahl rechts oben, eine **ISS-Nummer** und die fortlaufende Nummer 35.

Die **International Standard Serial Number (ISSN)** wird vom **Centre International d'Enregistrement des Publications en Series** in Paris vergeben. Diese Registrierung ermöglicht die Aufnahme in internationale Zeitschriftenverzeichnisse.



35: Wenn Sie mit der vorigen (blauen) **PC-NEWS** vergleichen, wird Ihnen eine Nummer abgehen: Auf Grund einer kurzfristigen Einladung der **PC-EXPO-Messeleitung** wurde eine **Werbeausgabe mit Mehrzweckcharakter** geschaffen (Auflage 10.000), die die Nummer 34 erhielt.

PC-NEWS-34

Einerseits sollen in dieser Sondernummer Interessenten über die Tätigkeit der **PC-NEWS**, die Tätigkeit der Vereine **PCC-TGM**, **MCCA** und **ADIM** informiert werden, andererseits dient sie allen Lesern und Clubmitgliedern als **Gesamtverzeichnis** für alle bisherigen Ausgaben.

Alle Leser werden gebeten, bei Bedarf diese Sonderausgabe mit der Bestellkarte anzufordern und damit zur Kostendeckung dieses Werbeprojekts beizutragen.

Aus dem Inhalt:

Herausgeber:

ADIM, His Master's Voice, MCCA, PCC-TGM.

PC-NEWS - Auf einen Blick:

Autoren, Sonderdrucke, Literatur, Tabellen, Disketten, Gesamthalt, Titelseiten.

44 Seiten, weiß.

Wenn Sie eine weitere Übersicht zum Weitergeben brauchen, kreuzen Sie das bitte bei der Antwortkarte an!

Disketten in den PC-NEWS

Bisher wurden alle in den **PC-NEWS** beschriebenen Disketten ausschließlich über den **PCC-TGM** vertrieben. Da der **PCC-TGM** nur an Mitglieder fakturiert, konnten andere Leser nicht versorgt werden. Durch eine verstärkte Zusammenarbeit mit der Mailbox **His Master's Voice** wurde diese Schwierigkeit umgangen:

- (1) Alle Disketten (siehe Gesamtverzeichnis in den **PC-NEWS-34**), die im Rahmen der Gestaltung der **PC-NEWS** gesammelt werden (Programme, Texte, Demoprogramme usw.), sind ab sofort auch über die Mailbox von Werner Illsinger zu beziehen. Entweder zum Downloaden im Area PCN oder als Diskette zu denselben Bedingungen, wie bisher beim **PCC-TGM** (Bestellschein am Ende des Heftes)
- (2) Alle Kopien des Literaturverzeichnisses (siehe Gesamtverzeichnis in den **PC-NEWS-34**) können ab sofort auch über den **PC-NEWS-Eigenverlag** bezogen werden.

CDROM

Vor Zeit zu Zeit geben wir einen Überblick über aktuelle CDs, heute, dem Leitthema entsprechend, zwei CDs mit Source-Codes. Die CDs sind, gemessen am Inhalt, nicht sehr teuer, doch wenn Sie noch keinen CD-Spieler besitzen, bieten Ihnen die **PC-NEWS** die Möglichkeit, daß Sie sich Teile daraus kopieren können.

Wie erfahren Sie, was die Programme im Einzelnen sind? Neben der Zusammenfassung in diesem Heft (1 Programm pro Zeile, bzw. 1 Diskette pro Zeile) können Sie genauere Darstellungen in den Dateien **CUG.ZIP** und **SCC.ZIP** nachlesen. Eine bestimmte Datei oder ein bestimmtes Programm können Sie über den Diskettendienst bestellen.

Rückschau, Fehlerberichtigung

Es ist unvermeidlich, daß sich bei größeren Auflagen Fehler beim Drucken oder Binden ergeben. Wenn in Ihrem Exemplar sinnstörende Fehler vorkommen, schickt Ihnen der Verlag gerne ein Ersatzexemplar zu (solange vorhanden), sonst eine Kopie der fehlerhaften Seiten.

PC-NEWS-33: In allen Fußzeilen war die **falsche** Nummer 32 statt **richtig** 33 eingetragen!

Seite 81 ("Zu dieser Ausgabe") **ADIM-BTX-Nummer richtig:** 912-218-106, **ADIM-FAX-Nummer richtig:** 369-88-59-7.

Die **INTERNET-Nummer** der Redaktion via CompuServe enthielt bis zur letzten Ausgabe einen Beistrich statt einem Punkt zwischen den Zahlen.

Die **Unterschrift als True-Type-Font** ist in den **PC-NEWS-33**, Seite 29 mißlungen. Der Grund: ein zu kleines virtual Memory des PostScript-Druckers. Mit der neuen Ausrüstung der Redaktion: kein Problem, wie Sie weiter unten sehen.

VORSCHAU auf die PC-NEWS-36, Windows

Da das Thema Programmierung naturgemäß ergiebig ist, wurde es geteilt; die nächste Ausgabe beschäftigt sich ausschließlich mit Windows und der Programmierung unter Windows.

- Object-Windows
- VISUAL-BASIC, ein Einstieg
- Hardwarenahe Programmierung Teil 6, (Lautsprecher/Timer)
- Menüs und andere Gerichte
- Ein Menüprogramm für Laborübungen
- WinWord-Kurs
- Textbearbeitung mit WordBasic
- Texttransfer aus DOS
- ACCESS-Lehrgang

Allen Lesern ein gutes neues Jahr 1994

Fraus Tiago

Piratenpreise für CDC-Computer,

DIE PIRATEN

Canon-Drucker, Philips-Bildschirme

KOMMEN

und Software von Borland, Corel,

MIT PIRATEN-

Lotus, Micrografx, Microsoft, Symantec

PREISEN!

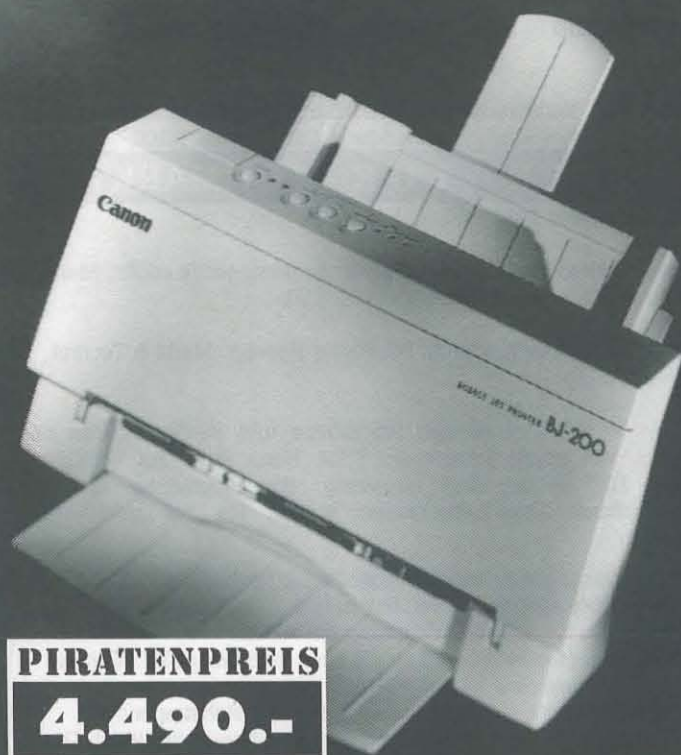
ab sofort bei Comp Delphin.

Piratenpreise gültig solange der Vorrat reicht. Alle Preise inkl. MwSt.



PIRATENPREIS
14.270.-

CDC 845ML 250 MB Prozessor: 486 DLC,
40 MHz, 4 MB Ram, 3.5" Floppy, Festplatte 250 MB
+ ARCHE 214 ML 14" Farb-VGA-Monitor,
800 x 600 Bildpunkte ni, 72 Hz um öS 14.270.-



PIRATENPREIS
4.490.-

CANON BJ 200

A4 Bubble-Jet Drucker,
173 Z/sec LQ, 360 x 360 dpi,
Druckerkabel um öS 4.490.-



Die Computerpiraten

1090 Wien • Berggasse 5
Tel.: 0 222 / 34 44 42

8010 Graz • Schubertstrasse 2-4
Tel.: 0 316 / 38 46 70-3

Leserbrief

PCAD45 oder P-CAD MD 6.06?

Schon bevor das TGM sein PCAD45-Menüsystem hatte (siehe **PC-NEWS** 33), war von mir ein sehr leicht bedienbares und effizienteres Menüsystem für die Versionen 4.x und 5.x jahrelang in Verwendung. Aber das interessierte niemanden. "Wir rufen P-CAD von DOS aus auf! Die Schüler sollen die Aufrufparameter lernen und die Zusammenhänge verstehen." (u.a. auch von einem TGM-Lehrer)

Jetzt, wo P-CAD in der Version 6 da ist und erstmalig wirklich leicht zu bedienen ist, wird plötzlich PCAD45 forciert! Warum? Weil da die Computer für die erhöhten Anforderungen von P-CAD 6 nicht vorhanden sind? Das glaub ich kaum. Weil da einige Kollegen viel Arbeit in IHR Menüsystem gesteckt haben? Oder wollen da vielleicht einige Kollegen nicht dazulernen, wie man es vielleicht besser machen könnte? Es wird uns ja mitunter vorgeworfen, daß wir nicht am Stand der Technik sind und Veraltetes unterrichten.

So will man uns jetzt PCAD45 als "... ein sehr gut verwendbares Menüprogramm, das Ungeübte gefahrlos durch den Dschungel dieses Programmpakets geleitet. ..." verkaufen. Es ist gelungen, "... dieses Programm einigermaßen zu zähmen. ..."! Siehe da, einigermaßen! Sicher hatte ich und die Schüler auch Probleme im Umgang mit P-CAD, aber für "Ungeübte" ist das ja auch nicht vorgesehen. Es soll ja unterrichtet und dabei geübt werden. (Zitate aus der **PC-NEWS** Nr.33, S.3)

Für mich ist es jedoch so, daß die Probleme daher kommen, daß viele der "Kollegen" ihre eigene Suppe kochen wollen. Nur ja nicht etwas machen, was ein anderer "Kollege" auch macht. Ich bin Ich! (Am TGM lautet meine Information nach das Hauptverzeichnis im Netzwerk I: = I bin I:?) So entstanden lauter Insellösungen, jeder löste die gleichen Probleme, jeder hatte viel Zeit dafür investiert. Das Ergebnis: eine Lösung, mit der es sich leben (unterrichten) läßt. Maulen im Hintergrund.

Frage: Wer hat denn P-CAD am Anfang so forciert? Wieso haben wir überhaupt P-CAD?

Eine mögliche Antwort: Auch in Hollabrunn wurden damals einige am Markt befindliche CAD-Systeme für Elektronik getestet. Das Ergebnis war ernüchternd. Zum großen Teil unbrauchbar: zu viele Einschränkungen, schlechte Bedienbarkeit, Ergebnis unbefriedigend usw. Nur P-CAD war als Zwischenstufe zu größeren Systemen (auch im Preis) ein Kompromiß. Die Funktionalität und Universalität war sehr gut. Die Bedienung war dementsprechend nicht ganz einfach. Man durfte aber nie das gesamte Programmpaket sehen, sondern mußte immer in kleinen Schritten vorgehen und Teilziele suchen und diese auch erreichen.

Derzeit gibt es auf dem Markt unter Umständen besser passende Programmpakete für die Elektronik. Aber können wir noch umsteigen? Oder ist es nicht besser, zusammenzuarbeiten, besser zu kommunizieren (miteinander reden) und gemeinsam Lösungen zu suchen, als immer nur zu nörgeln und den anderen die eigene Lösung aufzudrängen und so tun, als gäbe es nichts anderes?

Wolfgang Zelinka, HTBL Hollabrunn

Anm.d.Red.: Folgendes wurde per FIDO-Netmail 'ausgekundschaftet':

Die "P-CAD V6.06 Arbeitsanleitung", von Koll. Zelinka wird voraussichtlich 100-150 A4 Seiten Umfang haben und für Lehrer und Schüler, bei 15 Stk. Mindestmenge ca. öS 150,- bis 200,- +Versand kosten. Einzelstücke und Firmen ca. +50%.

Alle Interessenten werden um Voranmeldung an Kollegen Zelinka gebeten, das wäre sehr motivierend und auch hilfreich für die abschließende Kalkulation. Musterexemplare bzw. Vorabdrucke können nur in Einzelfällen abgegeben werden.

Außerdem kann eine "Kopierlizenz" für größere Stückzahlen erworben werden. Ebenso ist eine verkleinerte Kopie möglich (2*A4 auf eine A4-Querformat einseitig), wodurch sich die Kosten für Schüler halbieren lassen.

Weiter Details zu PCAD-6 entnehmen Sie dem gleichnamigen Beitrag weiter hinten.

BAZAR

HARDWARE FLOHMARKT

Biete **Alphatronic P3**, Triumph-Adler-PC, incl. 2 Laufwerke, Literatur, Monochrom-Monitor grün, VB: 2000,-.

Biete **Homecomputer COMMODORE VC-20**, inkl. Datasette, Spielmodule, Speichererweiterung 16k, Spiele, Zeitschriften. VB: 2000,-.

TEL: abends (0222)-47-05-351, Hr. Krumpschmied.

Suche **386er Hauptplatine 33 oder 40 MHz**, Angebote an Prof. Krann, TGM, (0222)-33-1-26

SOFTWARE FLOHMARKT

Alle folgenden Programme sind in Originalverpackung mit Dokumentation und auf 5-1/4"-Disketten.

• **Aporia**, ICON-Verwaltung unter Windows, ca. 100 ICONS, PC-SIG, Shareware, S 100,-.

• **Bunny's Programmschutz**, Boeder, Schützt COM und EXE-Programme mit einem Paßwort; S 100,-.

• **Flodraw**, PC-SIG, Flußdiagramme, Schemazeichnungen, Organigramme, DOS, S 100,-.

• **Computer Diplomacy**, Diplomatie im Jahre 1901.

• **3D Game Maker**, Entwurfsprogramm für 3D-Spiele; S 100,-.

• **MAGIC**, Ein kommerzielles, relationales Datenbank-Entwicklungssystem, S 100,-.

• **Fraktal-Generator**, DMV-Software; S 100,-.

• **Hacker**, Spiel rund um internationale Datennetze, S 100,-.

• **DEADLINE**, Spiel, 12 Stunden für Auflösung eines Krimis, INFOCOM, S 100,-.

• **HOLLYWOOD HI-JINX**, Spiel "Interactive Fiction", INFOCOM, S 100,-.

• **ERGO 3.03**, Komfortable DOS-Benutzeroberfläche, Adreßverwaltung, Echtbild-Textverarbeitung, Zeichneditor, DMV-Verlag, S 100,-.

LITERATUR FLOHMARKT

• **DOS-Jahrgang 1991**, komplett 12 Hefte, S 50,-.

• **Computerlexikon**, Olivetti, 650 Begriffe verständlich erklärt, Markt & Technik, 250 Seiten, Format 10x8 cm, S 50,-.

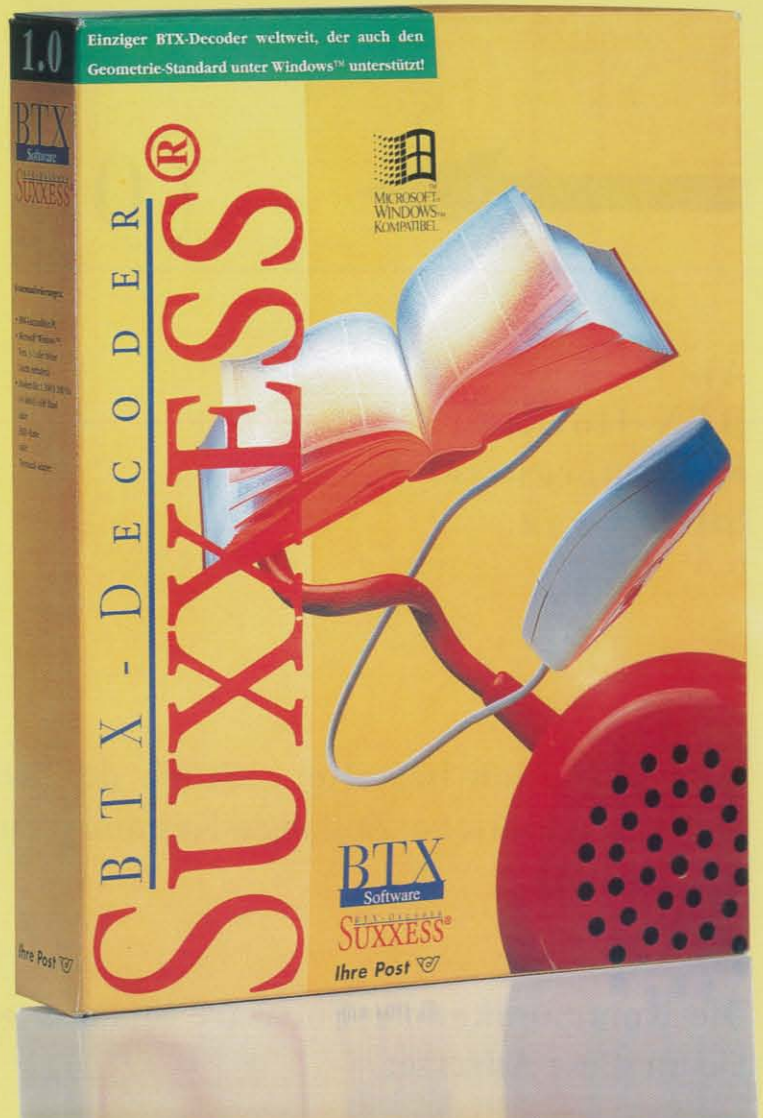
• **QuickC 2.0** Schnellübersicht PC, Rainer Haselier, Markt & Technik, 570 Seiten, S 50,-.

Alle mit '*' gekennzeichneten Programme und Bücher wurden der Redaktion von einem Schüler des TGM, Herrn Boys, zur Verfügung gestellt. Die Bücher sind neuwertig, die Programme sind im Originalzustand. Diese Programme können Sie auf zwei Arten bekommen:

- (1) Reservieren Sie jedenfalls bei der Redaktion (0222)-604-50-70, um Doppelbestellungen zu verhindern und
- (2a) berichten Sie über Ihre Erfahrung mit einem Gerät/Programm/Zubehör für die neue Reihe **angeTESTet** auf Diskette (Beispiele in diesem Heft) oder
- (2b) überweisen Sie den kleinen angegebenen Geldbetrag auf das Redaktionskonto. Die Versandkosten sind bereits inkludiert. Die Geldbeträge sind für unseren Schüler bestimmt.

Um nur öS 1080,-

Unter Windows™ in's größte Datennetz Österreichs



Ein Mausklick, und Suxcess baut die Verbindung zum BTX-Datennetz auf.

Ob beim Telebanking oder beim Recherchieren in einer der vielen Datenbanken: Sie brauchen auf den gewohnten Windows-Komfort mit Mausbedienung nicht mehr zu verzichten.

Für das Erstellen von Mitteilungen steht ein integrierter E-Mail Editor zur Verfügung.

Viele vorinstallierte Modems werden unterstützt, Telesoftware kann geladen werden, die Darstellungsgröße ist frei skalierbar, und vieles mehr.

Für alle, die mehr über die neue Kommunikations-Software Suxcess für Windows™ wissen wollen oder diese gleich *samt Zugangskennung* zum BTX-Datennetz zum Preis von nur **öS 1.080,-** inkl. MWSt. bestellen wollen: **Anruf oder Fax genügt! (Rückseite)**

Fax-Antwort-Formular auf der Rückseite ➔

BTX
willkommen im Datennetz
Ihre Post

Fax - Antwort



(0316) 91 73 53

An
BTX-Hotline
Karlauergürtel 1
8020 Graz

☐ **Ja**, ich bestelle.....Stück der Kommunikations-Software Suxxess und meine persönliche Zugangs-Kennung zum BTX-Datennetz um **nur öS 1.080,-** inkl. MWSt.

Die Kommunikations-Software Suxxess + persönliche Zugangs-Kennung senden Sie an diese Adresse:

**Für alle Fragen:
BTX-Hotline 0660 212**

Name, Firma

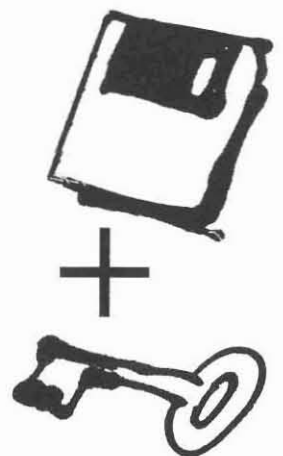
Anschrift

PLZ

Ort

Telefon (für allf. Rückfragen)

Unterschrift bzw. firmenmäßige Zeichnung



PC-NEWS-Disketten

Es bedeuten:

- (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-344(4): Point-Installationsdiskette (Update)

PCN-DSK-365(4): OS/2 Ver.2.1 Demodiskette

PCN-DSK-366(4): WinPrint Demodiskette, LaserMaster

PCN-DSK-367(3): Demos von Poet-Database,
Greenleaf Comm++, Silicon River

PCN-DSK-368(3): Unicom 3.1

Public-Domain-Version eines Terminal-Programms für Windows. Auszüge aus den Eigenschaften: Automatisierbare Funktionen von Anwahl, Up-Download mit Script-Sprache WINSRIPT. Hintergrundbetrieb möglich. Transferprotokolle: X-Y-ZModem, Ymodem Batch, Ymodem-G, ZModem Resume, Kermit, CompuServe B, QuickB and ASCII. 16 COLOR TERMINAL EMULATION, DEC VT102 (ANSI), DEC VT52, ANSI-BBS and TTY. GIF-Bildbetrachter. HOST MODE, Darstellung der Modem-Signallampen am Bildschirm, Buffer für bis zu 5000 Zeilen. Aufnahme wiederkehrender Bedienungssequenzen; DDE SERVER CAPABILITY; DOWNLOAD FILE PROCESSOR (Automatische Ausführung von Befehlen nach dem Downloaden); uvam.

PCN-DSK-369(4): PCC-Magazin 19 (7-8/93)

CALLP	EXE	gesidenter Adressmanager m. Wählfunkt.
GIFD	EXE	Gif-Bilder betrachten+katalogisieren
INSTALL	BAT	PCC 19 Installation INSTALL A: C:
NEOP	EXE	Grafikbearbeitungsprogramm
PARA	EXE	Parallele Datenübertragung PC<->PC
SCAN105	EXE	SCAN + CLEAN V 105 auch f. Windows
SDRAW	EXE	Coreldraw - Einstellhilfe
TERMIN	EXE	Terminplaner
UTIL	EXE	8 kleine Assemblertools

PCN-DSK-370(4): PCC-Magazin 20-1 (9-10/93)

CROW	EXE	Kreuzworträtselgenerator
FLIPPER	EXE	Flipperspiel am PC
INSTALL	BAT	PCC 20 Installdatei: Install A: C:
MI-CAD	EXE	3D-CAD Programm
NSHOWPRO	EXE	Slideshow mit vielen Effekten + Sound
PCCDISK	EXE	Inhalt der PCC-Disk 1-19 als Textdatei
PRN_SCR	EXE	Bildschirminhalt als Datei sichern
QMENU	EXE	Grafische Menüoberfläche für DOS
SYSLOCK	EXE	Paßwortschutz bei PC-Start
WAVUTIL	EXE	WAV Musik auch ohne Soundkarte spielen
WOUTIL	EXE	In allen Laufwerken Dateien suchen

PCN-DSK-371(3): PCC-Magazin 20-2 (9-10/93)

KNACKS	EXE	Kartenspiel (Hosen runter, Schnauz)
VOKSTAR	EXE	Vokabelstar (Lernprg. mit Tonausgabe)

PCN-DSK-372(4): PC-NEWS-35, Programme

HARDS	LZH	303550	Hardwarenahes Programmieren, Teil 5
SCC	LZH	264809	Source-Code-CD-ROM, Gesamtinhalt
AMMU	LZH	2395	DERIVE-Beispielprogramme
HDBENCH	LZH	20304	Testprogramme Programme für Controllervergleich
CUG	LZH	320750	C User Group CD-ROM, Gesamtinhalt Bestellformular

PCN-DSK-373(3): Spiel Waldgeister

Beschrieben in PC-NEWS-35, Seite 39.

Die Autoren

Fangl, Walter, Firma SYWA, Gentzgasse 9, 1180 Wien, TEL: 470 70 05, FAX: 470 70 06.

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: fiala@email.tgm.ac.at

Fleck, Eduard, Dipl.-Ing., Lehrer an der höheren Abteilung für Nachrichtentechnik und Elektronik im TGM. TEL: 33-1-26. EMAIL: fleck@email.tgm.ac.at

Gloner, Klaus, Firma Gloner, Dr. Hainstraße 8, 5620 Schwarzach, TEL: 06415/7215, FAX: 06415/7218.

Grausgruber, Thomas, Student, TU-Wien, email: tgn@eichow.tuwien.ac.at.

Ilsinger, 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.

Kadan, Dieter, Mag., AHS-Professor für Mathematik, Physik und Informatik am Privatschulzentrum KOLLEGIUM KALKSBURG in Wien 23.

König, Rudolf, Dipl.-Ing., Vorstand der höheren Abteilung für Nachrichtentechnik und Elektronik am TGM; Obmann des PCC-TGM. EMAIL: koenig@email.tgm.ac.at.

Lohninger, Hans, Dr., Institute of General Chemistry, TU-Wien, Lehargasse 4/152, A-1060 Vienna, Austria, email: hlohning@email.tuwien.ac.at, FAX: ++43-1-587-4835, voice: ++43-1-58801-5048.

Pöschko, Friedrich, Elektrotechnik, TU-Wien.

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: riemer@email.tgm.ac.at.

Sautner, Johannes, Student der Betriebsinformatik an der UNI-Wien. 1989 Einstieg in die BTX-Szene mit DECODIX 1.02. Autor mehrerer BTX-Programme basierend auf DECODIX. Seit 1992 im erweiterten Vorstand des MCCA. BTX: 912-222-628.

Sautner, Margit, Juristin im Bundeskanzleramt, ist seit 1991 in der Abteilung für Datenschutz tätig.

Scharl, Wolfgang, Dipl.-Ing., Lehrer an der höheren Abteilung für Nachrichtentechnik und Elektronik im TGM. BTX: 912-218-218. EMAIL: scharl@email.tgm.ac.at.

Schlögl, Helmuth, Jahrgang 1940, Obmann des Btx- u. Microcomputer Clubs 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.

Schuieler, Dominik, Student der Elektrotechnik an der TU-Stuttgart, FIDO: 2:2407/60.

Schüller, Peter, Mag. Dr., Arbeitsgruppe moderner Mathematikunterricht, TEL: (022-36)22-602.

Schwarz, Horst, Mag., Professor an der HTBL-Wien-10, Betreuer des **PCC-TGM**.

Weissenböck, Martin, Dipl.-Ing. Mag. Dr., Direktor der HTL Wien IV, Leiter der **ADIM** und Autor von **ADIM**-Skripten. Vorstandsmitglied des **PCC-TGM**. BTX: 912 213 458. FIDO: 2:310/1.35.

Zelinka, Wolfgang, Ing., Lehrer an der HTL-Hollabrunn; Beirat für PCAD beim **PCC-TGM**. BTX: 912 216 424, c/o HTBLA-Hollabrunn, Tel: (02952) 33 61 -0, FAX: Durchwahl -(02952) 33 61-215.

Liebe Mitglieder des PCC-TGM!

Aus aktuellem Anlaß wende ich mich an Sie persönlich mit der Bitte, unseren Club durch eine **intensive Mitgliederwerbung** aus seiner prekären finanziellen Situation zu helfen.

Durch unsere Clubleistungen, wie

- kostenlose Seminare (z.B. DOS, ACAD, PASCAL, C, EXCEL, WINDOWS, WINWORD),
- kostenlose Clubzeitschrift **PC-NEWS** 4 bis 5 mal pro Jahr,
- kostenlose technische Auskünfte bez. Hard- und Softwareprobleme während der Sekretariatszeiten,
- Vermittlung von kostengünstigen Hard- und Softwareangeboten von seriösen Händlern,
- Angebot von kostengünstigen Schul-Software-Lizenzen durch Fa. COMP-DELPHIN,
- bis Frühjahr 1993 sehr kostengünstiges Diskettenservice,
- sehr kostengünstiger Scannerdienst,
- kostenlose Mailboxdienste **His Master's Voice**

wurde unser bisheriges Budget so beansprucht, daß wir derzeit schon **fast im Minus** sind. Und dies, obwohl alle Clubfunktionäre und Sie als Betreuer die Tätigkeit für den Club ehrenamtlich und ohne irgendeine finanzielle Zuwendung ausüben.

Eine der Möglichkeiten, unser Budget zu sanieren, sehe ich nun in einer Vergrößerung unseres Mitgliederstandes und eine weitere darin, Software-Lizenzen zu ähnlich günstigen Bedingungen wie früher anbieten zu können. Mit den Softwarelieferanten scheint sich so eine Lösung anzubahnen, mit weiteren Anbietern werden wir noch verhandeln. Des weiteren wollen wir die Weitergabe von PD-Software forcieren. Ich werde Sie nach Abschluß der Gespräche sofort informieren.

Weiters hat der Club bei der Generalversammlung 1993 beschlossen, zur Platzsicherung von Kursplätzen **bei der Anmeldung** einen Beitrag von 25.-/Abend einzuheben, um allfälligen Anmeldungen "auf Verdacht" entgegenzuwirken. Gleichzeitig dient dieser Betrag zur Instandhaltung der verwendeten Club-Hardware. Wir müssen nun diesen Betrag auf 50.-/Abend erhöhen, um auch die Honorare der Vortragenden zum Teil abdecken zu können, da das Pädagogische Institut des Bundes nicht mehr mitzahlt.

Abschließend ersuche ich Sie noch, mir gegebenenfalls Ihre weiteren Vorschläge für kostendeckende beziehungsweise sogar einnahmenserhöhende Clubleistungen zu übermitteln und damit die künftige Existenz unseres Clubs zu sichern und verbleibe

Mit den besten Grüßen

Ihr Rudolf König

Anm.: **PC-NEWS** und **His Master's Voice** sind das Vorfeld der Clubs. Die Aktivitäten dieser beiden non-profit-Unternehmungen sind die Quelle zukünftiger Clubmitglieder, daher sollte es unser aller Interesse sein, diese zu erhalten. Finanzprobleme des Clubs drücken sich durch Reduktion der Aufträge oder Unterstützungen aus und schmälen indirekt deren Gewinn, nämlich die Anzahl der **Leser** und **User**. Es schwinden die Möglichkeiten, schwinden die Leistungen, verschwindet der Anreiz, **Leser** zu sein, **User** zu sein, **Mitglied** zu sein.

In welcher Form der **PC-NEWS-Eigenverlag** zu der Problemlösung beitragen will, können Sie symbolisch an dieser Ausgabe ablesen: durch ein **maximales Informationsangebot**, damit Sie, liebes Mitglied, mit dieser Ausgabe in der Hand, für das Anliegen der **PC-NEWS**, von **His Master's Voice** und von **PCC-TGM** bei Freunden eintreten können. Da die **PC-NEWS** auch für Clubmitglieder ein wichtiges Identifikationsmerkmal sind, werden wir mit gemeinsamen Anstrengungen die gewohnte Qualität und Quantität auch im nächsten Jahr zu erreichen versuchen; eine 'Meinungsumfrage' im Vorstand des **PCC-TGM** hat das ergeben. Wir werden alles in diesem Sinne unternehmen.

Franz Fiala

SEMINARE

Im November 1993 beginnt wieder 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.

Teilseminar S1:

INHALT:

ACCESS, Datenbank unter WINDOWS, Einführung für Anfänger: Aufbau eines Datenbanksystems. Erzeugen von Abfragen, Formularen und Berichten mit Hilfe des Assistenten. Serienbriefherstellung und automatisieren mit Macros.

TERMINE:

Mi 3.11./10.11./17.11./24.11./1.12.1993

VORTRAGENDER:

Dipl. Ing. Fleck SAAL: 1400

UNKOSTENBEITRAG:

S 250.-

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

Neues von der ADIM

Vor wenigen Wochen wurde eine Probenummer der **PC-NEWS** an alle **ADIM**-Interessenten versandt. Damit soll diese Zeitschrift immer mehr auch als Sprachrohr der **ADIM**-Freunde dienen und über aktuelle Pläne informieren.

Modemaktion

Die neuesten Geräte der Type 1414AX werden mit der Version 7 der Software ausgeliefert. (Sie können die Version mit dem Befehl `ati0i1i9` auf den Bildschirm rufen.) Nach Aussagen der Firma wird damit eine Verbesserung des Verhaltens bei schlechten Leitungen erreicht. Sie können diese Version auch aus der Mail-Box, Area **ADIM**, Dateiname `1414AX70.LZH` laden.

Zum Modem 1414AX: das Ziel der Modemaktion war ein Modem zu finden, das möglichst viel "kann" und ein optimales Preis-Leistungsverhältnis aufweist. Es war nicht das Ziel, das billigste 14400-Baud-Modem zu organisieren. Für Interessenten, die nicht alle Features des 1414AX benötigen, wird es etwa ab Jänner 1994 den Typ 1414CX geben: Datenübertragung und Fax sind selbstverständlich, ebenso alle Betriebsarten. Eingespart werden Dinge wie das Einstellen der Parameter über die Leitung usw. Erwarteter Preis:

um 3000 Schilling inklusive Mehrwertsteuer.

Bei Interesse schreiben Sie bitte an die **ADIM**.

Mini-Turbo

Im **ADIM**-Band 47, Turbo-Pascal, werden alle Elemente der Sprache dargestellt. Durch die größere Schrift und einige Erweiterungen hat das Skriptum nun 180 Seiten und kostet 120 Schilling. Eine der Ideen der Skriptenreihe war, eine möglichst vollständige Lernhilfe zu einem sehr günstigen Preis anzubieten. Dies ist zweifellos immer noch der Fall - ein im Umfang vergleichbares Buch würde wohl mindestens 500 Schilling kosten.

Die Sprache soll, wie erwähnt, vollständig dargestellt werden. Von zwei Schulen wurde nun bemerkt, daß der gesamte dargebotene Stoff nicht in 2*2 Jahreswochenstunden unterzubringen ist und das Skriptum nicht verwendet wird, um keinen Schüler zu frustrieren. Abgesehen davon, daß auch bei der Verwendung des halben Skriptums das Preis-Leistungsverhältnis wohl immer noch gut ist, möchte ich einen Vorschlag für ein "Mini-Pascal-Skriptum" vorlegen. Es würde die Kapitel **0** (Editor), **1** (Allgemeines, Anweisungen, Kommentare), **2** (Verzweigungen), **3** (Wiederholungen, Schleifen), **4** (Datentypen, beschränkt auf die wichtigsten und ohne selbstdefinierte Typen), **5** (Unterprogramme und Funktionen, ohne Rekursion usw.), **6** (Grafik, eventuell erweitert), **7** (Objektorientierte Programmierungen, aber nur die Idee dazu) und **8** (Arrays) enthalten. Der Umfang wäre etwa die Hälfte und damit würde auch der Preis rund 60 Schilling betragen.

Was meinen Sie dazu? Bitte schreiben Sie an die **ADIM**, Postfach 23, 1191 Wien.

Von Turbo-Pascal zu C++

Die Bedeutung von C++ steigt ständig. "Von Turbo-Pascal zu C++" ist ein neues Projekt für ein kleines Skriptum: die wichtigsten Unterschiede beider Sprachen. Gute Pascal-Kenntnisse werden vorausgesetzt. Durch einen zweispaltigen Druck sollen die beiden Sprachen unmittelbar verglichen werden können. Was meinen Sie dazu? Briefe bitte an die **ADIM**.

ADIM-Area in der Mailbox

In der Mailbox **His Master's Voice** ist eine Area **ADIM** eingerichtet; von dort können die aktuellen Modem-EPROM-Versionen, **ADIM**-Beispieldisketten und README-Dateien (Druckfehlerberichtigungen) geladen werden.

In diesem Heft finden Sie nach einer vergleichenden Darstellung einiger **ADIM**-Bände eine Fehlerberichtigung zu **ADIM**-Band 50 (C++).

Martin Weissenböck

Liebe Mitglieder des MCCA

Wir wollen Sie daran erinnern, daß Sie ihr **PC-NEWS**-Abo beim **MCCA** jährlich erneuern müssen. Benützen Sie bitte die Antwortseite *255091#. Wenn Ihnen die Zeitschrift gefällt, geben Sie sie an Freunde weiter, vielleicht werden auch sie **MCCA**-Mitglieder. Als **MCCA**-Mitglied haben sie einen äußerst vergünstigten Abopreis von nur ÖS 50.- pro Jahr!

Ein neuer Dienst im BTX

Wie schon einmal in den **PC-NEWS** erwähnt, hat die Post seit Kurzem den "X.29"-Dienst als Testbetrieb gestartet. Nunmehr ist es mit fast jedem Terminalprogramm möglich, über BTX externe Rechner über Gatewayseiten zu erreichen.

Dies ist zur Zeit leider nur identifiziert möglich. Die Zugangstelefonnummern sind für 1200 und 2400 Baud "06611", für 9600 und 14400 "06612". Für den ISDN Zugang gibt es die Nummer 03289. Auch für diesen Dienst ist der BTX-Rechner von ganz Österreich zum Telefon-Ortstarif (ÖS 0,67/Minute) zu erreichen.

Was sind nun die Unterschiede zu CEPT-BTX?

Beim Anwählen der BTX-Zentrale wird von ihr eine Identifikation des Endgerätes erfragt. Dieser **Facility-Identifyer** besagt ob es ein "MUPID", ein PC mit einer BTX-Softwareemulation oder ein ASCII-Terminal ist. Ist es ein ASCII-Terminal, wird man zu einem anderen Sektor des BTX-Zentralenrechners weitergeleitet. Hier gibt es nur den ASCII Zeichensatz, aber eine 80-Zeichenanzeige. Natürlich sind hier BTX-Bilder sinnlos, weil sie falsch interpretiert werden.

Ein Verzeichnis von angeschlossenen Rechnern und Netzwerken (CompuServe, Internet etc.) wird von der Zentrale zur Verfügung gestellt. Nach dem altbewährten Prinzip der Gatewayseiten, wird man über X.25 und X.29 an den jeweiligen externen Rechner herangeführt.

Nach dem Verbindungsaufbau, werden mit diesem Daten ausgetauscht, wobei die BTX-Zentrale auf durchgehende Transparenz umschaltet. Es kann nun mit dem jeweiligen Rechner ein DFÜ-Protokoll vereinbart werden und Up- u. Download können durchgeführt werden..

Ich kann mir nun gut vorstellen, da der Anwählvorgang an eine BTX-Zentrale (flächendeckend über ganz Österreich) leicht automatisierbar und äußerst kostengünstig ist, daß Banken, Großkaufhäuser, der Kraftfahrzeughandel und viele Zubehörfirmen an diesem Dienst Gefallen finden werden. Applikationen, die auf Rechnern, Fileservern laufen, brauchen nur um die Zugangsberechtigung erweitert werden. (Dies könnte ein Vorschalt-PC bewerkstelligen) Für diese Applikationen kann der Betreiber des Rechners einen zusätzlichen zeitgesteuerten Tarif einsetzen, der eventuell monatlich zu zahlende Gebühren abdeckt. Ein Staffeltarif für Kommerzkunden und private Kunden wäre so zu realisieren. Eine Benutzungsgebühr z.B. für Telebanking könnte so einer Leasingrate für das jeweilige Programm bis zum Erreichen des Kaufpreises gleichgesetzt werden.

Bei der Erst-Installation von Telebanking könnten verschiedene Masken, Utilities und Eingabeformulare zum Download bereitgestellt werden, die das weitere Arbeiten mit Nettdaten beschleunigen. Lesen Sie dazu weitere Details im Beitrag von Johannes Sautner.

Herzliche Grüße, Ihr H. Schlögl

Neues aus der Mailbox His Master's Voice

Betreuung

In den **PC-NEWS** 3/93 (blau) wurde darüber berichtet, daß ich ab 1. Oktober meinem Vaterlande dienen werde. Die Situation hat sich mittlerweile so entwickelt, daß ich nach 13 Tagen aus gesundheitlichen Gründen wieder abgerüstet bin. Die Betreuung ist also wieder so, wie Sie das alle gewohnt sind.

Interessiert es Sie, wieviel Zeit so in die Betreuung der Mailbox fließt? Nun, ich sitze jeden Tag ca. vier bis fünf Stunden vor der Mailbox, manchmal auch mehr, und das 7 Tage in der Woche - auch Feiertags. Das macht also im Durchschnitt etwa eine Ganztagsbeschäftigung aus.

Dafür erreicht die Mailbox eine Verfügbarkeit von ca. 99%. Solche Verfügbarkeitsdaten sind derzeit nur im Großrechnerbereich Standard und kosten den Betreiber sehr viel Geld.

Bei Mailboxen, die in Firmen oder etwa Schulen installiert sind, kann der Sysop nicht eingreifen, wenn die Mailbox am Freitag Nachmittag abstürzt. Das war auch der Grund, warum die Mailbox seit einigen Jahren zu Hause in meinem Wohnzimmer steht.

Fakten

Manche können sich unter der Mailbox einfach nichts vorstellen. Da steht nun irgendwo so ein Computer, der viel Geld kostet. Was beschäftigt diesen Computer den ganzen Tag?

Er hat für

150	Anrufer täglich zu sorgen, die auf
3	Telefonleitungen 24 Stunden am Tag anrufen können. Gesamt gibt es im System
2.500	Benutzer , die im Jahre 1993 angerufen haben. Davon sind ca.
1.000	Benutzer mit Namen und Adresse bekannt . Ungefähr
200	Mitglieder des PCC-TGM nehmen diesen Dienst in Anspruch. In den letzten beiden Monaten haben sich
70	Benutzer neu eintragen lassen, von diesen haben sich ca.
40	für den PCC-TGM interessiert und sind somit potentielle neue Clubmitglieder

An downloadbaren Dateien stehen aus den verschiedensten Bereichen dem Anrufer auf derzeit

1.600	Megabyte gesamter Festplattenkapazität
9.213	Dateien zum download zur Verfügung, die gesamt
1.000	Megabyte der Festplatte belegen.

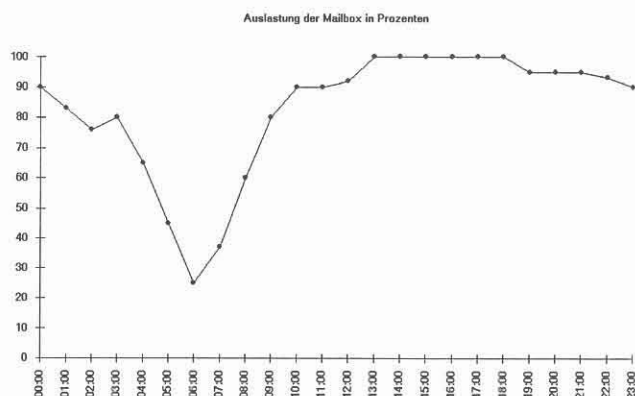
Außerdem hat der Benutzer die Möglichkeit aus

70	verfügbaren Diskussionsforen , die gemeinsam
20	Megabyte an Platz belegen und derzeit
12.216	Mitteilungen beinhalten, zu wählen.

Außerdem dient der Computer noch nebenbei als Arbeitsrechner. So wird zum Beispiel gerade dieser Beitrag auf ihm geschrieben.

Auslastung

Die folgende Grafik zeigt die Auslastung der Mailbox in Prozent. Dort, wo die Kurve am niedrigsten ist, ist die Chance am höchsten, daß Sie durchkommen. Dort, wo die Grafik 100% erreicht, ist am öftesten besetzt. Daher haben Sie zwischen 13h und 18h am wenigsten Chancen durchzukommen.



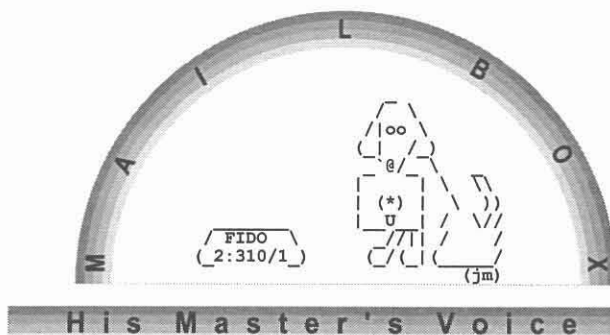
Finanzierung

Die **Mailboxkosten** unterteilen sich vor allem in Telefonkosten, Hardwarekosten, und sonstige Kosten.

An **Telefonkosten** laufen pro Telefonrechnung durchschnittlich ca. ÖS 7.000,- an, das sind im Jahr daher ca. ÖS 40.000,-. Diese Kosten entstehen durch den Import der Echomail und der Dateien, die via FidoNet verteilt werden. Das ist notwendig, um die Mailbox ständig mit den neuesten Dateien zu versorgen. Die Telefonkosten wurden bisher vom **PCC-TGM** getragen.

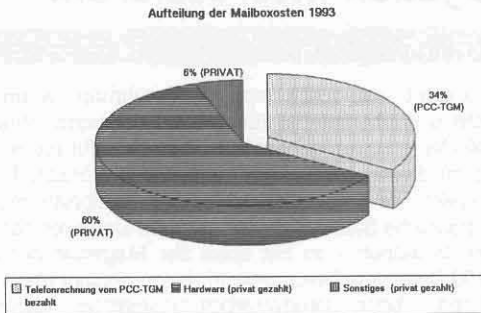
Die Telefonrechnung macht jedoch nur einen kleineren Teil der Gesamtkosten der Mailbox aus. Seit Februar 1992 kauft der Club keine neue Hardware mehr für die Mailbox. Die notwendigen Neuanschaffungen wurden von mir privat mit Spendenunterstützung der Mailboxbenutzer finanziert. Im Jahre '92 haben die zusätzlichen, nicht vom Club getragenen Investitionen ÖS 72.000,- ausgemacht. Dieses Jahr wurde ein neues Modem mit höherer Geschwindigkeit angeschafft, um die Telefonrechnung nicht proportional der steigenden Datenmenge ansteigen zu lassen und ein neuer Streamer zur Datensicherung. Für heuer ist als Anschaffung noch ein Laserdrucker und eventuell eine zusätzliche Festplatte geplant. Es wird daher für 1993 mit einem ähnlich hohen Betrag wie 1992 gerechnet.

Die **sonstigen Kosten** wie Büromaterial und Briefporto machen einen relativ geringen Anteil von ca. 6.000 Schilling jährlich aus.



Kostenaufteilung

Daraus ergibt sich in etwa folgende Aufteilung der Kosten:



Aus dieser Grafik sieht man, daß nur ca. 1/3 der Gesamtkosten der Mailbox vom **PCC-TGM** getragen werden. Die restlichen 2/3 der Kosten werden von mir privat mit Spendenunterstützung der Mailboxbenutzer abgedeckt. Für ein persönliches Hobby ist mir das jedoch zu viel.

Auf einer Vorstandssitzung des **PCC-TGM** im Oktober 1993 wurde berichtet, daß im TGM eine eigene Mailbox als HTL-Mailbox aufgestellt werden soll. Diese Mailbox soll auch die Clubmailbox ersetzen. Daher wurde vom Vorstand beschlossen, die Mailbox **His Master's Voice**, sobald die Schulmailbox den Betrieb aufnimmt, nicht mehr zu unterstützen. Genauer kann ich Ihnen leider zu diesem Thema noch nicht bekanntgeben, da ich diese Information ebenfalls nur aus zweiter Hand habe. Zu hinterfragen wäre auf jeden Fall noch, ob die HTL-Mailbox gleichzeitig auch die Clubmailbox sein sollte, aber das müssen Sie, liebes Clubmitglied entscheiden.

Ich plane daher, um die Mailbox in ihrem derzeitigen Umfang weiterführen zu können, ab 1994 Beiträge zur Erhaltung der Mailbox einzuheben. Wie hoch diese Beiträge sein werden, bzw. in welcher Form sie abgerechnet werden, steht jetzt noch nicht fest. Zu diesem Thema wird derzeit eine Befragung der Benutzer der Mailbox via Fragebogen durchgeführt. Über die Ergebnisse dieser Befragung werde ich Sie in der nächsten Ausgabe der **PC-NEWS** informieren.

Eine andere Gruppe der Benutzer der Mailbox, nämlich die Points, haben bei einem Pointtreffen einstimmig zugestimmt, daß für die bezogene Echomailmenge eine Bezahlung an die Mailbox erfolgen soll. Gedacht wird momentan an ein Modell, bei dem öS 20,- pro Megabyte bezogener Echomail verrechnet wird. Dabei würden sich für die Points durchschnittliche Kosten von S 50,- pro Monat ergeben.

Point Installation

Im Heft 2a/93 wurde die Point Installation mit einer zugehörigen Diskette (**PC-NEWS** Diskette 344) beschrieben. Auf der Diskette haben sich einige Fehler eingeschlichen, so hat zum Beispiel ein Verzeichnis gefehlt. Außerdem wurde die Mailbox von mir auf 4-D umgestellt. Das heißt die beschriebenen FakeNet-Adressen gibt es nicht mehr. Um dem allen Rechnung zu tragen, wurde die Diskette 344 upgedatet.

Die Diskette ist als **PCN_344.DIM** von der Mailbox aus dem Dateibereich **PCNDISK** downloadbar. Dieses File enthält ein Diskimage, daß mittels des Programmes **DSKEXP** auf eine Diskette überspielt werden kann. Der Befehl steht unter der Annahme, daß das File auf die Diskette im Laufwerk "A" überspielt werden soll, wie folgt aus:

```
DSKEXP PCN_344.DIM A:
```

In einer der folgenden Ausgaben der **PC-NEWS** wird es eine Fehlerberichtigung und weitere Ausführungen zum Thema FidoNet-Points geben. Wenn Sie dazu Anregungen haben, senden Sie diese bitte an mich.

ADIM

Die Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik (**ADIM**) hat nun einen eigenen Dateibereich **ADIMin** der Mailbox. Dort werden die neuesten Beispiele zu den Skripten und auch Informationen zur **ADIM** zum Download bereitgestellt. Der Leiter der **ADIM**, Herr Dir. Weißenböck ist über die FidoNet Adresse 2:310/1.35 erreichbar.

PCC-TGM

FidoNet Echomail Area PCC-TGM

Dem in der Mailbox **His Master's Voice** gegründeten und aufgrund der nicht-Teilnahme der Clubverantwortlichen, die es wohl als einzige mit Information füllen hätten können, schlummernden lokalen Mitteilungsbereich (Echomail Area) **PCCTGM** wurde neues Leben eingehaucht. Dieser Mitteilungsbereich ist jetzt als FidoNet Echomail Area verfügbar. Diese Area wird jetzt auch nach Graz geroutet und ist bei diversen Grazer FidoNet Boxen erhältlich.

Die **ADIM**-Graz, die Redaktion der **PC-NEWS**, sowie einige Clubmitglieder nehmen an den Diskussionen in diesem Echomail Bereich schon recht fleißig teil. Dieser Bereich dient dazu, die Mitglieder des **PCC-TGM** und andere interessierte Personen immer mit den neuesten Informationen über den Club zu versorgen.

Falls Sie ein **PCC-TGM**-Mitglied in der Steiermark sind, fragen Sie Ihre nächstgelegene FidoNet Mailbox. Ansonsten richten Sie Anfragen an mich. Wir können, falls das gewünscht wird, diesen Bereich sicher auch an andere Mailboxen in den anderen Bundesländern verteilen.

Der **PCC-TGM** ist leider seit dem Ausscheiden von Bernhard Leeb aus dem Clubsekretariat nicht mehr direkt über FidoNet erreichbar.

P-CAD

Der verwaiste P-CAD-Bereich wird hoffentlich nun auch wieder eine Aufwertung erleben. Die Firma **Rekirsch-Elektronik** hat in diesem Area durch eine einmalige Diskettensendung teilgenommen. Danach gab es keine Unterstützung mehr. Hilfesuchende mußten vergebens auf Antworten zu Ihren Fragen warten. Das wird sich nun hoffentlich alles bessern. Der P-CAD-Bereich wurde von Hr. Ing. Wolfgang Zelinka übernommen, der sich nun fleißig Ihrer Probleme annehmen wird. Er ist auch als Point im FidoNet unter der Nummer 2:310/1.37 erreichbar.

PC-NEWS

PC-NEWS Disketten

Die **PC-NEWS** Disketten sind seit dieser Ausgabe der **PC-NEWS** im Dateibereich **PCN** ab der Diskette Nummer 200 vollständig lagernd. Sollten Sie eine ältere Diskette benötigen, hinterlassen Sie mir bitte eine Nachricht. Die Disketten sind alle als Disketten-Images abgelegt. Das heißt, daß die Disketten 1:1 in einer Datei abgelegt wurden. Das Programm, das Sie zum Verwandeln der Dateien in Disketten benötigen, heißt **DSKEXP.ZIP** und steht im gleichen Bereich zur Verfügung. Einer der besonderen Vorzüge dieses Programmes ist auch, daß sowohl eine DOS, als auch eine OS/2 Version verfügbar ist. Die Verwandlung einer *.DIM-Datei in eine Diskette funktioniert folgendermaßen:

```
für DOS:      DSKEXP PCN_xxx.DIM A:
```

```
für OS/2:     DSKEXP2 PCN_xxx.DIM A:
```

Die Disketten sind alle als 3,5"-Images abgelegt. Das Programm **DSKEXP** verfügt aber auch über eine Option zur Konvertierung dieser Images auf 5 1/4" Disketten.

Bestellung der PC-NEWS über die Mailbox

Seit kurzem besteht die Möglichkeit, die **PC-NEWS** über die Mailbox **His Master's Voice** zu bestellen. Dazu wurde im Hauptmenü der Menüpunkt **PC-NEWS** eingerichtet. Sie können von dort aus ein Probeheft, ein ABO (150,- für 5 Ausgaben) oder auch einzelne Hefte der **PC-NEWS** bestellen. Eine Clubmitgliedschaft beim **PCC-TGM** ist dafür nicht erforderlich. Bei diesem Vorgang wird automatisch eine sogenannte Netmail an den Redakteur der **PC-NEWS** (Franz Fiala) generiert, der die Nachrichten dann an seinen Point (2:310/1.36) zugestellt bekommt.

Ich hoffe Ihnen mit diesen Informationen aus der Welt der Mailbox gedient zu haben und verabschiede mich mit herzlichen Grüßen

Werner Illsinger

Urheberrecht_s-Novelle 1993:

Die neuen Rechte des Anwenders

Walter J. Jaburek, Diebold

Die Urheberrechtsnovelle 1993 hat dem EDV-, bzw. Software-Anwender nicht nur Schlechtes gebracht-- etwa das Verbot der Vervielfältigung zum eigenen Gebrauch--sondern auch explizit definierte Rechte.

Diese Rechte werden allerdings in eher von der Herstellerseite motivierten "kritischen" Aufsätzen meist vergessen. Kernstück der Softwareregelung zum Thema "Schutz des Softwarekonsumenten" sind die Absätze 2 bis 4 des Paragraphen 40d des neuen Urheberrechtsgesetzes:

Paragraph 40 d.

(2) Computerprogramme dürfen vervielfältigt und bearbeitet werden, soweit dies für ihre bestimmungsgemäße Benutzung durch den zur Benutzung Berechtigten notwendig ist; hiezu gehört auch die Anpassung an dessen Bedürfnisse.

(3) Die zur Benutzung eines Computerprogramms berechnete Person darf

1. Vervielfältigungsstücke für Sicherungszwecke (Sicherungskopien) herstellen, soweit dies für die Benutzung des Computerprogramms notwendig ist;

2. das Funktionieren des Programms beobachten, untersuchen oder testen, um die einem Programmelement zugrunde liegenden Ideen und Grundsätze zu ermitteln, wenn sie dies durch Handlungen zum Laden, Anzeigen, Ablaufen, Übertragen oder Speichern des Programms tut, zu denen sie berechnete ist.

(4) Auf die Rechte nach Abs. 2 und 3 kann wirksam nicht verzichtet werden; dies schließt Vereinbarungen über den Umfang der bestimmungsgemäßen Benutzung im Sinn des Abs. 2 nicht aus.

Dies bedeutet: Solange keine gegenteiligen Verträge mit dem Softwareverkäufer, also zB dem PC-Händler, abgeschlossen wurden, darf Software ohne weiteres auch auf verschiedenen EDV-Systemen kopiert werden. Die Grenze der Vervielfältigung bildet einerseits das Recht des Urhebers, Veröffentlichungen zu verbieten, und andererseits die "bestimmungsgemäße Benutzung" von Software.

Veröffentlichungen liegen nach der Rechtsprechung dann vor, wenn durch die Vervielfältigungen die Software einem großen (über 100) Kreis von Personen auch innerhalb eines Unternehmens zugänglich gemacht wird oder andererseits, wenn die Software durch die Vervielfältigungen einem kleineren Kreis (ab etwa 10 Personen) zugänglich gemacht wird, die untereinander durch keinerlei Beziehungen gekennzeichnet sind, wie dies bei frei zugänglichen Veranstaltungen der Fall ist.

Was schließlich die "bestimmungsgemäße Benutzung" von Software ist, daran werden noch Generationen von Juristen herumräteln. Zumindest das Einspielen von dazu geeigneter Software für PCs auf Fileservern ist sicherlich auch bestimmungsgemäße Benutzung.

Es ist also keineswegs so, daß das neue Urheberrecht besagt, eine Softwarekopie dürfe nur auf einem PC verwendet werden. Die Abgrenzung des Erlaubten vom Unerlaubten wird sich erst im Laufe der Zeit herauskristalisieren bzw. durch (schriftliche von beiden Seiten unterschriebene) Verträge definiert werden. Einseitig formulierte Beipackzettel zur Software reichen dazu keineswegs aus. Positiv für den Softwareanwender wirkt sich auch aus, daß jetzt diese freien Werknutzungsrechte und auch das Recht der Anpassung der Software an die Bedürfnisse des Anwenders vertraglich nicht ausgeschlossen werden können. Im Softwarevertrag kann laut Bemerkungen des Justizministeriums nur die Anzahl der Anwender, die eine Software nutzen dürfen, eingeschränkt werden. □

Datenschutz

sollte jeden interessieren!

Margit Sautner, MCCA

Mit Datenschutz kommt man meist dann in Berührung, wenn man persönlich adressiertes Werbematerial zugeschickt bekommt, ohne daß man mit dem Absender je zu tun gehabt hat und sich dafür interessiert, wie der Absender zu dieser Adresse gekommen sein könnte. Das ist aber nur ein Teilaspekt. Die Einhaltung der Datenschutzbestimmungen ist aber auch für öffentliche Stellen sehr wichtig, die Datenverarbeitung im großen Rahmen betreiben - so hat etwa der Magistrat der Stadt Wien allein über 100 Datenverarbeitungen, in denen personenbezogene Daten gemeldet sind, beim Datenverarbeitungsregister gemeldet! Auch im Schulbereich werden Daten von Schülern und Lehrern gespeichert; für die Datenverarbeitungen an den Wiener Schulen ist etwa der Stadtschulrat für Wien verantwortlich. Datenschutz sollte darüber hinaus aber jeden interessieren, der eine EDV-Anlage oder einen PC betreibt und mit personenbezogenen Daten arbeitet, weil dies etwa die Pflicht zur Registrierung beim Datenverarbeitungsregister und der Führung der DVR-Nummer begründet.

In diesem Artikel sollen nur die Rechte der "Betroffenen" (das sind im "Datenschutzjargon" jene, deren Daten verwendet werden) gegenüber solchen Unternehmen behandelt werden, die mit ihren Adressen Werbung betreiben.

Das **Datenschutzgesetz**, das 1980 in Kraft trat und seither schon mehrmals novelliert wurde, schützt **personenbezogene Daten**; das sind etwa Name, Adresse, Telefonnummer, Geburtsdatum, aber auch sensiblere Informationen wie die Religionszugehörigkeit, medizinische Diagnosen, Beurteilungen etc., vor mißbräuchlicher Verwendung.

Erhält man persönlich adressiertes Werbematerial, obwohl man mit der absendenden Firma noch nie Kontakt hatte, so steht hinter dieser Werbekampagne zumeist ein sogenannter **Adressenverlag** oder ein **Direktwerbeunternehmen**. Seit dem 1. Juli 1993 gelten für diese gewerblichen Unternehmen Bestimmungen, die detailliert regeln, was sie mit personenbezogenen Daten anfangen dürfen und welche Rechte den Betroffenen gegen solche Unternehmen zustehen (§ 262 Gewerbeordnung).

Üblicherweise läuft das "Geschäft mit den Adressen" folgendermaßen ab: ein Unternehmen, mit dem man bereits einmal Kontakt als Kunde oder Interessent hatte, z.B. ein Versandhaus, dessen Katalog man angefordert hat, oder bei dem man etwas bestellt hat, "vermietet" die Adresse (in der Regel einen ganzen Adressensatz) an ein Direktwerbeunternehmen oder einen Adressenverlag, der diese Adressen dann für die Durchführung von Werbeaktionen auch für andere Firmen, die bei ihm Kunden sind, verwendet. Das ist gewerberechtlich zulässig. Allerdings dürfen keine anderen Daten als Name, Titel und akademischer Grad, Adresse, und, falls bekannt, Geburtsjahr, Berufs- oder Geschäftsbezeichnung sowie die Zugehörigkeit zu einer bestimmten Kunden- oder Interessentendatei, verwendet werden. Auf gar keinen Fall dürfen diese Unternehmen "sensible Daten", die etwa auf den Gesundheitszustand oder die politische Überzeugung eines Menschen rückschließen lassen, verwenden.

Will man als Betroffener wissen, woher nun so ein Adressenverlag oder ein Direktwerbeunternehmen die Adresse hat, so kann man ein **Auskunftsbegehren** über die Herkunft der Daten stellen. Das Unternehmen ist dann verpflichtet, kostenlos binnen vier Wochen schriftlich Auskunft zu geben, woher es die Adresse hat. Das Auskunftsbegehren muß allerdings innerhalb von drei Monaten gestellt werden. Außerdem empfiehlt es sich, eine Kopie der Werbeaussendung mitzuschicken.

Darüber hinaus kann man die **Löschung** seines Datensatzes bei einem Adressenverlag oder Direktwerbeunternehmen verlangen; diesem Wunsch muß das Unternehmen innerhalb von vier Wochen nachkommen.

Schluß auf der nächsten Seite unten rechts.

IBM

Neue PS/1: Local Bus, Multimedia-Tower, "Koma" spart Energie

Die PS/1 der IBM - bedienungsfreundliche Komplett-Computer - werden nun kraft "Smart Energy System" umweltbewußt. Der Bildschirm und die Festplatte gehen, wenn nicht benutzt, bescheiden in den "Standby"-Modus und sparen Energie. Noch drastischer bremst "Rapid/Resume" den Stromdurst: In Arbeitspausen schaltet sich der Computer freiwillig ab. Bevor er so ins "Koma" fällt, speichert er noch seinen Status Quo auf Festplatte. Per Knopfdruck erweckt, legt er dann einen "rapiden" Schnellstart hin und zeigt binnen weniger Sekunden wieder den Zustand, in dem man ihn verlassen hat.

Jedes PS/1 kommt komplett mit 14-Zoll-Farbschirm (Super-VGA), Tastatur und Maus und enthält umfangreiche, fertig installierte Software-Pakete. Auch ein Neuling braucht den PC nur einzuschalten und kann sofort loslegen. Die Software umfaßt DOS 6.0, Windows 3.1, Works 2.0 für Windows (Textverarbeitung, Tabellenkalkulation, Grafik), Lernprogramme, Online-Lexikon, Virenschutz u.ä. Ein neuer Multimedia-Tower besitzt zusätzlich CD-Laufwerk, Soundblaster-Audiokarte, Stereoboxen sowie viele Megabyte Multimedia-Programme und Spiele wie "Battlechess".

Alle neuen PS/1 verfügen über 486er-Prozessoren und sind mit 32-bit breitem VESA Local Bus ausgestattet. Diese Bus-Technik überträgt die Daten zwischen Prozessor und Grafik-Chip nicht mit dem normalen Takt des AT-Bus von nur 8 MHz, sondern mit der hohen Taktfrequenz der Platine, z.B. bei einem i486DX2/66 Prozessor mit 33 MHz. Die PS/1 sind gut ausbaubar, selbst das kleinste bietet drei freie Steckplätze, und die Modelle ab 486DX lassen sich auf Pentium-Overdrive aufrüsten.

Technische Daten und Preise

Die Auflösung von 640 x 480 Bildpunkten der SVGA-Farbbildschirme erlaubt eine flimmerfreie Bildfrequenz von mindestens 70 Hertz. Die größeren Modelle bieten auch eine Auflösung von 1024 x 768 Bildpunkten in flimmerfreier Qualität. Sie sind strahlungsarm und entsprechen der schwedischen Norm MPR II.

Die kompakteren Desktop-Modelle haben wahlweise einen i486SX25 oder i486DX33 Prozessor und eine Festplatte mit 85 bzw. 129 MB; die größeren Desktop-Modelle sind mit i486DX2/50 oder i486DX2/66 ausgestattet und bieten Festplatten mit 170 bzw. 253 MB Kapazität. Vier Tower-Geräte arbeiten mit denselben Prozessoren und besitzen Festplatten von 129 bis 253 MB.

Der Multimedia-Tower verfügt über einen i486DX33 Prozessor und 170 MB Festplatte. Die Multimedia-Zusätze wie CD-Laufwerk (für Photo-CD in Multisession, Audio-CD und CD-ROM), Audiokarte Soundblaster 16, zwei Lautsprecher und Multimedia-Software gehören zum Lieferumfang. Mitgeliefert werden auch eine Reihe CDs mit Multimedia-Software und Spielen: "PS/1 Photo Gallery", "Multimedia-Guide", "CD-Player", "Alone in the Dark", "CD-Katalog", "Stunt Island", "ESS Mega", "Battlechess".

Die neue PS/1-Familie kommt im November in den Handel, die Preise reichen von öS 14.990 für das kleinste Desktop-Modell bis zu öS 29.990 für den größten Tower. Der Multimedia-Tower kostet ebenfalls öS 29.990 (unverb. empf. Richtpreise inkl. MWSt.).

IBM

PowerPC-Technologie für Notebooks und Stromspar-Computer

Motorola und IBM stellen die ersten Muster des Mikroprozessors PowerPC 603 vor, der stromsparenden Variante des PowerPC-Chips. Der Prozessor entstammt der Entwicklungssallianz von IBM mit Motorola und Apple. Er ist bereits das zweite Produkt aus dieser Allianz, den ersten Computer mit einem PowerPC 601 brachte IBM bereits im September heraus. Der 603 eignet sich ideal für Notebooks und für umweltfreundliche, da energiesparende Computer. Der PowerPC-Prozessor läuft bis zu viermal so schnell wie der Pentium und wird unterschiedliche Arten von Software nebeneinander betreiben können: Apple, Unix, Windows, DOS, OS/2...

Der PowerPC 603 ist ein RISC-Prozessor, gefertigt in CMOS-Technologie. Er enthält zusätzliche Schaltkreise, die speziell im Batteriebetrieb den Stromverbrauch kontrollieren und reduzieren. Er wird von IBM im Werk Burlington (Vermont/USA) und von Motorola in Austin (Texas) produziert.

IBM und Motorola verfolgen mit der Entwicklung der PowerPC-Prozessorfamilie einen ehrgeizigen Entwicklungsplan: den PowerPC als kommenden Industriestandard zu etablieren. Wenige Monate nach der Ankündigung des Prozessors 601 liegen nun schon die ersten Fertigungsmuster des 603 vor. Die gesamte Prozessorfamilie zeichnen hohe Geschwindigkeit und günstige Preise aus. Die Familie dieser Mikroprozessoren wird das gesamte Computer-Spektrum versorgen - von Notebooks und Personal Computern über Workstations und Midrange-Systemen bis zu Supercomputern. Im gemeinsamen Design-Zentrum Somerset, Austin, Texas arbeiten mehr als 300 Ingenieure von IBM, Motorola und Apple an der Entwicklung der Prozessorfamilie.

improve-it-technologies

AT-286 wird zum 386 oder 486

Es gibt eine einfache Möglichkeit, einen AT-286 zu einem 386er oder gar 486er zu erweitern, ohne gleich das liebgewordene und gut funktionierende Motherboard oder gar das ganze Gerät tauschen zu müssen. Unter dem Begriff **Make-it 386** und **Make-it 486** verstecken sich kleine Module, die in den 286er-Sockel eingesetzt werden und die gewünschte Metamorphose bewirken. Die folgenden Angaben stammen aus dem Prospekt und wurden nicht verifiziert:

Beachten Sie, daß es drei verschiedene Sockelformen für den 286er gibt, die man bei der Bestellung angeben muß:

PGA (Pin Grid Array) Stifte des Chip schauen senkrecht zur Platine, grau-schwarzes Keramik, Sockel etwa so groß wie die CPU.

PLCC (Plastic Leaded Chip Carrier): Anschlußstifte an den Seiten des Gehäuses, schwarzes Gehäuse. Sockel etwas größer als CPU.

LCC (Leadless Chip Carrier): Flache Kontakte an goldfrabener Unterseite des Chip, graus Gehäusefarbe oben, Chip wird mit Metall-Teil gehalten.

Make-It Upgrades sind mit PLCC-Sockel versehen, mit einem PGA-Adapter oder gegen Aufpreis mit einem LCC-Adapter.

Eine typische Geschwindigkeitserhöhung bei Make-it 486 ist etwa ein Faktor 4.

Make-it 386 (\$ 149,- oder \$179,- mit Taktverdoppelung) **Make it 486** \$299,- optionaler Koprozessor \$ 99,-, LCC-adaptor \$ 49,-.

Anschrift: improve-it-technologies, 3707 North Canyon Road, Provo, Utah 84604, TEL: 001-801-224-0088, FAX: 001-801-224-0355.

Fortsetzung und Schluß von Datenschutz sollte jeden interessieren!

Was tut man aber, wenn man nicht nur aus dem Datenbestand eines bestimmten Adressenverlages gelöscht werden will, sondern generell von adressiertem Werbematerial verschont bleiben will? In diesem Fall bietet sich die Eintragung in die sogenannte **"Robinson-Liste"** an. Darunter versteht man eine Liste, die vom Fachverband Werbung und Marktkommunikation in der Sektion Gewerbe und Handwerk der Bundeskammer der gewerblichen Wirtschaft geführt wird und in die alle Personen kostenlos einzutragen sind, die die Zustellung von Werbematerial für sich ausschließen wollen. Man wendet sich am besten schriftlich unter Angabe von Name und Adresse an die genannte Stelle und begehrt die Aufnahme in die "Robinson-Liste". Adressenverlage und Direktwerbeunternehmen dürfen an die in diese Liste eingetragenen Personen keine adressierten Werbemittel versenden oder verteilen und deren Daten auch nicht vermitteln.

Wie behilft man sich aber in jenen Fällen, in denen diese Unternehmen ihren Pflichten nicht nachkommen und z.B. keine oder nur eine unvollständige Auskunft erteilen oder trotz Lösungsbegehrens den Datensatz nicht aus ihrer EDV löschen? Das Gesetz sieht hier den Gang zum Richter vor. Um aber möglicherweise kosten- und zeitintensive Rechtsstreitigkeiten vermeiden zu können, wurde im Bundeskanzleramt eine **Schlichtungsstelle-Datenschutz** eingerichtet, die sich - zumeist mit Erfolg - solcher Fälle annimmt. □

P-CAD Version 6

Wolfgang Zelinka, HTBLA-Hollabrunn

PCAD ist ein CAD-System, spezialisiert auf die Entwicklung von elektronischen Printplatten. Es besteht aus einer Vielzahl von Teilprogrammen (zusammen 35MB), jedes für eine bestimmte Aufgabe (z.B. Schaltplan-Eingabe, Symbole-Bearbeiten, usw.). Die Zusammenarbeit der einzelnen Programme und der zugehörigen Daten regelt der P-CAD Master Designer V6.06 (im weiteren Text nur mehr MD genannt) in der jeweils aktuellen Version.

Vom Konzept her besteht P-CAD aus einer Vielzahl von Programmen, jedes für eine spezielle Funktion (zB: Schaltplan eingeben, Netzliste erstellen, Ausdrucken, Plazieren usw.) und vielen unterschiedlichen, teils temporären Dateien. Aus diesem Grund war es bisher notwendig, die exakten Zusammenhänge zwischen den einzelnen Programmen und Dateien zu kennen, um den Weg zu der fertigen Printplatte erfolgreich gehen zu können. Es war notwendig, die einzelnen Programme in der richtigen Reihenfolge mit den richtigen Parametern und der richtigen Datei aufzurufen. Das Ergebnis mußte richtig interpretiert werden, um den nächsten Schritt durchführen zu können. Deswegen gab es einige mehr oder weniger brauchbare Zusätze wie z.B. die original p-cad shell bis zur Version 5 und andere menüorientierte Hilfen.

Mit der P-CAD Version 6 wurde Abschied genommen von dieser überholten Art der Programmaufrufe. P-CAD wurde vollkommen auf Mausbedienung und graphische Bedienoberfläche umgestellt. Das reibungslose Zusammenarbeiten der einzelnen Programme wird vom MD geregelt. Auch sind die Anforderungen an die Hardware gestiegen. Die Mindestanforderungen sind: ein 386SX-Prozessor, 3MB Hauptspeicher und eine VGA-Karte.

Die Bedienoberfläche P-CAD Master Designer V6 ist erstmalig wirklich brauchbar geworden und leicht bedienbar. Dennoch gibt es in der englischen Originalversion 6.x einige Haken, die den reibungslosen Ablauf behindern. Diese, und Lösungen dafür sind in den folgenden Punkten aufgelistet:

1. Die grafische Bedienoberfläche ist nur mit der Maus bedienbar! Wenn diese ausfällt, muß der Rechner neu gestartet werden (Booten).
- **Hotkeys werden in Version 7 kommen (Ende 1993)!**
2. Bei der erstmaligen Erstellung eines P-CAD-Projektes müssen für den einwandfreien Betrieb einige Dateien im aktuellen Arbeitsverzeichnis vorhanden sein, die sich normalerweise wo anders befinden. Ein Ausstieg ins DOS ist unvermeidlich und der Ablauf ist gestört.
- **Auch hier wird die Version 7 Verbesserungen bringen!**
3. Ein Punkt für Datensicherung ist nicht vorgesehen.
- **Hier habe ich PCAD erweitert.**
4. Der erweiterte Zeichensatz wird zwar theoretisch bereitgestellt, funktioniert aber nicht. Die deutschen Umlaute sind nicht verfügbar.
- **Die Zeichensatz-Dateien wurden von mir einer radikalen Veränderung unterzogen. Das Problem mit den Umlauten 'ä' und 'ü' ist gelöst und alle Formate sind zu fast 100% im Zeichensatz identisch geworden. Änderungen sind jetzt jederzeit leicht möglich. Ein Programm zur Restaurierung der Pointertabelle ist verfügbar.**
5. Für den Schulbetrieb (speziell im Netzwerk) und bei der Verwendung der Voll-Version und gleichzeitig der Evaluation-Version ist eine saubere Lösung hinsichtlich der Verzeichnisse und Pfade nicht möglich.
- **Hier habe ich PCAD erweitert.**
6. Eine Vielzahl von EDV-Begriffen ist bereits im Wortschatz des durchschnittlichen EDV-Anwenders (zB: EDV-User) selbstverständlich geworden. In P-CAD V6 gibt es aber einige englische Begriffe, die speziell dem Anfänger große Probleme bereiten.
- **Hier habe ich PCAD sanft übersetzt und erweitert.**

Dank der sehr offenen Architektur im Bereich des P-CAD MD konnten viele Anpassungen erfolgreich durchgeführt werden und Ergänzungen wurden in Form von BAT-Dateien und Hilfsprogrammen gemacht.

Die P-CAD Dateien sind integraler Bestandteil des P-CAD Master Designers und damit auch Copyright by P-CAD - CADAM Inc. Eine Offenlegung der Struktur und der Regeln sowie Unterstützung wird vom Hersteller nicht gegeben. Es besteht aber die mündliche Zusage, daß die Dateien geändert und weitergegeben werden dürfen. Somit kann und darf auch damit kein Geschäft gemacht werden. Die Weitergabe erfolgt ohne irgendwelche Garantie. Jede Gewährleistung ist ausgeschlossen!

Die Evaluation-Version

Der einzige Unterschied zur Voll-Version ist die Beschränkung auf 35 Bauteile in PCCAPS.EXE und PCCARDS.EXE, wobei das mit 35 nicht ganz exakt ist (scheint auch mit 36 noch zu gehen!). Es zählen selbstverständlich auch alle PADSTACKS, die geladen werden (in der *.SSF-Datei) als Bauteil! Überschreitet man die 35er-Grenze, so kommt beim Abspeichern eine Fehlermeldung: "Error Message 20389 does not exist". Laden, arbeiten und Plotdatei erstellen funktionieren aber scheinbar auch mit größerer Bauteilanzahl! Weitere Einschränkungen bestehen im Auto-Router. Es sind alle ROUTE*. * Dateien betroffen.

Die PCC-TGM Mailbox

In der PCC-TGM-Mailbox wurde eine Area PCAD installiert. Diese soll allen P-CAD Anwendern zum Austausch von Informationen dienen. Näheres dazu erhalten Sie vom Autor.

Die Arbeitsanleitung für P-CAD V6

Die Arbeitsunterlage beschreibt die Bedienung der 'eingedeutschten' und erweiterten Version des "P-CAD Master Designer V6.06" (abgekürzt als "MD 6.06"). Vorausgesetzt wird eine funktionierende P-CAD-Programm Installation auf einer lokalen Festplatte oder auf einem Netzwerk-Server. Installationshinweise und der Aufruf des MD werden im Anhang beschrieben. Ebenso die zusätzlichen Hilfsprogramme inkl. der BAT-Dateien. Es wird eine mögliche Art der Installation beschrieben, die der P-CAD-Originalinstallation entspricht und diese voraussetzt. Darüberhinaus wurden Funktionserweiterungen vorgenommen.

P-CAD ist so umfangreich, daß im 1. Teil nur die grundlegenden Arbeitsschritte gezeigt werden, um einfach und schnell zu einem befriedigenden Ergebnis zu kommen und den Anfänger nicht zu überfordern:

- Erstellen eines neuen Projekts und alle erforderlichen Konfigurationen durchführen
- Eingeben der Schaltung und Packen der Bauteile auf eine Printplatte
- Verlegen der Verbindungsleitungen (wire) auf der Printplatte
- Überprüfen der Übereinstimmung von Schaltung (SCH) und Printplatte (PCB) + DRC
- Ausdrucken des Schaltplans und der Printplattenzeichnungen.
- Datensicherung

Die Behebung von Fehlern und die Lösungsmöglichkeiten beim Auftreten von Problemen, sowie weiterführende Tips und Tricks werden im 2. Teil behandelt.

Wenn Sie Fragen, Wünsche oder Anregungen haben, dann wenden Sie sich bitte an die

Fa. Rekirsch, Tel. (0222) 25 36 26 - 31 DW Hr. Wiesauer

oder direkt an den Autor. □

High-Tech-Unterricht: *mehr als ein Kind und ein Computer*

Walter Riemer, EN/NA, TGM

Der nachfolgende Aufsatz erschien in der Zeitschrift "The Institute" 7/8-1993 des **IEEE**.

An der National Academy of Sciences in Washington D.C. wurde im Mai 1993 eine Konferenz "Neudefinition der Schulen: Technologie jetzt" abgehalten, in der gefordert wurde, endlich die moderne Informationstechnik in die Klassen zu bringen. Das Nachfolgende ist ein Kurzbericht über diese Konferenz.

Diese Technik geht weit über den Personal Computer hinaus in Richtung neuer Methoden der Zugänglichmachung und Verteilung von Information. Mit Computernetzen, beispielsweise, könnten Lernende Zugriff auf Datenbanken und elektronische Bibliotheken haben, mit anderen an anderen Orten kommunizieren und sogar Kontakt mit Experten an Universitäten und Forschungszentren herstellen. Satelliten-Kommunikation könnte die besten Lehrer in jedem Fachgebiet Studenten in fernliegenden Gebieten zur Verfügung stellen.

Wie geht es weiter?

Obwohl die Computertechnik für das Unterrichtswesen als vielversprechend bekannt ist, wurde darüber, Computer in die Schulen zu bringen, 10 Jahre lang nur gesprochen. Woher kommt diese Verzögerung?

Die horrenden Kosten sind nur ein Hindernis. Die Kosten, Computer in all U.S. Klassen zu bringen, werden zwischen 20 und 200 Milliarden US-\$ geschätzt.

Unzureichende Lehrerausbildung ist ein weiteres Problem. Während manche Schulen Computerräume und erfolgreiche Pilotprogramme haben, werden anderswo Computer- und Netzausrüstungen gar nicht ausgenutzt. Mit anderen Worten, den Lehrern wird nicht beigebracht, die Technik wirklich auszunutzen - "Lehrer lehren so, wie sie selbst gelehrt wurden".

Es wurde vorgeschlagen, Lehrer auf entsprechende Kurse zu schicken und sie auch dafür zu bezahlen.

Pädagogische Revolution

Unabhängig davon verändert die Informationstechnik die Unterrichts-Philosophie; wir befinden uns in einem frühen Stadium einer Revolution in der Art und Weise, wie Kinder lernen und denken.

In zunehmendem Maße wird Wissen eher durch Erfahrung als durch passives Aneignen von Tatsachenwissen zu vermitteln sein. So werden es zum Beispiel Physik- und Mathematik-Software-Pakete und Umgebungen virtueller Realität möglich machen, daß Studenten Hypothesen aufstellen und sie direkt testen.

Weiters wird die Konkurrenz zwischen Einzelnen zunehmend durch Gruppenarbeit und Kooperation ersetzt werden. Dies wird auch Einfluß darauf haben, wie Lehrer lehren. Sie werden nicht mehr nur Führer und Vortragende sein, sondern als Coaches und Mentoren fungieren.

Diese Entwicklungen werden die Anforderungen an die Software bestimmen. Nicht Drill-Programme werden gefragt sein, sondern Programme, die Dialog zwischen Lehrern und Schülern ermöglichen.

Zweischneidiges Schwert

Zumindest teilweise kommt die Dringlichkeit dieser Anliegen davon, daß die mangelhafte Effektivität des heutigen Sekundarunterrichts allgemein erkannt ist. Arbeitgeber wie auch Eltern sind unzufrieden. Die Technologie-Schere zwischen Schulen und der Wohn- und Arbeitsumgebung besteht und wächst.

Auch die Kosten-Schere besteht: Es besteht die Befürchtung, daß finanziell besser gestellte Schulen mehr in die Informationstechnik investieren werden und so ihren Schülern noch bessere Chancen auf dem Arbeitsmarkt verschaffen. Es sollte jedoch vermieden werden, die Chancengleichheit auf diesem Weg noch weiter zu vermindern, um die Spaltung der Gesellschaft (Stichwort: 2/3-Gesellschaft) nicht noch zu vergrößern, was die Gefahr der Zerstörung der Gemeinden und der Förderung von Gewalt mit sich brächte.

Andererseits werden positive Aspekte hervorgehoben, etwa daß körperlich behinderte Personen von den neuen Techniken entscheidend profitieren könnten. Nie sollte jedoch vergessen werden, daß die Technik nicht Selbstzweck, sondern nur Werkzeug sein kann. Demonstrierend wird es eine wesentliche Aufgabe sein, die geeigneten Lehrpläne herzustellen. Wenn auch die Technik schon zur Verfügung steht, die Aufgabe, sie vernünftig in den Klassen einzusetzen, ist zum Großteil noch unbewältigt. □

Ziele des elektrotechnischen Ingenieurwesens im 21. Jahrhundert

Walter Riemer, EN/NA, TGM

IEEE (Institute of Electrical and Electronics Engineers, kurz I-Triple-E) ist eine der führenden Vereinigungen von Elektroingenieuren. Von ihrem Sitz in den USA aus wird eine weitgespannte, auch international orientierte Tätigkeit ausgeführt.

In einer der zahlreichen Fachzeitschriften des "Institute" ("The Institute", July/August 1993) fanden wir im Rahmen eines

umfangreichen Aufsatzes über "The **IEEE** in the 21st century" eine Zusammenstellung, die auch sicher für unsere Leser von Interesse ist. Die folgenden sieben großen Herausforderungen für die Elektrotechnik wurden vom **IEEE**-Komitee "Neue Technologie-Richtungen" formuliert:

- Jede Person irgendwo in der Welt erreichbar zu machen, auf ihren Wunsch, zu jeder Zeit, mittels drahtunabhängiger Kommunikationseinrichtungen.
- schnellen weltweiten Zugriff zu Informationsquellen bereitzustellen mittels Techniken wie Computer-Datenbanken, Hochgeschwindigkeits-
- Kommunikationsverbindungen und flachen Displays und Schnittstellengeräten.
- Einer Person auf ihren Wunsch zu ermöglichen, an einem beliebigen Ort zu beliebiger Zeit mittels der Techniken der virtuellen Präsenz und der virtuellen Wirklichkeit präsent zu sein.

- Jeder Person den Zugriff zu einer ausreichenden Quelle sauberer, sicherer und erschwinglicher Energie zu ermöglichen.
- Intelligente Schnellstraßen und Transportsysteme zu schaffen, welche Personen jeden gewünschten Ortswechsel auf weltweiter Ebene ermöglichen.
- Das papierlose Büro in die Realität umzusetzen, indem Einrichtungen wie flache Displays und elektronische Schreibtische verwendet werden.
- Ein bargeldlose Gesellschaft zu verwirklichen, in der der Geldverkehr durch elektronische Einrichtungen wie die elektronische Geldbörse ersetzt wird. □

PAN - Public Access Network - oder

BTX mal 2: "ASCII-BTX" und die neuen X.29-Host-Zugänge

Johannes Sautner, MCCA

ASCII-BTX ist ein neuer Datendienst der Post, der sowohl für Informationsanbieter als auch für BTX-Teilnehmer einige interessante neue Möglichkeiten bietet. ASCII-BTX ist seit Anfang Oktober 1993 im Testbetrieb; die offizielle Dienst Einführung ist voraussichtlich im Jänner 1994.

Was bringt ASCII-BTX ?

Mit ASCII-BTX soll das BTX-System zum öffentlichen Datennetz (PAN = Public Access Network) heranwachsen. Da der Teilnehmerzugang zu ASCII-BTX mit einem Terminalprogramm mit **VT100-Emulation** (z.B. Telix, Procom, ...) erfolgt, ist endlich die Übertragung und Darstellung eines Standardbildschirmes mit **24 Zeilen zu je 80 Zeichen** möglich. Deshalb können bereits bestehende Datenbanken (Hosts), wie z.B. Bibliotheksinformationssysteme, ausländische BTX-Systeme, Banken, Internet (mit 1.3 Mill. Rechnern weltweit), Reuters, Compuserve, Mailboxen, ..., die über einen Datex-P-Anschluß verfügen, ohne Softwareänderung, an ASCII-BTX angeschlossen und damit allen BTX-Teilnehmern zugänglich gemacht werden.

Vergebührung: Zugangsentgelt und Zeittakt

Informationsanbieter im ASCII-BTX sind also ausschließlich Betreiber von Hosts bzw. Datenbanken. Sie können für die Benutzung ihrer Datenbank ein (beliebiges) Zeitentgelt verlangen (Zeittakt-Vergebührung). Die Verrechnung des Zeitentgeltes erfolgt über die österreichische Post, die die aufgelaufenen Entgelte einerseits den BTX-Teilnehmern in Rechnung stellt und andererseits dem Informationsanbieter gutschreibt.

Der Zugang zu ASCII-BTX selbst ist (derzeit) kostenlos. Es fallen nur die Telefongebühren für die Modemverbindung von derzeit öS 0.67/Minute an. Bei Verbindungsaufbau zu einem Host können aber noch folgende Entgelte dem Teilnehmer verrechnet werden:

- ein Zugangsentgelt, je Verbindungsaufbau zum Host
- ein Zeitentgelt; Einheit: je Verbindungsminute; Verrechnung aber nach
- tatsächlicher Nutzungszeit (auch Bruchteile von Minuten).

Der Teilnehmerzugang mit BTX- oder VT100-Terminal

Der Zugang zu ASCII-BTX ist, wegen einer etwaigen Zeittakt-Vergebührung, nur für identifizierte BTX-Teilnehmer möglich und erfolgt über dieselben Zugangsports wie für CEPT-BTX:

- über die Modem-Telefonzugänge
06612 - Ortstarif, 9.600-14.400bps
06611 - Ortstarif, 1.200-2.400 bps
- über den ISDN-Zugang
03289 - Ortstarif, Euro-ISDN, 19.200 bps
- sowie über alle Modem-Telefonzugänge mit 1200/75 bps

Für den Einstieg in das ASCII-BTX verwendet man ein VT100-Terminalprogramm (z.B. Telix, Procom, Unicom, Bitcom,...). Nach der Anwahl wird von der BTX-Zentrale der sog. "Terminal Facility Identifier" (TFI) des Terminalprogrammes abgefragt. Je nach Terminaltyp gelangt man schließlich in einen der zwei BTX-Sektoren:

- mit einem VT100-Terminal: in den ASCII-Sektor, bzw.
- mit einem BTX-Terminal (BTX-Softwaredecoder): in den CEPT-Sektor.

Der CEPT-Sektor mit Prestel- und CEPT-Hosts

Das ist das schon bekannte "alte" BTX-System mit vielen grafischen Möglichkeiten und 40 Zeichen/Zeile. Aus dem CEPT-Sektor kann man jederzeit (sofern man identifiziert ist) in den ASCII-Sektor wechseln. Der Wechsel in den ASCII-Sektor ist aber erst sinnvoll wenn man einen Multistandarddecoder (siehe weiter unten) verwendet: Eingabe: *OPT:ASCII#.

Über den CEPT-Sektor kann man einerseits auf die bereits vorhandenen "Externen Rechner" (Prestel-Host) als auch auf neuen, in der X.29-Technik realisierten CEPT-Hosts zugreifen. Ein interessantes Beispiel

für einen X.29-CEPT-Host findet man auf Seite *41025#. Die neuen BTX-Auslandszugänge werden ebenfalls als X.29-CEPT-Hosts realisiert.

Der Beenden einer Verbindung mit einem X.29-Host erfolgt durch dreimaliges Drücken der Escape-Taste und nicht mit *0#!

Was Sie eventuell auch noch ausprobieren könnten: Wechseln Sie in den ASCII-Sektor mit *OPT:ASCII#. Der ASCII-Systeminhalt verwendet nur Steuerzeichen, die auch mit einem BTX-Decoder problemlos darstellbar sind. Rufen Sie z.B. bei TESTIIG (siehe unten) (nur) die CEPT-Anwendungen auf. Mit OPT:CEPT kehren Sie dann wieder zurück in den CEPT-Sektor.

Der ASCII-Sektor mit ASCII-Hosts

Nach erfolgter Anwahl mit einem VT100-Terminal und der Identifikation wird der Systeminhalt des ASCII-Sektors angezeigt. Durch Menüauswahl kann man seine BTX-Gebühren abfragen oder auch eine Liste der angeschlossenen Hosts anzeigen. Der Verbindungsaufbau zu den Hosts erfolgt einfach durch Eingabe Hostnamens, z.B.:

COMPASS<Eingabetaste> (Börsenkurse) oder

PSK<Eingabetaste> (Telebanking) oder

TESTIIG<Eingabetaste> (Bibliothek, Internet).

Das Angebot im ASCII-Sektor ist derzeit noch klein. Aus dem ASCII-Sektor kann man auch jederzeit durch Eingabe von: OPT:CEPT<Eingabetaste> in den CEPT-Sektor wechseln. Der Wechsel ist aber nur mit einem Multistandarddecoder sinnvoll.

Die derzeit faszinierendste Anwendung ist sicher TESTIIG. Es wird eine Verbindung zu einem Grazer Universitätsrechner und zum System "Hypertext" hergestellt. Über dieses Informationssystem hat man den weltweiten Zugriff auf ausgewählte Datenbanken. Sogar der aktuelle Wellengang vor der Insel Hawaii läßt sich abfragen (Wetterwerte). TESTIIG ist derzeit nur für eine geschlossene Benutzergruppe zugänglich. Interessierte können sich jedoch in die GBG aufnehmen lassen. Bitte wenden Sie sich an **Dipl.Ing. Gerhard Greiner, BTX 913110861**. Die IIG behält sich jedoch das Recht vor, Teilnehmer ohne Angabe von Gründen nicht in die GBG aufzunehmen.

Der Beenden einer Verbindung mit einem X.29-Host erfolgt durch dreimaliges Drücken der Escape-Taste!

Multistandarddecoder

Für die optimale Nutzung von BTX wäre ein Multistandarddecoder erforderlich, der sowohl die Funktionen eines BTX-Decoders als auch eines VT100-Terminals vereint. Es wird daran gedacht in einer der nächsten SUXXESS-Versionen die Funktionen eines VT100-Terminals zu integrieren.

Was manchmal irrtümlich unter ASCII-BTX verstanden wurde

Unter dem bisher noch etwas unklaren Begriff "ASCII-BTX" wurde manchmal auch irrtümlich die Fähigkeit der BTX-Zentralen verstanden, BTX-Seiten des CEPT-Sektors zu dekodieren und auf einem VT100-Terminal darstellbar zu machen. Diese Fähigkeit wird aber bewußt nicht unterstützt, da keine befriedigende Umsetzung der CEPT-Seiten auf VT100-Codes möglich ist. Probleme würden sich insbesondere bei Telesoftware und den häufig verwendeten Seitennachfragesequenzen ergeben.

Weitere Informationen über X.29 und ASCII-BTX

- BTX-Hotline, Tel. 0660212 (Ortstarif)
- Dipl.Ing. Hofbauer, Fa. Infonova, Graz
- Fernmeldetechnisches Zentralamt Wien (Ing. Zachara)
- Post-Generaldirektion Wien (Mag. Lechner)
- für Applikationen: Dipl.Ing. Greiner, IIG, TU-Graz

Der **MCCA** (Anbietergruppe), BTX *2550# , steht Ihnen auch gerne für weitere Auskünfte, Anfragen und bei der Beschaffung von Unterlagen zur Verfügung.

Übersicht Prestel-ER, X.29 CEPT-Host und X.29 ASCII-Host

	Prestel-Externer Rechner	X.29 CEPT-Host	X.29 ASCII-Host
Terminaltyp	BTX-Decoder oder Multi-standarddecoder	BTX-Decoder oder Multi-standarddecoder	VT100-Terminalprogramm oder Multistandarddecoder
Bildschirmformat	24 * 40	24 * 40	24 * 80
Zeichensatz	genormter CEPT-Zeichensatz	genormter CEPT-Zeichensatz	genormter 8-bit-Zeichensatz; einige 8-bit-Zeichen auf IBM-PCs nicht darstellbar; mit Problemen bei der Darstellung von Umlauten ist zu rechnen
Grafik, Farben	zahlreiche grafische Möglichkeiten, bis zu 32 Farben gleichzeitig	zahlreiche grafische Möglichkeiten, bis zu 32 Farben gleichzeitig	keine Grafik, keine Farben
Vergebühungsmöglichkeiten	Seitenentgelt: damit realisierbar auch Zugangs- und Zeitgebühr für Hosts	Zugangs- und Zeitgebühr	Zugangs- und Zeitgebühr
Datenübertragung	CEPT-ER- und LL2-Protokolle: Host ↔ BTX-Zentrale ↔ BTX-Decoder	Völlig transparente (Direkt-) Verbindung: Host ↔ Terminal	Völlig transparente (Direkt-) Verbindung: Host ↔ Terminal
Sicherung der Datenübertragung	mit LL2-Sicherungsprotokoll	keine Datensicherung. Empfohlen wird die Verwendung eines Modems mit MNP4/5 bzw. V.42/V.42bis Datensicherung; Ev. LL2-Protokoll verwendbar	keine Datensicherung. Empfohlen wird die Verwendung eines Modems mit MNP4/5 bzw. V.42/V.42bis Datensicherung
Dateiübertragung (Telesoftware)	gesichert mit LL2-Protokoll, Dateiformat nach TSW51	Ev. LL2-Protokoll und Dateiformat nach TSW51	gesichert mit (privaten) End-to-End Protokollen (z.B. ZModem, Kermit)
Zulassung als BTX-Host	Software-Zulassungsprüfung	keine Zulassungsprüfung; nur Abstimmung der X3-Parameter mit dem FZA	keine Zulassungsprüfung; nur Abstimmung der X3-Parameter mit dem FZA
Kostenvergleich Prestel-ER, X.29: mittlere Host-Antwortseite	1	* 4	* 4
Kostenvergleich Prestel-ER, X.29: Dateiübertragung	1	* 1.06 (mit TSW51) * 1.12 (mit LL2 und TSW51)	* 0.8

□

AMMU, Arbeitsgruppe Moderner Mathematikunterricht

Peter Schüller, AMMU

Die letzten Jahre haben **markante Entwicklungen auf dem Gebiet der elektronischen Arbeitshilfen** für Mathematik mit sich gebracht. Die bisher kaum vorstellbare Leistungsfähigkeit der neuesten Taschenrechnergenerationen, das Auftauchen von ungemein leistungsfähigen und bereits relativ komfortabel handhabbaren Computer-Algebra-Systemen sowie die laufend (auch für Schüler) erschwinglicher werdende HARDWARE haben eine Entwicklung eingeleitet, an welcher der Mathematikunterricht mit Sicherheit nicht vorbeigehen kann und es wohl auch nicht darf.

Aus diesem Anlaß hat das Bundesministerium für Unterricht und Kunst für den Bereich der Höheren Technischen Lehranstalten eine **österreichweite Arbeitsgruppe** ins Leben gerufen, die die **Aufgabe** hat, sich mit den **Auswirkungen dieser Entwicklungen auf den Unterricht** auseinanderzusetzen.

Sie hat die Aufgabe, **didaktische Inhalte und Ziele auszuarbeiten**, die diesen neuen Medien gerecht werden können, und Überlegungen anzustellen, wie und in welchem Ausmaß diese Mittel im Mathematikunterricht der Zukunft sinnvoll eingebaut werden können. Ferner soll die Gruppe bei Kollegen, die mit diesen Medien noch nicht vertraut sind, das Interesse fördern und die nötigen **Einstiegshilfen bereitstellen**. Die Arbeitsgruppe ist weiters als **offenes Forum** gedacht, in welchem Kollegen, die in diesen Bereichen bereits aktiv sind, ihre schon geleisteten Arbeiten unter Wahrung der Autorenschaft der Allgemeinheit zugänglich machen können.

Dabei ist in keiner Weise beabsichtigt, die "klassische Mathematik" ad acta zu legen oder den bestehenden Unterricht völlig umzugestalten. Die Gruppe hat vielmehr das Ziel, unter gleichmäßiger Beachtung unterschiedlicher Mittel (PC, Taschenrechner, Videorecorder, Overheadprojektoren), **Ideen anbieten** und mögliche **Ergänzungen zum normalen Unterricht aufzeigen**. Dabei will sie auch Kollegen, die hier Neuland betreten, in das Arbeiten mit modernen Medien einführen und für Interessierte als Ansprechpartner zur Verfügung stehen. So sollte es möglich sein, Schwellenängsten abzubauen und ein möglichst hohes Maß an Hilfe anzubieten.

Hauptziel ist das **Erarbeiten inhaltlicher Konzepte**, die von einer bestimmten Hard- und Software unabhängig sind, also die völlige Orientierung an den inhaltlichen Möglichkeiten, welche die modernen Medien bieten. Bestimmte angesprochene Software-Produkte dienen lediglich der konkreten Darbietung der Musterausarbeitungen, damit diese unmittelbar nachvollzogen werden können.

Jedes Jahr werden **zwei Aussendungen folgenden Inhaltes** versandt: mindestens 6 konkrete Musterausarbeitungen zu Themenbereichen aus dem praktischen Unterricht (jeder Beitrag bedient sich der angesprochenen Medien), dazu ein begleitender didaktisch-methodischer Artikel als vertiefender Rahmen, weiters eine Diskette, die Daten und gegebenenfalls selbsterstellte Software enthält. Der Schwerpunkt der Beiträge liegt naturgemäß, wenn auch nicht ausschließlich, bei Lehrinhalten, welche die Höheren Technischen Lehranstalten betreffen. Im Laufe der nächsten Jahre soll so eine recht umfangreiche Sammlung von Arbeitsunterlagen entstehen.

Zur Zeit (Oktober 93) beziehen **309 Lehrer in Österreich** und **4 Institute im Ausland** die Aussendungen der Arbeitsgruppe. Unter den 309 inländischen Beziehern befinden sich auch etwa 30 Lehrer aus dem Bereich der AHS und 5 Hochschulinstute für Mathematik. **Bis dato** erfolgten **3 Aussendungen** in halbjährlichen Abständen (Oktober 92, Mai 93, Oktober 93) mit insgesamt **21 methodischen und 3 didaktischen Beiträgen**, dies ist ein Gesamtumfang von ca. 240 Seiten. Eine detailliertere Inhaltsübersicht finden Sie am Ende dieses Artikels. Die einzelnen Arbeiten beschäftigten sich mit konkreten Einsätzen der Medien **Computeralgebra, Tabellenkalkulation, selbstverfertigter Software, programmierbare Taschenrechner, Overheadfolien und Videorekorder**. Jede Aussendung war zusätzlich mit einer Diskette versehen, die notwendige Daten und Programme zu den Aussendungen enthielten.

Die **Mitglieder der Arbeitsgruppe**, allesamt Mathematiklehrer an Höheren Technischen Lehranstalten, sind zur Zeit:

Gruber Martin, Prof. Mag.	(HTL Klagenfurt)
Pichler Roland, Prof. Mag.	(HTL Kapfenberg)
Rohm Wilfried, Prof. Mag.	(HTL Saalfelden)
Schweitzer Christian, Prof. Mag.	(TGM - Wien)
Schüller Peter, Prof. Mag. Dr.	(HTL Mödling)
Dir. DI. Dr. Weissenböck Martin	(HTL Wien 4)

Betreuung durch das Bundesministerium:

DI. Dr. Christian Dominger

Kontaktadresse:

Peter SCHÜLLER
Arbeitsgruppe Moderner Mathematikunterricht
2351 Wr. Neudorf, Reisenbauerring 6 /1/35
Tel.: 022 36 / 22 602

Unter dieser Adresse können die Aussendungen der Arbeitsgruppe bestellt werden. Die **Aussendungen** sind **kostenlos**, lediglich für die **Diskette** ist jeweils nach Erhalt ein Kostenersatz in der Höhe von **S 40,-** zu entrichten. Es sind auch Nachbestellungen der bereits erschienenen Ausgaben möglich, in diesem Falle müssen jedoch die Kopier- und Versandkosten in der Höhe von S 100,- je nachbestellter Ausgabe verrechnet werden.

Ebenso besteht auch die Möglichkeit der Kontaktaufnahme für Personen, die eventuell im Rahmen einer Aussendung selbst einen Beitrag veröffentlichen wollen. Die Arbeitsgruppe ist an einer Vielzahl von guten Beiträgen verschiedener Autoren interessiert, um ein möglichst breitgestreutes Spektrum anbieten zu können. Die Auswahl der Beiträge, die veröffentlicht werden, erfolgt in den halbjährlich stattfindenden Arbeitssitzungen.

In der Folge eine Übersicht über die bereits erschienenen Ausgaben:

Aussendung 1 - Oktober 92

1) Kurvendiskussion von Biegelinien (Roland Pichler - DERIVE): Die elastische Linie eines halbeingespannten, durch eine Kraft F symmetrisch belasteten Trägers mit Auflager setzt sich aus zwei Kurven mit verschiedenen Funktionsgleichungen zusammen. Diese sollen auf technisch wichtige Bereiche untersucht werden. Außerdem sind die Steigungen und Krümmungen an der Stelle der einwirkenden Kraft anzugeben.

2) Näherungsweise Lösen von Gleichungen Iterationsverfahren (Peter Schüller - Taschenrechner): An Hand einer einfachen Aufgabe aus dem technischen Alltag werden aufbauend verschiedene Methoden zum Lösen algebraischer Gleichungen gegenübergestellt und ihre Vor- und Nachteile ausführlich besprochen. Am Ende sollte auch noch eine direkte Anwendung fertiger SOLVE-Programme von modernen Taschenrechnern stehen. In Rahmen der Entwicklung der einzelnen Methoden bietet sich weiters die Möglichkeit, eventuell vorhandene programmierbare Taschenrechner nach Maßgabe einzusetzen und ihre Handhabung an Hand dieser konkreten Aufgaben zu schulen.

3) Eine erste Einführung in den Begriff "Numerische Stabilität" (Wilfried Rohm - selbstverfaßte Programme): Das Thema "numerische Stabilität" kommt im herkömmlichen Mathematikunterricht zu kurz. Es werden im Hinblick auf später zu besprechende, kompliziertere numerische Algorithmen und als Schnittstelle zum EDV-Unterricht die Probleme und Lösungsmöglichkeiten an Hand einfacher Formeln und Iterationen aufgezeigt. Zentrales Thema ist das Problem der "numerischen Auslöschung".

4) Planungsmathematik - Lineare Optimierung (Martin Gruber - programmierbare Taschenrechner): Ergebnisse von Optimierungsaufgaben werden in der Technik oft als Entscheidungshilfen eingesetzt. Im Mathematikunterricht des ersten Jahrganges bietet die lineare Optimierung für den zweidimensionalen Fall Gelegenheit, wesentliche Teile des Lehrstoffes und den Einsatz des Taschenrechners in Anwendungsbeispiele einzubinden.

5) Die allgemeine Sinusfunktion (Klaus Scheiber - Multimedia): Im vorgestellten Unterrichtsbeispiel werden die mathematischen Zusammenhänge zwischen verschiedenen Koeffizienten und

dem jeweiligen Kurvenverlauf durch ein Computerprogramm visualisiert und demonstriert.

Folgende Möglichkeiten der Darbietung sind denkbar:

- Interaktiver Betrieb mittels Demonstrations-Rechner mit/ohne Einbeziehung von praktischen Übungen durch die Schüler (Abhängig von der Geräteausstattung!)
- Durcharbeiten der Videoaufzeichnung

Bei beiden Konzeptionen wird der Lehrstoff aktiv mit der Klasse unter Einbeziehung von vorbereiteten Overhead-Transparenten und Hardcopies der Bildschirmhalte erarbeitet bzw. wiederholt.

6) Vektoralgebra (Analyt. Geometrie mit DERIVE)

(Horst Schwarz): Einführung in die Darstellung und Berechnung von Vektoren, Vektorgleichungen, innerem und äußerem Produkt mit Hilfe des Softwarepaketes DERIVE. Anwendung auf einfache, nicht geübte Aufgabenstellungen in Einzelarbeit oder Kleingruppen am PC: Vergleich und Diskussion verschiedener Lösungswege, Folgerungen.

7) Gew. Lin. Differentialgleichung 1. Ordnung (Heinz Stegbauer - selbstverfaßtes Programm): Für jede der Differentialgleichungen $y' - y = 0$, $y' - y = x$, $y' - y = x^2$, $y' - y = e^x$ und $y' - y = \sin x$ wird ein Ausschnitt des Richtungsfeldes dargestellt, dem anschließend beliebig viele spezielle Lösungen überlagert werden können.

Es werden zwei Einsatzmöglichkeiten empfohlen:

- Nach erstmaliger Einführung in Differentialgleichungen zur Demonstration von Richtungsfeld und speziellen Lösungen.
- Bei Besprechung Gew.Lin. Differentialgleichungen 1.Ordnung zur Demonstration der Wirkung unterschiedlicher Störfunktionen.

8) Auswerten von Meßergebnissen aus Stichproben (Horst Schwarz - SuperCalc): Vorgabe: Einer Fertigung von Bolzen wird eine Stichprobe vom Umfang $n=120$ entnommen. Es wird der Durchmesser (in mm) gemessen. Die Werte sind in der Reihenfolge der Ermittlung angeführt.

Aufgabenstellung: a) Liegt eine normalverteilte Grundgesamtheit vor? b) Es sollen Schätzungen für μ und σ dieser Grundgesamtheit ermittelt werden.

- Gewinnung von Information über die Verteilung der Grundgesamtheit durch sinnvolle Klassierung der Stichprobenwerte;
- Überprüfung auf das Vorliegen einer Normalverteilung;
- Grafische Ermittlung von Schätzungen für μ und σ im Wahrscheinlichkeitsnetz.

9) Moderne Medien im Mathematikunterricht der Zukunft (Peter Schüller - Didaktikbeitrag)

Aussendung 2 - Mai 93

1) Taylorreihen und Computereinsatz - 1.Teil (Wilfried Rohm - DERIVE): Demonstration der Approximation einer Funktion durch ihre Taylorreihe - graphischer Vergleich von Taylorreihen bei Wahl unterschiedlicher Entwicklungspunkte - Berechnung nicht elementar integrierbarer Funktionen.

2) Simplexverfahren (Martin Gruber - Tabellenkalkulation): Ausgehend von den mit den erforderlichen Informationen beschriebenen linearen Optimierungsproblem wird ein entsprechendes mathematisches Modell, ein Lineares Programm, entwickelt. Das Simplexverfahren bietet dabei eine effiziente Berechnung der zu untersuchenden Basislösungen. Mit Hilfe eines Tabellenkalkulationsprogrammes ist es nun möglich, auch umfangreichere, praxisnahe Aufgabenstellungen für den Schüler transparent durchzurechnen.

3) Funktionen in Parameterdarstellung (Roland Pichler - DERIVE): Im ersten Beispiel werden bei der gewöhnlichen Zykloide Evolute, Bogenlänge und Flächeninhalt eines Ganges mit Hilfe der Infinitesimalrechnung berechnet. Beim zweiten Beispiel wird die Parameterdarstellung einer Parabel in gedrehter Lage angegeben sowie deren Vertikal- und Horizontaltangente bestimmt.

4) Fourier - Reihen (Gerald Kaiser - DERIVE): Es werden periodische Funktionen zunächst "händisch" in Fourierreihen entwickelt. Den Schülern wird hier klar, welcher enormer Rechenaufwand für komplizierte Aufgaben notwendig ist und daß eine grafische Darstellung der Reihe viel zu aufwendig ist. Mit Hilfe eines CAS wird nun nicht nur die Berechnung ver-

einfacht, sondern vor allem auch eine übersichtliche graphische Darstellung ermöglicht.

5) Programmierbare SHARP-Rechner im Unterricht (Erich Zott - BASIC-programmierbare Taschenrechner): An einem konkreten Beispiel wird der Einsatz BASIC-programmierbarer Taschenrechner demonstriert. Koll. Zott stellt dabei eine Programmsammlung vor, die er mit den Schülern entwickelt, regelmäßig benützt und im Laufe der 4 Jahrgänge ständig ausbaut.

6) Differentialgleichungen der Elektrotechnik (Horst Schwarz - DERIVE): Einführende Beschreibung des Typus der Aufgabenstellung und Erläuterung der Methode, den mathematischen Lösungsansatz zu finden. - Einsatz der DERIVE-Hilfsdateien ODE1.MTH und ODE2.MTH, um die Lösungen spezieller Aufgabenstellungen zu finden. - Herleitung der allgemeinen Lösungen für die freie gedämpfte elektrische Schwingung und die erzwungenen elektrische Schwingung mit Hilfe von DERIVE. Einsetzen spezieller Werte für die einzelnen Fälle; graphische Darstellung der Lösungsfunktionen. - Hinweis auf andere Lösungsmethoden, die nicht in diesem Beitrag behandelt werden.

7) Die Notebooks des BMfUK im Unterricht - Einsatz- und Organisationsformen (Peter Schüller - Didaktikbeitrag): Der Artikel setzt sich mit den Einsatzformen dieser Geräte im praktischen Unterricht auseinander. Es sollen Ideen aufgezeigt werden, welche Möglichkeiten die neuen Geräte bieten, um den Unterricht effizienter und interessanter zu gestalten.

Aussendung 3 - Oktober 93

1) Numerische Integration (Roland Pichler - selbsterstelltes Programm [Turbo-PASCAL]): Verschiedene Möglichkeiten der numerischen Integration werden gezeigt. Dabei werden die relativen Genauigkeiten der einzelnen Verfahren mit Hilfe des PC gegenübergestellt.

2) Iteratives Lösungsverfahren für eine Differentialgleichung (Friedrich Kliement - Tabellenkalkulation [Excel 4.0]): Eine Aufgabe, die bei exakter Behandlung auf eine Differentialgleichung führt, wird in Iterationsschritten näherungsweise gelöst. Bei jedem Schritt wird eine Variable, die sich kontinuierlich ändert, während eines Zeitintervalls konstant gehalten. Die exakten und iterativ gefundenen Werte werden in einer Tabelle und einem Diagramm verglichen.

3) Rekonstruktion einer Funktion durch eine Fourier-Reihe (Gerald Kaiser - DERIVE): Dazu muß das Signal in bestimmten Abständen abgetastet werden. Im folgendem Beispiel geschieht dies, indem man in bestimmten Abständen senkrechte Geraden zeichnet und mit dem Cursor die Koordinaten der Schnittpunkte ermittelt. Die Berechnung der Fourier-Koeffizienten erfolgt numerisch.

4) Vertiefung des Funktionsbegriffes mittels DERIVE (Peter Schüller - DERIVE): In diesem Beitrag finden Sie einen Weg, mittels eines CAS (ev. auch grafikfähiger Taschenrechner) das Vorstellungsvermögen der Schüler bezüglich mathematischer Funktionen auf eine sehr effiziente, gleichzeitig für Schüler aber auch sehr reizvolle Weise zu schulen.

5) Anwendung der komplexen Rechnung in der Wechselstromtechnik (Franz Thimary/Klaus Scheiber - DERIVE): Anhand zweier einfacher Schaltungsaufgaben aus der Wechselstromtechnik soll dem Schüler die Anwendung der komplexen Rechenmethode demonstriert werden.

6) Taylorreihen und Computereinsatz - Teil 2 (Wilfried Rohm - selbsterstellte Programme [Turbo-PASCAL]; programmierbare Taschenrechner; DERIVE): Zunächst wird am einfacheren Beispiel der Sinusfunktion unter Ausnutzung der Periodizität ein Algorithmus zur Berechnung von $\sin(x)$ für beliebige x mit Hilfe der Taylorreihe um $x=0$ hergeleitet. Danach wird der etwas schwierigere Fall der Exponentialfunktion untersucht, wobei dort das Problem der numerischen Stabilität wegen numerischer Auslöschung zur Entwicklung eines numerisch stabilen Algorithmus führt. Die Aufgabenstellungen sollten von Schülern selbstständig (nach Anleitung) gelöst werden können. Der Artikel kann auch als Grundlage für Schülerreferate verwendet werden.

Schluß auf der nächsten Seite, rechts unten

Vektoralgebra (Analyt. Geometrie mit DERIVE)

Horst Schwarz, HTBL Wien-10

DSK-372\AMMU.LZH

Mathematische Inhalte:

Vektoralgebra im 3-dimensionalen Raum

Anwendung:

Anwendung der Vektoralgebra (Betrag eines Vektors, Skalarprodukt, Vektorprodukt, vektorielle Darstellung einer Geraden im Raum) auf einfache Dreiecksberechnungen.

Kurzzusammenfassung:

1. Einführung in die Darstellung und Berechnung von Vektoren, Vekorgleichungen, innerem und äußerem Produkt mit Hilfe des Softwarepaketes DERIVE.
2. Anwendung auf einfache, nicht geübte Aufgabenstellungen in Einzelarbeit oder Kleingruppen am PC:
geg.: die Koordinaten der Eckpunkte eines Dreiecks im 3-dimensionalen Raum
ges.: die Länge der Seiten, die Größe der Innenwinkel, die Koordinaten des Höhenschnittpunktes, der Winkel zwischen der Flächennormalen auf die Dreiecksfläche und einem gegebenen Vektor.
3. Vergleich und Diskussion verschiedener Lösungswege, Folgerungen.

Lehrplanbezug:

Höhere Lehranstalt für Maschinenbau und Höhere Lehranstalt für Elektronik:
2. Jg. Vektoralgebra und Berechnung des Dreiecks
Höhere Lehranstalt für Elektrotechnik:
2. Jg.: Vektorrechnung und Berechnung des Dreiecks; 4. Jg.: Vektoralgebra

Anmerkung:

Im Lehrplan der HLA f. Elektrotechnik wird im II. Jahrgang unter Algebra "Vektorrechnung (Skalarprodukt)" angeführt, das Vektorprodukt also in die Vektoralgebra des vierten Jahrgangs verschoben. Dies findet auch in den approbierten Lehrbüchern seinen Niederschlag. Für eine sinnvolle Umsetzung mathemat. Wissens in den theoret. Fachgegenständen ist jedoch dieser Zeitpunkt eindeutig zu spät.

Zeitaufwand:

zweimal 2 aufeinanderfolgende Unterrichtseinheiten

Mediales Umfeld:

verwendete Medien: ein PC mit LCD-Overhead-Display und ein Overheadprojektor zur Diskussion der Ergebnisse, je Arbeitsgruppe ein PC und eine formatierte Arbeitsdiskette

verwendete Software: DERIVE Vers. 2.51 (Generallizenz)

Dateinamen auf der Diskette zu diesem Beitrag:

SW_VEALA.MTH (Demofile der Vektorrechnung mit DERIVE),

SW_VEALB.MTH (Dokumentation des Lösungsweges in DERIVE),

Anmerkungen:

Mathematisches Wissen, das ohne tieferes Verständnis nur für Schularbeiten und Prüfungen **gelernt** worden ist, wird relativ leicht **vergessen**. Das Anwenden mathematischer Kenntnisse beim Lösen **nicht geübter** Aufgabenstellungen trägt wesentlich zu einem tieferen Verständnis bei. Durch den Einsatz von Computer-Algebra-Systemen (CAS) wie z. B. DERIVE können auch durchschnittlich motivierte Schüler in vertretbarer Zeit durchaus brauchbare Ergebnisse erzielen, die darüber hinaus mühelos dokumentiert und für die weitere Auswertung auf Diskette gespeichert werden können. Dadurch wird das subjektive Erfolgserlebnis gesteigert und eine objektivere Bewertung möglich. Dort wo es jedoch darum geht, mathematische Standardroutinen für die Anwendung im theoretischen Fachunterricht bereitzustellen, ist DERIVE weniger geeignet. Hier ist der programmierbare Taschenrechner und in zunehmendem Maß der Einsatz von Tabellenkalkulationsprogrammen sinnvoller.

1. Vektordarstellung und -berechnungen mit DERIVE:

Vektoren werden in DERIVE als geordnete Liste von Elementen in eckigen Klammern dargestellt:

$oa := [ax, ay, az], [-3, 2, -1]$.

Das Skalarprodukt wird durch den (Infix-)Operator "." berechnet:

$a.b, [1, 2, 3].[3, 4, 5]$,

das Vektorprodukt muß mit der Funktion CROSS berechnet werden: $CROSS(a, b)$.

Für nähere Informationen wird auf das DERIVE-Handbuch [1], Seite 133 ff verwiesen.

Um nicht deklarierte Variable (so wie oben $oa := [ax, ay, az]$) mit mehr als einem Buchstaben Länge verwenden zu können, muß mit **OPTION/INPUT/WORD** auf den WORD-Eingabemodus umgeschaltet werden [1], Seite 42

Beispiel siehe nächste Seite.

2. Anwendung der Vektoralgebra auf einfache Dreiecksberechnungen im 3-dimensionalen Raum:

Unter der Voraussetzung, daß die Schüler

- mit dem Softwarepakete DERIVE grundlegend umgehen können,
- mit Dreiecksberechnungen in der Ebene vertraut sind und
- die Grundbegriffe der Vektorrechnung und deren Anwendung in DERIVE, wie unter Punkt 1 erläutert, beherrschen,

können eine odere mehrere Aufgaben folgenden Typs gestellt werden, die in Kleingruppen am PC mit Hilfe des Softwarepaketes DERIVE gelöst werden sollen:

geg. sind die Eckpunkte eines Dreiecks

$A(4/-3/-3), B(2/4/4), C(-7/-1/1)$

ges. sind

- a) die Länge der Seiten und
- b) die Größe der Innenwinkel
- c) die Koordinaten des Höhenschnittpunktes und
- d) der Winkel, den der Vektor $v=[0,-3,2]$ mit der Normalen auf die Dreiecksebene einschließt.

Vorstellung AMMU (Schluß)

7) Programmierbare SHARP-Rechner im Unterricht ("Scharfe Programme") - Teil 2 (Erich Zott - BASIC-programmierbare Taschenrechner): Die 1. und 2.Ableitung wird durch numerisches Differenzieren gebildet; Anwendung bei der Newtonschen Nullstellensuche und bei der Krümmung. Erweiterung der BASIC-Programmsammlung für den 3.Jahrgang.

8) Black Boxes im Mathematikunterricht (Jochen Maaß/Wolfgang Schlögelmann - Didaktikbeitrag): Der Didaktikbeitrag kommt diesmal vom mathematischen Institut der Universität Linz und befaßt sich sehr ausführlich mit dem Thema der Black-Box im Mathematikunterricht. Sie finden darin auch einen konkreten Vorschlag zur Aufarbeitung dieses Themas im realen Unterricht. Die Autoren haben dazu ein Spiel gewählt, das den Schülern die Thematik näherbringen soll. □

Beispiel einer einführenden Demonstration der Vektorrechnung mit DERIVE:

Protokoll der Berechnung¹

```

1: v:=[vx,vy,vz]
2: w:=[wx,wy,wz]
3: v:=[ 1, 1, 1]
4: |v|
5: 1.73205
6: w:=[ 1, 0, 0]
7: v . w
8: 1
9: ACOS[  $\frac{v \cdot w}{|v| \cdot |w|}$  ]  $\frac{180}{\pi}$ 
10: 54.7356
11: CROSS(v,w)
12: [ 0, 1, -1]

```

erreicht über Menüpunkt

Kommentar

Author
 Author
 Manage/Substitute #1
 Author ABS(v)
 approX #4
 Manage/Substitute #2
 Author
 Simplify #7
 Author
 approX #9
 Author
 Simplify #11

Die vektorielle Darstellung einer Geraden im Raum erfordert beim Einsatz von DERIVE kaum zusätzlichen Erklärungsaufwand, da die Schreibweise analog zu der in den approbierten Lehrbüchern ist. Das Beispiel 2 aus [2] Seite 289 kann mit DERIVE so gelöst werden:

```

13: "geg.: Gerade g durch P(1/4/-5), r=[2,-1,8] liegt auf g"
14: "ges.: Parameterdarstellung der Geraden"
15: p:=[ 1, 4, -5]
16: r:=[ 2, -1, 8]
17: x := p + μ r
18: [2 μ +1, 4 - μ, 8 μ - 5]

```

Author Ortsvektor OP
 Author
 Author
 Expand #17 (alle Variable)
 liefert die gesuchte
 Komponenten-Darstellung

Didaktische und methodische Hinweise:

Das Lösen nicht geübter Aufgaben mit einem im bisherigen Mathematikunterricht neuem Medium ist meist mit Ratlosigkeit und Unruhe verbunden. Man sollte diese an und für sich "positiven emotionalen Kräfte" durch geeignete Maßnahmen als "Antriebsenergie" zur Lösung der Aufgabe nutzen, etwa dadurch, daß der/die unterrichtende Lehrer(in)

- je nach Bedarf der Klasse Hinweise für die ersten Lösungsschritte gibt (z.B. "ein Dreieck kann als geschlossenes Vektorpolygon gesehen werden") und kurz die sich daraus ergebenden Lösungsvorschläge in einer Art "brain-storming" sammelt,
- für vorhersehbare Probleme (z.B. der Lösung eines linearen Gleichungssystems von 3 Gleichungen mit 2 Unbekannten mit DERIVE) bereits schriftliche Hinweise vorbereitet, die er/sie dann der jeweiligen Gruppe geben kann und dadurch wieder für die "Betreuung aller Gruppen" frei wird.
- Jene Kolleg(inn)en, die mit der Bedienung der "Hardware" nicht so vertraut sind, sollen dafür sorgen, daß zumindest zu Beginn ein(e) EDV-Versierte(r) "zur Hand ist".

Von den verschiedenen Möglichkeiten, die Geradengleichungen der Dreieckshöhen aufzustellen werden im anschließend aufgelisteten Dokumentationsfile zwei behandelt:

a) Normale auf die Dreiecksseite über das Vektorprodukt:

Hier wird die Eigenschaft von $a \times (a \times c)$ genutzt, in der Dreiecksebene zu liegen und normal auf die Seite a zu stehen (#49), gleichzeitig erhält man den Vektor der Flächennormalen für die letzte Teilaufgabe (#74).

b) Normale auf die Dreiecksseite ohne Vektorprodukt:

Hier wird der Vektor vom Eckpunkt A zum gegenüberliegenden Fußpunkt der Höhe auf die Seite a über trigonometrische Beziehungen der ebenen Geometrie

$$\vec{n}_a = \vec{c} + \vec{a} \frac{|\vec{c}| \cos(\beta)}{|\vec{a}|}$$

aufgestellt (#93):

Es muß jedoch für die letzte Teilaufgabe der Normalvektor auf die Dreiecksebene (über ein Vektorprodukt) berechnet werden.

Die Vorteile des Lösungsweges über das Vektorprodukt werden also bei vollständiger Lösung der gestellten Aufgabe unmittelbar einsichtig, eine ausführliche "mathematische Aufbereitung" ist aber insbesondere auch im Hinblick auf weitere Aufgabenstellungen wie z. B. die Berechnung des Normalabstandes eines Punktes von der (Dreiecks-)Ebene nötig (Punkt 3):

¹ Die folgenden berechnungen finden Sie auf der Diskette unter dem dateinamen SW_VEALA.MTH (DERIVE-Format)

Lösung eines linearen Gleichungssystems von 3 Gleichungen mit 2 Unbekannten mit DERIVE:

Das Gleichsetzen der beiden Geradengleichungen der Höhe auf a (#51 bzw. #52) und der Höhe auf c (#54 bzw. #55) ergibt folgendes lineares Gleichungssystem (#60)

$$\begin{aligned} 4 + 0.525925 \mu &= -7 - 0.970427 \tau \\ -3 - 0.565461 \mu &= -1 - 0.238213 \tau \\ -3 - 0.635339 \mu &= 1 - 0.0390514 \tau \end{aligned}$$

Der **mathematische Teil des Problems** sollte, zumindest nach kurzer Auffrischung der Kenntnisse über lineare Gleichungssysteme (1. Jahrgang), von den Schülern selbst gelöst werden können, denn aufgrund des geometrischen Sachverhaltes hat das Gleichungssystem genau eine Lösung und aus der Komponentendarstellung (#60) ist ersichtlich, daß die ersten beiden Gleichungen keine proportionalen Koeffizienten für μ bzw. τ haben, das Gleichungssystem aus den beiden Gleichungen genau eine Lösung hat. Ob diese Lösung auch die dritte Gleichung erfüllt kann mit DERIVE leicht überprüft werden:

Nachdem durch #64 und #65 den Variablen μ und τ die Lösungswerte zugewiesen worden sind, liefert der Befehl `approX #59` den Vektor
 $[0.440990 = 0.440986, 0.826556 = 0.826560, 1.29943 = 1.29943]$,
 der im Rahmen der für `approX` eingestellten Genauigkeit gleiche Ergebnisse für die linken und rechten Seiten der Gleichungen liefert.

Es bleibt daher nur noch das Problem, aus der Komponentendarstellung des Gleichungssystems (#60) in DERIVE - ohne lästiges Abtippen von Zahlenwerten - eine mit dem Befehl `soLve` lösbare Form des aus zwei der drei Gleichungen gebildeten linearen Gleichungssystems zu erstellen.

Das leistet die Funktion **ELEMENT(v,n)**, die die **n-te Komponente des Vektors v** liefert. Näheres kann in [1], Seite 136, "8.3 Das Zugreifen auf Elemente" nachgelesen werden. Der Schnittpunkt der beiden Geradengleichungen #51 und #56 wird durch Einsetzen der Lösungswerte des Gleichungssystems #62 in diese Geradengleichungen gefunden.

Kommentierte Auflistung des Lösungsganges der gestellten Aufgabe mit Hilfe von DERIVE:

Lassen Sie sich von der Länge der Auflistung der Anweisungen, wie sie bei Einsatz von DERIVE am Bildschirm aufscheinen, nicht abschrecken, weil, um den Rechengang transparent und leichter lesbar erscheinen zu lassen, zusätzlich Kommentarzeilen ("...") und Befehle zur Berechnung von Zwischenwerten eingefügt worden sind, die zur Lösung der Aufgabe nicht unbedingt erforderlich wären.

Protokoll der Berechnung ²

erreicht über Menüpunkt
Kommentar

```

1: "Vektorielle Dreiecksberechnungen"
2: "geg.:A(4/-3/-3),B(2/4/4),C(-7/-1/1)"
3: "ges.:a) Länge der Dreiecksseiten und b) die Innenwinkel"
4: "      c) die Koordinaten des Höhenschnittpunktes"
5: "      d) der Winkel, den der Vektor v=[0,-3,2] mit der"
6: "          Normalen auf die Dreiecksebene einschließt"
7: "Nützliche Funktionsdefinitionen:"
8: "Winkel zwischen 2 Vektoren v1 und v2:"
9: WINKEL(v1,v2):=ACOS[  $\frac{v1.v2}{|v1|. |v2|}$  ]  $\frac{180}{\pi}$            Author
10: "Normalvektor der Länge 1 auf die durch die Vektoren v1 und v2"
11: "aufgespannte Ebene"
12: NVEKTOR(v1,v2):= $\frac{CROSS(v1,v2)}{|CROSS(v1,v2)|}$            Author
13: "Eingabe der Ortsvektoren:"
14: "Zu beachten: OPTIONS/INPUT/WORD setzen !"
15: "Bei Wahl des Vorgabewertes Insensitive erscheinen Funktionen"
16: "in Großbuchstaben und Variablen in Kleinbuchstaben"
17: oa:=[ax,ay,az]           Author
18: ob:=[bx,by,bz]           Author
19: oc:=[cx,cy,cz]           Author
20: "Spezielle Werte mit MANAGE/SUBSTITUTE einsetzen!"
21: oa:=[4, -3, -3]           Manage/Substitute #17
                                Ersetzen von oa mit
                                Eingabetaste überspringen
22: ob:=[2, 4, 4]           Manage/Substitute #18
                                ob überspringen
23: oc:=[-7, -1, 1]           Manage/Substitute #19
                                oc überspringen
24: "Dreiecksseiten durch Vektoren darstellen"
25: a:=oc - ob           Author
26: [-9, -5, -3]           Simplify #25
27: b:=oa - oc           Author
28: [11, -2, -4]           Simplify #27
29: c:=ob - oa           Author
30: [-2, 7, 7]           Simplify #29
31: "LÖSUNG a):Länge der Seiten durch ABS-Funktion"
32: la:= |a|           Author ABS(a)

```

² Die folgenden Berechnungen finden Sie auf der Diskette unter dem Dateinamen SW_VEALB.MTH


```

33: 10.7238                                approX #32
34: |b:= |b|                                Author ABS(b)
35: 11.8743                                approX #34
36: 10.0995                                Author lc:=ABS(c)
                                STRG+EINGABE-Taste
                                liefert sofort das Ergebnis
37: "LÖSUNG b):Innenwinkel des Dreiecks mit vordef. Funktion WINKEL"
38:  $\alpha$ := WINKEL(c, -b)                    Author
39: 57.7464                                approX #38
40:  $\beta$ := WINKEL(a, -c)                    Author
41: 69.4600                                approX #40
42: "Γ bzw. gamma ist schon für die Gamma-Funktion in Verwendung, daher "
43: w_gamma:= WINKEL(-a, b)                Author
44: 52.7935                                approX #43
45:  $\alpha + \beta + w\_gamma$                 Author
46: 180                                    Simplify #45
47: "Lösung c): Höhenschnittpunkt:"
48: "Schritt 1:Geradengleichung für ha: x1:=oa+ $\mu$ .na"
49: na:= NVEKTOR(a, NVEKTOR(a, c))          Author
50: [0.525925, -0.565461, -0.635339]        approX #49
51: x1:= oa +  $\mu$  . na                      Author
52: [4,-3,-3]+ $\mu$ . [0.525925,-0.565461,-0.635339] approX #51
53: "Schritt 2:Geradengleichung für hc: x2:=oc+ $\tau$ .nc"
54: nc:= NVEKTOR(c, NVEKTOR(a,c))          Author
55: [-0.970427, -0.238213, -0.0390514]    approX #54
56: x2:= oc +  $\tau$  . nc                      Author
57: [-7,-1,1]+ $\tau$ . [-0.970427,-0.238213,-0.0390514] approX #56
58: "Nun die beiden Geraden schneiden ..."
59: x1 = x2                                Author
60: [4,-3,-3]+ $\mu$ . [0.525925,-0.565461,-0.635339]~ approX #59
    ~[-7,-1,1]+ $\tau$ . [-0.970427,-0.238213,-0.0390514]
        { ~ = Fortsetzung nur hier in der nächsten Zeile, Befehl wird in DERIVE
          fortlaufend geschrieben, ~ gehört nicht zum Befehl ! }
61: "Lösung des Gleichungssystems aus den beiden ersten Komponenten/SOLVE"
62: [ELEMENT(oa,1)+ $\mu$  ELEMENT(na,1)~
    ~ELEMENT(oc,1)+ $\tau$  ELEMENT(nc,1),~
    ~ELEMENT(oa,2)+ $\mu$ .ELEMENT(na,2)~
    ... ~ELEMENT(oc,2)+ $\tau$ .ELEMENT(nc,2)]
63: [ $\mu$  = -6.76714,  $\tau$  = -7.66773]          solve #62
64:  $\mu$ := -6.76714                          Author kopiert aus #63
                                                wie in [1] Seite 27
65:  $\tau$ := -7.66773                          Author kopiert aus #63
66: "Berechnen des Ortsvektors oh zum Höhenschnittpunkt"
67: oh:=oa +  $\mu$  na                          Author
68: [0.440990, 0.826556, 1.29943]          approX #67
69: "Probe..."
70: oh:=oc +  $\tau$  nc                          Author
71: [0.440990, 0.826556, 1.29943]          approX #70
72: "Lösung d):Winkel zwischen Vektor v und Normaler auf Dreiecksebene"
73: v:=[vx, vy,vz]                          Author
74:  $\delta$ := WINKEL(v, NVEKTOR(a, b))          Author
75: "Spezielle Werte für v mit MANAGE/SUBSTITUTE einsetzen"
76: v:= [0, -3, 2]                          Manage/Substitute #73
                                                Ersetzen von v mit
                                                Eingabetaste überspringen
77: 15.1293                                approX #74
78: "Probe.."
79:  $\delta$ := WINKEL(v, NVEKTOR(a, c))          Author
80: 164.870                                approX #79
81: "Fehler ?"                            Richtungsabhängigkeit des Vektorproduktes !
82:  $\delta$  := WINKEL(v, NVEKTOR(a, -c))        Author
83: 15.1293                                approX #82

```

```

84: "Variante der Berechnung des Höhengschnittpunktes ohne Verwendung"
85: "des Vektorproduktes - die Berechnung verläuft analog zur"
86: "vektoriellen Berechnung in der Ebene:"
87: "Die Berechnung der Seiten und der Innenwinkel verläuft so wie oben"
88: "der Klammerausdruck (1/c COS(β _ rad)/1/a) ist die Länge der Strecke vom"
89: "Eckpunkt A bis zum Fußpunkt der Höhe auf a."
90: "Der Vektor na steht daher normal auf die Seite a."
91: β_rad:= $\frac{\beta\pi}{180}$  Author
92: 1.2123 approX #91
93: na:=c + a  $\frac{1/c \cos(\beta\_rad)}{1/a}$  Author
94: [-4.97391, 5.34782, 6.00869] approX #93
95: nc:=-a - c  $\frac{1/a \cos(\beta\_rad)}{1/c}$  author
96: [9.74509, 2.39215, 0.392156] approX #95
97: "Der weitere Rechengang verläuft wie oben - es wird nur"
98: "μ durch ε und τ durch σ ersetzt:"
99: x1:= oa + ε na Author
100: x2:= oc + σ nc Author
101: [ELEMENT(oa,1)+ε ELEMENT(na,1)=~
~ELEMENT(oc,1)+σ ELEMENT(nc,1),~
~ELEMENT(oa,2)+ε.ELEMENT(na,2)=~
~ELEMENT(oc,2)+σ.ELEMENT(nc,2)] Author
102: [ε = 0.715535, σ = 0.763562] solve #101
103: ε := 0.715535 Author kopiert aus #102
104: σ := 0.763562 Author kopiert aus #102
105: oh:= oa + ε na Author
106: [0.440990, 0.826556, 1.29943] approX #105
107: oh:= oc + σ nc Author
108: [0.440986, 0.826560, 1.29943] approX #107

```

3.Vergleich und Diskussion verschiedener Lösungswege, Folgerungen:

Der abschließende Vergleich und die Diskussion der verschiedenen Lösungswege soll zwei Ziele verfolgen:

Die einzelnen Arbeitsgruppen sollen das Gefühl erhalten, daß ihre Mühe gewürdigt und ihre Arbeit bewertet wird - muß nicht Benotung sein.

Die Lösungswege und -versuche der einzelnen Gruppen müssen "mathemat. aufgearbeitet" werden, d.h. offene Fragen klären, häufiger auftretende Fehler korrigieren aber auch durch weitere gezielte Fragen aus dem Gelernten Neues erarbeiten - hier die Vektordarstellung der Hesse-Normalform einer Ebene und ihre Anwendung.

Didaktische und methodische Anmerkungen:

Aus zeitlichen Gründen und auch um die Aufmerksamkeit der Schüler zu erhalten, ist davon abzuraten, daß alle Lösungswege der einzelnen Gruppen detailliert besprochen werden. Um aber die zuvor angeführten Ziele trotzdem anzustreben, empfiehlt sich folgende Vorgangsweise: Die Schüler kopieren mit dem Befehl **Transfer/Save/Derive** ihren Lösungsgang auf ihre Arbeitsdisketten. Der/die unterrichtende Lehrer(in) kann dann (nötigenfalls auch außerhalb des Unterrichtes) die einzelnen Lösungswege sichten, kopieren und dann im folgenden Unterricht zu den einzelnen Gruppenergebnissen Stellung nehmen, diese also verbal bewerten, bzw. die interessantesten Lösungswege von einem Gruppenmitglied der gesamten Klasse erläutern lassen. **Als Gerätekonfiguration reicht hier ein PC mit LCD-Overhead-Display und ein Overhead-Projektor aus.**

Hesse-Normalform und deren Anwendung:

Nach der Besprechung ist den Schülern bekannt, daß die DERIVE-Funktion **NVEKTOR(v₁,v₂)** (wobei v₁ und v₂ zwei nicht parallele

Vektoren in der Dreiecksebene sind) einen Vektor $\vec{n}=[n_x, n_y, n_z]$ der Länge 1 liefert, der normal auf die Dreiecksebene und damit auf alle darin liegenden Vektoren steht, und daß aufgrund der Eigenschaften

des Skalarproduktes gilt::

$$\vec{n} \cdot \vec{a} = 0, \quad \vec{n} \cdot \vec{b} = 0, \quad \vec{n} \cdot (\vec{a} + \vec{b}) = 0, \quad \vec{n} \cdot (\vec{b} + 2\vec{c}) = 0 \dots$$

Dies kann in DERIVE mit den speziellen Daten der Aufgabenstellung sofort überprüft werden. Die Richtungsabhängigkeit des Normalvektors ist den Schülern ebenfalls aus den Berechnungen zur Teilaufgabe d) (#74, #79, #82) bekannt. Darauf aufbauend kann verallgemeinert

werden: $\vec{n} \cdot (\vec{x} - \vec{x}_0) = 0$, mit $\vec{x} = \vec{OP}$, $\vec{x}_0 = \vec{OP}_0$

wobei $P_0(x_0, y_0, z_0)$ ein fester Punkt und $P(x, y, z)$ ein beliebiger Punkt in der Ebene ist. Durch Umformung erhält man die

Gleichung $\vec{n} \cdot \vec{x} = \vec{n} \cdot \vec{x}_0$, die verbal etwa so interpretiert werden

kann: "Projiziert man den Ortsvektor \vec{OP} vom Ursprung zu einem beliebigen Punkt der Ebene auf die Richtung der Flächennormalen, so erhält man stets den gleichen Betrag, nämlich den Normalabstand des Ursprunges von der Ebene".

Einsetzen der Werte eines beliebigen Punktes liefert einen Wert $\neq 0$.

Eine Betrachtung des Ausdrucks $\vec{n} \cdot \vec{x} - \vec{n} \cdot \vec{x}_0$ führt zur Interpretation des Wertes als Normalabstand des Punktes von der Ebene. Ersetzt man die Funktion NVEKTOR durch ihre auf dem Vektorprodukt basierende Definition (#12), so erhält man eine von DERIVE-Notationen unabhängige Form der Gleichung der Ebene:

$$(\vec{v}_1 \times \vec{v}_2) \cdot (\vec{x} - \vec{x}_0) = 0, \text{ mit } \vec{x}_0 = \vec{OP}_0, \vec{x} = \vec{OP}$$

wobei \vec{v}_1 und \vec{v}_2 "die Ebene aufspannen".

Protokoll der Berechnung

erreicht über Menüpunkt
Kommentar

```

109: "n=[nx,ny,nz] ...Normalvektor auf die Dreiecksebene"
110: n:=NVEKTOR( a, b)
111: "OPTION/PRECISION/EXACT einstellen!"
112:  $\left[ \frac{7\sqrt{10286}}{5143}, -\frac{69\sqrt{10286}}{10286}, \frac{73\sqrt{10286}}{10286} \right]$ 
113: n . a
114: 0
115: "approX liefert Ergebnisse mit beschränkter Genauigkeit"
116: [0.138039, -0.680342, 0.719780]
117: -1.33854 10-8
118: n . b
119: 0
120: n . (a + b)
121: 0
122: n . (a + 2 c)
123: 0
124: "Hesse-Normalform der Dreiecksebene"
125: x:=[xx, xy, xz]
126: x0:= oa
127: n . (x - x0)
128: "Überprüfen ob Koordinaten von B die Gleichung erfüllen"
129: x:= ob
130: 0
131: "ebenso C..."
132: x:= oc
133: 0
134: "Normalabstand des Ursprungs..."
135: n . x0
136: 0.433839
137: "Einsetzen der Koordinaten eines beliebigen Punktes"
138: x:=[2, -3, 5]
139: 5.48215

```

4..Literaturhinweise:

- [1] **Handbuch DERIVE**, Version 2, Soft Warehouse, Inc. Honolulu, Hawaii, Deutsche Übersetzung von Soft Warehouse GmbH Europe, Schloß Hagenberg, Österreich, Vierte Ausgabe, dritte Auflage: November 1990
- [2] **J. Schärf, MATHEMATIK, Band 4**, vierte, neubearb. Aufl., Oldenbourg Verlag Wien 1989
- [3] **Schalk, Mathematik 4**, 2., erweiterte Auflage, Reniets Verlag Wien 1992
- [4] **J. Schärf, MATHEMATIK, Band 2**, siebente, verbesserte. Aufl., Oldenbourg Verlag Wien 1987
- [5] **Schalk, Mathematik 2**, Reniets Verlag Wien 1987 □

Bretter-Recycling

Friedrich Pöschko, TU-Wien

Wer kennt sie nicht, die "Bretter" (gemeint sind PC-Motherboards, aber auch Peripherie-Steckkarten) aus den Wühlkisten diverser Händler: Manche stammen aus Konkurs-Aufkäufen und sind vielleicht sogar noch ganz intakt, manche haben vielleicht die Endkontrolle nicht geschafft, wieder andere wurden zur Reparatur eingeschickt und aus Zeit- oder Kostengründen nie repariert, sondern den Kunden gegen neue ausgetauscht und landeten dann einfach in der "Kiste".

Lohnt sich der Kauf? Besteht überhaupt eine Chance, so ein Brett "wieder hinzukriegen"? Oder ist es nur wegen der darauf befindlichen Bauteile interessant?

Der folgende Artikel zeigt einige Möglichkeiten auf, welche bei der Fehlersuche in PC-Motherboards angewendet werden können. Sein Ziel ist es, den in eine Reparatur gesteckten Zeitaufwand zu verringern und die Chance auf das Gelingen einer Reparatur zu steigern.

Folgende **Vorgangsweise** empfiehlt sich: Zunächst sollte man versuchen, ein **Handbuch** (Manual) zum Board zu bekommen. Zwar sind oft viele Jumper, DIP-Schalter und Steckverbinder auf den Boards beschriftet, leider aber meist nicht alle, sodaß ein ungeklärter Rest und damit ein Gefühl der Unsicherheit bleibt. Leider erhält man in den wenigsten Fällen ein Board "aus der Kiste" samt Manual, sodaß man gut beraten ist, sich um eine 'second source' für ein solches (Freunde, Bekannte, andere Händler) umzusehen.

Aber **auch ohne Manual** kann man sich an die Inbetriebnahme wagen. Dazu muß das Board alle zum Betrieb notwendigen Bauteile enthalten:

Prozessor (CPU), RAM (üblicherweise mindestens 1 MB in der richtigen Bauform und Organisation wie SIM-Modul, SIP-Modul oder DRAM) Keyboard-Controller, BIOS-ROM(s) oder -EPROM(s). Bei manchen Boards ist im CMOS-Setup der Parity-Check abschaltbar, diese können dann mit 8-Bit-RAM-Modulen bestückt werden. Boards ohne dieses Feature brauchen mal-9-organisierte RAM-Module. Weiters sollte man grundsätzlich die schnellsten zur Verfügung stehenden RAMs einsetzen.

Die Verwendung des **BIOS-ROMs** oder -EPROMs eines anderen Boards ist nur dann möglich, wenn auf beiden (abgesehen von der Taktfrequenz) der gleiche CPU-Typ (z. B. 386DX) UND dasselbe Chipset sitzen!

So das Board **Cache-RAM** besitzt, darf dieses eventuell fehlen, nur müssen dann meist einige Jumper richtig (auf 0 kB Cache) gesetzt sein.

Jedenfalls verzichtbar ist ein **Coprozessor**.

Für den Test reicht ein **Netzteil**, ein **Floppy-Controller** (üblicherweise in einem IDE-Controller enthalten, kann aber auch on-board sein) samt über ein 34poliges Floppykabel angeschlossenes Floppylaufwerk, ein **Keyboard** und eine **Bildschirmkarte** samt **Monitor**. Dazu braucht man noch eine **Boot-Floppy-Disk** (auf einem anderen PC mit SYS A: erzeugbar), auf die man zweckmäßig auch einige andere Tools kopiert. Das Board braucht nicht in ein Gehäuse eingebaut zu werden, es sollte aber während des Tests auf einer gut **isolierenden Unterlage** liegen (am besten auf Distanzbolzen in der Luft).

Zunächst setzt man alle **Jumper/DIP-Schalter** möglichst auf die richtigen Werte (oder was man dafür hält); ungefährlichere Werte sind immer besser, so z. B. kann man bei einigen Boards den Bustakt per Jumper einstellen; natürlich nimmt man für den Test den langsameren.

Danach empfiehlt sich eine **optische Kontrolle**, etwa auf Lötbrücken, verbogene IC-Beine, vergessene Schrauben im Slotstecker, Stecknadeln, die IC-Pins leitend verbinden, etc. (nicht lachen, alles schon erlebt); aber auch, ob alle ICs wirklich richtig ausgerichtet (Kerbe oder Aufdruck kennzeichnet Pin 1) in den Sockeln sitzen.

Nach dem **Zusammenbau** des Ganzen schaltet man zunächst den Strom ein (und geht dabei vorsichtshalber in Deckung). Bleibt die Sicherung im Netzteil intakt (Lüfter läuft), dann mißt man mit einem Voltmeter die Versorgungsspannungen am Power-Stecker des Motherboards (5V, 12V, -5V, -12V). Sind diese OK und erscheint ein

Bild auf dem Monitor, dann versucht man, ins CMOS-Setup des Boards zu gelangen.

Jedes Board hat ein batteriegepuffertes RAM (das sog. **CMOS-RAM**), in dem grundlegende Einstellungen gespeichert sind (etwa, welche Floppy-Laufwerke vorhanden sind). Damit die Einstellungen nach dem Ausschalten des Netzteils nicht verlorengehen, wird das RAM aus einer Batterie oder einem Akku (kann on board sein oder auch extern) versorgt. Die meisten Boards haben ein im ROM befindliches Setup-Programm, das mit einer speziellen Tastaturkombination erreichbar ist (etwa: DELETE-Taste nach dem Booten gedrückt halten, oder CTRL+ALT+ESC, etc.). Nur wenige Boards haben dieses Feature nicht, sie benötigen eine Startup-Diskette, bei dem dieses Programm von Disk geladen werden muß - was zumindest einen funktionierenden Floppybetrieb voraussetzt, weswegen die Hersteller davon abkamen.

Wenn das Board nicht hochfährt, kann es daran liegen, daß im CMOS-Setup gänzlich falsche Einstellungen gespeichert sind; gute Boards haben für einen solchen Fall einen Jumper, der das CMOS-RAM von der Batterie/dem Akku trennt; gibt es keinen solchen Jumper, hilft das Abziehen des Akku-Kabels bei externem Akku bzw. das Loslöten des Onboard-Akkus. Dadurch wird das CMOS-RAM geleert und es werden beim nächsten Booten Default-Einstellungen aus dem ROM geholt, mit denen ein intaktes Brett jedenfalls hochfährt. Das Trennen des CMOS-RAMs von der Versorgungsspannung sollte sicherheitshalber einige Minuten andauern, da ein Stützkondensator die Spannung am CMOS-RAM für etwa diese Zeit aufrechterhalten kann.

Oft ist nur zufällig (oder nach Murphy wohl eher die Regel?) einer der Slots, in die der Controller oder die Grafikkarte gesteckt wurden, defekt, man kann einen anderen Slot probieren. Andere Fehlerquellen sind: Floppy verkehrt herum angesteckt (Pin 1 nicht an der Markierung des Flachbandkabels, Kennzeichen: Floppy-LED leuchtet dauernd); Netzteil-Stecker verkehrt (erkennlich am wütenden Summen oder der kaputten Sicherung des Netzteils).

Kommt man ins Setup, so sollte man dort zunächst wieder die "ungefährlichsten" Varianten wählen, also nur eine (!) Floppy, keine Harddisk, maximale Verzögerungszeiten (Wait States) für RAM und Cache-RAM, keinen Parity-Check, Booten von A:, etc.

Wenn das Board nach dem nächsten Reset von der Floppy bootet, kann man schrittweise darangehen, mehr Peripherie zu installieren (wohl am Notwendigsten ist eine Harddisk mit Prüf-Tools wie dem legendären Programm CHECKIT oder den ebenfalls bekannten NDIAGS (NORTON Diagnostics) von den NORTON Utilities, Version 7).

Um eine **Harddisk** betreiben zu können, ist oft die Angabe von Parametern (wieviele Zylinder/Köpfe/Sektoren) im Setup erforderlich, die man dem Manual der Harddisk entnimmt, ferner muß die Harddisk meist eine spezielle Jumper-Einstellung haben, um als Single Drive zu funktionieren. Hat man kein Manual, empfiehlt es sich zur Ermittlung unbekannter Plattenparameter, auf einem funktionierenden PC zunächst irgendeinen (!) Plattentyp im CMOS-Setup einzustellen und ohne (!) Schreibzugriff auf die Platte die Plattenparameter auszulesen (Ausleseprogramme heißen z. B. ATBUS.EXE oder IDEID.EXE und sind als Shareware erhältlich). Es existieren aber auch Datenbanken mit Plattenparametern und Jumper-Settings wie etwa die Kauf-Software DISKBASE.

Moderne Boards lesen die **Platten-Parameter von AT-Bus-Platten** vollautomatisch aus (sogenanntes Auto-Detect-Feature), sodaß man sich darum nicht kümmern muß. Besitzer von SCSI-Controllern haben dieses Problem nicht; zum Test sollte man aber besser eine AT-Bus- oder eine MFM-Platte vorziehen, da SCSI-Controller mehr an funktionierendem Equipment voraussetzen.

Viele Boards kommen bis zu dem Punkt, daß alle Funktionen wie Floppy- und Harddisk-Datentransfer, CPU und Bildschirm etc. funktionieren, daß aber **Abstürze nach einigen Minuten** oder auch erst nach Stunden zu beobachten sind. Diese können vielerlei Ursachen haben; nur durch Änderung der Betriebsverhältnisse des Boards kann man diese herausfinden.

Zunächst versucht man, ob sich ein **Absturz durch Rütteln** an Komponenten des Boards herbeiführen läßt. Oft ist es eine schlechte Lötstelle eines Slotsteckers (Nachlöten hilft) oder ein schlecht im Sockel sitzender Baustein. Manchmal ist auch ein Pin schlichtweg abgebrochen (kommt häufig nach dem unfachmännischen Einstecken von CPUs vor).

Unangenehmer ist es, wenn sich der Absturz beim Versuch, das Board durchzubiegen einstellt. Dann handelt es sich fast immer um einen **Haariß** in einer Leiterbahn, der - frei nach Murphy - meist nicht an der Oberfläche, sondern in einer der inneren Leiterbahn-Layers auftritt (die meisten Motherboards haben 4 oder 6 Layers). Aber auch ein Haariß einer Leiterbahn in einer der beiden Oberflächen-Layers ist so gut wie nicht zu lokalisieren - zu zahlreich sind die Leiterbahnen und Durchkontaktierungen, als daß man alle mit dem Ohmmeter überprüfen könnte.

Die **Oxidation von Kontakten** kann als Fehlerursache meist ausgeschlossen werden. Kontaktflächen sind auch bei billigsten Komponenten entweder verzinnt oder vergoldet und gewährleisten über einige Jahrzehnte hinweg Oxidationsfreiheit.

Sehr häufig sind es **thermische Ursachen**, aufgrund derer ein Board abstürzt. Deshalb wird diesen im folgenden breiter Raum gewidmet. Fast immer macht dabei nur ein einziger Baustein Probleme (etwa, weil er durch Hitzestau früher "angeknackst" wurde). Praktisch immer gilt, daß ein Baustein streikt, weil ihm zu heiß wird (kaum je, weil ihm zu kalt ist).

Man testet zunächst die **Oberflächentemperatur** aller Bausteine mit dem Finger. Manche dürfen ruhig wärmer als handwarm sein (etwa hochgetaktete CPUs); brennheiße Bauteile deuten aber ziemlich sicher auf Fehler hin.

Bei **brennheißen Cache-RAMs** hilft es oft, einfach das Cache im CMOS-Setup (sofern möglich) abzuschalten; auf Dauer ist das keine gute Lösung, da die Board-Rechenleistung darunter empfindlich leidet (z. B. auf 30% von vorher sinkt). Schnellere Cache-RAMs oder ein zusätzlicher Cache Wait State wirken oft Wunder.

Für einen Halbleiter-Baustein ist die Sperrschicht-Temperatur (im Inneren) wesentlich; diese kann, muß sich aber nicht unbedingt durch eine erhöhte Oberflächen-Temperatur bemerkbar machen. In diesem Fall versucht man, durch **kleinflächige Änderung der Umgebungstemperatur** den Übeltäter zu lokalisieren. Dabei ist es sowohl denkbar, durch Erhitzen einzelner ICs einen beschleunigten Absturz herbeizuführen, als auch durch zusätzliche Kühlung diesen hinauszuzögern zu versuchen.

Man sollte beachten, daß die meisten herkömmlichen ICs nur im Temperaturbereich von 0 Grad bis 70 Grad Celsius richtig arbeiten müssen. Außerhalb dieses Bereichs dürfen sie ruhig versagen (und gelten trotzdem als intakt).

Die erste Möglichkeit (**Erwärmen**) ist leichter anwendbar und zur Lokalisierung besser geeignet. Man braucht dazu einen Heißluft-Revolver oder einen Haar-Fön mit regelbarer Temperatur (350 Grad sind auf jeden Fall zu heiß, da schmilzt das Lötzinn und der IC hüpf von der Platine) und mit kleinflächig konzentrierbarem Luftstrom (Düse mit kleiner Öffnung). Auch ein in die Nähe gebrachter Lötkolben oder eine Kerze sind, wenn auch schlechter, geeignet. Bei Erwärmung eines bestimmten Bausteins stürzt der Rechner immer ab? Gefunden!

Zur zweiten Möglichkeit (**Abkühlen**) kann gesagt werden: Als Kühlmittel ist etwa flüssiger Stickstoff mit -196 Grad denkbar ungeeignet (zu kalt und steht wohl auch nicht jedem zur Verfügung). Aber auch die im Handel erhältlichen Kältesprays sind nicht anzuraten, sie sind meist elektrisch leitfähig und "kriechen". Nur Preßluft ist halbwegs geeignet, die fördert aber bei zu hohem Druck das Kondensieren von Wassertropfen (elektronische Bauteile hassen Wasser in jeder Form!). Es bleibt also wohl nur noch die Luftkühlung. Ein Ventilator hat üblicherweise keine Düse, produziert daher meist keinen definierten Luftstrom und kann daher nicht zur Auffindung des problematischen Bausteins dienen. Wohl aber kann er zur Vermeidung der thermischen Probleme im Dauereinsatz beitragen.

Hat man den Problememacher lokalisiert, dann ist zu überlegen, ob ein einfacher Kühlkörper reicht (bei ICs in Flat Package ist ein dafür geeigneter Fingerkühlkörper anzuraten). Kühlkörper sollte man entweder mechanisch aufkleben (je fester, desto geringer ist der Luftspalt und desto besser ist die Wärmeleitung zwischen IC-Oberfläche und Kühlkörper. Außerdem ist ein sicherer Halt wichtig

wegen allfälligen Verrutschens.). Noch besser ist es, ihn vorher mit Wärmeleitpaste zu bestreichen. Diese Paste ist zwar ein schlechter Wärmeleiter, aber ein besserer als Luft. Am allerbesten ist das Aufkleben des Kühlkörpers mit einem speziellen Wärmeleitkleber (der im Wesentlichen eine Wärmeleitpaste mit Klebstoffeigenschaften ist). Dieser ist zwar im Einzelhandel erhältlich, aber sehr teuer; die geringste erhältliche Menge reicht aber dafür gewöhnlich für Hunderte von Kühlkörpern. Vom Aufkleben mit normalen Haushaltsklebern (etwa UHU) ist stark abzuraten, da die getrocknete Klebeschicht thermisch gut isoliert; das gleiche gilt für Zweikomponenten- oder Superkleber, die aber ohnehin kein vernünftiger Mensch verwenden wird (weil man den Kühlkörper nur unter Zerstörung des ICs wieder von diesem lösen kann).

Meist reicht aber ein Kühlkörper zur Bausteinkühlung allein nicht aus; die Bestrahlung des ICs mit einem Ventilator (oder eine Kombination aus Kühlkörper und Ventilator) ist dann anzuraten.

Elektromagnetische Einstrahlungen aus der Umgebung sind nicht zu vernachlässigen; ein Board ohne Gehäuse direkt in der Nähe eines Monitors ist ein potentieller Absturzskandidat.

Andere Faktoren, wie etwa die **Feuchtigkeit der Umgebungsluft** bewegen sich (außer im Badezimmer) ohnehin meist im Rahmen des Unproblematischen.

Aber auch auf dem **Software-Weg** kann manches Problem behoben werden. So kam eine alte, sehr langsame Harddisk (die über 500 MB hatte und darum den Aufwand wert schien) mit dem Bustakt eines 486-Boards nicht zurecht, obwohl dieser sich mit 8 MHz noch im erlaubten Bereich befand (der Bustakt wurde nie so richtig offiziell spezifiziert, sollte aber irgendwo zwischen 6 und 8 MHz liegen). Das eingebaute Setup hatte keine Möglichkeit zur Einstellung des Bustaktes. Oft läßt einen das eingebaute Setup nicht alle Parameter verändern, die man im Chipset wirklich verändern kann. Manchmal schafft ein von Disk ladbares Setup-Programm (bei Boards mit neuerem AMI-BIOS heißt es AMISETUP) Abhilfe: es ließ mich den Bustakt senken - und seitdem funktioniert die Platte an diesem Board einwandfrei.

Als weiteres **Beispiel** sei angeführt: Der Verfasser dieses Artikels bekam ein 386DX33-Motherboard (ohne CPU) geschenkt. Der Vorbesitzer hatte es entnervt aus seinem Rechner entfernt, weil das Board ca. jede Viertelstunde einen Reset durchführte. Nach der Investition in eine CPU (gebraucht um 400 öS) wurde auf dem oben beschriebenen Weg festgestellt, daß ein bestimmter Baustein des Chipsets thermische Probleme machte, die aber durch dauernde Bestrahlung dieses Chips mit Luft aus der Nähe (kleiner Ventilator um 90 öS) gelöst werden konnten. Seitdem arbeitet dieses Board mit nicht mehr Abstürzen als andere auch - um insgesamt 490 öS wohl wirklich preiswert.

Es sei nicht verhehlt, daß es der Wunder viele gibt: das Board eines meiner Freunde stürzt etwa in der ersten Viertelstunde nach dem Power-On gerne ab - danach aber nicht mehr. Im Sommer fühlt es sich wohl, tödlich hingegen ist das Fenster-Öffnen im Winter. Offensichtlich braucht da irgendein IC eine Mindest-Temperatur (dieser Fall ist der einzige mir bekannte, wo es ICs gerne warm haben - ich lerne aber gerne dazu).

Kryptischeste Beeinflussungen existieren (bestimmte Platte will nicht zusammen mit 70ns-SIMMs, funktioniert aber mit langsameren 80ns-SIMMs). Oft hilft ein simples Austauschen von Teilen gegen andere, scheinbar gleichwertige oder "schlechtere" (langsamere). Kaum jemand hat das Equipment (Logic-Analyser, Speicher-Oszilloskop), das Know-How und vor allem die Zeit, um wirklich Messungen durchzuführen und den Fehler systematisch zu suchen. Heuristische Vorgangsweisen führen viel eher zum Ziel. Motto: Probieren geht über Studieren - nur der Erfolg zählt.

Noch ein anderer - zugegeben extremer - Fall aus der Praxis sei geschildert, der den Recycling-Gedanken ein wenig hervortreten lassen soll: Windows hat die noch vor wenigen Jahren als Standard geltenden 286er dermaßen wertlos werden lassen, daß ein Kollege auf einem PC-Meeting (ca. 80 potentielle Interessenten) ein funktionierendes 286er-Motherboard mit 12 MHz nicht 'mal um 200 öS an den Mann bringen konnte - worauf er es kurzerhand einem Wildfremden zum Ausschachten schenkte, der bei Lötversuchen den Power-Connector seines 386er-Motherboards beschädigt hatte und einen solchen als Ersatzteil suchte....

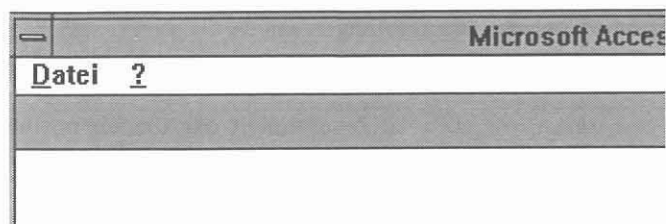
Schluß auf der übernächsten Seite.

ACCESS, eine Datenbank unter Windows

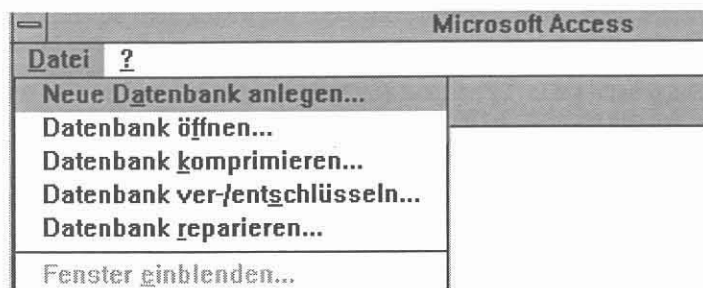
Helmut Schlögl, MCCA

MS-ACCESS ist eine neue, leistungsstarke Datenbank unter der Windowsoberfläche. Die Vorteile liegen in den rasch zu erstellenden Formularen, Berichten und Abfragen. Sie ist auch als Front-End Tool zu SQL-Datenbanken, wie Oracle oder Informix gut geeignet. Bei der Installation von MS-ACCESS wird eine eigene Programmgruppe im Programmanager eröffnet. Mit Doppelklick wird MS-ACCESS gestartet und gibt die Versionsnummer und den Lizenznehmer aus. Zum Ausblenden des Hilfebildschirmes bei weiteren Aufrufen von MS-ACCESS ist ein Feld zum Ankreuzen vorhanden.

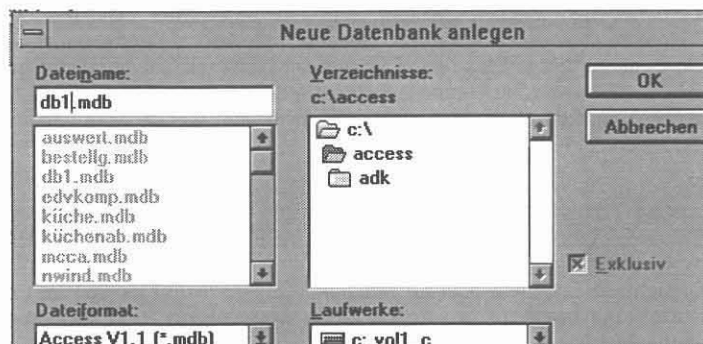
Und so meldet sich MS-ACCESS:



Das Fragezeichen (ALT ?) eröffnet die Onlinehilfe.



Im Menüpunkt "Datei" sind vorerst 2 Einträge von Bedeutung "Neue Datenbank anlegen .. und Datenbank öffnen....."

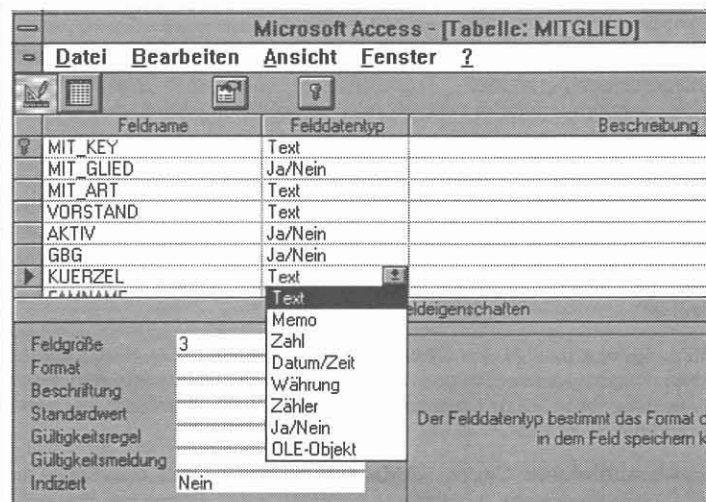


Nun muß ein Laufwerk/Pfad und Name für diese Datenbank = Projekt eingegeben werden. Projekt deshalb, weil alle **Tabellen, Formulare, Berichte, Abfragen, Macros** und **Module** unter diesem Projektnamen.mdb zusammengefaßt sind.



Ich habe die Datenbank (Projekt) MCCA benannt. Wir sehen nun die verschiedenen Schaltflächen, wobei die letzte, hier fast nicht sichtbar, die MS-ACCESS Basic-Modul Schaltfläche ist. In der Menüleiste sind einige Punkte entweder durch Mausklick oder mit ALT und unterstrichenem Buchstaben auswählbar. Darunter befindet sich die Zeile "Symbolleiste" auf die wir später eingehen werden.

Zunächst erstellen wir eine neue Tabelle. Dazu klicken wir auf die Schaltleiste **Tabelle** und dann auf **Neu**.



So sieht nun das Formular für die Eingabe der Feldnamen und Feldtypen aus.

Für das Erstellen einer Datenbank bedarf es vorerst einer gründlichen Erstellung eines Pflichtenheftes: was soll erreicht werden, welche Daten werden wo benötigt, kann man alle Daten in einer Tabelle verwalten und warten, welche Auswertungen (Berichte, Abfragen) werden gewünscht ?

Leichter ist es natürlich aus einer schon bestehenden, selbstentwickelten Datenbank unter dBase III+ oder IV oder einer anderen Datenbank, Tabellen (.dbf) "einzubinden" oder "anzubinden". Tatsächlich gibt es zwischen beiden Worten einen Unterschied. Beim "Einbinden" werden die Struktur und die Daten in eine ACCESS-Tabelle umgewandelt und können daher frei editiert werden; beim "Anbinden" werden nur die Daten in einer anderen Datenbank aktualisiert, die Struktur kann in ACCESS nicht verändert werden.

Die Feldbezeichnungen können 64 Zeichen lang eingegeben werden und sind daher recht übersichtlich zu gestalten. Feldtypen gibt es lt. Popup eine Reihe mehr als in anderen Datenbanken. Zur Dokumentation kann jeder Zeile eine Beschreibung mitgegeben werden.

Mit Mausklick oder F6 (Funktionstaste) werden nun die Eigenschaften jeder Eingabezeile festgelegt. Feldlänge, Format (Zahlen), Standardwert, Gültigkeitsregeln und die Beschriftung des Eingabefeldes in einem Formular. Unterläßt man die Beschriftung, werden in einem automatisch generierten Formular (z.B. zur Eingabe der Daten) nur die Feldnamen eingetragen. Je sorgfältiger eine Tabelle erstellt wird, umso einfacher werden die nachfolgenden Schritte.

Jeder Tabelle sollte ein eindeutiges Feld als Schlüsselwort beigegeben werden, um spätere Beziehungen (**Relation**) zwischen anderen Tabellen herstellen zu können. Ein Klick auf die Zeilenmarkerleiste, die Zeile wird schwarz unterlegt und ein Klick auf das Schlüsselsymbol (Schlüsselloch) erzeugt den **Primary Key**. Es sind auch mehrere **Primary Keys** möglich, nach denen indiziert wird. Diese **Primary Keys** dürfen in der Tabelle dann in unterschiedlichen Kombinationen nur einmal vorkommen. In den Eigenschaften wird vermerkt "Indiziert" (ohne Duplikate). Zusätzlich kann über jedes andere Feld ein Index, der auch Duplikate enthalten kann, angelegt werden. Hiefür gibt es die Eigenschaft **Index Duplikate** möglich. Aus der Menüleiste, Punkt **Ansicht** und weiter mit **Tabelleneigenschaften**, oder Klick auf das Tabelleneigenschaftssymbol (3. von links), können 5 Mehrfeldindizes angelegt werden. (z.B. Index1 mit_key; eintritt) . *Schluss auf der nächsten Seite.*

Dokumentation eines Datenbankentwurfs

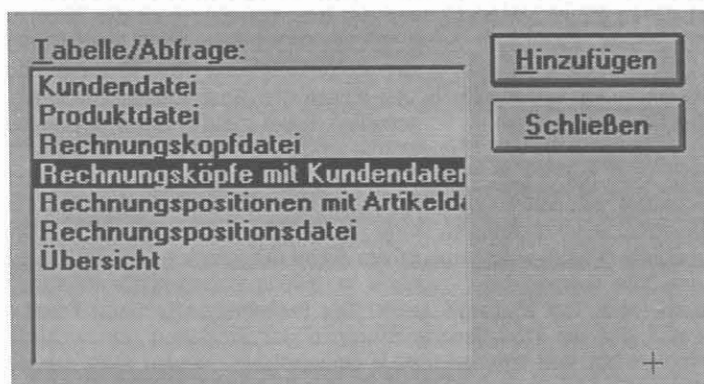
Eduard Fleck, N, TGM

Die Entwurfsansicht der Abfragen in Access bieten eine ausgezeichnete Möglichkeit, den gesamten Datenbankentwurf grafisch darzustellen. Es kann dabei sowohl die Struktur der Tabellen (Dateien), als auch die Beziehungen (Relationen) übersichtlich dargestellt werden. Besonders in Hinblick auf die Realisierung von normierten Datenbanken ist eine grafische Darstellung überaus hilfreich!

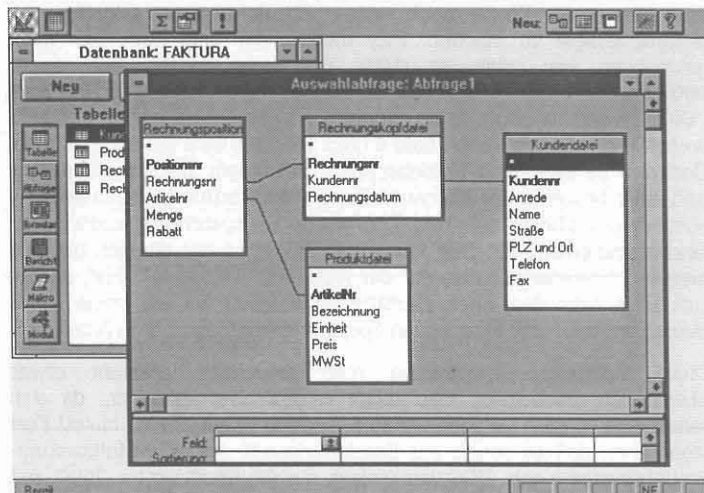
Ich empfehle die folgende Vorgangsweise bereits in der Entwurfsphase, um eventuelle Fehler bei den Relationen oder Verstöße gegen diverse Normalformen rechtzeitig zu erkennen.

Nach dem im ersten Schritt alle Tabellen erstellt sind, müssen zweitens die Relationen zwischen den Tabellen erzeugt werden. Dabei kann es bereits hilfreich sein, sich in der Entwurfsansicht der Abfragen einen Überblick zu verschaffen.

Öffnen Sie nun eine neue Abfrage und ziehen Sie aus dem Datenbankbereich alle Tabellen in den Entwurfsbereich oder wählen Sie aus der Liste der Tabellen alle aus.



Access zeigt nun alle Tabellen mit den zugehörigen Relationen an. Durch entsprechendes Verschieben der Tabellen läßt sich ein übersichtliches Bild der gesamten Datenbank erzeugen.



Leider bietet Access nicht die Möglichkeit, diese Darstellung direkt auszudrucken.

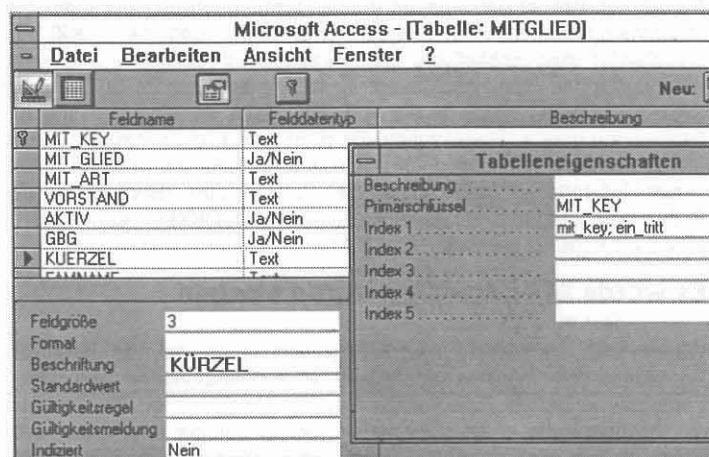
Sie können aber den Inhalt des aktuellen Fensters mit der Tastenkombination <ALT> <PrintScreen> in die Zwischenablage kopieren und von dort zum Beispiel in Winword weiterverwenden oder in Paintbrush weiterbearbeiten. □

Schluß des Beitrags Bretter-Recycling

Zusammenfassend kann festgehalten werden: Ein Board aus der Kiste sollte möglichst vollständig sein, um gute Chancen auf Wiederbelebung zu haben. Fehlende Handbücher oder ICs mindern diese Chance dementsprechend. Damit sich aber der Erfolg einstellt und ein Board wirklich zu einem "Schnäppchen" wird, ist noch etwas Probieren (und oft Improvisationsgeist) gefordert. Aber auch Mißerfolge bei der Reparatur sollten nicht entmutigen: Das Board kann immer noch ausgeschlachtet werden. Ein ausgeschlachteter funktionsfähiger

Keyboard-Controller etwa kann nicht nur ein im Notfall begehrtes Ersatzteil sein (Murphy: Defekte passieren immer Samstag mittag, sobald alle Elektronik-Läden geschlossen sind), sondern reduziert auch den Müllberg, was vom umweltpolitischen Standpunkt aus sicher zu begrüßen ist. Und für wieviele Stunden mißglückter Reparaturen wird man entschädigt, wenn man ein einziges Teil wirklich selbst repariert hat! □

Schluß des Beitrags ACCESS, eine Datenbank unter Windows



Ist die Tabelle sorgfältig erstellt, wird sie mit Datei speichern und Eingabe eines Namens abgelegt und erscheint im Datenbankfenster. Durch zweimaliges Anklicken mit der linken Maustaste oder Klicken der Schaltfläche Öffnen wird die Tabelle in der Dateiansicht, durch zweimaliges Anklicken mit der rechten Maustaste die Entwurfsansicht, die wir soeben erstellt haben, geöffnet.



Wenn Sie Interesse an der Weiterführung dieser Lektion haben, schreiben Sie mir bitte unter "Fido 2:313/9.30" oder über BTX 912916327, H.Schlögl. □

Das Ende einer Odyssee... (mein Kampf mit OS/2)

Dominik Schuierer, TU-Stuttgart

PROLOG:

Am Anfang war das Nichts, dann kam Bill Gates und es gab DOS, das ging eine Weile lang gut, bis der böse Gott der PC-Hersteller auf die Idee kam, seine Kinder nach seinem Ebenbild immer schneller und leistungsfähiger zu machen. Billy merkte, daß sein Kind dem nicht gewachsen war, aber es störte ihn nicht weiter. Daß dieses Betriebssystem nur 640k verwalten konnte, war für ihn kein Handicap (*"640k of memory ought to be enough"*). Es gingen Jahre ins Land. DOS wurde älter und war schon bald 4 (.01). Plötzlich kam ein anderer böser Gott auf die Idee, sein Betriebssystem mit einem akademischen Titel und einer besseren Speicherverwaltung auszustatten. Plötzlich fiel Billy auch dieses Manko auf, und Dos wurde 5. Irgendwann dazwischen fand irgendjemand ein paar verstaubte Disketten (für die sich bis jetzt niemand interessiert hatte) mit der Aufschrift "Windows". "Ha", dachte sich Billy, *"aus dem alten Grafiktreiber machen wir ein prima neues Betriebssystem - mit einer neuen Speicherverwaltung und Multitasking"*.

Diese historische Darstellung mag manchem vielleicht etwas übertrieben erscheinen; man möge es mir aber verzeihen, da sich selten eine so gute Gelegenheit zum Einstieg in ein Thema bietet! Fest steht aber, daß es schon zur Entstehungszeit der PCs Multitasking-Betriebssysteme gab. DOS war einfach nur ein überhasteter Unfall, mit dessen Risiken und Nebenwirkungen wir heute noch zu kämpfen haben.

Und - welch ein Wunder - die Firma, die Mitschuld an all diesem Leid trägt (IBM), hat sich auf Abhilfe besonnen und - jetzt sind wir endlich beim Thema - OS/2 entwickelt! Über dieses Betriebssystem hört man ja wahrlich viele Horrorstories: Teuer, langsam, inkompatibel, unzuverlässig, lange (mehrfache) Installation, DFÜ klappt nicht uva. Gerade deshalb schreibe ich diesen Text! Denn ich habe es gewagt, OS/2 ohne Vorkenntnisse auf meinem Rechner zu installieren und bin damit in Speicherregionen vorgedrungen, die nie ein Mensch zuvor gesehen hat (oder war das etwa ein anderer Film?). Also Sinn (oder Unsinn) dieser Schreibung war es, einmal einen unabhängigen Erfahrungsbericht über die Installation und Performance von OS/2 zu bringen. Zum verwendeten Rechner: 486DX33/16MB RAM/210MB HDD/ET4000W32 Grafik-Karte (Local Bus) also ideal für OS/2! Ich habe mich entschlossen, OS/2 mit DOS zusammen auf einer FAT Partition zu installieren. Das OS/2 eigene HPFS würde zwar mehr Leistung bringen, läuft aber nicht mit meinem Floppy Streamer. Also stürzen wir uns gleich in den Urwald der Installation, hier mein ultimativ subjektiver Tatsachenbericht. Wir beginnen bei der Stunde NULL:

0.00

Am Anfang war der Schraubenzieher und den brauche ich, um meinen Floppy-Streamer abzuklemmen (das haben viele empfohlen, scheint während der Installation nötig zu sein). **0.10** Der Drucker rumpelt an der README-Datei. Nach deren Anweisungen setze ich im Setup den secondary Level Cache zurück und nehme den Turbo raus. Diese Prozedur verlängert zwar die ganze Gaudi ziemlich, soll aber dafür sorgen, daß es während der Installation und auch später keine Probleme gibt (na hoffentlich). **0.22** Die erste Diskette verschwindet im Laufwerk **0.27** Es erscheinen farbige Bildschirme, die erste Hürde scheint genommen. **0.29** Fehlermeldung! Aufstöhnen "Jetzt schon!" Ich war so ungezogen, in meiner CONFIG.SYS den Befehl `command.com/p` und in der `autoexec.bat` `setcomspec...` zu vergessen. **0.30** es geht weiter.

14 Disketten und eine Tina Turner CD später...

1.10 fertig - es geht los mit der Konfiguriererei - problemlos. **1.13** Er will mir helfen, alte DOS Programme einzubinden. Ich mache ihm die Freude, und nach einer fröhlichen Plattenrödelei identifiziert er Turbo C und Turbo Pascal (sonst nichts). Naja, muß der große Meister wohl wieder einmal selber den Zauberstab schwingen! **1.27** Installation endgültig abgeschlossen! Ich starte das Lemprogramm, inzwischen rumort es auf der Platte, der Rechner braucht noch etwas Verdauungszeit und ich kann inzwischen (dank Multitasking) etwas üben! **1.50** genug gelernt für heute - auf in die Praxis! **1.56** ET4000 Treiber installieren. **2.10** Fertig - diesmal hoffentlich endgültig.

Das war's eigentlich. Die Installation ist beendet, alles klappte problemlos. Der Turboschalter rastet wieder ein und der Cache nimmt seinen Dienst wieder auf - schnallen wir uns an... es geht los!

Erstmal ein oberflächlicher Blick auf die DOS-Programme. Von diesen laufen (anscheinend) folgende: GSADRESSEN, LANDMARK, CORETEST, FRACTINT, MATHEASS, TURBOPLOT, TURBO PASCAL, XRS Offline Mailreader, MERCURY, CHEMVIEW, SCANEDIT (das Programm zu meinem Handscanner), ASEASYAS, ELVPROM (Programm zum Eprommer). Folgende Programme verweigern aufgrund irgendwelcher fauler Ausreden den Dienst: 3D-Designer, Turbo C++ (war zu erwarten, kriegte ich ja unter DOS kaum zum Laufen!), VGACOPY, LABY (selbst geschriebenes Spiel).

Das war sicherlich keine berauschende Statistik. Von wegen DOS-Kompatibilität! Die Anfangseuphorie schwindet dahin. Gleichzeitig flimmert der Bildschirm und im DOS-Fenster wurden keine ESCAPE-Sequenzen dargestellt (zuständig für Farb- und Cursor-Steuerung). Da selbst ich (hüstel) nicht alles auf einmal machen kann (der Mensch ist eben nur Singletask-fähig), versuche ich mich zuerst an den Escape-Sequenzen. "Klarer Fall", denkt sich der (Möchtegern)Profi, *"da fehlt der ANSI.SYS"*. OS/2 hat ja auch eine AUTOEXEC und CONFIG Datei. Und da stellen wir uns ganz naiv und binden den Ansi wie unter DOS ein. Der User macht einen Systemabschluß, Reset - und wundert sich, denn es erscheinen im DOS-Fenster die gewünschten Farben! Momentan liegt OS/2 mit **3:1** in Führung (1 von 3 Problemen gelöst). Nächster Härtestest. Ich hatte gehört, daß es bei der DFÜ zu Problemen mit der Übertragungsrate kommen kann (jedenfalls, wenn man sich keine spezielle Schnittstellenkarte kaufen will). Auf geht's - Telix rein und eine Datei downgeloadet. Ergebnis ca. 1600 cps. Besser liefs unter DOS auch nicht. Der Mutpegel steigt! Der Performance-Test mit Fractint ergibt 95% der DOS-Rechenleistung in der Emulation. Ein weiteres Problem hat sich ergeben. Einige Grafikzeichen werden nicht korrekt dargestellt. Versuche ergeben als Grund die falsch gewählte Codeseite 850. Direkt mit CHCP 437 auf die richtige Seite 437 zu schalten, scheitert. Nach zwei Stunden bemerkt der entnervte User den Grund für das Vergebliche seines Tuns. Die CHCP Anweisung gehört in die AUTOEXEC und nicht in die CONFIG. Ein Griff in die Gegend der Denkkzentrale, und ein unterdrücktes Stöhnen begleiten diese Erkenntnis. Lohn der Mühe: alle Zeichen werden jetzt vollkommen korrekt dargestellt. (Es steht **4:2**) Einer bösen Eingebung folgend starte ich Telix und versuche einen Upload - mit böser Überraschung! Nach 10kB hängt alles. Der Task konnte zwar beendet, aber nicht geschlossen werden und so boote ich die Kiste ohne Systemabschluß (Todsünde!!!) neu. Jetzt wird's ernst, denn DFÜ ohne Upload Möglichkeit ist wie Telefonieren ohne reden - redundant. Deshalb muß nun ein OS/2-Terminal-Programm herhalten. Das Ergebnis ist verblüffend. Das System stürzt beim Upload ab! Mit einem Wort: riesig! Inzwischen sind einige Stunden ins Land gegangen und der User sitzt zusammengesunken vor seinem Häufchen Unglück. (**5:2**) Er entscheidet sich schließlich, die Bildflimmerei abzustellen. In den entsprechenden Fido-Echos stand ja, man müsse nur im DOS-Fenster `SVGA OFF` danach `VMODE xxxxx` und dann `SVGA ON` eingeben und die Sache wäre geritzt. Pustekuchen. Da rührt sich nix! Es vergeht wieder etwas Zeit mit Probieren, doch schließlich erinnere ich mich an meine "Ansi" Erfahrungen und binde die Befehle zum Hochsetzen der Bildwiederholffrequenzen einfach in die AUTOEXEC ein. Und - Frechheit siegt, es geht! (Es steht **5:3**).

Es wurde Abend und es wurde Morgen - der zweite Tag...

Warum läuft der Upload nicht? Man hört ja so viel von Interrupt-Konflikten. Ich habe ja schließlich eine Scannerkarte und einen Eprommer eingebaut. Das könnte doch die Ursache sein! Ein Zerlegen des Rechners und anschließender Check der Jumper ergibt aber kein Indiz für diese Vermutung. Auch eine Spielerei an den COM-Einstellungen kann da nicht helfen! Dafür hat der gute Geist der Programmierer ein paar andere Überraschungen bereit: Mit Hilfe der DOS-Einstellungen ist es auf einmal möglich, Turbo C, 3D-Designer und Laby zum Laufen zu bringen. Anscheinend brauchen einige DOS-Programme den Hardwaretimer und direkten Bildschirmzugriff. VGACOPY streikt beharrlich und nachdem im Handbuch selbst etwas von "wild programmiert" steht, gebe ich diese Versuche auf und

verbuche es als einen Sieg für mich: **5:4**. Ein wildes Herumhantieren an den DOS-settings für Telix löst mein DFÜ Problem auch nicht.

Es wurde Abend und es wurde Morgen - der dritte Tag...

Das Upload-Problem hat es mir angetan. Ich zerlege den Rechner schon wieder und entferne auf Verdacht die Scannerkarte (eine wahre Verzweiflungstat!) Hoffnungsvolles Blättern im Handbuch bringt Erfolg. Da steht etwas von 9600 Baud (Schnittstellengeschwindigkeit) pro Task. Naja, mit einem 14400 bps Modem geht das wohl nicht, aber man könnte ja mal versuchsweise von 38400 auf 9600 runterschalten.... Murphy war gnädig und läßt für heute ein Wunder zu! Upload und Download laufen absturzfür! Also kein Kartenproblem. Gut! Aber 19200 brauche ich mindestens um mit 14400 auf die Leitung zu gehen. Also wüßte ich Telix die 19200 ins Setup und - Upload Absturz. Das darf doch nicht wahr sein, ich sehe mich schon lebenslänglich mit 9600 bps auf der Leitung! Das Menü der DOS-Settings spendet Trost und rät, den direkten Schreibzugriff auf die Schnittstelle (ohne Puffer) zuzulassen. Danach läuft der Upload extrem schnell, aber ohne Empfangspuffer fehlen beim Download zu viele Zeichen => Ärger. Das bringt also nichts. Eine letzte Chance bietet das Handbuch, es rät unter anderem die Priorität **IO_DISK** auf **NO** und **WAITMAX** auf **1** zu stellen (was immer das auch bedeuten soll). Man mag es nicht für möglich halten, von da an ging's. Aber wehe, wenn die Schnittstelle auf 38400 steht. Da läuft nix! Das war der Ausgleich **5:5**.

So, der Kampf ist gewonnen. Eigentlich hatte ich es mir viel schlimmer vorgestellt. Wie sich gezeigt hat, funktioniert die Konfiguration ähnlich DOS mit den entsprechenden Dateien. DOS-Programme laufen problemlos und ein klarer Vorteil für OS/2 ist, daß für jedes einzelne Programm andere Settings gewählt und eine eigene **AUTOEXEC.BAT** erstellt werden kann. So kann jedem Task soviel EMS und XMS zugewiesen werden, wie nötig. Keine faulen Kompromisse mehr! Ein abgesammelter Task läßt sich abbrechen und auf der Arbeitsoberfläche (meist) auch löschen. Wirkliche Probleme treten nur bei DFÜ auf. Wer sich eine Karte mit FIFO sparen will, sollte auf jeden Fall kein 19200er Modem haben und sich wie beschrieben ans Handbuch halten. In der Hinsicht gibt es dann keine Probleme mehr. Allerdings sollte man schon etwas "Kommandozeilenerfahrung" mitbringen. Oberflächengeschädigten Usern würde ich einen Umstieg nicht

empfehlen. Eine Eigenheit von OS/2 will ich hier auch nicht verschweigen: Der "Systemabschluß" ist Pflicht. D.h. es ist verboten, dem Rechner am Ende der Tipperei einfach den Saft abzudrehen. Vorher möchte das System gerne noch seine Daten sicher abspeichern. Sollte man das vergessen, oder aufgrund eines (i.a. sehr seltenen) Systemcrashes nicht dazu kommen, können die INI-Dateien fehlerhaft werden. Für solche Fälle hat der Flachmann immer ein Backup der wichtigsten Systemdateien zur Hand.

Vielen wird aufgefallen sein, daß ich nichts über die Windows 3.1 Emulation unter OS/2 gesagt habe; der Grund ist einfach: ich habe momentan keine Windows Programme am laufen (s. Prolog). Und da IBM ja den original Microsoft-Code verwendet, kann da soviel nicht anders sein (Abstürze mitgerechnet).

Ein Erfahrungsbericht, wie ich ihn hier fabriziert habe, krankt natürlich immer am gleichen Problem: das Negative fällt sofort auf, das Positive wird vergessen. Dem kann ich hier nur kurz versuchen, Abhilfe zu schaffen: Das Betriebssystem wird mit einigen (Online-)Handbüchern zusammen installiert; mit deren Suchfunktionen kann man fast alles ziemlich hurtig finden und so viele Probleme online lösen. Die Oberflächenbedienung hat mich anfangs überrascht. Hier geht es wirklich zu, wie auf meinem Schreibtisch! Ordner können blindlings abgelegt, verändert, kopiert und wahllos in andere Ordner oder auf die Arbeitsoberfläche verschoben werden. Alles kein Problem. Und für uns Kommandozeilen-Fanatiker gibt es sowohl den DOS als auch OS/2 Bildschirm, mit denen sich (zwecks Unterhaltungswert) auch schon mal zwei Disketten gleichzeitig formatieren lassen... Last but not least ist OS/2 das erste Betriebssystem, das die PCs mit einer vernünftigen Speicherverwaltung fahren kann und trotzdem in der Lage ist, DOS und Windows Programme vernünftig zu verarbeiten.

Normalerweise versucht man zum Schluß, den Leser mit rhetorischen Tricks auf seine Seite zu ziehen. Das will ich beiden Seiten hier (er)sparen. Ich habe versucht, die Probleme und Lösungsansätze, die sich bei mir ergeben haben, zu schildern. Ob man sich in ein solches "Abenteuer" stürzen will, soll jeder selbst entscheiden, für mich war es auf jeden Fall das Richtige! □

CD - Topliste, 27. 10. 1993, alle Preise inkl. MWSt. Klaus Gloner, Dr. Hainstrasse 8 - 5620 Schwarzach - Tel. 06415/7215 - Fax 06415/7218

Platz	Art.Nr.	CD-ROM Titel:	Sparte:	Preis	Sonderpreis
1	CD-050	Clipart Goliath	Grafik	414,00	352,00
2	CD-153	Seedy Six-Pack	Erotik	4.429,00	3.765,00
3	CD-022	Pegasus 2.0	Shareware	657,00	558,00
4	CD-047	GIF's Galore	Grafik	357,00	303,00
5	CD-049	Our Solar System	Grafik	414,00	352,00
6	CD-304	German Only Vol. 1	Shareware	544,00	462,00
7	CD-401	AniMazing	Grafik	254,00	216,00
8	CD-135	Ecstasy Hot Pics	Erotik	1.159,00	985,00
10	CD-331	Bertelsmann Lexikon	Lexika	1.342,00	1.208,00
11	CD-028	WinWare Vol. 4	Shareware	338,00	287,00
12	CD-233	Videos für Windows	Grafik	470,00	400,00
13	CD-175	CD-Globe USA,	Reiseführer	733,00	623,00
14	CD-002	CICA Windows	Shareware	357,00	303,00
15	CD-113	Microsoft Cinemania	Kinodatenbank	2.423,00	2.060,00
16	CD-174	Sounds für Windows	Musik auf CD	414,00	352,00
17	CD-227	Raytrace Magic	Shareware	414,00	352,00
19	CD-001	SIMTEL 20	Shareware	357,00	303,00
20	CD-012	Night Owl 9	Shareware	788,00	670,00
21	CD-073	7th Guest	Spiel	2.048,00	1.741,00
22	8002	Kodak Access Foto CD-ROM	Grafik	525,00	446,00
23	CD-318	Global Explorer	Landkarten	3.551,00	3.018,00
24	CD-312	Jennifer Aktfotografie privat	Erotik	1.674,00	1.423,00
25	CD-182	Publishers Paradies	Grafik	1.163,00	989,00
27	CD-225	VGA Magic	Grafik	1.107,00	941,00
29	CD-326	ROMWARE Edition	Shareware	1.315,00	1.118,00
30	CD-016	PD-ROM 2, deutsche Shareware	Shareware	544,00	462,00

SCSI-Controller - ISA oder VLB?

Thomas Grausgruber, TU-Wien

DSK-372\HDBENCH.LZH

Nachdem in den heutigen *Normalverbraucher PC's* fast keine MFM oder RLL Harddisks mehr zu finden sind, sondern fast nur mehr AT-Bus (IDE) oder sogar SCSI Controller, und auch die Bussysteme immer leistungsfähiger werden - Stichwort: Windows und 8088 - soll hier an dieser Stelle ein Vergleich zwischen einem etablierten SCSI-Controller (**Adaptec 1542B, ISA**) und einem neuen VESA-Local-Bus SCSI-Controller (**VLS-6360**) gebracht werden.

Für alle, die den **VL-Bus** noch nicht kennen: Der **VL-Bus** ist ein direkt mit der CPU verbundener Bus - Daten, Adressleitungen und Kontrollsignale werden direkt an die VLB-Stecker geführt. Gegenüber dem ISA-Bus ist sowohl der Daten- als auch der Adressbus in einer Breite von 32 Bit ausgeführt. Weil die Taktgeschwindigkeit am VL-Bus dieselbe ist wie die der CPU (ISA: normalerweise 8 MHz) erreicht man hierbei einen größeren Durchsatz als bei ISA und EISA. Der Nachteil dabei ist, daß man maximal 3 solche VLB-Slots auf einem Motherboard unterbringen kann, da ansonsten die Leitungslängen und Kapazitäten zu unerwünschten Effekten führen.

Sobald man den **VLS-6360** aus seiner Verpackung befreit hat, sticht einem gleich zwei mal ein bekannter Name entgegen: Adaptec. Des Rätsels Lösung: die ganze Hardware dieser Steckkarte basiert auf einem von Adaptec hergestellten SCSI-Single-Chip-Controllers, auch die beiliegenden Treiber sind von Adaptec lizenziert. Für jeden, der schon einmal versucht hat, Wechselharddisks (z.B. SyQuest) zu lesen, die mit einem Controller eines anderen Herstellers formatiert wurden, der weiß sicher, daß meistens jeder sein eigenes Süppchen kocht. Hier die wenn auch nicht ganz unerwartete positive Überraschung: Harddisks, die von einem Adaptec 1542B formatiert worden sind, können ohne Problem vom VLS-6360 bearbeitet werden. Auch der Einbau erwies sich als unproblematisch, das beiliegende Handbuch (leider nur in Englisch) erklärt ausführlich, wie die Jumper und DIP-Switches auf dem Board einzustellen sind. Im Normalfall gibt es auch hier nichts zu verstellen, ich wollte die Karte aber als zweiten Controller parallel zum Adaptec 1542B betreiben, auch das war dank der guten Beschreibung problemlos möglich. Positiv zu vermerken ist auch, daß die Schalterstellungen nicht nur im Handbuch, sondern auch direkt auf dem Controller aufgedruckt sind. Besonders für Zeitgenossen, die gerne einmal das Handbuch verlegen und nach einiger Zeit doch noch einige Parameter ändern wollen, erweist sich das als außerordentlich nützlich. Dann kam aber auch schon die erste negative Überraschung. Das beigelegte SCSI-Kabel zur Verbindung zwischen Controller und Harddisk hat keine verpolsicheren Steckverbinder. Die meisten Harddisks haben leider keine Numerierung der Steckerleiste, sondern nur einen Schlitz auf einer Seite, sodaß ein mit einer Nocke versehener Stecker nur in einer Position angesteckt werden kann. Das kann dann böse Folgen für Harddisk und Controller haben. Auch auf das beiliegende Floppy-Anschlußkabel trifft das zu.

Ab jetzt wirds aber wieder besser. Die Installation der Treiber verursacht keine Probleme, der VLS-6360 verhält sich wie ein Adaptec 1522. Der Unterschied zwischen 1542 und 1522 liegt darin, daß der 1542 Busmaster-DMA fähig ist, der 1522 diese Spielart jedoch nicht beherrscht.

Busmaster-DMA heißt in diesem Falle, daß der Controller von der CPU lediglich eine Anforderung bekommt, Daten von der Harddisk nach einem angegebenen Speicherbereich zu transferieren. Dazu ist keine Rechenleistung der CPU erforderlich - der Controller übernimmt in dieser Zeit die Kontrolle über den Bus (daher auch Busmaster), die CPU kann sich in der Zwischenzeit mit etwas anderem beschäftigen. Beim 1522 hingegen übergibt der Controller sämtliche Daten der CPU, die dafür sorgt, daß diese richtig in den Speicher transferiert werden. Unter Betriebssystemen wie DOS wird man daher von Busmaster-DMA nicht allzuviel bemerken, wenn man aber Multiuser- und/oder Multitaskingfähige Betriebssysteme benutzt, merkt man sehr schnell eine nicht unerhebliche Leistungssteigerung - die CPU hat ja auch "mehr Zeit".

Ich möchte mich hier aber hauptsächlich auf DOS beschränken, das ja das (noch) hauptsächlich verwendete Betriebssystem ist.

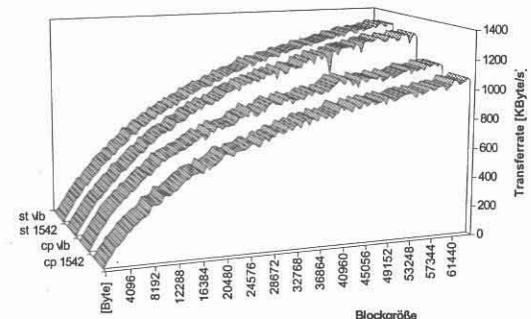
Zur Ermittlung der Geschwindigkeit der Controller wollte ich mich nicht auf Programme wie Coretest, Checkit und Norton's Sysinfo verlassen. Diese Programme ermitteln häufig Werte, die dann leider im

praktischen Betrieb nicht erreichbar sind. Welches Programm liest schon 100 mal hintereinander einen 64kB großen Block ein. Daher habe ich auf ein von der deutschen Computerzeitschrift c't entwickeltes Programm zurückgegriffen, daß vier verschiedene Datentransferraten ermittelt:

- **Write Linear** - dabei werden Datenblöcke hintereinander auf Platte geschrieben
- **Read Linear** - aufeinanderfolgende Datenblöcke werden eingelesen, hier werden die "Höchstgeschwindigkeiten" erzielt
- **Write Random** - der Realität näher - Datenblöcke werden durcheinander auf Platte geschrieben und bei
- **Read Random** - in zufälliger Folge eingelesen

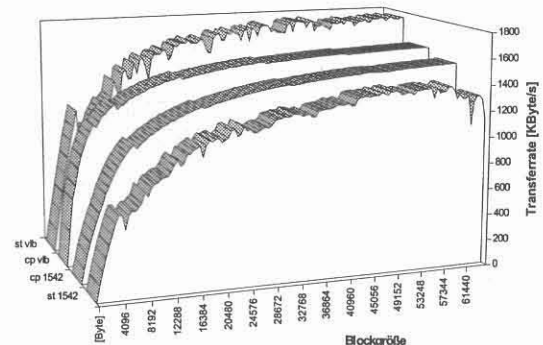
Damit nicht durch den Cache auf den Harddisks eine Platte Vorteile gegenüber der anderen hat, werden dazu Blöcke von 512 Byte bis 64 kByte (um genau zu sein: 65024 Bytes) benutzt. Das ganze ergibt einen Berg von Zahlen, die besser in grafischer Form darzustellen sind:

Bild 1: Write Linear



In **Bild 1** sind die Transferraten bei **Linearem Schreiben** dargestellt. Zu vergleichen sind jeweils die ersten beiden und letzten beiden Graphen. Ganz vorne sind die Ergebnisse einer Conner CP3540 (540MB) mit einem Adaptec 1542B, dahinter die mit dem VLS-6360. Im Vergleich dazu noch eine Seagate ST3283N (240MB) mit dem Adaptec 1542B dahinter und ganz hinten mit dem VLB-Controller. Der VLS hat hier nur sehr kleiner Geschwindigkeitsvorteile gegenüber dem 1542B.

Bild 2: Read Linear



Beim **linearen Lesen** (**Bild 2**) hingegen sind die Unterschiede schon deutlich sichtbar. Hier kommt auch eine Eigenheit der Seagate Platte zum Vorschein. Die starken Einbrüche im Kurvenverlauf sind nicht durch die Controller bedingt, sondern durch den auf der Platte integrierten Cache. Obwohl die ST3283N mit dem VLB Controller schneller ist als die Conner HD, ist die ST3283N mit dem 1542B deutlich langsamer als die CP3540.

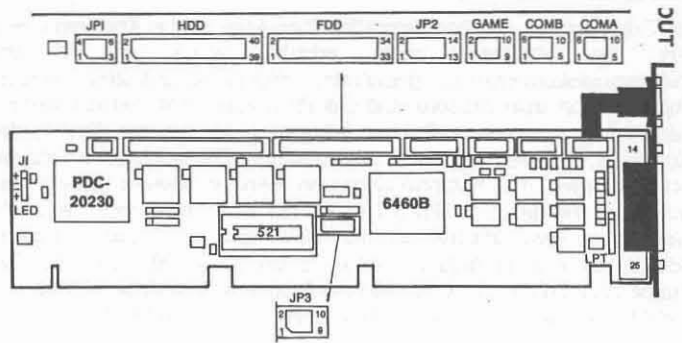
Schluß in der Mitte der nächsten Seite

VLB-Universal-Adapter

Was der Enduser nicht weiß, er aber beim nachträglichen Einbau zusätzlicher Platinen leidvoll erfahren muß, ist die Tatsache, daß keineswegs jede Kombination von Erweiterungskarten im VLBus funktioniert. Der hier vorgestellte VLB-Adapter vereinigt alle wichtigen Schnittstellen auf einer Karte und auf Grund guter Erfahrungen der Lieferfirma übertrifft er die bisher üblichen Adapter bei weitem, was die Kompatibilität mit anderen Karten insbesondere bei höheren Taktgeschwindigkeiten betrifft.

Seine Besonderheit ist ein eigenes BIOS, das sich beim Booten mit der Frage meldet, ob man einen TURBO-Modus oder den Standard-Modus verwenden will. Der Unterschied liegt darin, daß der TURBO-Modus die im BIOS vorhandenen Routinen zur Kommunikation mit der Festplatte durch eigene ersetzt und auch beschleunigt. Als zusätzliches Feature kann das Programm auch in einem schnelleren RAM ablaufen.

Im Aufbau überwiegt die Mechanik bereits die eigentliche Elektronik. Die Karte erreicht nicht einmal mehr die volle Bauhöhe. Alle Schnittstellen sind vielfältig konfigurierbar und natürlich über Jumper ausschaltbar.



Bezugsquelle: excon, Röggersgasse 6-8, 1090 Wien, TEL: (0222)-310-99-74, FAX: (0222)-310-99-74, Bestell-Nr.: 436VJH26, Preis S 576,-.

Schluß des Testberichts über SCSI-Controller

Da aber diese beiden Zugriffsarten in der Praxis seltener vorkommen, will ich das Hauptaugenmerk auf die nächsten beiden Tabellen legen.

Bild 3 zeigt die Transferraten beim **Schreiben in zufälliger Reihenfolge**, **Bild 4** beim **Lesen**. Beim Schreiben kann der VLB-Controller seine Geschwindigkeit nicht voll ausnützen, er ist wieder nur geringfügig schneller als der ISA-Controller. Die Lesegeschwindigkeiten hingegen sind deutlich höher. Bemerkenswert hierbei ist allerdings, daß die ST3283N im Vergleich zu Bild 2 gegenüber der CP3540 beim Lesen in zufälliger Reihenfolge langsamer ist. Hier kann der VLS-6360 seine 32-Bit Busbreite voll ausspielen.

Bild 3: Write Random

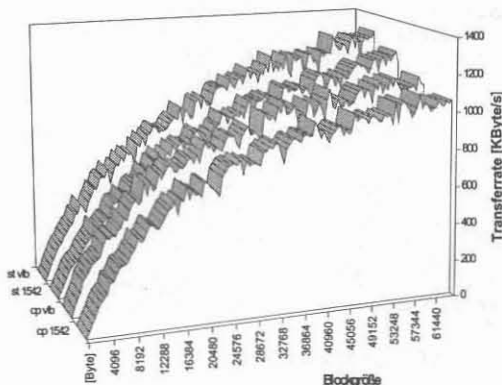
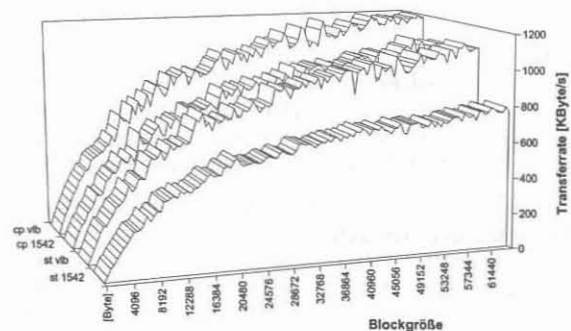


Bild 4: Read Random



Fazit: Unter DOS ist der VLC-6360 ein ernstzunehmender Konkurrent für den Adaptec 1542B. Interessant wäre noch ein Vergleich unter Linux oder OS/2. Wie man aus den Bildern aber entnehmen kann, ist nicht nur unbedingt der Controller ein Kriterium, sondern auch in ebensolchem Maße die Harddisk. Wer also aus seiner SCSI-Harddisk das Letzte herausholen will, der ist mit diesem Controller diesem Ziel ein Stück näher gerückt. □

Bezugsquelle: excon, Röggersgasse 6-8, 1090 Wien, TEL: (0222)-310-99-74, FAX: (0222)-310-99-74, Bestell-Nr.: DC403VLB, Preis S 2.220,-.

Public-Domain - Shareware - Freeware

Klaus Gloner

Im Gegensatz zu den kommerziellen Produkten ist das Kopieren dieser Art von Software jedem erlaubt, wenn die in der Programmdokumentation genannten Regeln eingehalten werden. Hierzu gehört insbesondere, daß das Programm nicht verkauft werden darf, sondern nur eine Gebühr erhoben wird, um die Kosten des Kopierens, der Werbung usw. auszugleichen. Die wachsende Zahl der professionellen und halbprofessionellen Händler beweist jedoch, daß mit dem Vertrieb von PD-Software und Shareware sehr wohl ein Gewinn zu erwirtschaften ist. So finden sich dann auch schwarze Schafe, die eine Programmdiskette für bis zu \$ 400,- anbieten. Der Kunde glaubt dann meist, er habe das Programm bereits bezahlt und ist nicht mehr gewillt, die Sharewaregebühren (s.u.) zu begleichen.

Praktisch alles, was an kommerziellen Programmen angeboten wird, ist auch in der Shareware zu finden, also: Utilities, Anwendungsprogramme, Spiele, Speziallösungen ... es gibt mehrere Zehntausend (!) Disketten aus den USA und einige Tausend aus deutschsprachigen Ländern. Natürlich existiert kein "Ventura Publisher der Sharewareszene", andererseits gibt es aber auch Anwendungen, mit denen kommerziell kein Profit zu erwirtschaften wäre, besonders im Bereich der Hilfsprogramme (Tools), die den Umgang mit dem DOS und den Anwenderprogrammen oft erheblich vereinfachen können. Teilweise wurden - aus Protest gegen die hohen Preise - kommerzielle Produkte nachprogrammiert (sog. "Clones"). Ein bekanntes Beispiel hierfür ist "As-Easy-As", ein weit verbreiteter "Lotus 1-2-3"-Clone, dessen Qualität auch in der Fachpresse gelobt wird.

Ein Teil der PD/Shareware stammt aus den Händen von Hobbyprogrammierern, die keinen Kommerz machen wollen. Besonders wenn der Quellcode des Programmes vorliegt, lassen sich natürlich Anregungen und Problemlösungen für eigene Software finden.

Begriffe

Die Autoren von **PD-Programmen** (PD steht für Public Domain) gestatten die kostenlose Weitergabe ihrer erstellten Software, ohne jegliche finanzielle Forderungen an den Benutzer zu stellen und verzichten sogar auf ein Copyright für ihre Programme.

Im Gegensatz zur Public-Domain-Software erbittet der Programmautor von **Sharewareprogrammen** von allen, denen das Programm gefällt und die es dauerhaft einsetzen wollen, eine Gebühr. Teure Fehlkäufe werden dadurch vermieden, daß der Anwender das Programm vorher testen kann. Für diese sog. Registriergebühr erhält der Anwender dann oft ein ausführliches, gedrucktes Handbuch, aktuelle Versionen des Programmes oder Unterstützung bei Fragen und Problemen mit der Software.

Bei der **Freeware** handelt es sich um Programme, die zwar ebenfalls kostenlos benutzt werden können, aber im Gegensatz zu den PD-Programmen darf die Software und die Dokumentation nicht verändert werden.

Windows Highlights Disk 1

KIRK100S.EXE - Captain Kirk Version 1.0

Captain Kirk ist ein neuer deutschsprachiger Dateimanager (oder sollte man besser sagen: eine Kommando-Brücke), mit dem alle wichtigen Dateioperationen komfortabel ausgeführt werden können.

Auffällig ist die Nähe zum Norton Commander, was Bildschirmaufbau und Befehle anbelangt. Von allen Dateimanagern, die uns bekannt sind, kommt Captain Kirk dem Commander am nächsten. Dabei weist das Produkt aber auch sinnvolle Erweiterungen zum Symantec-Klassiker auf, wie zum Beispiel eigene Buttons zum Anwählen/Deselektieren aller Dateien und zum Tauschen der Markierungsmerkmale.

SBUTILEXE SoftBlox Resource Monitor for Windows

Hauptaufgabe von SuperRM ist die Anzeige des noch für Windows zur Verfügung stehenden Speicherplatzes und der Systemressourcen. Aber ebenso wie die freien Kilobytes wird auch die aktuelle Zeit in einer, wie wir meinen, sehr schönen Digitalanzeige im 24-Stunden-Format dargestellt.

Das Fenster ist beliebig verschiebbar und kann auch ständig über den anderen Tasks liegen. Alle Anzeige-Elemente können farblich den

Wünschen des Users angepaßt werden. Und das Beste: SuperRM ist "Free EnjoyWare" (und kostet somit nichts). Da in der Dokumentation kein Copyrightvermerk ausfindig zu machen ist, kann es sogar in die Rubrik "Public Domain" eingeordnet werden.

SPEAKDRV.EXE - Lautsprechertreiber für Windows 3.1 (Freeware)

SPEAKER.DRV ist ein Treiber für den internen Lautsprecher, den Microsoft kostenlos zur Verfügung gestellt hat, damit auch Windows 3.1 - Anwender, die keine Soundkarte in ihren PC eingebaut haben, zumindest rudimentär in den Genuß der Soundmöglichkeiten von Windows kommen können. Konkret sind zumindest zwei Anwendungen als sinnvoll anzusehen:

1. Abspielen und Editieren von .WAV-Dateien mit dem Klangrecorder.
2. Zuordnung von Klängen zu bestimmten Systemereignissen, wie Start und Ende von Windows

Hierzu muß in der Systemsteuerung der Treiber SPEAKER.DRV (Sound driver for PC Speaker) gemäß Handbuch eingebunden werden. Unter dem Menüpunkt "Einrichtung" können gewisse Grundeinstellungen getroffen werden, wie die Wahl der Geschwindigkeit (Speed) und der Lautstärke (Volume). Außerdem kann die Wiedergabe auf beliebige Sekunden begrenzt werden (Seconds to limit playback). Unter "Test" können die aktuellen Parameter ausgetestet werden.

Die Option "Enable interrupts during playback" bitte nicht aktivieren, da sonst die Wiedergabe spürbar schlechter wird! Allerdings kann dann während des Abspielens z.B. der Mauszeiger nicht bewegt werden.

"Default" aktiviert die Standardeinstellungen, "Cancel" bricht das Setup ab.

Nach Anklicken von "OK" und Neustart von Windows müßte vor dem Öffnen des Programm-Managers eine Fanfare ertönen. In der Systemsteuerung sind nach Anklicken des "KLANG"-Icons aber auch andere Klänge den Ereignissen zuordbar. Im Prinzip können alle .WAV-Dateien eingebunden werden, obwohl die Wiedergabequalität über den Lautsprecher natürlich gewisse Grenzen setzt.

Beispiel für die Einbindung in die SYSTEM.INI (falls die Installation nicht automatisch klappt)

```
....
[drivers]
Wave=speaker.driv
MidiMapper=midimap.driv
Timer=timer.driv
....
```

Werte für 40Mhz 386. CPU-Speed bezeichnet das Sampling Delay. Für schnellere ; Rechner müssen größere Zahlen verwendet werden.

```
[speaker.driv]
CPU Speed=45
Volume=500
Version=774
Enhanced=1
Max seconds=10
Leave interrupts enabled=0
```

SPICKER2.EXE - Spicker 2.0

Spicker ist eine grafisch sehr gelungene Applikation, die Notizen, Termine u.a. verwaltet. Somit können die kleinen Zettelchen, die wohl bei vielen Anwendern irgendwo am Monitor kleben oder auf dem Schreibtisch (neudeutsch: Desktop) verstreut sind, getrost vergessen werden. Ebenfalls integriert ist eine komfortable Rückruf-Funktion.

Die Terminverwaltung erinnert auch an fällige Termine, wobei optional beliebige Dateien im WAV-Format als akustisches Signal eingebunden werden können.

TBWIN110.EXE - TB-WIN 1.10

TBWIN ist eine Startoberfläche für die sehr guten Antivirusprogramme TBSCAN / TBCLEAN der Firma Thunderbyte (DOS Diskette Ausgabe 8). Viele User ziehen die genannten Programme dem bekannten Viruscan vor, da sie u.a. schneller arbeiten und einen "intelligenteren" Suchalgorithmus besitzen. Wie bei "Viruscan for Windows" werden aber

leider auch hier nur die DOS-Scanner / Cleaner von der Windows-Oberfläche gestartet.

Das Programm benötigt neben den erwähnten Antivirentools auch die Borland-Bibliothek **BWCC.DLL**, die auf der ersten Windows-Diskette zu finden ist.

WBPRO1G.EXE - WinBüro Professional Version 1.0g

WinBüro Professional ist wohl die zur Zeit umfangreichste und zugleich ansprechenste Adressen-Verwaltung für Windows mit vielen Zusatzfunktionen.

Auffällig ist die Bedienung über Buttons ähnlich einem Kassettenrekorder, hier kann mit z.B. mit "Play", "Record", (schnellen) Vor-/Rücklauf komfortabel in den Datensätzen "geblättert" werden.

Alles in allem ein Programm, das Spaß macht!

Hinweis: Das Programm benutzt eine modifizierte Borland-Bibliothek namens **WINPRO.DLL**. Sollte sich im Windows-Verzeichnis bereits eine deutsche Version dieser Bibliothek befinden (z.B. **BWCCDEU.DLL** der letzten Ausgabe), so kann es zu Problemen kommen (z.B. Meldung: **WBPRO.DLL** nicht gefunden). Als Abhilfe kann **WBPRO.DLL** ggf. gelöscht und die Datei **WBPRO.INI**, wie in **PROBLEMS.DOC** beschrieben, entsprechend modifiziert werden.

WCONTR13.EXE - Windows Control Version 1.3

Viele Windows-Anwendungen speichern die Position und die Größe des aktuellen Fensters nicht und erscheinen beim Start immer an der gleichen Fensterposition und in der gleichen Fenstergröße und verdecken somit die übrigen Fenster. Hier schafft Windows Control Abhilfe, indem es in einer speziellen INI-Datei den Zustand des letzten geöffneten Tasks abspeichert. Ein kleines, aber sehr nützliches Programm!

Der Setup lässt sich auch vom Programm-Icon aus gut bewältigen, so daß Windows Control selbst nur sehr wenig Platz auf dem Desktop beansprucht.

WHOOPT20.EXE - Whoop it Up Version 2.0

Wer die internen Variationsmöglichkeiten von Windows 3.1 als zu gering empfindet, für den ist das Sharewareprogramm **WHOOPT IT UP** zu empfehlen, das ebenfalls notfalls mit dem Lautsprechertreiber zusammenarbeitet (natürlich werden alle Soundkarten unterstützt, die in Windows eingebunden werden können) und die Möglichkeiten der Klanguntermalung noch einmal bedeutend erweitert: Beispielsweise kann nun das Bewegen, Schließen und Vergrößern von Fenstern mit einem Klang begleitet werden. Außerdem können auch MIDI-Dateien (Endung .MID) als akustisches Signal verwendet werden.

WOL11.EXE - Wheel of Luck Version 1.1

Wheel of Luck ist eine grafisch ansprechende (und leider auch sehr viel Speicherplatz beanspruchende) PC-Version des bekannten Glücksrad-Spieles aus der "Fernsehwerbung". Besonders bei eingeschalteter Soundoption (das Klicken des Rades und die weiteren Klänge hören sich auch auf dem PC-Lautsprecher gut an, wenn der Soundtreiber - s.o. - eingebunden ist!) kommt viel Spielspaß für die ganze Familie auf. Ein Spiel auch für User, die ansonsten sich nur "ernsthafteren" Aufgaben am PC widmen.

Hinweise:

1. Das Game benötigt die Visual Basic - Laufzeitbibliothek **VBRUN200.DLL**. Diese ist auf dieser Diskette enthalten, bitte in das Windows-Hauptverzeichnis kopieren, damit auch andere Applikationen darauf Zugriff haben.

2. Die vom Spiel benutzten Begriffe können durch einfaches Editieren der ASCII-Datei **PHRASES.TXT** modifiziert und somit auch komplett eingedeutscht werden.

WSK303.EXE - WIN-Sock Version 3.03 (Freeware)

WinSock ist ein Freeware-Benchmarkprogramm, das die Geschwindigkeit der Grafikkarte (bzw. des installierten Grafiktreibers), des Speichers und der Festplatte mißt und in einer grafisch sehr schönen Anzeige darstellt.

Das ideale Programm, um Vergleiche zwischen verschiedenen Konfigurationen zu erhalten, zumal das vorliegende Produkt noch relativ unbekannt ist und sich somit den Manipulationsversuchen einiger Grafikkartenhersteller (vgl. Artikel in der DOS International 2/93, S. 28) erfolgreich entziehen kann.

Windows Highlights Disk 2

BWCC.DLL - Borland Windows Custom Control Library

BWCC.DLL ist die Standardbibliothek, die von vielen Programmen benötigt wird, die mit den Compilern der Firma Borland erzeugt wurden. Damit alle Programme auf die Bibliothek Zugriff haben, sollte sie in das Windows-Standardverzeichnis (meist C:\WINDOWS) oder in das SYSTEM-Verzeichnis kopiert werden.

Das vorliegende Archiv enthält ebenfalls die deutsche Version der Bibliothek (**BWCCDEU.DLL**), die aber offenbar mit der US-Version nicht ganz kompatibel ist.

CHANGER.EXE - Changer Version 1.0 (Freeware)

Changer erlaubt es, das Aussehen des Desktops (Farben, Hintergrundmuster, Wallpapers und Bildschirmschoner) automatisch bei jedem Start zu verändern, wobei das Programm sowohl eine festgesetzte als auch eine zufällige Auswahl treffen kann.

CLYSB170.EXE - Clysmic Icon Bar Version 1.70

ClysBar ist gedacht als Ersatz für den Programm-Manager. In einem schmalen Ikonmenü am rechten Bildschirmrand (Standard) stellt es einen schnellen Zugriff auf die wichtigsten Applikationen zur Verfügung. Somit findet das Programm neben (oder über/unter) jedem Fenster Platz. In klar gegliederten Untermenüs können ca. 200 Sinnbilder (.ICO, .EXE, .DLL) verwaltet werden.

Die Knöpfe werden in einem optisch ansprechenden 3D-Design dargestellt. Das Aussehen des Balkenmenüs kann auf vielfältige Weise angepaßt werden.

COMPOS20.EXE - CERES Composer Version 2.0

Der CERES Composer ist ein Stereo-Komponierprogramm für Adlib, Soundblaster, Soundblaster Pro und kompatible Soundkarten. Der CERES Composer arbeitet unter Windows 3.x oder OS/2 2.0. Der Real-Modus von Windows wird jedoch nicht unterstützt.

Zur Darstellung der Musikstücke wird die Sequenznotation (keine Noten) verwendet. Das erweist sich als besonders günstig, wenn man Effektinstrumente verwenden will, oder ein "unmögliches" (z. B. 9/4) Taktmaß benötigt.

Für die Speicherung der Musikstücke und Instrumentenbänke werden eigene, besonders speichersparende Formate verwendet. Diese Formate sind in der Entwicklerversion des CERES Composers offengelegt. Sie sind durch beigefügte Beispielprogramme und TURBO-PASCAL-UNITS sehr einfach zu programmieren.

Es existieren umfassende Blockoperationen (Ausschneiden, Kopieren, Einfügen, Ersetzen, Löschen, in Datei schreiben, aus Datei lesen, transponieren usw.) zur Bearbeitung der Musikstücke und Instrumentendateien. Diese sind sowohl für die aktuelle Stimme als auch für alle Stimmen vorhanden.

Ein Musikstück kann bis zu 32 Stimmen enthalten, auch wenn diese von der Soundkarte nicht unterstützt werden. In Verbindung damit, daß die Stimmen für Mono- und für Stereokarten unterschiedlichen Kanälen zugeordnet werden können, bietet der CERES Composer die Möglichkeit, ein und dasselbe Musikstück für Mono- und Stereokarten anders zu setzen.

Die Abspielprogramme und die zur Entwicklerversion mitgelieferten Routinen erkennen automatisch, ob das Musikstück auf einer Mono- oder Stereokarte abgespielt wird. Alle notwendigen Informationen stehen dann in der Musikdatei. Dadurch müssen nicht für jede Soundkarte andere Musikdateien verwendet werden. Die Musikstücke können ausgedruckt werden. Dabei ist es möglich, nur die aktuelle Stimme oder alle Stimmen auszudrucken.

Zur Shareware-, Standard- und zur Entwicklerversion werden Demonstrations-musikstücke mit ausgeliefert. Diese können auch mit Hilfe eines beigefügten Programmes von der DOS-Ebene abgespielt werden.

GWSWIN11.EXE - Graphic Workshop for Windows Version 1.1

GWS für Windows ist die Umsetzung der Shareware-Klassikers Graphic Workshop für den Betriebssystem-Aufsatz. Alle Vorteile von Windows fließen in das Programm ein, so daß der Bedienkomfort höher ist als in der DOS-Version.

Neben der Möglichkeit, Grafiken vieler Formate zu betrachten und zu konvertieren, bietet die vorliegende überarbeitete Version die Option, für jedes Bild einen sog. Thumbnail erzeugen zu lassen. Diese ca. 9K

großen Dateien bieten sozusagen eine Vorschau auf das eigentliche Bild und können statt der Dateiliste angezeigt werden. Dieses ist insbesondere für Dateiverwaltung (umbenennen, kopieren, löschen u.a.) eine große Hilfe.

Besonders erwähnenswert ist, daß das neue Bildformat .JPG in das Programm integriert wurde. Somit wird das platzsparende JPEG-Format erstmals von einem Standard-Grafikprogramm unterstützt.

MLK104.EXE - MicroLink Version 1.04

MicroLink ist ein DFÜ-Programm für Windows und ersetzt das völlig unzureichende integrierte Terminalprogramm TERMINAL.EXE.

Alle wichtigen Funktionen werden unterstützt. Die optische Aufmachung ist sehr ansprechend. Besonders erwähnenswert ist die Anzeige der Betriebszustände des Modems in einer simulierten LED-Anzeige (abschaltbar), was besonders bei internen Modemkarten eine große Hilfe darstellt.

Auch Highspeed-Modems werden unterstützt (getestet mit einem ZyXEL U-1496E), allerdings sollte auf langsameren Rechnern die Schnittstellengeschwindigkeit nicht höher als 19200 bps eingestellt werden.

Während die maximale Geschwindigkeit eher mit DOS-orientierten Terminalprogrammen zu erreichen ist (z.B. mit dem neuen Telix 3.20 oder mit Telemate 4.0), ist der Komfort ein wichtiges Kriterium für Windows-Anwender. Insbesondere können natürlich unter Windows zeitraubende Transfers oder auch das Anwählen einer Verbindung bequem im Hintergrund durchgeführt werden.

PCS10.EXE - Professional Capture Systems Version 1.0

Das Professional Capture System stellt jeweils ein Schnappschuß-Programm für Windows und für die DOS-Ebene zur Verfügung. Hierbei ist die DOS-Version weniger erwähnenswert, obwohl sie die Möglichkeit bietet, beliebige Textbildschirme (auch erweiterte Modi) in einer PCX-Datei einzufangen. Alle Standard-VGA-Modi werden vom DOS-Programm unterstützt, nicht jedoch die erweiterten Grafikauflösungen (SVGA).

Die Windows-Version trägt allerdings Attribut "professional" zu recht. Installation und Bedienung sind denkbar einfach. Auf Tastendruck kann ein beliebiger Bildschirmausschnitt, der ganze Bildschirm oder auch das aktive Fenster (auch mit aufgeklappten Menüs!) in optimaler Qualität "abfotografiert" werden. Die Ausgabe ist in das Clipboard (!), auf Drucker oder in eine Datei möglich, wobei viele Ausgabeformate unterstützt werden (.BMP, .PCX, .GIF u.a.).

PLGIN111.EXE - Plug in Version 1.11

Plug-In verbessert auf vielfältige Weise die Funktionalität des Programm-Managers, ohne ihn zu ersetzen. Das Programm klinkt sich gewissermaßen in den Programm-Manager ein und stellt innerhalb und außerhalb der Menüs neue Optionen zur Verfügung. Hier eine Auswahl der wichtigsten Neuerungen:

- Verbesserte Gruppen-Behandlung (Kopieren, (de-)aktivieren u.a.)
- Verbesserte Icon-Verwaltung

- Neue Cursorformen
- Sofortiges Speichern der Werte, Rücksetzen auf die Standardeinstellungen
- Umfangreiche System-Informationen
- Anzeige der System-Kapazität
- Alarm beim Unterschreiten eines definierten Mindestwertes der Ressourcen
- Automatische Installation von Applikationen
- Quick-Run-Menü: Schnelles Starten von Programmen
- Schneller Kommandozugriff mit History- / Browse-Funktion
- Kontextsensitive Hilfe
- Schnell Tasten für die wichtigsten Funktionen
- Automatische De-Installation

u.a.m.

Fazit: Wer viel mit dem Programm-Manager arbeitet, sollte sich Plug-In unbedingt anschauen. Es lohnt sich!

S&D24.EXE - Search & Destroy Version 2.4

Search & Destroy ist die Umsetzung des Spieles "Schiffe versenken" auf dem PC unter Windows. Die Grafik ist gut, bei installierter Soundkarte können die einzelnen Aktionen auch akustisch untermalt werden.

Zu Spielziel und Taktik braucht hier wohl nichts angemerkt werden, oder gibt es jemanden, der noch nie "Schiffe versenken" gespielt hat? Viel Spaß!

WRU311.EXE - WRU Version 3.11 (Freeware)

WRU (where are you) ist ein Freeware-Dateifinder unter Windows, der auch Dateien innerhalb von Archiven (.ARC, .ARJ, .LZH, .ZIP, .ZOO) findet.

Weitere Einstellmöglichkeiten sind die Dateigröße und das -Datum, wobei nach Gleichheit, kleiner, größer oder innerhalb eines Variationsspielraumes gesucht werden kann.

*** Irrtümer vorbehalten ***

Shareware Highlights Disk 1

- AMISSETUP:** Konfigurations-Programm für Rechner mit AMI-Bios (D)
- BACKINFO:** Wichtige Festplattendaten auf einer Rettungsdiskette speichern
- CONFIG:** Hardware / Software Diagnoseprogramm der Extraklasse (D)
- CRAZY CROSS:** Rubik's Cube zweidimensional, aber ebenso schwierig (VGA/D)
- DC:** Sehr guter Norton Commander - Clone (Public Domain!)
- DSKCV384:** Kurvendiskussion in Vollendung (D)
- FOCUS:** Neues Diskcopy-Programm mit VGA- oder Textoberfläche (D)
- JPEG:** Grafiken konvertieren (GIF -> JPG) mit Speicherplatzgewinn
- KEN'S LABYRINTH:** Super 3D-Labyrinthspiel mit Soundblaster-Option
- AVIEW:** Kleines aber leistungsstarkes Packer-Utility

Bezugsbedingungen

Einzelpreis pro Diskette S 140,-, im Abo: 6 Diskette um S 550,-, 12 Disketten S 950,-. Klaus Gloner, Dr. Hainstraße 8, 5620 Schwarzach, 06415/7215, FAX: 06415/7218.

Aktuelles vom Modem - Markt

Echte Neuerungen gibt es wenige in der DFÜ - eine aber steht vor der Tür: der lange erwartete neue Standard **V.fast** wurde von der CCITT verabschiedet. **V.fast** erlaubt Datenübertragungsgeschwindigkeiten von bis zu 28.800 bps! V.fast ist damit der schnellste, in einem herkömmlichen Telefonnetz mögliche Modus - was dann kommt, ist ISDN. Als einer der ersten Anbieter bringt die bekannte Firma ACTIVE Modems nach diesem neuen Standard auf den Markt.

Das Modell **144 UF plus** von ACTIVE soll besonders hervorgehoben werden: Eine effektive Übertragungsrate von 3500 Zeichen/Sekunde läßt einen leichten "Geschwindigkeitsrausch" aufkommen. Der Verbindungsaufbau erfolgt problemlos, auch die Verbindung mit anderen Modems läßt sich mit der praxisgerechten Fabrikeinstellung ohne Schwierigkeiten herstellen.

Die Modems arbeiten mit dem Hayes-Befehlssatz, der sich als Quasi-Standard etabliert hat und "verstehen" auch den erweiterten Hayes - Befehlssatz. Alle aktuellen CCITT- und MNP-Normen werden unterstützt.

Das externe Modell kommt in einem formschönen Aluminiumgehäuse, mit den üblichen Betriebsleuchten auf der Frontseite, die interne Steckkarte ist gleich mit einem UART 16550, dem gepufferten Baustein für Hochgeschwindigkeitsmodems, ausgestattet. An Software wird **Bitfax** und **Bitcom** für DOS (in Englisch) mitgeliefert.

Alle jetzt ausgelieferten ACTIVE-Modems sind für V.fast ausgelegt und können problemlos aufgerüstet werden. Das Modell 144 UF plus steht ab Ende November in Stückzahlen zur Verfügung.

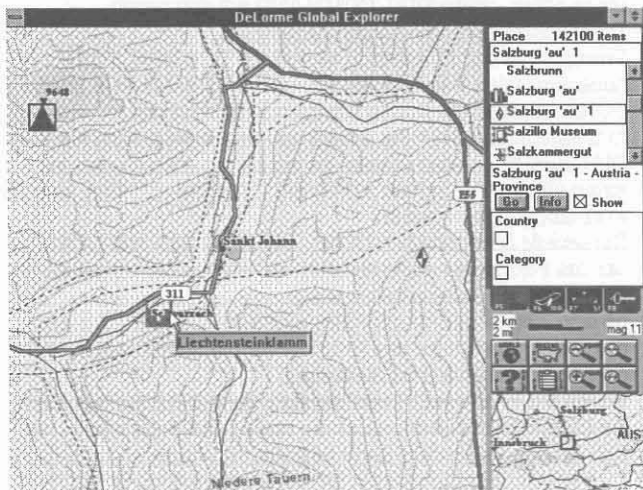
Für die **BTX-Anwender** unter den Lesern: Das Anwählen der BTX-Zugänge klappte sowohl bei den "alten" Zugängen wie auch bei den neuen, schnellen Zugängen einwandfrei.

Die Modems sind angesichts der gebotenen Ausstattung als sehr preiswert zu bezeichnen. Für Um- oder Aufsteiger gibt es bis Jahresende ein spezielles Programm, das Sie im Inseratenteil nachlesen können.

Klaus Gloner

GLOBAL EXPLORER, eine CD, die Spaß macht

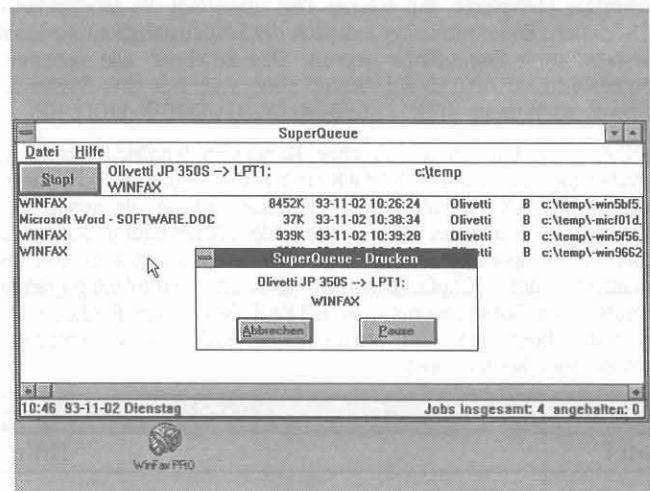
Ist der "GLOBAL EXPLORER" von DeLorme. Unglaublich, was die Leute an Daten zusammengetragen haben: 120.000 Orte, Flugverbindungen, Strassenkarten aus aller Welt usw. Hervorragend die ZOOM - Einrichtung, die es erlaubt, Vergrößerungen bis 2 km zu wählen. Schauen Sie sich das Bild an: Es zeigt einen Ausschnitt aus der Gegend, in der der Rezensent wohnt und arbeitet. Damit kann man schon etwas anfangen! Ganz billig ist die Silberscheibe zwar nicht, aber dafür ein wirklich brauchbares Stück "Computerfutter"!



Weiter aktuelle CDs finden Sie auf Seite 29.

SuperQueue, ein Ersatz für den Druckmanager von Windows

Der WINDOWS - Druckmanager gehört zweifellos nicht zu den Glanzlichtern von Windows. Einen Ersatz haben wir in SuperQueue 3.0 gefunden, den wir Ihnen nur wärmstens empfehlen können: SuperQueue lenkt die Druckausgabe eines beliebigen WINDOWS-Druckers in eine Spooldatei um und druckt dann im Hintergrund. Eine dreiseitige .doc - Datei blockiert jetzt die Arbeit am PC für 10 Sekunden, dann wird im Hintergrund gedruckt und Sie arbeiten im Vordergrund (ruckfrei) weiter! Mit SuperQueue können Sie aber noch mehr: Zeitversetzt drucken, Druckjobs anhalten, entfernen, hinzufügen. Auch Netzwerkdrucker werden unterstützt! Der Preis für SuperQueue ist sehr günstig. Für Vieldrucker und/oder Ungeduldige ist SuperQueue ein Muß!



Virenschutz total

verspricht ein Produkt, das bis dato in Österreich noch nicht erhältlich ist: **PC-cillin** aus dem Hause **TREND Micro Devices**.

Der Hersteller, in seinen bisherigen Märkten alles andere als ein Unbekannter (die uns vorliegende amerikanische Kundenliste liest sich wie eine Aufstellung der größten amerikanischen Unternehmen und Banken, angeführt vom Chemieriesen DuPont, der **PC-cillin** 40000 mal im Einsatz hat. In Japan hat Trend Micro Devices nach eigenen Angaben einen Marktanteil von 95 %), bietet ein Produkt mit überzeugenden Leistungsmerkmalen und einfacher Bedienbarkeit.

Neben der Suchfunktion bietet **PC-cillin** auch eine vierstufige Reinigungsfunktion an. Auch ein Wächtermodul ist im Lieferumfang enthalten, das mit Tastendruck ein- und ausschaltbar ist (Diese Funktion wird jeder schätzen, der z.B. mit Nortons Speedisk oder Central Points Compress arbeitet).

Zwei interessante Zusätze runden das Produkt ab: Zum einen ist im Paket ein Schalter enthalten, der auf den parallelen Druckerausgang aufgesteckt wird und zwei Funktionen in sich vereint: Zum einen werden wichtige Daten wie die Partitionstabelle und der Bootsektor der Festplatte in einem Eprom gespeichert, zum anderen ist ein Schutz vor Bootsektor-Viren schon vor dem Abarbeiten der config.sys - Datei gegeben.

Der andere Zusatz ist eine Suchmethode, die Trend als "DeepScan" bezeichnet. DeepScan prüft Programme auf verdächtige Befehlsfolgen z.B. dem Verändern des Bootsektors.

Das Programm hat uns durch seine Schnelligkeit und Zuverlässigkeit beim Scannen sehr gut gefallen: Die Tests mit "Standardviren" wie Stoned, Tequila, Michelangelo, Tremor wurden einwandfrei bestanden.

Das übersichtliche Handbuch (derzeit noch in englisch, eine deutsche Ausgabe ist in Arbeit) bietet nicht nur eine Installationsanleitung, sondern gibt dem Interessierten auch weiterführende Hinweise zu Computerviren.

Was nützt das beste Antiviren-Programm, wenn es veraltet ist? **PC-Cillin** geht auch hier einen eigenen, kundenfreundlichen Weg: Registrierte Benutzer können im ersten Jahr ihre Updates kostenlos aus einer Mailbox beziehen! Wer kein Modem hat, erhält seine Updates gegen einen mäßigen Kostenersatz per Diskette zugesandt.

Ein "Schmankerl" haben wir für Leser der **PC-NEWS** bereit: Sie erhalten **WinRx**, die WINDOWS-Version des Scanners und Cleaners zu einem besonders günstigen Vorzugspreis!

□

Bestellung

Ich bin Leser der **PC-NEWS** und bestelle hiermit zum Vorzugspreis

- | | | |
|------------------------------------|---|----------------------------------|
| Stück PC-cillin 2 in 1 | bestehend aus: Stecker, Wächter, Scan- und Reinigungsmodul, Windows - Version | à S 1.955.-- (Empf. VK 2.835.--) |
| Stück WinRx | Scan- und Reinigungsprogramm für Windows | à S 650.-- (Empf. VK 1.250.--) |
- Alle Preise inklusive Mehrwertsteuer - Angebot gültig bis 15. 12. 1993

Meine Adresse:

Ich zahle

- ☐ mit beigelegtem Scheck und spare die Versandkosten
☐ gegen Rechnung nach Lieferung (zuzüglich S 50.-- Porto und Verpackung)

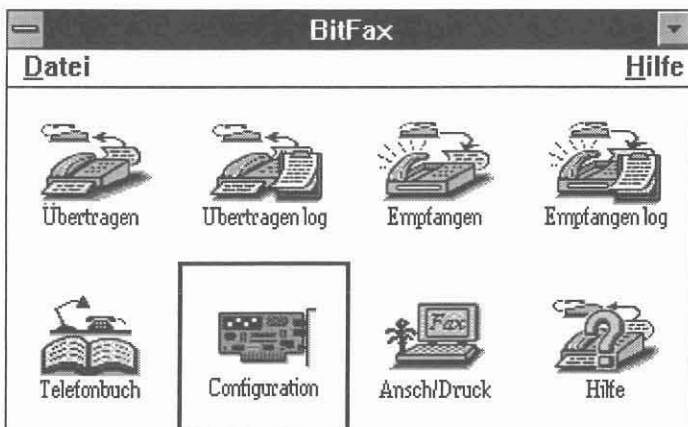
Datum und Unterschrift:

FAX-Pocket-Modem 14000 bit/s

Für alle, die einen Laptop besitzen und auch unterwegs, unabhängig von der Steckdose on-line sein wollen, ist dieses Fax-Modem gedacht. Mit 4 1.5-Volt-Batterien ausgerüstet, ist es ein praktisch vollwertiger Ersatz für Standgeräte. Einige Leuchtdioden zeigen die wichtigsten Schnittstellensignale an. Für den stationären Betrieb wird ein Steckernetzteil mitgeliefert.

Wie wurde getestet? Echte Modemvergleichstest erfordern einige hochwertige Meßgeräte. Ein solcher Test übersteigt die Möglichkeiten der Redaktion. Es wurde daher lediglich die Betriebsfähigkeit als Ersatz eines bewährten Standgeräts geprüft. Das Ergebnis: alle getesteten Datendienste funktionierten auf Anhieb ohne auch nur eine Einstellung korrigieren zu müssen. (BTX, TELEBOX, COMPUSERVE, FIDO, HP).

Mitgeliefert werden ein ordentliches Handbuch (englisch) sowie die Datenübertragungssoftware BITFAX und BITCOM (deutsch), beide für Windows. BITFAX installiert sich in Senderichtung als zusätzlicher Druckertreiber und kann daher aus jeder Anwendung aufgerufen werden. In Empfangsrichtung wird bei eingeschaltetem Rechner automatisch der Empfangsmodul aktiviert. Protokollprogramme überprüfen den Datentransport über BITFAX. Mit einem Bildbetrachter kann man abgehende und ankommende FAXe in verschiedenen Vergrößerungen kontrollieren.



Zur Abrundung der Informationen über dieses Modem finden Sie im Anschluß:

Informationen für die Benutzung von FaxModems

Walter Fangl, SYWA

Ihr FaxModem ist für den Betrieb an der österreichischen Telefonleitung nicht zugelassen. Die Verwendung erfolgt auf Ihr eigenes Risiko.

Alle FaxModems unterstützen den Standard-Hayes-Kommandosatz und werden mit Software ausgeliefert, die amerikanische Telefonsysteme als Standard annehmen. Die Standard-Installation (default values) der Software-Pakete stellt also auf TONWAHL und amerikanischen Wahlton ein. Der Wahlton ist der Ton oder die Tonfolge aufgrund der das FaxModem die Freileitung erkennt. Fehlermeldung bei falscher Einstellung: NO DIAL TONE

Bezugsquelle: sywa, Getzgasse 9, 1180 Wien, TEL: (0222)-470-70-05, FAX: (0222)-470-70-06. Bestellbezeichnung: Pocket FaxModem 1414P, Preis: 4.260,-.

Einstellungen aus der nachfolgenden Liste wurden bei 99,9% der Reklamationen nicht korrekt gemacht:

- Bei unseren älteren Telefonen ist die Pulswahl (= PULSE DIAL) einzustellen - siehe DIAL COMMAND bei den neuen Digitaltelefonen ist dies nicht umzustellen.
- Prinzipiell ist das Kommando x4 = WAIT FOR DIAL TONE (amerikanischer Wahlton) auf x1 = DO NOT WAIT FOR DIAL TONE umzustellen. Dieses Kommando kann in der Initialisierungszeile (= init string) und im Vorwahl-Kommando (= DIAL PREFIX) enthalten sein. Ist ein Eingabefeld für den init string vorhanden und mit ATX4 (oder nicht) beschrieben, so sollte ATX1 eingetragen werden.
- Die serielle Schnittstelle = COM1, COM2, COM3 oder COM4, an der das FaxModem angeschlossen ist, muß in der mitgelieferten Software richtig eingestellt werden.
- Ist eine Netzwerkkarte und/oder eine Soundkarte im Computer eingebaut, kann diese den IRQ3 der für die Schnittstellen COM2 oder COM4 reserviert ist blockieren.
- Wurde MS Windows 3.1 vor Installation einer FaxModem-Karte am Computer installiert, so ist die neue COM-Schnittstelle, die sich auf der FaxModem-Karte befindet, in Windows zu aktivieren. Dies erfolgt über Menüpunkt 'Systemsteuerungen', weiter bei 'Anschlüsse', dort die neue Schnittstelle und dafür 'weitere Anschlüsse' wählen. Für die gewählte Schnittstelle ist nun die Adresse und der IRQ einzutragen: COM1 = 3F8 IRQ4, COM2 = 2F8 IRQ3, COM3 = 3E8 IRQ4, COM4 = 2E8 IRQ3; Oder man definiert eine vorhandene COM-Schnittstelle auf eine andere um - z.B.: COM2 wird zu COM3, und das FaxModem wird auf COM2 eingestellt. Dazu sind die entsprechenden Jumper oder DIP-Schalter gemäß Bedienungsanleitungen der Schnittstellenkarte (= I/O-Karte) und der FaxModem-Karte zu ändern.
Warnung: COM1 + COM3 verwenden IRQ4, COM2 + COM4 verwenden IRQ3 - das bedeutet, daß das gleichzeitige Verwenden von COM1 + COM3 oder COM2 + COM4 zu Problemen führt!
- Manche COM-Schnittstellen von Standard-Multifunktionskarten sind für die neuen, schnellen externen FaxModems zu langsam. Neue Schnittstellen-Chips wie der 16C550 haben damit keine Probleme mehr.
- Bei mit Batterien betriebenen Pocket-FaxModems prüfen Sie, ob alle Batterien auch wirklich richtig eingelegt sind oder nicht etwa die Batterien leer sind!
- Bei FaxModems sind die typischen amerikanischen RJ1 1-Kontakte zum Anschluß an die Telefonleitung zu finden. Meistens sind 2 Kontakte, 1 x LINE (oder auch WALL) und 1 x PHONE zu sehen. Um dies an unserer TST-Buchse richtig anzuschließen, ist ein spezielles Kabel notwendig.
Die TST-Dose hat 3 Buchsen für die typischen 10-pol Flachstecker: Fax, Telefon, Anrufbeantworter. Diese 3 Buchsen sind parallel geschaltet, solange kein Stecker in einer Buchse steckt und liegen in der Reihung: Fax (links) = 1 (= Beginn), Anrufbeantworter (rechts) = 2, Telefon (Mitte) = 3 (Ende).
Wird ein Flachstecker eingesteckt, unterbricht das die Leitung zum nachfolgenden Kontakt. Deshalb ist es auch notwendig, die 2. Leitung (PHONE) am FaxModem anzuschließen, damit das Telefon und/oder Anrufbeantworter normal funktionieren, wenn diese an der TST-Dose angeschlossen sind.
- Internes Schnittstellenkabel (im Computer!) zur I/O-Karte oder externes serielles Kabel überprüfen.
- Soll das Programm CIM (CompuServe Informations-Manager) verwendet werden, muß bei FAX-Modems zuerst der selbsttätige Faxempfang ausgeschaltet werden.

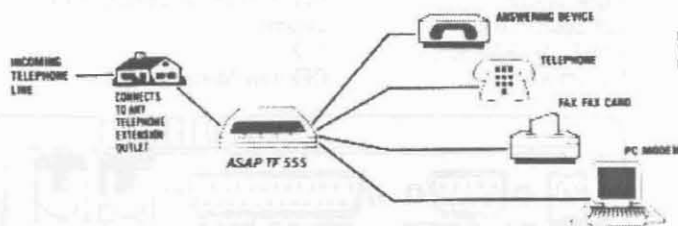
Für häufigen Empfang von Faxen mit dem FaxModem ist ein automatischer Fax-Umschalter sehr zu empfehlen. Damit wird bei jedem eingehenden Telefonat geprüft, ob es ein Fax oder ein normales Gespräch ist und entsprechend zum Fax (FaxModem) oder zum Telefon verbunden. □

TF555 - automatischer Fax-Umschalter

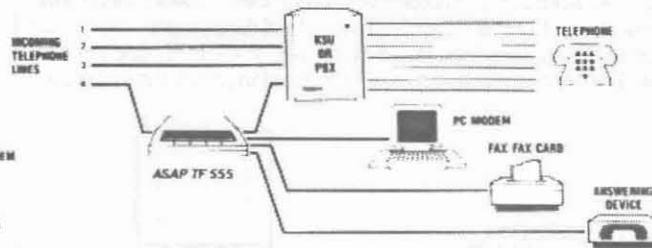
Walter Fangl, SYWA

Der TF555 schaltet ein Faxgerät, einen Anrufbeantworter, ein Modem und ein Telefon an eine Telefonleitung.

Home Office Solution



Large Office Solution



TF555 arbeitet bei **Puls- und Tonwahlsystem**. Ein anrufendes Faxgerät sendet einen speziellen Pfeifton - das CNG-Signal. Bei einem Anruf hebt TF555 ab und prüft, ob der Anrufer das CNG-Signal sendet. Falls ja, schaltet TF555 zum Faxgerät, andernfalls zum Telefonapparat. Erhält man ein Fax zum Telefonapparat weil ein Anrufer manuell gewählt hat oder das anrufende Faxgerät kein CNG-Signal sendet (altes Gerät), so kann man zum Faxgerät verbinden.

TF555 eignet sich besonders für private Anwender, kleine und mittlere Betriebe, bei einer einzelnen Telefonleitung oder bei einer Leitung innerhalb einer Seriennummer (man nennt z.B. die letzte Nummer der Serie als Fax-Nummer).

Einfache Installation: TF555 hat serienmäßig amerikanische RJ11-Kontakte - LINE, FAX, PHONE, TAD und AUX.

Für alle diese Kontakte gibt es passende Adapter-Kabel/Stecker.

- Das Fax wird von der Dose abgesteckt und mit dem passenden Adapter an TF555 angeschlossen. Wird eine Fax-Karte (USA/Taiwan/HongKong etc.) mit RJ11-Kontakten verwendet, so ist nur ein RJ11-Kabel zum direkten Anschluß an TF555 notwendig.
- TF555 wird bei den Kontakten PHONE und LINE mit einem besonderen Adapter-Kabel an der Dose angeschlossen, wo vorher das Faxgerät angeschlossen war.
- Verwenden Sie einen Anrufbeantworter oder ein Schnurlostelefon ist damit nichts zu ändern.
- Wollen Sie Fax, Telefon, Anrufbeantworter und Schnurlostelefon verwenden und es fehlt Ihnen ein Anschluß, so können Sie den Anrufbeantworter auch am TF555 (TAD-Kontakt) anschließen. Die nun freie Buchse der Dose können Sie dann für das Schnurlostelefon verwenden.

VERWENDEN SIE NUR POSTGENEHMIGTE SCHNURLOSTELEFONE !

- Am Modem-Kontakt AUX kann man ein Modem anschließen um den Zugang von Außen zu Ihrem Computer zu ermöglichen. Das Umschalten zum Modem erfolgt im Reverse-Modus oder durch Nachwahl eines Codes.
- Verwenden Sie ein Faxgerät ohne eigene Wahlvorrichtung, so benötigen Sie ein zusätzliches Telefon (parallel am Faxgerät) zum Wählen. Der neue Fax-Umschalter trennt ja die Leitung zwischen Fax und Telefon, d.h. diese sind nicht mehr parallel angeschlossen wie vorher. Bei Stromausfall läutet TF555 beim Telefon und Anrufbeantworter. TF555 hat einige Schalter, die ein- oder ausschaltbar sind und Funktionen die durch Eingabe eines Codes mit einem angeschlossenen Telefon (nur mit Tonwahl!) einstellbar (programmierbar) sind:

Schalter:	Bezeichnung	standard
[0]	Nachtschaltung	an
[1]	'freie Leitung'-Signal	aus
[2]	Nebenstellen-Erkennung	an
[3]	Anrufbeantworter zu Fax	aus
[4]	Pulswahl-Erkennung	an
[5]	autom. Faxeinschaltung	an
[6]	autom. Faxerkennung	an
[7]	Reverse-Modemerkenkung	aus

Funktion	Bezeichnung	standard	möglich:
[2]	Fax-Umschaltcode	11	0-9999
[3]	Modem-Umschaltcode	22	0-9999
[4]	Läutanzahl vor Abheben	1	1-99
[5]	Läutanzahl Telefon/Anrufer	6	0-99
[6]	Läutanzahl Fax/Modem	3	0-99
[8]	Läutsignal für Anrufer	1	1-7
[9][9][9]	Rücksetzen aller Werte auf Standard-Wert		

Programmier-Schritte:

- Telefonleitung (Line-In) vom TF555 abstecken
- Telefonhörer abheben
- zwischen dem Klicken innerhalb 1 Sekunde [#] wählen, Klicken hört auf

für Schalter:

- [0] oder [1] tippen: [1] = an, [0] = aus
- Schalter-Nummer tippen: [0] - [7]

für Funktion:

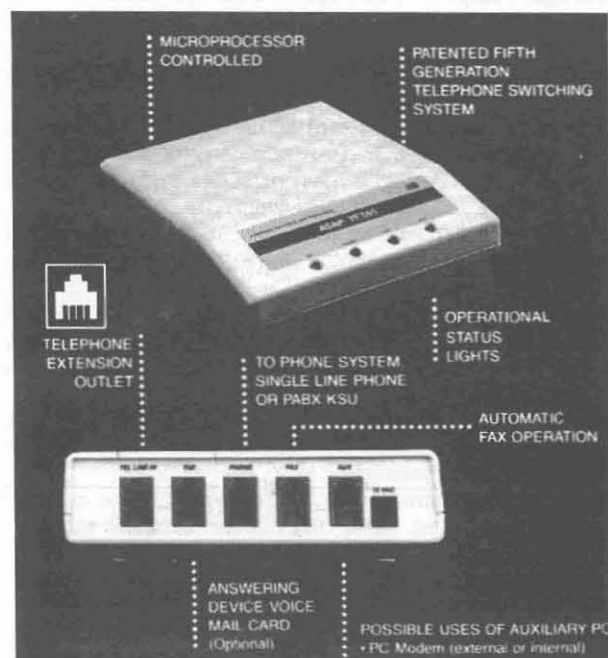
- Funktions-Nummer tippen: [2] - [8] oder [9][9][9]
- mögliche Ziffer tippen

danach bei beiden:

- Eingabe bestätigen = [*] tippen
- danach zu Punkt 4 für weitere Änderung oder

nochmals mit Taste [*] Eingaben beenden.

Sind drei kurze Töne zu hören, ist die Eingabe akzeptiert worden. Wir empfehlen, die Standard-Werte zu belassen, nur Code 8 sollte auf Typ 6 = US Style 2 gestellt werden (sehr ähnlich zu unserem Läutsignal).

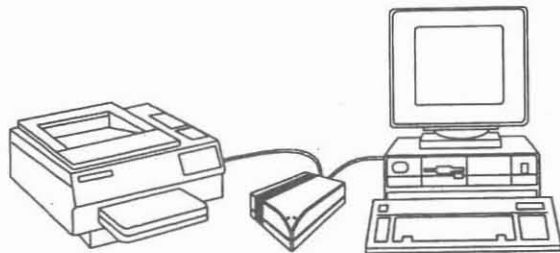


weiter auf der nächsten Seite mitte

FaxPak • FAXempfang ohne FAXmodem

Walter Fagl, SYWA

Viele Benutzer eines FAX-Modems bemängeln, daß zwar das Senden problemlos möglich ist, man für den FAXempfang den Rechner aber ständig eingeschaltet haben muß. Viele Besitzer eines hekömmlichen Thermopapier-FAX-Geräts beklagen die mangelhafte Papierqualität und den Preis von Normalpapierfaxgeräten. Dafür bietet dieses Gerät eine ungewöhnliche Lösung an: Der FAX-Empfänger in der Verbindungsleitung zwischen PC und Drucker. Vorteil: PC und Drucker bleiben im Prinzip ausgeschaltet, der FAX-Empfänger ist auf Empfang:



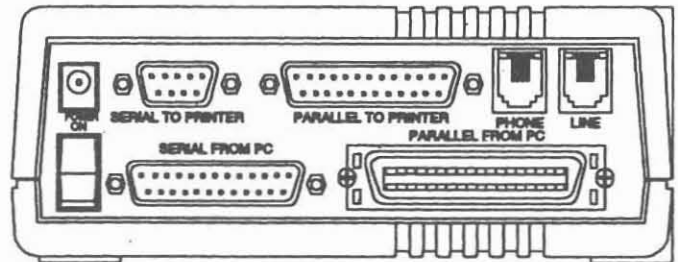
Wird der Drucker eingeschaltet, überträgt der FAX-Empfänger die Daten zum Drucker. Mit welchen Druckern gehts?

Drucker

- * HP II,III,IV, Deskjet
- * Canon Laser (CaPSL), Bubble Jet
- * Epson 9pin, 24pin
- * IBM 9pin, 24pin

Wichtige Optionen

Drucker-Auflösung:	150x150, 300x300 dpi
Papierformate:	Letter, Legal, A4
Identifikation:	max. 20 Zeichen
Papierzufuhr:	Upper, Manual, Lower, Deck
Fax/Sprachumschalter:	ja/nein
Anzahl der FAX-Kopien:	1..9
Lautsprecher:	OFF/low, Medium, High



Installation

Da das zusätzlich erforderliche Druckerkabel mitgeliefert wird, ist die Installation weiter kein Problem. Was man aber zusätzlich braucht, ist ein Tonwahl-Fernsprechapparat zur Einstellung der diversen Optionen des FaxPak. □

Bezugsquelle: sywa, Getzgasse 9, 1180 Wien, TEL: (0222)-470-70-05, FAX: (0222)-470-70-06. Bestellbezeichnung: FaxPak/FaxBuffer, Preis: 4.470,-.

Fortsetzung von der vorigen Seite

Einstellungen

alle Schalter und Funktionen sind nur mit einem Tonwahl-Telefon programmierbar!

Schalter

[0] Nachtschaltung (an) Nach 6x Läuten des Telefons ohne Abheben (die Läutanzahl ist mit Funktions-Programmierung von 0-99 programmierbar) schaltet TF555 auf Fax um. Alle nachfolgenden Anrufe (zum Telefon) läuten nur 2x vor dem Umschalten. Nach dem ersten Gespräch vom Telefon wird zum Grundzustand geschaltet.

[1] 'freie Leitung'-Signal (aus) Hebt man den Hörer ab und die Leitung zum Wählen ist besetzt, weil gerade ein Fax gesendet/empfangen wird, so legt man einfach auf. Ist die Leitung wieder frei, klingelt kurz das Telefon.

[2] Nebenstellen-Erkennung (an) Ist die eingehende Telefonleitung mit TF555 und parallel mit einem zusätzlichen Telefon verbunden, so erkennt TF555 das Abheben des parallelen Telefons bei einem eingehenden Anruf und beendet das Läuten.

[3] Anrufbeantworter zu Fax (aus) Ist ein Anrufbeantworter aktiv, werden alle Anrufe nach der Ansage zum Fax umgeschaltet. Man sollte dafür eine Info am Band aufsprechen.

[4] Pulswahl-Erkennung (an) Damit ist TF555 für das alte (Pulswahl) und das neue (Tonwahl) Telefonsystem gerüstet. Bei Pulswahl-Telefonen ist der Umschalt-Code fix vorgegeben und zwar zum Fax '3' und für das Modem '4'.

[5] automat. Faxeinschaltung (an) Wird die Läutanzahl beim Telefon (s. Funktions-Code [5]) überschritten, wird der Anruf automatisch zum Fax geschaltet.

[6] automat. Faxerkennung (an) Will man Faxe nur von autorisierten Personen erhalten, so ist dieser Schalter auf 'aus' zu stellen. Man erreicht das Faxgerät nur mehr durch Nachwahl der Fax-Umschaltcodes nach der Telefonnummer. (Siehe Funktion [2] - Fax-Umschaltcode)

[7] Reverse-Modem-Erkennung (aus) Damit kann man anstatt der Nachwahl-Option (siehe Funktions-Code [3]) mit einem Modem das die Reverse-Funktion unterstützt ein BBS-System aufbauen. Nach der Anwahl zum BBS wird mit [R] der Reverse-Mode aktiviert - z.B.: ATDT3102032R

Funktionen:

[2] Fax-Umschaltcode: Damit kann man Anrufe zum Fax umschalten. Anrufer die den Code kennen können zum Fax durchwählen. Siehe Schalter [6].

[3] Modem-Umschaltcode: Zur Durchwahl zum Modem von außen. In Verbindung mit BBS-. Betrieb. Siehe auch Schalter [7]. Mögliche Anwahl mit Modem von Außen: z.B.: ATDT3102032,,22 oder ATDP für Pulswahl. Der Beistrich steht für 2 Sekunden Zeitverzögerung zum Abheben von TF555.

[4] Läutanzahl vor Abheben: Läutanzahl bevor TF555 bei einem eingehenden Anruf die Leitung. Eine Änderung ist nicht sinnvoll!

[5] Läutanzahl Telefon/Anrufbeantw.: Läutanzahl von TF555 zum Telefon+Anrufbeantworter. Bei Schalter [5] = an wird danach automatisch zum Fax verbunden, andernfalls der Anruf unterbrochen.

[6] Läutanzahl Fax/Modem: Hebt das Fax/Modem mit der eingestellten Läutanzahl nicht ab, dann muß man den Wert erhöhen.

[8] - Läutsignal für Anrufer: Hebt TF555 bei einem eingehenden Anruf ab, hört der Anrufer ein Standardwert = 1, einstellbar von 1-7. Läutsignal. Dieses Läuten kann je nach Land in Frequenz und Dauer eingestellt werden:

- | | |
|-------------------|--|
| [1] = U.S. Typ 1 | 2 Sekunden Läuten, 4 Sekunden Pause - Wechselton |
| [2] = Belgien | 1 Sekunde Läuten, 3 Sekunden Pause |
| [3] = Europa gen. | 0,4 Sek. Läuten, 0,2 Sek. Pause, 0,4 Sek. Läuten, 2 Sek. Pause |
| [4] = Spezial | 1 Sekunde Läuten, 5 Sekunden Pause |
| [5] = England | 1,5 Sekunden Läuten, 1,5 Sekunden Pause |
| [6] = U.S. Typ 2 | 2 Sekunden Läuten, 4 Sekunden Pause - tiefer Ton |
| [7] = U.S. Typ 3 | 2 Sekunden Läuten, 4 Sekunden Pause - hoher Ton |

Wir empfehlen den Wert [6] - das Läutsignal ist unserem am ähnlichsten. □

Bezugsquelle: sywa, Getzgasse 9, 1180 Wien, TEL: (0222)-470-70-05, FAX: (0222)-470-70-06. Bestellbezeichnung: TF555, Preis: 2.970,-.

LAPTOP STATT SCHLEPPTOP

Olivetti QUADERNO PC-XT-20

Mag. Dieter KADAN, Kollegium Kalksburg

Eigentlich war ich schon immer an so einem Ding interessiert: Eine Art Minicomputer für unterwegs, den man wirklich IMMER dabei hat. Eine Art elektronischer Organizer für mein persönliches Timemanagement und ein Notizbuch für die vielen kleinen Ideen (z.B. zur Unterrichtsvorbereitung). Bislang hatte ich für Notizen entweder zuwenig Platz in meinem Taschenkalender oder Zettel verwendet, die ich dann nicht wieder finden konnte, wenn ich sie gerade benötigte.

Und dann ist da noch ein Punkt: Ich habe zu Hause am Schreibtisch zwar einen PC stehen, aber Ideen habe ich immer dann, wenn ich gerade nicht davor sitze. Wem - außer vielleicht einem Schriftsteller - geht es nicht so oder so ähnlich?: Wenn ich vor meiner Textverarbeitung sitze, nimmt meine Kreativität und Schreiblust rapide ab. Fern vom PC im Arbeitszimmer entstehen dann scheinbar mühelos Konzepte, Entwürfe, Texte aller Art, die erst wieder in den PC getippt werden müssen.

Kurzum: Ich habe für mich eine passende Lösung gefunden. Eigentlich schieden viele Notebooks schon allein wegen Gewicht und Größe aus. Oder schleppen Sie gerne ein drei Kilogramm schweres Gerät in A4-Größe mit sich herum? Im Herbst 92 gab es eine Club-Einkaufs-Aktion für den SHARP 3100. Dieser XT-Palmtop kostete damals ca. 10000.- (heute als Auslaufmodell 3000.-). Er hatte bereits eine ergonomische Tastatur (kein Knopferl) und ließ sich als echter 500 Gramm-Palmtop im 60% vom A5-Format überallhin mitnehmen. Seine Nachteile waren der extrem kleine Bildschirm (Zeichen ohne Unterlängen), der batteriefreie Betriebsmodus und die teuren Speicherkarten.

"Mein" Quaderno hingegen hat 1kg im A5-Format und eine eingebaute 20MB-Platte, Betriebssystem ROM-DOS 5.0. Das interessanteste ist aber sein Preis: Als Auslaufmodell verschleudert die Firma ESCOM diesen XT-Kompatiblen (mit NEC V30HL-Prozessor) um 5500.- ÖS(!). Ich habe das geschrieben, um alle Clubmitglieder auf das gute Preis-Leistungsverhältnis aufmerksam zu machen. Und das angenehme ist, daß der Datentransfer zum PC mit dem mitgelieferten Kabel bei mir wirklich auf Anhieb funktioniert hat (seriell, parallel auch möglich, aber nicht getestet). Im Lieferumfang ist auch das Netzgerät, das Akkupaket und eine Staubschutztasche enthalten. Wer an diesem von Olivetti "Palmtop PC-XT-20 QUADERNO" genannten Spielzeug interessiert ist, sollte hier zu lesen aufhören und gleich zur Bestellung - an der ich übrigens keinen Groschen verdiene - schreiben; wer weiß, wie lange es dieses Angebot noch gibt.

(Übrigens: Der neue QUADERNO33 ist genau so fein: A5, 1,3kg, 386er, 60MB-Festplatte, VGA, Trackball, Windows3.1. Gesehen im Mediamarkt um 24000.-). >>>

Von Waldgeistern und goldenen Blättern

Wolfgang Scharl & Sons, N, TGM DSK-373

Ein Computerspiel, das Unterhaltung und Spannung bietet und dabei ganz ohne Mord und Totschlag auskommt, hat das Kooperationsabkommen "Forst-Platte-Papier" entwickelt. Bei diesem Spiel sind mit der Unterstützung freundlicher Waldgeister eine Vielzahl von Aufgaben zu lösen (siehe Bild unten). Es gilt sinnvolle Waldpflege zu betreiben, Papier und Spanplatten zu erzeugen und auf der Skipiste Umweltbewußtsein zu zeigen. Zwischendurch ist noch ein Labyrinth zu durchqueren.

Bei allen Aufgaben ist vor allem Grips sowie das nötige Wissen über die Zusammenhänge gefragt. Mit Gewalt oder schnellen Reaktionen ist bei den Waldgeistern nichts zu holen. Wenn die Gespräche mit den Bäumen, Geistern und sonstigen Zeitgenossen nicht weiterhelfen, so gibt es immer ein kluges Buch (F1) und eine Landkarte und darüber hinaus eine Spielanleitung mit Tips und Ratschlägen. Als Lohn für all die Mühe winkt die Aufnahme in die ehrenvolle Runde der Waldgeister sowie eine Auszeichnung in Form eines goldenen Blattes.

Ganz nebenbei und unaufdringlich wird eine Menge Wissen über die Bedeutung des Waldes und der Forstwirtschaft, über industrielle Produktion und Umweltschutz vermittelt. Vom Unterrichtsministerium wurde sogar der Einsatz des Computerspiels im Unterricht erlaubt.

Raubkopien sind leider nicht möglich, denn kopieren ist ausdrücklich erlaubt. Das Spiel kann auch kostenlos vom Kooperationsabkommen FPP, Gumpendorfer Str. 6, 1061 Wien, Frau Dipl. Ing. Herzog bezogen werden. Die Installation ist einfach, die Originaldiskette ist viresgeschützt. Als Voraussetzung ist ein IBM-kompatibler PC mit 3MB Platz auf der Harddisk und mindestens EGA Grafik erforderlich.

Für Kinder ab etwa 10 Jahren ist das Spiel bereits problemlos zu bewältigen. Es empfiehlt sich, den Spielstand mit der Datei "SPIELSTA.DAT" ab und zu zu sichern und geben Sie acht, daß Sie im dunklen Wald nicht aber zu viel sollte auch nicht verraten werden. □

>>> Auch wer nur einen leistungsfähigen "Taschenrechner" sucht, ist mit dem Quaderno gut bedient: Das Programm DERIVE - wer kennt es nicht - findet auf der Festplatte mühelos Platz. Mit ein bißchen DOS-Kenntnissen kann man den QUADERNO dem eigenen Bedarf anpassen, z.B. die Organizer-Funktionen abschalten, ein Softwarepaket installieren, u.s.w.. Daß man ihn auch als Diktiergerät verwenden kann, macht ihn neben Größe und Gewicht einzigartig. Diesen, meinen Beitrag habe ich übrigens auch nicht am Schreibtisch sitzend getippt, sondern abends im Bett mit den Füßen unter der warmen Decke und meinem Quaderno am Schoß. Ich bin gespannt, wie lange meine Frau das akzeptiert. □



An alle Freunde der ADIM!

Martin Weissenböck, ADIM

Wenn Sie die **ADIM** noch nicht kennen, wollen wir uns kurz vorstellen:

Die **Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik** (kurz: **ADIM**) wurde mit dem Ziel gegründet, **preiswerte und qualitativ gute Medien** für alle Formen des Unterrichts in den Bereichen "Informatik" oder "Elektronische Datenverarbeitung" oder ähnlicher Unterrichtsgegenstände bereit zu stellen. Die meisten Mitarbeiter der **ADIM** sind Lehrer, die in ihrer Freizeit an Skripten und anderen Hilfsmitteln arbeiten. (Genauere Angaben über die **ADIM** finden Sie am Bestellblatt weiter hinten, sowie im Impressum.

Schon vor mehr als 10 Jahren wurden - vorerst nur für kleinere Gruppen - Unterlagen über den Apple II, über Hardware, über Studienvorschriften und Lehrpläne und ähnliche Themen zusammengestellt.

Einige Skripten - wie etwa eine Bedienungsanleitung für den Apple II - sind nicht mehr interessant. Andere sind seit Jahren immer wieder gefragt: so zum Beispiel die Unterlagen über Turbo-Pascal oder C. Natürlich werden die Skripten immer wieder überarbeitet und an die aktuellen Versionen der Software angepaßt. Aber auch die Erfahrungen beim Einsatz im Unterricht führen dazu, daß immer wieder Beispiele angepaßt oder Teile verändert werden. Deshalb werden die Skripten in der Regel auch einmal pro Jahr neu aufgelegt.

Ein durchgehendes **wichtiges didaktisches Prinzip** ist der einheitliche Aufbau der Programmiersprachenbände. Alle Beispiele werden in den einzelnen Sprachen möglichst detailgetreu so formuliert, daß sie als Beispiele für die Übersetzung von einer Sprache in eine andere verwendet werden können. Dazu zählen nun: Turbo-Pascal, C, C++, Turbo/Power-Basic, Quick-Basic und in Kürze auch Modula. Sie finden nach diesem Artikel Mustertexte aus einzelnen Bänden.

Die **objektorientierte Programmierung** hat vor einiger Zeit die umfangreichsten Änderungen gebracht. Gerade das Beispiel dieser modernen Programmieretechnik zeigte, daß die Umsetzung im Unterricht gar nicht so einfach war und die dabei gewonnenen Erkenntnisse "von Jahr zu Jahr" eingebaut werden konnten.

ADIM-Skripten werden bereits an etlichen Schulen für den EDV-Unterricht werden. Für höhere technische Lehranstalten stehen sogar vier Bände als **Gratis-Fachbuch** zur Verfügung.

Als Übersicht über derzeit verfügbare Bände finden Sie in auf Seite 45 einen Bestellschein.

Eine weitere Aktivität der **ADIM** ist die Organisation von **Sammeleinkäufen**. In den Anfängen der PC-Zeit waren Sammelkäufe sehr günstig; inzwischen hat der allgemeine Preisverfall und auch die Konkurrenz verschiedener Anbieter zu erfreulich niedrigen Preisen geführt. (Wer weiß schon noch, daß die ersten Disketten vor 10 Jahren 140,- Schilling (!) gekostet haben und damals als Clubaktion um 70,- Schilling zu kaufen waren?) Die Hardwareaktivitäten der **ADIM** beschränken sich nun auf besondere Komponenten. Derzeit sind **Modems** mit besonders vielen Eigenschaften nach wie vor aktuell.

Seit einiger Zeit wollen wir Sie über ein eigenes **Mitteilungsblatt** über unsere aktuellen und neuesten Aktivitäten informieren. Der große damit verbundene Aufwand hat aber dieses Projekt bisher verhindert. Nun haben wir in den **PC-NEWS** einen guten Partner gefunden. In dieser Zeitschrift wollen wir Sie regelmäßig über aktuelle Neuigkeiten, unsere Ziele und pädagogische Ideen informieren und damit auch ganz allgemein über Fragen des EDV-Unterrichts diskutieren.

Wir laden daher alle ADIM-Freunde sehr herzlich ein, die PC-NEWS zu abonnieren! Auf ein regelmäßiges "Wiederlesen" freut sich

Martin Weissenböck

Ausschnitt aus dem Band 40, Turbo C:

Beispiel 5.1F:

Man verwandle Sekunden in Stunden, Minuten und Sekunden. Die Eingabe wird mit der Zahl 0 abgeschlossen.

umwandeln

Eingabeparameter: ein

Ein/Ausgabeparameter: h, m, s

h <- ganzzahliges Divisionsergebnis
von ein durch 3600

ein <- Divisionsrest von
ein durch 3600

m <- ganzzahliges Divisionsergebnis
von ein durch 60

s <- Divisionsrest von ein durch 60

main

Eingabe einlesen

Abbruch, wenn Eingabe 0 ist

umwandeln(Eingabe, Std, Min, Sek)

Eingabe, Std, Min, Sek ausgeben

```
/* Beispiel 5.1F */
/* Umwandeln einer Sekundenzahl in
   Stunden, Minuten und Sekunden */
#include <stdio.h>
#include <conio.h>
void umwandeln
(long ein, int *h, int *m, int *s);

void umwandeln
(long ein, int *h, int *m, int *s)
{
    *h=(int) (ein/3600);
    ein%=3600;
    *m=(int) ein/60;
    *s=(int) ein%60;
} /* umwandeln */

void main ()
{
    int std,min,sek;
    long eingabe;

    printf("Umwandeln SEK -> STD MIN SEK\n");
    for (;;)
    {
        printf("Zeit in SEK (0...Abbruch)");
        scanf("%ld",&eingabe);
        if (!eingabe) break;
        umwandeln(eingabe,&std,&min,&sek);
        printf
            ("%7ld SEK =%4d STD%3d MIN%3d SEK\n",
             eingabe,std,min,sek);
    }
} /* b051f */
```

Anmerkung: Da ein Tag bereits 86400 Sekunden hat, wurde für die Eingangsgröße der Typ **long** verwendet. Mit dem Typ **int** wäre 32767 die größte Zahl.

Jeweils ein **aktueller Parameter** und ein **formaler Parameter** müssen paarweise zusammenpassen. Daher ist auf

- den Typ der Parameter,
- die Reihenfolge der Parameter und
- die Anzahl der Parameter

zu achten.

Anmerkung: in C können auch Unterprogramme mit einer variablen Zahl von Parametern vereinbart werden.

Die Wahl der Namen der formalen Parameter ist nicht von Bedeutung!

Eine **return-Anweisung** kann verwendet werden, um das Unterprogramm sofort zu verlassen.

Ausschnitt aus dem Band 41, Turbo/Power-Basic:

Beispiel 5.1F:

Man verwandle Sekunden in Stunden, Minuten und Sekunden. Die Eingabe wird mit 0 abgeschlossen. Achtung: 32767 ist die größte INTEGER-Zahl, daher werden die Sekunden als LONG-Wert gespeichert.

b051f

eingabe einlesen
umwandeln(eingabe, std, min, sek)
eingabe, std, min, sek ausgeben
bis eingabe gleich 0

umwandeln

Eingabeparameter: ein

Ein/Ausgabeparameter: h, m, s

h <- ganzzahliges Divisionsergebnis von ein durch 3600
ein <- Divisionsrest von ein durch 3600
m <- ganzzahliges Divisionsergebnis von ein durch 60
s <- Divisionsrest von ein durch 60

' Beispiel 051F

' Umwandeln einer Sekundenzahl in Stunden,
Minuten und Sekunden.

PRINT "Umwandeln Sek -> Std Min Sek"

DO

INPUT "Zeit in Sek (0...Abbruch)",_

eingabe%

IF eingabe% = 0 THEN EXIT LOOP

PRINT

CALL umwandeln_

((eingabe%), std%, min%, sek%)

PRINT eingabe%; " SEK = "; std%; " STD";

min%; " MIN"; sek%; " SEK"

LOOP

END ' B051F

SUB umwandeln (ein%, h%, m%, s%)

h% = ein% \ 3600

ein% = ein% MOD 3600

m% = ein% \ 60

s% = ein% MOD 60

END SUB ' umwandeln

Jeweils ein **aktueller Parameter** und ein **formaler Parameter** müssen paarweise zusammenpassen. Daher ist auf

- den Typ der Parameter
- die Reihenfolge der Parameter und
- die Anzahl der Parameter

zu achten.

Die Wahl der Namen der formalen Parameter ist nicht von Bedeutung!

Eine EXIT SUB - Anweisung kann verwendet werden, um das Unterprogramm sofort zu verlassen.

Wenn das gesamte Programm beendet werden soll, ist einer der Befehle STOP, END oder SYSTEM zu verwenden.

Ausschnitt aus dem Band 47, Turbo Pascal:

Beispiel 5.1F:

Man verwandle Sekunden in Stunden, Minuten und Sekunden. Die Eingabe wird mit 0 abgeschlossen. Achtung: 32767 ist die größte integer-Zahl, daher werden die Sekunden als longint-Wert gespeichert.

umwandeln

Eingabeparameter: ein

Ein/Ausgabeparameter: h, m, s

h <- ganzzahliges Divisionsergebnis von ein durch 3600
ein <- Divisionsrest von ein durch 3600
m <- ganzzahliges Divisionsergebnis von ein durch 60
s <- Divisionsrest von ein durch 60

b051f

Eingabe einlesen
Abbruch, wenn Eingabe 0 ist
umwandeln(Eingabe, Std, Min, Sek)
Eingabe, Std, Min, Sek ausgeben

PROGRAM b051f;

(* Umwandeln einer Sekundenzahl in Stunden,
Minuten und Sekunden *)

PROCEDURE umwandeln(ein:longint;

VAR h,m,s:integer);

BEGIN

h:=ein DIV 3600;

ein:=ein MOD 3600;

m:=ein DIV 60;

s:=ein MOD 60

END; (* umwandeln *)

VAR

eingabe:longint;

std,min,sek:integer;

BEGIN

writeln('Umwandeln s -> std min s');

REPEAT

write('Zeit in Sek (0...Abbruch)');

readln(eingabe);

umwandeln(eingabe,std,min,sek);

writeln(eingabe:5,' s =',

std:4,' std',min:3,' min',sek:3,' s')

UNTIL eingabe=0

END. (* b051f *)

Jeweils ein **aktueller Parameter** und ein **formaler Parameter** müssen paarweise zusammenpassen. Daher ist auf

- den Typ der Parameter
- die Reihenfolge der Parameter und
- die Anzahl der Parameter

zu achten.

Die Wahl der Namen der formalen Parameter ist nicht von Bedeutung!

Eine exit-Anweisung kann verwendet werden, um das Unterprogramm sofort zu verlassen. Mit halt wird das gesamte Programm verlassen. Wird halt mit einem Parameter verwendet, wird dessen Wert auf die DOS-Ebene übertragen und kann dort als ERRORLEVEL abgefragt werden.

Ausschnitt aus dem Band 49, Quick-Basic:

Beispiel 5.1F:

Man verwandle Sekunden in Stunden, Minuten und Sekunden. Die Eingabe wird mit 0 abgeschlossen. Achtung: 32767 ist die größte INTEGER-Zahl, daher werden die Sekunden als LONG-Wert gespeichert.

b051f

Eingabe einlesen
Abbruch, wenn Eingabe 0 ist
umwandeln(Eingabe, Std, Min, Sek)
Eingabe, Std, Min, Sek ausgeben

umwandeln

Eingabeparameter: ein

Ein/Ausgabeparameter: h, m, s

h <-ganzzahliges Divisionsergebnis von ein durch 3600
ein <-Divisionsrest von ein durch 3600
m <-ganzzahliges Divisionsergebnis von ein durch 60
s <-Divisionsrest von ein durch 60

' Beispiel 051F

' Umwandeln einer Sekundenzahl in

' Stunden, Min. und Sek.

DECLARE SUB umwandeln (ein&, h%, m%, s%)

PRINT "Umwandeln Sek -> Std Min Sek"

DO

INPUT "Zeit in Sek (0...Abbruch)",_

eingabe&

IF eingabe& = 0 THEN EXIT DO

PRINT

CALL umwandeln_

((eingabe&), std%, min%, sek%)

PRINT eingabe&: " SEK ="; std%; " STD";_

min%; " MIN"; sek%; " SEK"

LOOP

END ' B051F

SUB umwandeln (ein&, h%, m%, s%)

h% = ein& \ 3600

ein& = ein& MOD 3600

m% = ein& \ 60

s% = ein& MOD 60

END SUB ' Umwandeln

Jeweils ein **aktueller Parameter** und ein **formaler Parameter** müssen paarweise zusammenpassen. Daher ist auf

- den Typ der Parameter
- die Reihenfolge der Parameter und
- die Anzahl der Parameter

zu achten.

Die Wahl der Namen der formalen Parameter ist nicht von Bedeutung!

Eine EXIT SUB Anweisung kann verwendet werden, um das Unterprogramm sofort zu verlassen.

Wenn das gesamte Programm beendet werden soll, ist einer der Befehle STOP, END oder SYSTEM zu verwenden.

Ausschnitt aus dem Band 50, Borland C++:

Beispiel 5.1F:

Man verwandle Sekunden in Stunden, Minuten und Sekunden. Die Eingabe wird mit der Zahl 0 abgeschlossen.

umwandeln

Eingabeparameter: ein

Ein/Ausgabeparameter: h, m, s

h <-ganzzahliges Divisionsergebnis von ein durch 3600
--

ein <-Divisionsrest von ein durch 3600

m <-ganzzahliges Divisionsergebnis von ein durch 60
--

s <-Divisionsrest von ein durch 60

main

Eingabe einlesen

Abbruch, wenn Eingabe 0 ist

umwandeln(Eingabe, Std, Min, Sek)

Eingabe, Std, Min, Sek ausgeben

// Beispiel 5.1F

/* Umwandeln einer Sekundenzahl in
Stunden, Minuten und Sekunden */

#include <iostream.h>

#include <iomanip.h>

void umwandeln

(long ein,int &h,int &m,int &s)

{

h=(int) (ein/3600);

ein%=3600;

m=(int) ein/60;

s=(int) ein%60;

} // umwandeln

void main ()

{

int std,min,sek;

long eingabe;

cout << "Umwandeln SEK -> STD MIN SEK\n";

for (;;)

{

cout << "Zeit in SEK (0...Abbruch)";

cin >> eingabe;

if (!eingabe) break;

umwandeln(eingabe,std,min,sek);

cout << setw(7) << eingabe << " SEK ="

<< setw(4) << std << " STD" <<

setw(3) << min << " MIN" <<

setw(3) << sek << " SEK\n";

}

} // b051f

Anmerkung: Da ein Tag bereits 86400 Sekunden hat, wurde für die Eingangsgröße der Typ **long** verwendet. Mit dem Typ **int** wäre 32767 die größte Zahl.

Jeweils ein **aktueller Parameter** und ein **formaler Parameter** müssen paarweise zusammenpassen. Daher ist auf

- den Typ der Parameter,
- die Reihenfolge der Parameter und
- die Anzahl der Parameter

zu achten.

Anmerkung: in C++ können auch Unterprogramme mit einer variablen Zahl von Parametern vereinbart werden.

Die Wahl der Namen der formalen Parameter ist nicht von Bedeutung!

Eine return-Anweisung kann verwendet werden, um das Unterprogramm sofort zu verlassen.

Und so soll es auch in Zukunft weitergehen...

Korrekturen zu ADIM-Band-50

Martin Weissenböck, ADIM

Zum ADIM-Band 50, 3. Auflage (Juli 1993) liegen bisher Hinweise auf die folgenden Verbesserungsmöglichkeiten bzw. Druckfehler vor (auch zu finden in der Datei R50-3.LZH, **His Master's Voice**):

1. Seite 9, linke Spalte, 3. Absatz:

Statt

Zu den reservierte Wörtern gehören Wörter wie z.B. int.

richtig

Zu den reservierte Wörtern gehören Wörter wie z.B. void.
int ist kein reserviertes Wort.

2. Seite 27, linke Spalte, Beispiel b042c:

Statt

```
printf("*");
```

richtig

```
cout << "*";
```

3. Seite 38, rechte Spalte, Beispiel 5.1D:

Vor

```
int sum;
```

sollte der Prototyp eingefügt werden. Der Programm lautet dann:

```
void berechnung (int n); // Prototyp
int sum;
void berechnung (int n)
{ int i;
```

4. Seite 39, linke Spalte, Beispiel 5.1F:

Vor

```
void umwandeln
```

sollte der Prototyp eingefügt werden. Der Programm lautet dann:

```
void umwandeln
(long ein,int &h,int &m,int &s); // Prototyp
void umwandeln
(long ein,int &h,int &m,int &s) // Kopfzeile
{ h=(int) (ein/3600);
```

5. Seite 41, linke Spalte, letzter Absatz vor "5.2 Funktionen":

Statt

daß der Parameter im Unterprogramm

richtig

daß die Parameter im Unterprogramm

6. Seite 42, linke Spalte:

Zeile

B52B einfügen

streichen.

7. Seite 49, rechte Spalte, 1. Absatz, Mitte:

Statt

```
$A000:$0000
```

besser

```
0xA000:0x0000
```

8. Seite 51, linke Spalte, Mitte

Im Absatz

Mit line könnten die Zeichenbefehle des Beispiels 6.2A so aussehen:

ergänzen

(Beispiel 6.2B)

9. Seite 56 und Seite 57:

Das Beispiel 7.3B ist leichter zu verstehen bzw. zu erklären, wenn zuerst die Klasse Ort wie auf Seite 57 und dann erst wie auf Seite 56 behandelt wird.

10. Seite 57, rechte Spalte, 1. Absatz nach "7.4 Vererbung":

Statt

horhergehende Klasse

richtig

vorhergehende Klasse

11. Seite 58, rechte Spalte, Beispiel 7.4B und

Seite 60, linke Spalte, Beispiel 7.4D:

Statt

```
const char BGI [ ] = "c:\\s\\tc";
```

besser

```
const char BGI [ ] = "c:\\bc\\bgi";
```

12. Seite 64, rechte Spalte, letzter Programmabsatz:

Nach

```
#include <iostream.h>
```

fehlt

```
#include <math.h>
```

13. Seite 65, linke Spalte, Mitte:

Statt

```
long ggt(long x,long y)
{ x=abs(x);
  y=abs(y);
  ...
} /* ggt */
```

richtig

```
long Bruch::ggt(long x,long y)
{ x=labs(x);
  y=labs(y);
  ...
} // Bruch::ggt
```

14. Seite 66, linke Spalte, Beispiel 7.7B:

wie Punkt 12.

15. Seite 66, rechte Spalte, Beispiel 7.7C:

wie Punkt 12.

16. Seite 67, linke Spalte, oben:

wie Punkt 13.

17. Seite 67, linke Spalte, 2. Programmabsatz, im operator+ :

Statt

```
n=a.n*b.n,
```

richtig

```
n=a.n*b.n;
```

18. Seite 67, rechte Spalte, Mitte:

wie Punkt 17.

19. Seite 70, linke Spalte, Beispiel 8.1C, in void lesen(Feld z):

Statt

```
cin >> zahl[i];
```

richtig

```
cin >> z[i];
```

20. Seite 70, rechte Spalte, oben, 3. Zeile:

Statt

```
cout << groesser(zahl,m);
```

richtig

```
cout << zaehlen(zahl,m);
```

21. Seite 70, rechte Spalte, 3. Programmabsatz:

Statt

```
void lesen (int z[ ])
{
    ...
}
```

richtig

```
void lesen (float z[ ])
{
    ...
}
```

22. Seite 70, rechte Spalte, 3. Programmabsatz:

Statt

```
void lesen (int *z)
{
    ...
}
```

richtig

```
void lesen (float *z)
{
    ...
}
```

23. Seite 73, linke Spalte, Beispiel 8.2B:

Statt

```
platztausch(&reihe1[j],&reihe2[j]);
```

richtig

```
platztausch(reihe1[j],reihe2[j]);
```

24. Seite 73, linke Spalte, Beispiel 8.2B:

Statt

```
platztausch(&k[i][spalte1],&k[i][spalte2]);
```

richtig

```
platztausch(k[i][spalte1],k[i][spalte2]);
```

25. Seite 73, rechte Spalte, 1. Absatz:

Statt

```
scanf("%d",&i);
```

besser

```
cin >> i;
```

Sinngemäß genauso für m, j und n,

26. Seite 89, rechte Spalte, 3. Absatz von unten:

Statt

```
abgeleitet
```

richtig

```
abgeleitete
```

27. Seite 103, rechte Spalte, Beispiel 9.7A:

Das verbesserte Beispiel ist auf dieser Diskette zu finden.

28. Seite 119, rechte Spalte, vor

```
delete [ ] p;
```

ergänzen

(ab Version 3.1)

29. Seite 120, nach Operator []

// Beispiel 10.5A

einfügen.

30. Seite 124 (kommt fünfmal vor):

Statt

```
B111B.C
```

besser

```
B111B.CPP
```

31. Seite 126, rechte Spalte (kommt zweimal vor):

Statt

```
B113A.C
```

besser

```
B113A.CPP
```

32. Seite 126, Beispiel 11.3A:

Nach

```
#include <stdio.h>
```

einfügen

```
#include <math.h>
```

33. Seite 126, Beispiel 11.3A (kommt zweimal vor):

Statt

```
abs(
```

besser

```
labs(
```

34. Seite 126, Beispiel 11.3A:

Statt

```
/* drBruch */
```

richtig

```
/* druckeBr */
```

35. Seite 129, rechte Spalte, erster Textabsatz:

Statt

An den Methoden von Liste (einf_vorne und einf_hinten) und am Unterprogramm ausgabe ändert sich nichts, sie werden daher nicht wiederholt.

richtig

An den Methoden von Liste (aktion_vorw und aktion_rueckw) und am Unterprogramm drucke ändert sich nichts, sie werden daher nicht wiederholt.

36. Seite 135, linke Spalte, Beispiel 12.5B:

Das verbesserte Beispiel ist auf der Diskette zu finden.

37. Seite 136, linke Spalte, unterhalb der Mitte:

Statt

```
void interrupt (*alterTimerInt)();
```

richtig

```
void interrupt (*alterTimerInt)(...);
```

38. Seite 136, linke Spalte, unterhalb der Mitte:

Statt

```
void interrupt neuerTimerInt(void);
```

richtig

```
void interrupt neuerTimerInt(...);
```

Zum Bestellschein: die 5. Auflage des Bandes 47 (Turbo-Pascal, Juli 1991) ist ausverkauft. □

ADIM-Bestellschein

Bitte kopieren Sie dieses Blatt bei Bedarf.

An die Arbeitsgemeinschaft für
Didaktik, Informatik und Mikroelektronik
(ADIM)

Postfach 23
A-1191 Wien

Tel: 0222-369 88 59-8
Fax: 0222-369 88 59-7

Nr= Bestellung von **fertigen** Skripten,

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

Band Nr.	Bezeichnung Produkt (Hersteller)	Version	ISBN 3-85071- ohne mit Disk Disk		Auf- lage	Datum	nur Band		nur Disk		Band+Disk		Gesamt- preis: öS
							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/C Classic	ABVERKAUF!	2.03	000-9	001-7	1.	Jul87	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/Power Basic	6451*	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.	Sep93	120		50		160	
98	Peter Pfenicher: Turbo Pascal Anweisungssammlung mit Beispielen...	6.0	ISBN 3-	900985-00-6	4.	Okt91	130						
Gesamtanzahl der bestellten Skripten/Disketten:													
Gewünschte Freiemplare (für je 20 lieferbare Bände ein beliebiger Band), bitte Bandnummern angeben													0
Versandkostenanteil pro SKRIPTEN-Sendung (entfällt ab 1.000 S Bestellwert)													30
Endsumme (inklusive 10% bzw. 20% Umsatzsteuer)													

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!

Änderungen - insbesondere bei den Versandkosten - und Irrtum vorbehalten!

*) Fachbuchnummer (für Höhere technische Lehranstalten auch über die Schulbuchaktion zu beziehen) Verlag nr. 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. November 1993 (PC-NEWS-35)

Bitte beachten Sie:

- * Die Disketten enthalten die Programmbeispiele des jeweiligen Bandes. Alle Disketten werden im Format 5,25" (360 KByte) geliefert. Lösungsprogramme zu den Übungsaufgaben sind aus pädagogischen Gründen nicht erhältlich.
- * Werden nur Beispieldisketten bestellt, wird kein Versandkostenanteil berechnet. (Gilt nicht für Leerdisketten!)
- * Bände ohne Datum (Nummer in Klammern) sind zwar geplant; da die Fertigstellung vor allem vom Zeiteinsatz der **ADIM**-Mitarbeiter in deren Freizeit abhängt, kann ein exakter Termin nicht angegeben werden. Aus aktuellen Gründen können einzelne Bände auch vorgezogen werden. "*" bedeutet: geschätzter Preis für geplante Bände.
- * Die Umsatzsteuer ist in den Preisen enthalten: **ADIM**-Bände und **ADIM**-Bände+Disketten: 10%, Disketten allein und sonstiges: 20 %.

Auslandsbestellungen - nur gegen Vorauszahlung oder Verrechnung per Kreditkarte:

- * Postgiroamt München (BLZ 700 100 80), Konto 1209 14-800. Postcheckamt Chur, Konto 70-40051-3. Volksbank Brixen, Konto 37283.
- * Der Rechnungsbetrag verringert sich um die Mehrwertsteuer und das Versandkosten**pauschale**. Die Portospesen werden in ihrer tatsächlichen Höhe verrechnet. Wir bitten um Vorauszahlung oder Verrechnung per Kreditkarte: der Rechnungsbetrag wird Ihnen vor der Auslieferung mitgeteilt. Die Skripten werden sofort nach Zahlungseingang versandt. Größere Bestellposten werden geteilt, da Buchsendungen nur bis 5 kg zugelassen sind.

Zahlungstermine im Inland: Wir versenden üblicherweise die Skripten als Brief oder Paket und bitten um Überweisung binnen 14 Tagen bzw. (ab 10 Stück) binnen 3 Wochen. **Lieferung per Nachnahme vorbehalten.** Bei **Zahlungsverzug** können wir jedenfalls weitere Bestellungen nur gegen Nachnahme und Ersatz der Nachnahmespesen ausführen. Wir bitten um pünktliche Überweisung.

Ich wurde auf die **Aktivitäten** der **ADIM** aufmerksam durch:

Verwendung der Skripten:

ein Seminar des PI Wien	Kollegin/Kollegen:	als Lehrer/in
ein Seminar des PI Graz		als Schüler/in
ein anderes Seminar		als Student/in
die PC-NEWS	sonstige Informationen:	für mein Hobby
eine Mitteilung in einer Zeitschrift		freiberuflich

Abonnement: Neue Skripten können im Abonnement bestellt werden. Das Abo kann jederzeit gekündigt werden.

Ich bestelle neue Skripten bzw. neue und überarbeitete Auflagen im Abonnement (Ja/Nein).....

Adressen (bitte alle Angaben in BLOCKBUCHSTABEN):

Lieferung an (Vorname, FAMILIENNAME, Adresse) (bei Minderjährigen: des gesetzlichen Vertreters)	Rechnung (falls verschieden) an: (Name, Adresse):
Tel-Nr:	Tel-Nr:
BTX-Nr:	BTX-Nr:

Ein Service, vor allem für unsere Interessenten aus dem Ausland:

Skripten + Disketten können bei der **ADIM**-Wien auch mit folgenden Kreditkarten bezahlt werden (bitte ankreuzen):

Visa-Card	Kartennummer:	
Euro/Master-Card	lautend auf:	
American Express	gültig bis:	

Unterschrift des Bestellers (falls der Besteller noch nicht bei der Adresse angegeben ist, geben Sie bitte den Namen hier in **BLOCKBUCHSTABEN** an.) Bestellung von ganzen Klassen werden gerne bearbeitet. Angaben wie z.B. "3B" reichen aber nicht aus, der Name eines verantwortlichen Bestellers muß angegeben werden.

Ort, Datum:	Unterschrift:
-------------	---------------

Weitere Bestellmöglichkeiten:

Telefonisch: **ADIM**-Wien, Tel. 0222-369 88 59 8.
Wenn der Anruferbeantworter eingeschaltet ist, sprechen Sie bitte langsam und deutlich und geben Sie auch Ihre Telefonnummer für Rückfragen an.

Schriftlich: auch bei der **ADIM**-Graz,
Postfach 37, A-8028 Graz.

Über BTX (siehe auch BTX-Seite *56458#):

ADIM-Wien, BTX-Nr.: 912 218 106 oder

ADIM-Graz, BTX-Nr.: 913 110 525

Über Telefax: **ADIM**-Wien: . 0222-369 88 59 7

Über Telex: **ADIM**-Wien: 75 210 388 weim a
oder **ADIM**-Graz: 75 210 859 sber a

TURBO-PASCAL-TOOLBOX

Hans Lohninger, TU-Wien

Seit einigen Monaten ist nun die vierte Version meiner Turbo Pascal Toolbox in Form zweier Bücher auf dem Markt und erfreut sich großer Beliebtheit. Da ich selbst langjähriges Mitglied des **PCC-TGM** bin, möchte ich den Mitgliedern des **PCC-TGM** diese Toolbox in einer Sonderaktion verbilligt zur Verfügung stellen.

Jeder Band umfaßt ca. 320 Seiten und enthält eine umfangreiche Dokumentation einschließlich zahlreicher Beispielprogramme. Die Toolbox-Units und die Beispielprogramme sind als Source-Code auf der beigepackten Diskette enthalten.

Teil 1: Hans Lohninger
Turbo Pascal 7.0 Toolbox
IWT-Verlag 1993
327 Seiten, ISBN 3-88322-434-0

Teil 2: Hans Lohninger
Borland Pascal 7.0 Graphik Toolbox
IWT-Verlag 1993
324 Seiten, ISBN 3-88322-435-9

Jeder Band kann unabhängig vom anderen Teil benutzt werden und kostet öS 608,-. Für **PCC-TGM**-Mitglieder kann ich die Bücher um je öS 480,- zur Verfügung stellen.

Als erste Information für Sie ist im folgenden eine kurze englische Zusammenfassung des Inhalts wiedergegeben.

I want to announce a Turbo/Borland Pascal 7.0 library which may be of general interest for BP DOS-programmers. It supports a lot of various topics, such as XMS control, interrupt-driven serial lines, data structures, numerical table editors, mathematical routines, input masks, a graphical user interface for DOS-based programs, HPGL output, PostScript (EPS) hardcopy, a mouse interface, routines to display charts, etc. Part of the units are based on OOP technology.

The following gives a short summary of the features of the library:

Unit UTIL

A collection of 'mixed pickles' and basic routines: a bell, string handling routines, directory and file name support, alphanumeric cursor control, some binary de(en)coding routines, a stop watch, video attributes, system time and date,

Unit DSTRUC

This unit holds two classes to handle bit arrays and FIFO data structures.

Unit XMS

The unit XMS supports the usage of extended memory by DOS based programs. XMS-Memory as well HMA and UMBs can be allocated. This unit needs a 386 PC and HIMEM.SYS or something like that.

Unit MATH1

MATH1 contains a collection of simple mathematical routines which are not contained in the Borland Pascal libraries: format conversion (binary, octal, hexadecimal), cot, tg, tgh, cosh, sinh, decimal logarithms, some simple sorting routines, Min, Max, Sign, ...

Unit MATRICES

This unit defines two classes for handling real-valued vectors and two dimensional matrices without loading the data segment of BP programs. The matrices and vectors are maintained on the heap. Also a routine for matrix inversion is included.

Unit MATH2

The unit MATH2 contains some routines for more involved mathematical stuff like covariance matrices, eigenvectors, eigenvalues, simple statistics, linear regression, and principal components.

Unit ALPHAMSK

This unit provides a very simple mask generator for the alpha-numerical screen. It will be of little use if one is acquainted with Turbo Vison (tm).

Unit TABLES

This unit defines a class which is inherited from the class 'Matrix' (unit MATRICES) and supports the handling of numerical tables. In addition, a versatile table editor is contained.

Unit SERIAL

The unit SERIAL supports the use of interrupt-driven serial lines. In fact this is the only way to communicate via the RS-232 lines using a Pascal program.

Unit VGACARD

VGACARD supports the control of the color palette of VGA cards.

Unit GDRIVER

This unit supplies the basis (graphics drivers, HPGL and PostScript drivers, and other low level routines) of all other graphics routines as well as some self-contained routines like a hardcopy routine, the support of slides and the storing of graphics in the extended memory.

Unit GMOUSE

The unit supports the control of a mouse cursor on the graphics screen. Several high level features of mouse based user interfaces are supported (e.g. point snap, several types of cursors, rubber bands, confining the mouse motion, etc.).

Unit GRAFFONT

This unit supports a special graphics font which can be XORed with the graphics background.

Unit IM4

IM4 contains a lot of routines in order to draw on a hardware-independent virtual graphics screen. Using the routines of this unit (instead of the BP unit GRAPH) enables the program to adjust to a given graphics hardware (HGC,EGA,VGA) automatically WITHOUT changing the extents of the graphics (basically all graphics look like the same on each of the three above-mentioned adapters)

Unit IGWIND

The unit IGWIND defines two classes which support rectangular areas on the screen. These classes support automatic background saving.

Unit CHARTS

This unit provides a system of graphical windows (using classes inherited from IGWIND) which gives the user the possibility to display real-valued data within these windows. The scaling can be adjusted almost unrestricted. In addition, the automatic drawing of axes and several types of charts (lines, charts, contour plots, etc.) are supported.

Unit GRAFIN

Last not least the unit GRAFIN provides a system to create menus and command entry boxes on the graphical screen. Several input devices are supplied (e.g. menus, potentiometers, a pocket calculator, a file selection box, etc.).□

Anm.: Interessenten für die Turbo-PASCAL-Toolbox bitten wir, sich mit dem Autor direkt in Verbindung zu setzen. (siehe Autoren, Seite 5).

Nebeneffekt bei mit EOF zu beendenden PASCAL-Einleseschleifen

Walter Riemer, N/NA, TGM

Das folgende Programm hat die Aufgabe, eine nichtnegative, höchstens dreistellige ganze Zahl zu lesen und zu verarbeiten (das heißt, auszugeben); das Ganze soll wiederholt werden, bis es mit EOF (Eingabe von CTRL-Z) beendet wird. Ungültige Zahlen sollen zurückgewiesen werden; dies erledigt die Prozedur Lesen. Damit man überhaupt in die Lese-/Prüfeschleife in dieser Prozedur eintritt, wird (ein allgemein üblicher Kunstgriff!) die Zahl zunächst auf einen ungültigen Wert, nämlich 1000, gesetzt.

Das Programm funktioniert einwandfrei, wie das Ausführungsbeispiel zeigt:

```
VAR Zahl: INTEGER;
PROCEDURE Lesen;
BEGIN Zahl:=1000;
  WHILE (Zahl<0) OR (Zahl>999) DO
    BEGIN WRITE ('Zahl eingeben: ');
      READ (Zahl)
    END;
  END;
BEGIN Lesen;
  REPEAT WRITELN ('Die gelesene Zahl ist ',Zahl);
    Lesen
  UNTIL EOF
END.
```

```
Zahl eingeben: -5
Zahl eingeben: 12345
Zahl eingeben: 35
Die gelesene Zahl ist 35
Zahl eingeben: 1
Die gelesene Zahl ist 1
Zahl eingeben: 0
Die gelesene Zahl ist 0
Zahl eingeben: ^Z
```

Man könnte auch auf die Idee kommen, den Bereich der gültigen Zahlen auf positive Zahlen einzuschränken, also 0 auszuschließen. Man ändert also die fünfte Zeile auf

```
WHILE (Zahl<1) OR (Zahl>999) DO
```

In dieser Version wird EOF nicht mehr erkannt: Warum ?

```
Zahl eingeben: -5
Zahl eingeben: 12345
Zahl eingeben: 35
Die gelesene Zahl ist 35
Zahl eingeben: 1
Die gelesene Zahl ist 1
Zahl eingeben: 0
Zahl eingeben: ^Z
Zahl eingeben: ^Z
Zahl eingeben: ^Z
```

Begreiflicherweise neigt so mancher Programmierer, wenn er mit einem derartigen schwer durchschaubaren Sachverhalt konfrontiert wird, dazu, eine pragmatische Lösung zu suchen. Man könnte zum Beispiel "hoffen", daß die Sache mit einer WHILE-Schleife im Hauptprogramm besser funktioniert:

```
BEGIN Lesen;
  WHILE NOT EOF DO
    BEGIN WRITELN ('Die gelesene Zahl ist ',Zahl);
      Lesen
    END
  END.
```

Der Effekt ist aber genau der gleiche: Wenn 0 erlaubt ist, funktioniert alles, wenn 0 ausgeschlossen ist, wird EOF nicht mehr erkannt.

Die Ursache ist offensichtlich ein Nebeneffekt beim Einlesen von der Tastatur: Laut Standard-PASCAL (N. Wirth) sollte der Wert der zu lesenden Variablen undefiniert sein, wenn CTRL-Z eingegeben wurde; TURBO-PASCAL jedoch weicht davon ab und setzt den Wert einer Zahlenvariablen in diesem Fall auf 0.

Die Schleife in der Prozedur Lesen in der ersten Version des Programms wird daher nach Eingeben von CTRL-Z nicht aus Disziplin allein verlassen, sondern weil als Nebeneffekt Zahl auf 0 gesetzt wird; dies gilt als gültige Zahl.

In der zweiten Variante ist jedoch 0 eine ungültige Zahl, da nur positive Zahlen als gültig gelten; daher wird schon die Schleife in der Prozedur Lesen nicht verlassen (man bleibt in einer Endlosschleife) und CTRL-Z, das ja erst im Hauptprogramm bei UNTIL EOF geprüft werden würde, bleibt wirkungslos.

Man könnte nun dieses so erkannte Problem so lösen, daß man in der Prozedur Lesen auch auf EOF abfragt:

```
PROCEDURE Lesen;
BEGIN Zahl:=1000;
  WHILE ((Zahl<1) OR (Zahl>999)) AND NOT EOF DO
    BEGIN WRITE ('Zahl eingeben: ');
      READ (Zahl)
    END
  END;
```

```
-5
Zahl eingeben: Zahl eingeben: 12345
Zahl eingeben: 35
Die gelesene Zahl ist 35
Zahl eingeben: 1
Die gelesene Zahl ist 1
Zahl eingeben: 0
Zahl eingeben: ^Z
```

Nun funktioniert erwartungsgemäß zwar das Erkennen von EOF, aber beim ersten Aufruf der Prozedur Lesen wird die Eingabeaufforderung zunächst übersprungen (also nicht ausgegeben), dafür aber verspätet gemeinsam mit der nächsten Eingabeaufforderung ausgespuckt.

Dieses Verhalten erscheint schon recht ungewöhnlich: Immerhin basiert die Kunst des Programmierens auf dem Axiom, daß Anweisungen, die innerhalb eines Anweisungsblocks hintereinander stehen, auch in dieser Reihenfolge ausgeführt werden. PASCAL kehrt in diesem Fall die Reihenfolge der Ausführung um, und zwar gesteuert durch die Schleife:

```
Zahl:=1000;
WHILE ... (Prüfen der Laufbedingung)
  einlesen, ohne daß TRACE auf der READ-Anweisung stehen bleibt
  WRITE ('Zahl eingeben: ');
  READ (Zahl); bewirkt nichts, obwohl TRACE auf der READ-Anweisung steht !!
WHILE ... (Prüfen der Laufbedingung)
  Verlassen der Schleife, falls gültige Zahl eingegeben wurde.
```

Das Problem ist sofort bereinigt, wenn man vor dem Eingeben der ersten Zahl RETURN drückt; das ist natürlich sehr wenig benutzerfreundlich.

Man könnte zu der Annahme kommen, daß die Erscheinung nur auf die Standard-Eingabe beschränkt ist und vielleicht in der in TURBO-PASCAL ebenfalls verfügbaren, wesentlich besseren Konsolhandhabung nicht auftritt, die allerdings explizit vereinbart werden muß (das Unit CRT muß eingebunden werden). Neben wesentlich schnelleren Ein-/Ausgabeoperationen, Möglichkeit der Umleitung der Ein-/Ausgabe sowie ausgefeilter Steuerung des Bildschirms im Textformat (Attribute, Fenster usw.) bietet CRT für unseren Fall auch noch die Möglichkeit, den Eingabeweg "Tastatur" rückzusetzen. Man könnte meinen, daß damit das Problem gelöst ist; leider ist dies jedoch auch nicht der Fall: am Verhalten des Programms hat sich nichts geändert, obwohl vom Unit CRT aus implizit ASSIGNCRT(INPUT) ausgeführt, wodurch ein RESET(INPUT) erst ermöglicht wird:

```
BEGIN RESET (INPUT);
  CheckEOF:=TRUE;
  Lesen;
  WHILE NOT EOF DO
    BEGIN WRITELN ('Die gelesene Zahl ist ',Zahl);
      Lesen
    END
  END.
```

Selbstverständlich dürfte auch nicht darauf vergessen werden, die in CRT definierte Systemvariable CheckEOF auf TRUE zu setzen, da sonst CTRL-Z überhaupt nicht als gültige Eingabe akzeptiert wird und daher auch nicht erkannt werden kann.

Die nächste Idee könnte sein, die Situation mit einem Kunstgriff zu bereinigen, der nicht ganz den Prinzipien strukturierten Programmierens entspricht: Man setzt einfach die Variable `Zahl1`, falls sie durch CTRL-Z-Eingabe 0 wurde, auf einen gültigen Wert, zum Beispiel 1, damit die Schleife in der Prozedur `Lesen` verlassen wird.

```
PROGRAM LesProc1;
VAR Zahl1: INTEGER;
PROCEDURE Lesen;
BEGIN Zahl1:=1000;
  WHILE (Zahl1<1) OR (Zahl1>999) DO
  BEGIN WRITE ('Zahl eingeben: ');
    READ (Zahl1);
    IF Zahl1=0
    THEN Zahl1:=1 {Zahl gültig machen}
  END
END;
BEGIN Lesen;
  WHILE NOT EOF DO
  BEGIN WRITELN ('Die gelesene Zahl ist ',Zahl1);
    Lesen
  END
END.
```

Das gleiche Prinzip kann natürlich auch für die Variante mit REPEAT-Einleseschleife angewendet werden, jedoch zeigt sich hier für diesen speziellen Fall, daß die WHILE-Einleseschleife, weil sie kopfgesteuert ist, zu bevorzugen ist: die REPEAT-Schleife wird nämlich jedenfalls betreten, selbst wenn bei der allerersten Eingabe schon CTRL-Z eingegeben wird; dies führt zu unerwünschten Ausgaben:

Zahl eingeben: Die gelesene Zahl ist 1

Zahl eingeben: (nach der ersten Eingabeaufforderung wurde CTRL-Z eingegeben)

```
PROGRAM LesProc0;
VAR Zahl1: INTEGER;
PROCEDURE Lesen;
BEGIN Zahl1:=1000;
  WHILE (Zahl1<1) OR (Zahl1>999) DO
  BEGIN WRITE ('Zahl eingeben: ');
    READ (Zahl1);
    IF Zahl1=0
    THEN Zahl1:=1 {Zahl gültig machen}
  END
END;
BEGIN Lesen;
  REPEAT WRITELN ('Die gelesene Zahl ist ',Zahl1);
    Lesen
  UNTIL EOF
END.
```

Allen Verwirrspielen zum Trotz gibt es aber doch eine einwandfreie Lösung; der Schlüssel liegt gar nicht in der EOF-Behandlung! Der Eintritt in die Lese-/Prüfeschleife beruht, wie schon oben erwähnt, darauf, die Zahl zunächst auf einen ungültigen Wert zu setzen:

```
PROCEDURE Lesen;
BEGIN Zahl1:=1000;
  WHILE ((Zahl1<1) OR (Zahl1>999)) AND NOT EOF DO
  BEGIN WRITE ('Zahl eingeben: ');
    READ (Zahl1)
  END
END;
```

Wenn nun zusätzlich auch noch die Abfrage auf NOT EOF Bestandteil der Laufbedingung geworden ist, ist es entscheidend, wie der logische (BOOLEsche) Ausdruck in der WHILE-Anweisung ausgewertet wird. Im vorstehenden Beispiel ist die Zahl zunächst ungültig, der OR-Ausdruck ist daher TRUE. Er muß noch mit NOT EOF Und-verknüpft werden, also braucht das Programm einen Zugriff auf die Eingabedatei "Tastatur", um den Status "EOF oder nicht?" festzustellen; dies erfordert eine Eingabe, zumindest mittels RETURN-Taste.

Wenn man diesem unerwünschten Verlangen des Programms nach einer mit RETURN abzuschließenden Eingabe Rechnung trägt, ist das Problem mit einem Schlag gelöst:

```
PROCEDURE Lesen;
BEGIN WRITE ('Zahl eingeben: ');
  READ (Zahl1);
```

```
  WHILE ((Zahl1<1) OR (Zahl1>999)) AND NOT EOF DO
  BEGIN WRITE ('Zahl eingeben: ');
    READ (Zahl1);
  END
END;
```

Jetzt wird zunächst einmal gelesen und damit der Status "EOF oder nicht?" hergestellt. Nur wenn gleich bei der ersten Eingabemöglichkeit CTRL-Z eingegeben wurde, ist der Ablauf noch nicht perfekt: Der OR-Ausdruck ist TRUE, NOT EOF aber FALSE; TRUE AND FALSE ergibt natürlich FALSE, also wird die Schleife, wie gewünscht, nicht ausgeführt. Durch den Nebeneffekt des Nullsetzens der Variablen `Zahl1` wird jedoch in der REPEAT-Schleife eine Ausgabe durchgeführt "Die gelesene Zahl ist 0" und eine weitere Eingabe von CTRL-Z verlangt, was wohl unerwünscht ist; wenn man jetzt eine Zahl eingibt, kümmert sich das Programm überhaupt nicht mehr um das vorher eingegebene CTRL-Z! Diese Erscheinung (welche nicht auftritt, wenn beim ersten Lesen eine gültige Zahl eingegeben wurde) kann man unter Beibehaltung der REPEAT-Schleife nur mittels unschöner, unstrukturierter Abfrage beseitigen. In diesem Fall erweist sich die WHILE-Schleife als geeigneter:

```
VAR Zahl1: INTEGER;
PROCEDURE Lesen;
BEGIN WRITE ('Zahl eingeben: ');
  READ (Zahl1);
  WHILE ((Zahl1<1) OR (Zahl1>999)) AND NOT EOF DO
  BEGIN WRITE ('Zahl eingeben: ');
    READ (Zahl1);
  END
END;
BEGIN Lesen;
  WHILE NOT EOF DO
  BEGIN WRITELN ('Die gelesene Zahl ist ',Zahl1);
    Lesen
  END
END.
```

Und hiermit ist endlich die "perfekte" Lösung gefunden! Man sollte sich abschließend bewußt machen, daß die Hauptschwierigkeit eigentlich daher kam, daß man die von der Urversion stammende Bedingung für das Eintreten in die Schleife `Zahl1:=1000`, also Ungültigmachen der Variablen `Zahl1`, gar nicht mehr in Frage gestellt hat. Es lohnte sich jedenfalls, auch die Auswertung der Laufbedingung nach den Regeln der BOOLEschen Algebra zu analysieren, zum Beispiel bewirkte die Fassung

```
PROCEDURE Lesen;
BEGIN WRITE ('Zahl eingeben: ');
  READ (Zahl1);
  WHILE (Zahl1<1) OR (Zahl1>999) AND NOT EOF DO
  BEGIN WRITE ('Zahl eingeben: ');
```

, daß nach Eingabe von CTRL-Z die Lese-/Prüfeschleife nicht mehr verlassen wird, während

```
PROCEDURE Lesen;
BEGIN WRITE ('Zahl eingeben: ');
  READ (Zahl1);
  WHILE NOT EOF AND (Zahl1<1) OR (Zahl1>999) DO
  BEGIN WRITE ('Zahl eingeben: ');
```

einwandfrei funktioniert. Im ersten Fall ist der erste logische Operand wegen `Zahl1=0` schon TRUE; gleichgültig, ob er mit einem TRUE oder einem FALSE als Ergebnis der Und-Verknüpfung Oder-verknüpft wird, die Laufbedingung bleibt TRUE.

Abhilfe ist auch dadurch möglich, daß man die Oder-Verknüpfung nochmals in Klammern einschließt. Dann ergibt sie für sich TRUE, welches mit NOT EOF = FALSE Und-verknüpft jedenfalls FALSE ergibt, wodurch die Schleife nicht mehr betreten wird.

Die Erklärung für den zweiten Fall ist, daß nach Eingabe von CTRL-Z schon der erste logische Operand FALSE ist und die darauffolgenden Operanden können wegen der Und-Verknüpfung daran nichts mehr ändern.

Abschließend sollte noch darauf hingewiesen werden, daß TURBO-PASCAL normalerweise die Auswertung eines logischen Ausdrucks abbricht, sobald das Ergebnis feststeht; nur wenn man explizit den Compilerschalter `$B+` setzt, erfolgt immer eine vollständige Auswertung. □

Hard-Disk-Management im Novell-Netz des Rechenzentrums im TGM

Walter Riemer, N/NA, TGM

Eines der vielen Computer-Labors im Bereich der Abteilungen Nachrichtentechnik im TGM trägt noch heute den stolzen Namen "Rechenzentrum", weil dort früher eine Siemens-Rechenanlage mit dem Betriebssystem BS 2000 stand. Im dortigen Novell-Netz mit 25 Arbeitsplätzen findet insbesondere der Anfangsunterricht in EDV und Informatik statt. Aus pädagogischen Gründen ist dort nicht jedem Arbeitsplatz eine Benutzerkennung zugeordnet (wie in allen anderen vernetzten PC-Sälen), sondern jeder Benutzer (Schüler und Lehrer) hat eine individuelle Benutzerkennung, die sich bei Schülern aus der Jahrgangsbezeichnung und der zweistelligen Katalognummer zusammensetzt, zum Beispiel 3AN07, welche natürlich durch ein Password abgesichert ist.

Im Zuge des LOGIN wird dem Benutzer ein ihm zugeordnetes Unterverzeichnis auf der "Daten-Festplatte" des Servers U: (U für User) als Laufwerk I: (individuelles Laufwerk) ge"mapt", zum Beispiel U:\3AN\3AN07. Dort und nur dort hat er alle Rechte wie auf einem ihm gehörenden Laufwerk in einem stand-alone-Computer, allerdings begrenzt auf maximal 1,2 MB Speicherkapazität.

Diese Konfiguration bietet den pädagogisch wertvollen Vorteil, daß die Schüler mit dem ihnen zur Verfügung stehenden Massenspeicherplatz umgehen lernen, trotzdem aber auch das Sichern ihrer Daten und Programme auf einen zweiten Datenträger (Diskette) nicht vernachlässigen dürfen, da der Fortbestand der User-Daten nicht garantiert wird (etwa bei Festplatten-Crash).

Unter Berücksichtigung der Erfahrung, daß nicht alle Schüler die ihnen zur Verfügung stehende Kapazität von 1,2 MB wirklich ausnützen, ist das Produkt der Anzahl der Benutzer (ungefähr 600) mit 1,2 MB wesentlich größer als die Kapazität der User-Festplatte. Alle Schüler dürften und könnten also nicht je 1,2 MB abspeichern.

Solange überwiegend PASCAL unterrichtet wurde, bewährte sich dieses System sehr gut, da selten mehr als Quellfiles (.PAS) und Backup-Files (.BAK) auf I: bestehen blieben. Seit Beginn des Schuljahrs 1993/94 wird jedoch auch im Anfängerunterricht ausschließlich C unterrichtet. Dies hat den unangenehmen Nebeneffekt, daß laufend auch .OBJ- und .EXE-Files entstehen, von denen insbesondere die letzteren oft 20 oder 50 oder noch mehr kilo-Bytes groß sind. Da die Gesamt-Plattenkapazität nicht erlaubt, daß jeder Schüler tatsächlich 1,2 MB ausnützt, kann man das dadurch gelegentlich erforderliche Aufräumen auch nicht den Schülern überlassen (wenn das auch pädagogisch von Wert wäre). Ohne entsprechende Maßnahmen wäre die User-Festplatte auf jeden Fall nach etlichen Wochen voll, wobei aber die meisten Dateien (.BAK-, .OBJ- und .EXE-Files) unnötig wären.

Es erwies sich daher als notwendig, das Aufräumen zu automatisieren. Das LOGOUT wird daher mittels einer Batch-Datei vorgenommen, welche zunächst das nachstehende Programm KILLEOBX.EXE mit dem Parameter I: aufruft und dann erst LOGOUT ausführt.

Das Programm enthält einige interessante Besonderheiten und wird deshalb hier veröffentlicht:

Im Hauptprogramm wird nach vorhergehendem Einstellen des als Kommandozeilenparameter angegebenen Wurzelverzeichnisses, von dem an das Löschen der .BAK-, .OBJ- und .EXE-Files erfolgt, das Unterprogramm SubDirs aufgerufen. Dieses hat nun die Aufgabe, sich durch alle eventuell noch angehängten Unterverzeichnisse hindurchzuarbeiten. Dies wird auf rekursivem Weg bewerkstelligt, das heißt, das Unterprogramm SubDirs() ruft sich selbst immer wieder auf, nachdem es im gerade aktuellen Unterverzeichnis alle passenden Dateien desselben Tages gelöscht hat (Aufruf der Funktion KillFiles()). Dann wird geprüft, ob es noch ein angehängtes Unterverzeichnis beliebigen Namens (*.*) gibt; wenn nicht, dann wird zum aufrufenden Programm zurückgekehrt. Andernfalls werden alle Unterverzeichnisse nacheinander (mittels do-while-Schleife) bearbeitet und dabei insbesondere SubDirs() selbst immer wieder (rekursiv) aufgerufen.

Die Funktion KillFiles() eruiert zunächst für jede Datei einen Handle, um damit Zugriff zum Dateidatum zu erhalten. Das Dateidatum wird

anschließend mit dem aktuellen Datum verglichen; im Fall der Übereinstimmung wird ein DOS-Befehl DEL aufgebaut und die Datei mittels der system()-Funktion gelöscht. Anschließend wird noch eine entsprechende Meldung auf den Bildschirm geschrieben (die allerdings im Netz durch den Operanden >NUL im Batch-File unterdrückt wird).

Die Funktion RootEinstellen() übernimmt den Kommandozeilenparameter vom Hauptprogramm und stellt das gewünschte Verzeichnis ein, gleichzeitig wird auch die Zeichenkette für den in der Funktion KillFiles() auszuführenden DEL-Befehl vorbereitet.

KILLEOBX.C

löscht alle Dateien mit Erstellungsdatum = heutiger Tag aus dem angegebenen Verzeichnis sowie allen untergeordneten Unterverzeichnissen.

Aufruf: KILLEOBX wurzelverzeichnis

Beispiel KILLEOBX C:\TC2

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
#include <dos.h>
#include <dir.h>
#include <stdlib.h>
#include <io.h>
#include <fcntl.h>

#define IsSubDir 0x10

struct date Heute;

void SubDirs (void);
void KillFiles(void);
void RootEinstellen(char *RootPfad);

void main (int KdoCount, char **KdoZeile)
{ int Zaehler;
  char RootPfad[MAXPATH];
  getdate(&Heute);
  if (KdoCount == 2)
  { strcpy (RootPfad, KdoZeile[1]);
    strupr(RootPfad);
    RootEinstellen (RootPfad);
    SubDirs ();
  }
  else
    printf ("Aufruf: KILLEOB rootDirectory\n");
  puts("\n");
}

void SubDirs (void)
{ struct ffblk FFStruktur;
  KillFiles();
  if (findfirst("*.*", &FFStruktur, IsSubDir))
    return;
  do
  { if (FFStruktur.ff_attrib != IsSubDir ||
      !strcmp (FFStruktur.ff_name, ".") ||
      !strcmp (FFStruktur.ff_name, "..") )
    continue;
    chdir (FFStruktur.ff_name);
    SubDirs ();
    chdir ( ".." );
  }
  while (!findnext (&FFStruktur));
  return;
}
```

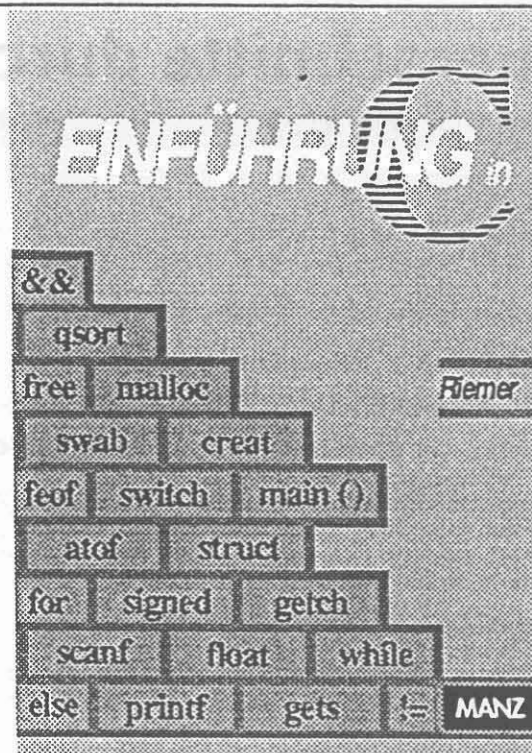
Schluß des Beitrags auf Seite 78

Walter Riemer

EINFÜHRUNG IN C

Das Lehr- und Arbeitsbuch wurde für alle geschrieben, die die heute führende Universalprogrammiersprache C bzw. C++ erlernen wollen, aber insofern Schwierigkeiten haben,

- als sie sich in den Handbüchern nicht zurechtfinden, weil diese zu wenig dem praktischen Programmieren entgegenkommen,
- aber andererseits auch mit den vorhandenen Lehrbüchern nicht zurechtkommen, weil sie in der Darstellung und Symbolik den Handbüchern zu ähnlich sind und hier zuviel ins Detail gehen, dort wieder zu wenig auf die praktische Umsetzung eingehen.



DIN A4, 233 Seiten, S 235,-

Diese Einführung geht an die recht komplexe Programmiersprache C in der Weise heran, daß sie sich an häufig vorkommenden Problemstellungen orientiert und dafür Lösungen anbietet. Es wird nicht der Versuch gemacht, die Programmiersprache C in allen Details präzise zu beschreiben, sondern eher wichtige und insbesondere unerwartete oder schwer verständliche Eigenschaften der Sprache C praxisgerecht (und daher in manchen Fällen eingeschränkt auf die wichtigste von allenfalls mehreren möglichen Varianten) darzulegen.

Hinsichtlich der Anführung von Parametern der C-Funktionen wird nicht ganz präzise, dafür aber für den Lernenden wesentlich anschaulicher als in den Handbüchern vorgegangen: die Parameter haben einfach sprechende Namen; ob jedoch der Name eines Datenobjektes direkt oder mittels Adreßschreibweise oder ob allenfalls ein Zeiger zu übergeben ist, ist der näheren Beschreibung und insbesondere dem zugeordneten Beispiel zu entnehmen. Sicherlich wird jedermann die Spezifikation

leichter verstehen als `qsort (schlüsselpointer, arraypointer, anzahl, größe, vergleich):`
`void qsort (void *base, size_t nelem, size_t width,`
`int (*fcmp) (const void *, const void *):`

Das Buch enthält 94 Beispiele, die auch auf Diskette verfügbar sind.

C++ wird am Schluß aufbauend auf C dargestellt. Hierbei wird auch auf die Techniken des objektorientierten Programmierens eingegangen.

Ein ausführliches **Sachverzeichnis** hilft beim Auffinden der gewünschten Problemlösung.

BESTELLCOUPON

MANZ

Verlags- und
Universitätsbuchhandlung
Verlag Schulbuch
Kohlmarkt 16
1014 Wien

Ich bestelle hiermit Exemplar(e) von
W. Riemer, EINFÜHRUNG IN C à S 235,-
sowie Stück Beispieldiskette(n) à S 160,-

Name: _____

Adresse: _____

Unterschrift: _____

Datum: _____

Querschnitte durch C

Franz Fiala, EN, TGM

Dieser Kurzbeitrag ist im Rahmen des C/C++-Seminars des PCC-TGM entstanden und beschreibt die verschiedenartige Bedeutung der Schlüsselwörter **const** und **static**. Nicht nur, daß den Autoren von C eine schreibökonomische Ausdrucksweise vorschwebte, auch verwenden sie dieselben Schlüsselwörter in einem zwar ähnlichen Zusammenhang, der aber doch - je nach Stellung - etwas anderes ausdrückt, sodaß es für Uneingeweihte zu einer unklaren Interpretation kommen kann.

const

(1) const zur Kennzeichnung unveränderlicher Variablen

```
const int SIZE = 5;
```

SIZE kann in der Folge wie eine Variable verwendet werden und wird auch beim Debuggen als Bezeichner erkannt. Bisher hat man in C dafür eine **#define**-Präprozessor-Anweisung verwendet.

```
#define SIZE 5
```

Der Nachteil: SIZE hat keinen Typ, es wird nur die Buchstabenkombination SIZE durch 5 ersetzt. Es kann keinerlei Typenprüfung erfolgen. SIZE ist beim Debuggen nicht sichtbar und wird durch 5 ersetzt.

(2) const zur Bezeichnung unveränderlicher Pointer

```
char *const ptr = mybuf;
```

Diese Zeile besagt, daß ptr nicht verändert werden kann, d.h. der Ausdruck `ptr=newbuf;` vom Compiler als unerlaubt eingestuft wird. Dagegen ist `*ptr='A';` erlaubt.

(3) const zur Bezeichnung eines Pointers auf eine Konstante

```
const char *ptr = mybuf;
```

Diese Zeile besagt, daß ptr auf die Speicherstelle mybuf zeigt, die nicht verändert werden darf. In diesem Fall ist der Ausdruck `ptr=newbuf;` erlaubt, dabei zeigt jetzt der Pointer ptr auf einen neuen Puffer newbuf, der ebenso wie mybuf konstant sein muß. Dagegen ist `*ptr='A';` nicht erlaubt, da dadurch der Puffer verändert werden würde.

(4) const bei Parameterübergabe

Bei der Parameterübergabe bei Funktionen dient **const** als Hinweis, daß eine Funktion einen übergebenen Parameter nicht verändern darf.

```
void fkt(const char a, char b);
```

In diesem Prototyp wird ausgedrückt, daß die Funktion den Parameter a nicht verändern darf, den Parameter b aber schon. Das ist aber bei dieser Übergabe als Wert nicht wichtig, da a und b lokale Parameter sind und ohnehin keinerlei Rückwirkung auf die rufende Funktion haben. Bedeutender wird **const** im folgenden Fall:

```
void fkt(char * const a, char *b);
```

In diesem Beispiel werden zwei Pointer übergeben; die Daten auf die a zeigt können durch die Funktion nicht verändert werden; die Daten, auf die b zeigt können verändert werden.

Ebenso wirkt der Bezeichner **const** im Zusammenhang mit Referenzen.

```
void fkt(const char &a, char &b);
```

Eine neue Dimension erhält **const** im Zusammenhang mit Klassen:

(5) const im Zusammenhang mit Objekten

Ebenso wie bei Variablen kann **const** auch bei Objekten angewendet werden. Eine Klasse DATUM kann folgendermaßen angewendet werden:

```
const DATUM Geburtstag (12,11,48);
```

Die Werte von Geburtstag können nicht geändert werden.

Hier kommt es aber zu einer zusätzlichen Schwierigkeit: Elementfunktionen von konstanten Objekten können nicht gelesen werden! Warum das, wird man sich fragen? Der Compiler kann bei konstanten Variablen alle Operationen identifizieren, die ihre Veränderung bewirken; er kann aber nicht wissen, welche der Elementfunktionen Veränderungen an den Daten des Objekts vornehmen. Daher verlegt er sich auf die sichere Seite und erlaubt gar keinen Zugriff auf Elementfunktionen. Um dennoch mit konstanten Objekten umgehen zu können, wird der Bezeichner zur Kennzeichnung jener Elementfunktionen verwendet, deren Verwendung auch bei konstanten Objekten sicher ist.

```
class DATUM
{
public:
    DATUM(int t, int m, int j);
    int LTag() const;
    int LMonat() const;
    int LJahr() const;
    void STag(int t);
    void SMonat(int m);
    void SJahr(int j);
private:
    int Tag, Monat, Jahr;
};
```

In diesem Beispiel können bei konstanten Objekten die mit **const** bezeichneten Funktionen LTag, LMonat, LJahr zu Lesen der Daten verwendet werden, dagegen können die Funktionen STag, SMonat und SJahr nicht angewendet werden.

Bei nicht-konstanten Objekten ist der Bezeichner **const** ohne Bedeutung. Der Bezeichner **const** muß gleichzeitig in der Deklaration und auch in der Definition angewendet werden. Würde eine der mit **const** gekennzeichneten Elementfunktionen eine private Variable verändern wollen, würde dies der Compiler ebenso ahnden, wie wenn ein konstantes Objekt dieser Klasse versuchen würde, eine nicht konstante Elementfunktion aufzurufen.

Regel: man sollte den Bezeichner **const** wo immer anwendbar verwenden, um dem Compiler die Möglichkeit zu geben, Flüchtigkeitsfehler beim Programmieren zu verhindern.

static

Es gibt einen weiteren Bezeichner, nämlich **static**, der in unterschiedlichen Stellen unterschiedliche Wirkung zeigt:

(1) static global

Der Bezeichner **static** drückt im Zusammenhang mit globalen Variablen (Variablen außerhalb von Funktionen) oder Funktionen aus, daß der Gültigkeitsbereich dieser Variablen oder dieser Funktion auf den aktuellen Modul(Datei) beschränkt ist und der Name der Variablen oder der Funktion nicht an den Linker weitergegeben wird. Das ist eine einfache Form von Kapselung einer Variablen oder einer Funktion in C und Schutz vor versehentlichem Zugriff in einem anderen Modul.

DATEI1	DATEI2
<code>static int i;</code>	<code>static int i;</code>
<code>static void func(void);</code>	<code>static void func(void);</code>

Die beiden `i` und die beiden `func()` kommen einander auch nach dem Binden der beiden Module nicht in die Quere.

(2) static innerhalb von Funktionen

Der Bezeichner **static** innerhalb von Funktionen bewirkt, daß die so gekennzeichnete Variable die Lebensdauer des Programms bekommt und nicht automatisch nach Ablauf der Funktion wieder verschwindet.

Beispiel:

Zählen, wie oft eine Funktion aufgerufen wurde:

<code>void fkt(void)</code>	<code>void fkt(void)</code>
<code>{</code>	<code>{</code>
<code> static count=0;</code>	<code> count=0;</code>
<code> count++;</code>	<code> count++;</code>
<code>}</code>	<code>}</code>
 <code>count in fkt()</code>	 <code>count in fkt() ohne static</code>
 0	 ?

<code>void main(void)</code>		
<code>{</code>		
<code> fkt();</code>	1	1 ?
<code> fkt();</code>	2	1 ?
<code>}</code>		

Die lokale Variable `count` wird ohne den Bezeichner **static** immer wieder auf den Wert 0 initialisiert, daher hat `count` nach `count++`; immer den Wert 1.

Hat die Variable `count` den Bezeichner **static**, wird bei Programmbeginn `count` auf 0 gesetzt. `count` existiert also schon bevor die Funktion gerufen wurde. Jeder Aufruf von `fkt()` bewirkt die Erhöhung von `count`, so als wäre `count` eine globale Variable. Gegenüber einer globalen Variablen genießt `count` jedoch den Vorzug, nur innerhalb der Grenzen von `fkt()` zu gelten.

(3) static in Klassen

Jedes Objekt einer Klasse enthält normalerweise einen Satz aller Variablen dieser Klasse; es gibt also ebenso viele Variablen als es Objekte dieser Klasse gibt. Ist aber eine Variable als **static** deklariert, existiert diese Variable nur einmal, unabhängig von der Zahl der Objekte.

Ähnlich, wie schon bei **static** innerhalb von Funktionen, kann **static** innerhalb von Klassen dazu benutzt werden, zu zählen, wie viele Objekte dieser Klasse existieren. Es sind zwei Fälle möglich:

Man kann sagen, daß eine mit **static** gekennzeichnete Variable innerhalb einer Klasse etwa so wirkt, wie eine gewöhnliche globale Variable, die aber den eingeschränkten Geltungsbereich der Klasse besitzt inklusive der Zugriffsbeschränkungen **public** oder **private**.

(3a) static als öffentliches Datum

```
class MYCLASS
{
public:
  static int count;
};
```

In diesem Fall kann die Variable `count` von aussen initialisiert werden:

```
MYCLASS m;
m.count = 0;
```

Diese Schreibweise ist allerdings zu vermeiden, erweckt man doch den Eindruck, `count` wäre eine Variable von `m` allein. Besser ist es, den Klassennamen zu Kennzeichnung zu verwenden, um anzuzeigen, daß es sich um eine Variable handelt, die allen Objekten dieser Klasse gemeinsam ist:

```
MYCLASS::count = 0;
```

Man kann eine Variable mit dem Bezeichner **static** nicht durch einen Konstruktor initialisieren, wie denn auch, es werden ja viele Konstruktoren gerufen aber eine Variable kann nur einmal initialisiert werden.

(3b) static als privates Datum

```
class MYCLASS
{
private:
  static int count;
};
```

Private Variablen mit dem Bezeichner **static** werden außerhalb von Funktionen und Klassen wie eine globale Variable mit voller Bezeichnung initialisiert:

```
int MYCLASS::count = 0;
```

Für diesen Sonderfall ist der Zugriff auf eine private Variable einer Klasse erlaubt. Es ist zu beachten, daß diese Zeile die Variable deklariert und initialisiert und nicht die entsprechende Zeile innerhalb der Klasse.

(3c) static als Bezeichner für eine Funktion

Eine Funktion einer Klasse, die lediglich die mit **static** bezeichneten Variablen bearbeitet kann ebenso den Bezeichner **static** erhalten. Das hat zur Folge, daß diese Funktion auch wirklich nur die Variablen mit **static** erreichen kann, nicht die anderen. Während alle anderen Funktionen einen `this`-Pointer besitzen, besitzen **static**-Funktionen keinen.

Folgendes Beispiel zeigt eine Klasse `BALL`, die zählt, wieviele Bälle kreiert wurden:

```
class BALL
{
public:
  BALL() { count++; }
  ~BALL() { count--; }
  static int WieViele() { return count; }
private:
  static int count;
};

int BALL::count = 0;
```

Die Funktion `WieViele()` gibt Aufschluß über die Anzahl existierender Bälle. □

Hardwarenahe Programmierung

Teil 5: Hauptspeicherzugriffe

Franz Fiala, N, TGM

DSK-372\HARD5.LZH

In dieser Reihe bereits erschienen:

- PC-NEWS-26: Teil 1 Eine Rundschau
- PC-NEWS-27: Teil 2 Der PC
- PC-NEWS-28: Teil 3 Sprache für hardwarenahe Programmierung
- PC-NEWS-30: Teil 4 Hardwareprogrammierung, IO-Zugriffe

Genauso, wie die Ein/Ausgabe in die Register der Peripheriebausteine, können wir den Speicher des PC direkt lesen und schreiben. Überlegen wir, wann wir wohin ungestraft zugreifen können (Lesen wird ja wohl überall erlaubt sein! oder?). Die Aufteilung des Hauptspeichers kennen wir schon aus unserem zweiten Teil (PC-NEWS-92/2, S.33). Unser Programm steht irgendwo in der TPA (Transient Program Area) aber wo?

Segmentadresse, HC05MM1.CPP (und HC05MM1.C)

Innerhalb eines C-Programms erfolgt der Zugriff auf Speicheradressen über dem Umweg über Variablen, deren Adresse man eigentlich gar nicht kennt. Wie kann man nun die **Adresse des eigenen Programms** feststellen? Dazu existieren in jeder Sprache eigene, i.a. maschinenabhängige Verfahren. C bietet die Funktion `segread()` zur Feststellung der Segmentadressen an.

Das folgende Programm zeigt den Wert der Segmentregister an. **Achtung:** Wenn man das Speichermodell unter 'Optionen' verändert, wird das Programm nicht noch einmal kompiliert, da aus der Sicht der Compiler das vorher kompilierte Programm durchaus noch gültig ist, daher nach dem Ändern des Modells die .EXE-Datei löschen oder mit Build-All neu compilieren.

```
/* HC05MM1.CPP */
/*
 * Wo bin ich im Hauptspeicher ?
 * =====
 */
#include <dos.h>
#define P_STREAMS
#include <portable.h>
#include <mytypes.h>

VOID main(VOID)
{
    SREGS s;
    constream con;
    con.clrscr();
    con << "Wo bin ich, im Hauptspeicher?\n"
        "Versuchen Sie dieses Programm mit "
        "allen Speicher-Modellen auszuführen!\n";
    segread(&s);
    con << "CS: " << hex << s.cs << endl;
    con << "DS: " << hex << s.ds << endl;
    con << "ES: " << hex << s.es << endl;
    con << "SS: " << hex << s.ss << endl;
    cin.get();
    con.clrscr();
}
```

Adresse einer Variablen

Die Adresse jeder C-Variablen kann über den Adreßoperator & bestimmt werden.

```
int i;
...
printf("Adresse von i ist %p\n", &i);
```

Je nach gewähltem Speichermodell ist diese Adresse ein near- oder far-Zeiger. Nur in den großen Speichermodellen (MEDIUM, LARGE, HUGE) kennt man mit dem Adreßoperator auch gleichzeitig die physikalische Adresse, da im Pointerwert eines far-Pointers sowohl das Segment als auch der Offset der Adresse enthalten sind. Bei den kleinen Code-Speicher-Modellen (TINY, SMALL, COMPACT) ist aber diese Adresse nur der Offset, das zugehörige Datensegment muß mit `segread()`, wie im vorigen Beispiel gebildet werden.

Wie ist der Compiler eingestellt?

Um feststellen zu können, in welcher Betriebsart ein Compiler arbeitet, verwendet man vordefinierte Makros, die entweder das Speichermodell (`__TINY__`, `__SMALL__`, `__COMPACT__`, `__MEDIUM__`, `__LARGE__`, `__HUGE__`) oder die Pointerlänge (`__SDATA__`, `__SCODE__`, `__LDATA__`, `__LCODE__`) spezifizieren. Während die Makros für das Speichermodell immer verfügbar sind, kennen manche Compiler die Makros für die Pointerlänge nicht. In der Headerdatei `portable.h` sind sie folgendermaßen definiert:

```
/*-----
 * Symbols for data-pointer-type
 * __SDATA__ __LDATA__
 * Symbols for code-pointer-type
 * __SCODE__ __LCODE__
 *-----*/
#ifndef __DOS_H
#include <dos.h>
#endif

#if !defined(__SDATA__) && !defined(__LDATA__)
#if defined(__MEDIUM__) || \
    defined(__SMALL__) || \
    defined(__TINY__)

#define __SDATA__ 1

#endif
#if defined(__COMPACT__) || \
    defined(__LARGE__) || \
    defined(__HUGE__)

#define __LDATA__ 1

#endif
#endif
#if !defined(__SCODE__) && \
    !defined(__LCODE__)

#if defined(__COMPACT__) || \
    defined(__SMALL__) || \
    defined(__TINY__)

#define __SCODE__ 1

#endif
#if defined(__MEDIUM__) || \
    defined(__LARGE__) || \
    defined(__HUGE__)

#define __LCODE__ 1

#endif
#endif
#endif
```


Modellunabhängige Adreßbestimmung

Ein Kode, der die physikalische Adresse unabhängig vom gewählten Speichermodell immer richtig bestimmt, muß daher in Abhängigkeit vom eingestellten Speichermodell kompiliert werden. Eine solche Technik benötigt man, wenn man Bibliotheken aufbaut, die für jeden späteren Betriebsfall geeignet sind.

```
/* HC05AD1.CPP */
/*
 * Modellunabhängige Bestimmung der physikalischen Adresse
 * einer Variablen
 */
#include <stdio.h>
#include <conio.h>

#define P_POINTER
#include <portable.h>

void main(void)
{
    int variable;

#ifdef __SDATA__
    struct SREGS s;
#endif

    printf("Die Adresse der Variablen variable ist ");

#ifdef __SDATA__
    segread(&s);
    printf("%4x:%4x\n", s.ds, &variable);
#endif

#ifdef __LDATA__
    printf("%lp\n", &variable);
#endif
    getch();
}
```

Der genaue Wert einer Variablenadresse ist - wenn es sich um in C definierte Variablen handelt - meistens ohne Bedeutung. Anders ist es, wenn man auf genau definierten Adressen im Speicher, etwa im BIOS-Datenbereich oder im Bereich der Interrupt-Vektoren lesen oder schreiben will.

Zugriff auf absolute Speicheradressen

Das in C verwendete Zugriffsverfahren auf absolute Speicheradressen basiert auf der Anwendung von far-Zeigern mit 20-bit-Adresse, bestehend aus Segment und Offset, deren Ziel mit dem dafür vorgesehenen Makro MK_FP() eingestellt wird. Umgekehrt können Segment und Offset eines far-Zeigers mit den Makros FP_OFF() und FP_SEG() wieder zerlegt werden. Das folgende Makro aus portable.h definiert MK_FP(), wenn es nicht existiert:

```
#if !defined(MK_FP)
#define MK_FP(seg, off) \
    ((void far *)(((long)(seg)<<16)|(UINT)(off)))
#endif
```

Die Speicherbereiche im PC werden von verschiedenen Programmteilen kontrolliert. Teilweise ist ihre Position CPU-bedingt (Interrupt-Vektoren 00000..003ff oder Reset-Vektor ffff0), teilweise BIOS-bedingt (BIOS-Daten 00400..004ff), teilweise Hardware-bedingt (diverse BIOS, Video-Bereich a0000..bffff), teilweise durch DOS (TPA xxxxx..9fff), teilweise durch neuere Programmkonzepte (HIMEM 100000..10ffff, Upper-Memory-Blocks z.B. EMM386 a0000..fffff).

DOS stellt dem aktuellen Programm einen Speicherblock zur Verfügung und steuert diesen Speicherblock durch einen Memory-Control-Block (MCB). Über diese MCBs werden wir später mehr erfahren. Wenn also das eigene Programm gestartet wird, könnte es interessant sein, zu wissen, wo es sich befindet. Wann das wichtig ist? Zum Beispiel, wenn man einer BIOS-Routine Parameter zu übergeben hat, die sich aus Adressen von Variablen oder Programmen eines C-Programmes zusammensetzen.

Speicherblöcke beginnen immer auf ganzzahligen Vielfachen von 16. Daher benötigt man nur eine Segmentadresse, der Offsetanteil des Programmbeginns ist immer 0.

Das Lesen und Schreiben im Hauptspeicher erfolgt immer über einen Pointer, der mit MK_FP() gebildet wird.

Betrachten wir beispielsweise den Bildschirm! Ab Adresse 0xb8000 beginnt der Bildschirmspeicher, mit jeweils einem Byte Zeichenstelle und einem Byte Attribut. Der Typ des zu definierenden Pointers wird durch die Daten bestimmt, die man zuzugreifen gedenkt. Je nach Typ

des Pointers ist die Wirkung der Pointerarithmetik unterschiedlich. In unserem Fall bieten sich drei Möglichkeiten an:

Man betrachtet den Inhalt des Bildschirmspeichers als eine Folge

a. ganzer Zahlen ohne Vorzeichen

Man erhält mit jedem Zugriff eine ganze Zahl, deren niederwertiger Teil das Zeichen und deren höherwertiger Teil das Attribut ist.

b. von Bytes

Man erhält bei einem Zugriff entweder das Zeichen oder das Attribut

c. von Strukturvariablen

Man erhält beides: Zeichen und Attribut und muß sich nicht mehr darum kümmern, wie man die Anteile zu trennen hat wie bei a.

```
/* HC05MM2A.CPP */
/*
 * Bildspeicher x 3
 */
#include <dos.h>
#define P_STREAMS
#include <portable.h>
#include <mytypes.h>

typedef struct
{
    UCHAR Text;
    UCHAR Attr;
} BILDP;

VOID main(VOID)
{
    UINT far * uifp; // Pointer auf ganze Zahlen
    UCHAR far * ucfp; // Pointer auf Bytes
    BILDP far * bfp; // Pointer auf Strukturen aus Text
    UCHAR c,a; // und Attribut

    constream con;
    con.clrscr();
    con << "A Zeichen links oben am Bildschirm: " << endl;
    uifp = (UINT far *)MK_FP(0xb800,0x0000);
    ucfp = (UCHAR far *)MK_FP(0xb800,0x0000);
    bfp = (BILDP far *)MK_FP(0xb800,0x0000);

    // a.) Bild besteht aus einer Folge ganzer Zahlen
    c = (UCHAR)(*uifp);
    a = (UCHAR)((*uifp)>>8);
    con << " Zeichen : " << hex << setw(2) << (UINT)c
    << " Attribut: " << hex << setw(2) << (UINT)a << endl;

    // b.) Bild besteht aus einer Folge von Bytes
    c = *ucfp;
    ucfp++;
    a = *ucfp;
    con << " Zeichen : " << hex << setw(2) << (UINT)c
    << " Attribut: " << hex << setw(2) << (UINT)a << endl;

    // c.) Bild besteht aus einer Folge von Strukturvariablen
    c = bfp->Text;
    a = bfp->Attr;
    con << " Zeichen : " << hex << setw(2) << (UINT)c
    << " Attribut: " << hex << setw(2) << (UINT)a << endl;

    con << "Ende mit Taste";
    cin.get();
}
```

Es ist nicht zu übersehen, daß es am übersichtlichsten ist, Strukturen für häufig vorkommende Variablenanordnungen zu definieren, statt einen Bereich geschlossen anzusprechen, wie unter a. oder sich auf eine byteweise Analyse, wie unter b. einzulassen.

In ähnlicher Weise verfährt man bei allen festen Speicheradressen.

Absolut adressierbare Speicherbereiche

Bevor wir das tun, sollten wir uns über die Speicherbereiche informieren, die es gibt: PC-NEWS-92/2, S.33-34. Jetzt ist es an der Zeit diese auch detaillierter zu behandeln.

Interruptvektoren

Zunächst sind da die 256 **Interruptvektoren** ab Adresse 0, jeweils 4 Bytes pro Vektor, daher 1 kByte insgesamt. Jeder Vektor besteht aus 2 Bytes Offset und Segment, hier wieder immer niederwertiges Byte zuerst. Diese 256 Vektoren werden in einem anderen Teil unserer Folge genauer behandelt.

Was sind Interruptvektoren?

Vektoren sind eine bewährte Möglichkeit, Tabellen und Programme auf verschieblichen Adressen unterzubringen und dennoch durch den Vektor eine ortsfeste Referenz darauf zu haben. Die ursprüngliche Aufgabe von Interruptvektoren ist, daß sie durch Hardwareereignisse ausgelöst werden und unabhängig vom gerade laufenden Programm ein kurzes Interruptserviceprogramm ausführen, welches das laufende Programm unmerklich unterbricht und danach wieder zu diesem zurückkehrt. D.h. der Benutzer bemerkt (fast) nichts von diesen Hintergrund-'Heinzelmännchen', es sei denn, es gibt zu viele dieser Unterbrechungen, dann kann die Vordergrundaktivität auch einmal stark verlangsamt werden.

Die Interruptvektoren bei 80x86 CPUs beginnen bei der absoluten Adresse 0x0000 und enden bei der absoluten Adresse 0x003ff. Diese Anordnung ist durch den Befehlssatz der CPU vorgegeben und kann nicht geändert werden. Jeder Vektor besteht aus 4 Bytes, es gibt daher 256 oder 0x100 Vektoren. Ein Interruptvektor besteht aus Segment und Offset in umgekehrter Anordnung und auch die Halbbytes von Segment und Offset sind jeweils umgekehrt, das niederwertige Byte kommt zuerst.

```

+=====+
|off lo| <---Adresse des Vektors = 4 * Interruptnummer = a
+-----+
|off hi| <--- a+1
+=====+
|seg lo| <--- a+2
+-----+
|seg hi| <--- a+3
+=====+

```

Jeder Vektor belegt 4 Bytes im Speicher. Normalerweise enthält ein Interruptvektor eine Adresse eines Interruptserviceprogramms.

Befehl zum Aufruf der Interruptserviceroutine in Assembler

```

INT Interruptnummer
oder in C

```

```

int86(int intno,...);

```

Die Aufgabe, die diesen Interrupts zukommt, ist bei den ersten bereits durch die CPU-Konzeption vorgegeben, alle weiteren zunächst durch das BIOS dann durch das jeweils geladene Betriebssystem, oder zusätzlich geladene Treiber (Maus, Netzwerk, EMS, XMS..). Jedes geladene Programm kann die Interrupts so beibehalten wie sie sind oder diese ändern und auf Routinen umlenken, die erst später geladen werden. Die frisch geladenen Routinen können vor/nach/statt dem alten Treiber ausgeführt werden.

Einige dieser Vektoren sind nicht Zeiger auf Programme sondern auf Daten. Diese wollen wir hier besprechen.

Interrupt Nummer	Adresse	Inhalt
0x1D	0x00074	Video Initialisierung
0x1E	0x00078	Diskette Parameter Table
0x1F	0x0007C	CGA Graphics Font
0x41	0x00104	Disk #1 Parm Table
0x43	0x0010C	EGA Parm Table Ptr
0x44	0x00110	EGA Graphics Font
0x46	0x00118	Disk #2 Parm Table

Diese Adressen wurden dem XT-AT-Handbuch (siehe **PC-NEWS**-31, Seite 10) entnommen.

Untersuchen von Speicheradressen mit DEBUG

Wie kann man nun diese Daten ansehen? Händisch ganz einfach: Mit DEBUG! Beispiel Disketten-Parameter-Tafel 2, die die Parameter der Festplatten enthält:

```

C:\>DEBUG
-D0:118
0000:0110          01 E5 00 F0 E7 98 00 F0 .....
0000:0120 E7 98 00 F0 E7 98 00 F0 E7 98 00 F0 .....
0000:0130 E7 98 00 F0 E7 98 00 F0 E7 98 00 F0 .....
0000:0140 E7 98 00 F0 E7 98 00 FF E7 98 00 F0 .....
0000:0150 E7 98 00 F0 E7 98 00 F0 E7 98 00 F0 .....
0000:0160 E7 98 00 F0 E7 98 00 F0 E7 98 00 F0 .....
0000:0170 E7 98 00 F0 E7 98 00 F0 E7 98 00 F0 .....
-df00:e500
F000:E500 00 01 03 05 00 00 2C 01-00 00 00 00 01 03 11 .....
F000:E510 00 01 03 07 00 00 FF FF-00 00 00 00 00 01 03 11 .....
F000:E520 00 00 04 07 00 00 00 02-00 00 00 00 00 FF 03 11 .....
F000:E530 00 00 02 05 00 00 2C 01-00 00 00 00 00 DC 02 11 .....
F000:E540 00 00 02 07 00 00 2C 01-00 00 00 00 00 DC 02 11 .....
F000:E550 00 00 02 05 00 00 2C 01-00 00 00 00 00 DD 02 11 .....
F000:E560 00 32 01 04 00 00 00 00-00 00 00 00 50 01 11 ..2.....P..
F000:E570 00 64 02 04 00 00 31 01-00 00 00 00 97 02 11 ..d...1.....
-q

```

Die ab F000:E500 stehenden Daten sind die Kennwerte der Festplattenparameter. Typisch ist die letzte Eintragung 0x11 = 17 = Anzahl der Sektoren.

Liste der Interruptvektoren, HC05MM4.CPP, MM4.C, MM4A.C

Da die Anzeige aller Interruptvektoren gleichzeitig auf einem Bildschirm nicht möglich ist, werden hier zwei unterschiedliche Lösungen gezeigt. In HC05MM4.CPP erfolgt die Anzeige der 256 Vektoren in 4 Bildschirmseiten, nach jedem Bildschirm wird auf eine Taste gewartet. HC05MM4A.C erlaubt die Eingabe einer Vektornummer (dezimal), ab der die nächsten 64 Vektoren angezeigt werden.

```

/* HC05MM4.CPP */
/*
 * Anzeige der Interruptvektoren
 * =====
 */

#include <mytypes.h>
#include <constrea.h>
#define P_POINTER
#include <portable.h>

#define COLW 13
#define COL 4
#define ROW 16

VOID main(VOID)
{
    INT num, row, col;
    ULONG far * ip;

    constream con;
    con.clrscr();
    con << "Interruptvektoren\n";

    ip=(ULONG far *)MK_FP(0,0);
    for (num=0; num<0x100; num++)
    {
        col=(num/ROW)%COL; row=num%ROW;
        con << setxy(1+col*COLW,row+2) << hex << setw(2)
            << num << ':'
            << setw(8) << setfill('0') << *ip;
        ip++;
        if ((col==(COL-1)) && (row==(ROW-1)))
        {
            con << setxy(1,wherey()+1) << "Taste";
            cin.get();
        }
    }
}

```

und wie geht's in C?

```

/* HC05MM4.C */
/*
 * Anzeige der Interruptvektoren
 * =====
 */
...
for (num=0; num<0x100; num++)
{
    col=(num/ROW)%COL; row=num%ROW;
    gotoxy(1+col*COLW,row+2);
    cprintf("%2x %Fp",num,*ip);
    ip++;
    if ((col==(COL-1)) && (row==(ROW-1)))
    {
        gotoxy(1,wherey()+1);
        cprintf("Taste");
        getch();
    }
}
...

```

```

/* HC05MM4A.C */
/*
 * Liste der Interruptvektoren
 * =====
 */
#include <stdlib.h>
#include <stdio.h>

void main (int argc,char *argv[])
{
    int far *p;
    int i,i0,off,seg;

    if (argc==1)
        i0=0;
    else
        i0=atoi(argv[1]);

    printf ("\n\nListe der Interrupt-Vektoren\n");
    printf ("===== \n");
    printf ("Aufruf: intvek      \n" \
        "      Die ersten 64 Vektoren \n" \
        "      intvek iii      \n" \
        "      Die Vektoren beginnend bei iii\n\n");
    for (i=i0;i<i0+64;i++)
    {
        p=(int far *) (long)(i*4);
        off=*p;p++;seg=*p;p++;
        printf ("%2x %3i %4x:%4x",i,i,seg,off);
        if ((i+1)%4) printf (" "); else printf ("\n");
    }
    printf ("\n");
}

```

BIOS-Datensegment

Nach den Interruptvektoren an der Adresse 00400=40:0=0:400 beginnt der 256 Byte große BIOS-Datenbereich, der in seiner Position niemals verändert wird, daher auf absoluten Adressen immer ansprechbar ist (anders als Variablen eines C-Programms, die ja immer andere Adressen aufweisen). Das Feststehen der Adressen hat auch unmittelbar zur Folge, daß die BIOS-Funktionen nicht 'reentrant' sind, sich also nicht selbst aufrufen können und auch nicht von verschiedenen Programmen gleichzeitig benutzbar sind (Multitasking).

BIOS-Datensegment

40:00	word	COM1 port address	These addresses are zeroed out in the OS/2 DOS Compatibility Box if any of the OS/2 COM???.SYS drivers are loaded.
40:02	word	COM2 port address	
40:04	word	COM3 port address	
40:06	word	COM4 port address	
40:08	word	LPT1 port address	
40:0A	word	LPT2 port address	
40:0C	word	LPT3 port address	
40:0E	word	LPT4 port address	(not valid in PS/2 machines)
40:0E	word	PS/2 pointer to 1k extended BIOS Data Area at top of RAM	
40:10	word	equipment flag (see int 11h)	

bits:	
0	1 if floppy drive present (see bits 6&7) 0 if not
1	1 if 80x87 installed (not valid in PCjr)
2,3	system board RAM (not used on AT or PS/2)
	00 16k
	01 32k
	10 48k
	11 64k
4,5	initial video mode
	00 no video adapter
	01 40 column color (PCjr)
	10 80 column color
	11 MDA
6,7	number of diskette drives
	00 1 drive
	01 2 drives
	10 3 drives
	11 4 drives
8	DMA present
	0 DMA present
	1 DMA not present (PCjr)
9,A,B	number of RS232 serial ports
C	game adapter (joystick)
	0 no game adapter
	1 if game adapter
D	serial printer (PCjr only)
	0 no printer
	1 serial printer present
E,F	number of parallel printers installed

note 1) The IBM PC and AT store the settings of the system board switches or CMOS RAM setup information (as obtained by the BIOS in the Power-On Self Test (POST)) at addresses 40:10h and 40:13h. 00000001b indicates "on", 00000000b is "off".

2) CMOS RAM map, PC/AT:

offset	contents
00h	Seconds
01h	Second Alarm
02h	Minutes
03h	Minute Alarm
04h	Hours
05h	Hour Alarm
06h	Day of the Week
07h	Day of the Month
08h	Month
09h	Year
0Ah	Status Register A
0Bh	Status Register B
0Ch	Status Register C
0Dh	Status Register D
0Eh	Diagnostic Status Byte
0Fh	Shutdown Status Byte
10h	Disk Drive Type for Drives A: and B:
	The drive-type bytes use bits 0:3 for the first drive and 4:7 for the other
	Disk drive types:
	00h no drive present
	01h double sided 360k
	02h high capacity (1.2 meg)
	03h-0Fh reserved
11h	(AT):Reserved (PS/2):drive type for hard disk C:
12h	(PS/2):drive type for hard disk D:
	(AT, XT/286):hard disk type for drives C: and D:
	Format of drive-type entry for AT, XT/286:
	0 number of cyls in drive (0-1023 allowed)
	2 number of heads per drive (0-15 allowed)
	3 starting reduced write compensation (not used on AT)
	5 starting cylinder for write compensation
	7 max. ECC data burst length, XT only
	8 control byte
	Bit
	7 disable disk-access retries
	6 disable ECC retries
	5-4 reserved, set to zero
	3 more than 8 heads
	2-0 drive option on XT (not used by AT)
9	timeout value for XT (not used by AT)
12	landing zone cylinder number
14	number of sectors per track (default 17, 0-17 allowed)
13h	Reserved
14h	Equipment Byte (corresponds to sw. 1 on PC and XT)
15h-16h	Base Memory Size (low,high)
17h-18h	Expansion Memory Size (low,high)
19h-20h	Reserved
	(PS/2) POS information Model 50 (60 and 80 use a 2k CMOS RAM that is not accessible through software)
21h-2Dh	Reserved (not checksummed)
2Eh-2Fh	Checksum of Bytes 10 Through 20 (low,high)
30h-31h	Exp. Memory Size as Det. by POST (low,high)
32h	Date Century Byte
33h	Information Flags (set during power-on)
34h-3Fh	Reserved

3) The alarm function is used to drive the BIOS wait function (int 15h function 90h).

4) To access the configuration RAM write the byte address (00-3Fh) you need to access to I/O port 70h, then access the data via I/O port 71h.

5) CMOS RAM chip is a Motorola 146818

6) The equipment byte is used to determine the configuration for the power-on diagnostics.

7) Bytes 00-0Dh are defined by the chip for timing functions, bytes 0Eh-3Fh are defined by IBM.

40:12	byte	number of errors detected by infrared keyboard link (PCjr only)
40:13	word	available memory size in Kbytes (less display RAM in PCjr) this is the value returned by int 12h
40:17	byte	keyboard flag byte 0 (see int 9h)
	bit 7	insert mode on 3 alt pressed
	6	capslock on 2 ctrl pressed
	5	numlock on 1 left shift pressed
	4	scrolllock on 0 right shift pressed
40:18	byte	keyboard flag byte 1 (see int 9h)
	bit 7	insert pressed 3 ctrl-numlock (pause) toggled
	6	capslock pressed 2 PCjr keyboard click active
	5	numlock pressed 1 PCjr ctrl-alt-capslock held
	4	scrolllock pressed 0

Danach ist es zunächst einmal aus mit absolut spezifizierbaren Adreßinhalten. Den folgenden Bereich, bis hinauf zur Adresse 09fff, verwaltet das jeweilige Betriebssystem, in unserem Fall MSDOS. Danach folgen Video-Adapter mit wechselnden Inhalten, siehe Tabelle in Folge 2.

Upper memory

Die folgenden Adressen sind jetzt nicht mehr so ganz verlässlich.

```
C000:001E EGA BIOS signature (the letters IBM)
F000:E05B loc Reset
F000:E2C3 loc NMI Entry Point
```

Hard Disk Information Tables

Jeder Tabelleneintrag enthält 16 Bytes für jede Festplattentypen. Diese Tafel kann mit BIOS-Hersteller und BIOS-Datum anders aufgebaut sein; die hier abgebildete hat als ersten Eintrag 0.

```
F000:E331 dw hdsk_cylinders Number of cylinders, hdsk_type_0
F000:E333 db hdsk_heads Number of heads
F000:E334 dw hdsk_lo_wrt_cyl Low write current cyl begin (XT)
F000:E336 dw hdsk_precomp_cyl Write pre-compensation cylinder
F000:E338 db hdsk_err_length Error correction burst length (XT)
F000:E339 db hdsk_misl_bits
Miscellaneous bit functions:
bits 0-2 disk option, XT only (XT)
0-2 unused, all others
3 = 1 if > 8 heads
4 unused
5 = 1 for bad map at last cylinder + 1
6 or 7 = 1 no retries
F000:E33A db hdsk_timeout Normal timeout (XT)
F000:E33B db hdsk_fmt_timeout Format timeout (XT)
F000:E33C db hdsk_chk_timeout Check timeout (XT)
F000:E33D dw hdsk_parking_cyl Parking cylinder number
F000:E33F dw hdsk_sectr_trac Number of sectors per track
F000:E340 db hdsk_unused Unused
F000:E331 ds hdsk_type_
F000:E6F2 loc Bootstrap Load
System Configuration Table
F000:E6F5 dw Config_tbl_size Size of table in bytes
F000:E6F7 db Config_model Model type
0F8h = 80386 model 70-80 types
0FCh = 80286 model 50-60 types, also most 80286/80386 compatibles
0FAh = 8088/86 model 25-30 type
F000:E6F8 db Config_sub_model Sub-Model type
F000:E6F9 db Config BIOS_rev BIOS revision number
F000:E6FA db Config_features Feature information
bit 7=1, hard disk uses DMA 3
bit 6=1, dual interrupt chips
bit 5=1, has real-time-clock
bit 4=1, int 15h, ah=4Fh is supported (keyboard)
bit 3=1, external wait support
bit 2=1, has extended BIOS RAM
bit 1=1, micro-channel
bit 0=1, unused
F000:E6FB db Config_info_bytes Information bytes (future use)
Baud Rate Table
Table of hex divisors for the serial ports. Table divisors for bauds 110 to 19,200.
F000:E729 dw baud_110, baud_rate_tbl1
F000:E72B dw baud_150
F000:E72D dw baud_300
F000:E72F dw baud_600
F000:E731 dw baud_1200
F000:E733 dw baud_2400
F000:E735 dw baud_4800
F000:E737 dw baud_9600
F000:E739 dw baud_19200
F000:E82E loc Keyboard Function Call
F000:E987 loc Keyboard Hardware Interrupt
F000:EC59 loc Floppy Disk Function Call
F000:EF57 loc Floppy Disk ISR
Floppy Disk Parameters
F000:EFC7 db dsk_info_1 Start of ROM BIOS data areas
hi nibble = stepping rate in ms
lo nibble = head unload time, ms
F000:EFC8 db dsk_info_2 2nd info byte bit 0 = 0 for DMA
F000:EFC9 db dsk_motor_delay
Delay after use for motor off
F000:EFCA db dsk_sectr_bytes
Bytes per sector 0 = 128 bytes
1 = 256 bytes
2 = 512 bytes
3 = 1024 bytes
F000:EFCE db dsk_sector_trac Number of sectors per track
F000:EFCC db dsk_head_gap Gap Length
F000:EFCD db dsk_data_length Data Length
F000:EFCE db dsk_format_gap Format Gap Length
F000:EFCE db dsk_format_byte Format write byte
F000:EFDD db dsk_settling_time Head load time, in ms
F000:EFDD db dsk_startup_time Motor startup wait time, 125ms
F000:EFDD LPT-Fuction-Call
F000:F065 Video Function Call
Video Hardware Registers
F000:F0A4 db video_hdwrbtbl1 mode CGA 40 columns x 25 lines
F000:F0B4 db video_hdwrbtbl2 mode CGA 80 columns x 25 lines
F000:F0C4 db video_hdwrbtbl3 mode CGA graphics
F000:F0D4 db video_hdwrbtbl4 mode MDA 80 columns x 25 lines
F000:F0E4 dw video_buf_size1 Video buffer bytes CGA 40x25
F000:F0E6 dw video_buf_size2 Video buffer bytes CGA 80x25
F000:F0E8 dw video_buf_size3 Video buffer bytes CGA Graphics
F000:F0EA dw video_buf_size4 Video buffer bytes CGA Graphics
F000:F0EC db video_columntbl Video columns per modes 0-7
F000:F0F4 db video_hdwrbtbl mode video hardware modes (0-7)
F000:F841 loc Memory size Function call
F000:F84D loc Equipment Check Function call
F000:F859 loc Cassette Function Call
F000:FA6E db video_char_tbl1 Video characters in graphic modes
F000:FE6E loc Timer Function Call
F000:FEA5 loc Timer Hardware Interrupt
F000:FEF3 dw int_vec_table Initial interrupt vectors
F000:FF1D dw int_data_table
F000:FF21 dw video_ptr
F000:FF23 dw int_vec_table_2
F000:FF53 loc Dummy Interrupt return
F000:FF54 loc Print Screen Function Call
F000:FFF0 loc power_on_reset SYSTEM RESET
```

Kalt- oder Warmstart-Einsprung

Ein Kaltstart enthält neben dem Test der CPU, des ROM und der Initialisierung der Hardware auch den Test der folgenden Komponenten:

Memory system
Timer/Counter (which is also used for RAM refresh)
Interrupt Controller(s)
DMA Controller(s)
Keyboard Controller
Video Controller & Video RAM
Floppy Controller
Hard Disk Controller (if present)

Fehler werden entweder durch Töne oder Fehlercodes am Bildschirm angezeigt.

Ein Warmstart benutzt den Inhalt des warm_boot_flag um den Speichertest zu überspringen (i.a. beim Drücken von Ctrl-Alt-Del).

```
F000:FFF5 BIOS release date
F000:FFFE PC model identification
date model byte submodel byte revision
04/24/81 FF = PC-0 (16k) -- --
10/19/81 FF = PC-1 (64k) -- --
08/16/82 FF = PC, XT, XT/370 (256k motherboard) -- --
10/27/82 FF = PC, XT, XT/370 (256k motherboard) -- --
11/08/82 FE = XT, Portable PC -- --
01/10/86 FB = XT 00 01
01/10/86 FB = XT-2 (early) 00 02
05/09/86 FB = XT-2 (640k) 00 02
06/01/83 FD = PCjr -- --
01/10/84 FC = AT -- --
06/10/85 FC = AT 00 01
11/15/85 FC = AT 01 00
04/21/86 FC = XT/286 02 00
09/13/85 F9 = Convertible 00 00
09/02/86 FA = PS/2 Model 30 00 00
11/15/86 FC = AT, Enhanced 8MHz 00 00
02/13/87 FC = PS/2 Model 50 04 00
02/13/87 FC = PS/2 Model 60 05 00
1987 F8 = PS/2 Model 80 00 00
2D = Compaq PC (4.77) -- --
9A = Compaq Plus (XT) -- --
00FC 7531/2 Industrial AT
06FC 7552 Gearbox
F000:FFFF db model_sub_type
```

BIOS-Erweiterungen

Das System überprüft den Speicher hinsichtlich installierter ROMs in 2k-Abständen beginnend bei 0C0000H. BIOS-ROMs beginnen mit dem Code AA55h. Danach folgt ein Längenfeld und danach der eigentliche Einsprungpunkt.

Externe ROM-Module können zwischen den Adressen c800:0000 und e000:0000 vorkommen. Jeder 2K-Block in diesem Adreßbereich wird für folgende ROM-Signatur abgesucht. :

Offset	Size	Contents
+0	1	[55h] Signature of BIOS-accessible ROM module (first word in segment is aa55h)
+1	1	[aaH]
+2	1	[len] length of ROM module in 512-byte increments
+3	?	[] executable code (often a NEAR jump to initialization code) (a dummy byte usually exists to validate checksum)

Wurde eine gültige ROM-Signatur gefunden, wird diese BIOS-Erweiterung mit einem far-CALL auf Adresse 3 des ROM initialisiert. Üblicherweise initialisiert das BIOS alle erforderlichen Interruptvektoren.

Lesen aus dem Hauptspeicher,

HC05MM2.CPP, HC05MM2.C

Das folgende Beispiel zeigt, wie man aus dem Bildschirmspeicher liest, wie man sich den Wert eines Interruptvektors ansieht und wie man die Adresse der seriellen Schnittstelle erfährt.

```
/* HC05MM2.CPP */
/*
 * Lesen im Hauptspeicher
 * =====
 */
#include <dos.h>
#include <mytypes.h>
#define P_POINTER
#define P_STREAMS
#include <portable.h>

VOID main(VOID)
{
    UINT far *uifp;
    UINT seg, off, ser;

    constream con;
    con.clrscr();

    con << "Lesen innerhalb des Hauptspeichers\n";
    uifp = (UINT far *)MK_FP(0xb800, 0x0000);
    con << "Zeichen links oben am Bildschirm: "
        << hex << setw(2) << (UINT)*uifp << ':'
        << setw(1) << (UCHAR)*uifp
        << " Attribut: "
        << hex << setw(2) << (UINT)((*uifp)>>8)
        << endl;

    uifp = (UINT far *)MK_FP(0x0000, 5*4);
    off = *uifp;
    uifp++;
    seg = *uifp;
    con << "Print-Screen Interruptvektor: "
        << hex << setw(4) << seg << ':' << off << endl;

    uifp = (UINT far *)MK_FP(0x0040, 0x0000);
    ser = *uifp;
    con << "Adresse der seriellen Schnittstelle: "
        << hex << setw(4) << ser
        << endl;

    con << "Ende mit Taste";
    cin.get();
}
```

Natürlich kann die Uhrzeit auch gleich richtig lesbar ausgegeben werden. Das Programm HC05MM3A.CPP berücksichtigt das Darstellungsformat und gibt in Stunden, Minuten und Sekunden aus:

```
/* HC05MM3A.CPP */
/*
 * Lesen der Uhrzeit aus den BIOS-Daten
 * =====
 */
#include <dos.h>
#define P_STREAMS
#include <portable.h>
#include <mytypes.h>

VOID main(VOID)
{
    ULONG far *time_p;
    UINT s;

    constream con;
    con.clrscr();
    con << "Uhrzeit aus dem BIOS-Datenbereich\n";
    time_p = (ULONG far *)MK_FP(0x0040, 0x006c);
    do
    {
        s = (UINT)((FLOAT)(*time_p)*0.053);
        con << s/3600 << ':'
            << s/60 << ':'
            << s%3600 << ' ';
        delay(1000);
    }
    while (kbhit() == 0);
    cin.get();
}
```

In beiden Programmen erfolgt der Hauptspeicherzugriff über Pointer, die zuerst gebildet werden müssen:

```
time_p = (ULONG far *)MK_FP(0x0040, 0x006c);
```

und auf deren Ziel mit `*time_p` zugegriffen wird. Einfacher wird der Zugriff durch eine eigene Klasse, die das Arbeiten mit Pointern erspart (HC05MM3B.CPP, HC05MM3C.CPP, siehe später).

So wie diese gezeigten Beispiele gibt es viele Adressen, deren Wert uns Aufschluß über den Zustand des Rechners gibt.

Darüber wollen wir uns jetzt eine Übersicht verschaffen:

Nachdem wir einige dieser Adressen exemplarisch analysiert haben, schreiben wir zwei Hilfsprogramme, die eine vollständige, wenn auch nicht dechifrierte, Liste der BIOS-Variablen am Bildschirm ausgeben.

Uhrzeit aus dem Speicher,

HC05MM3.CPP, HC05MM3A.CPP, HC05MM3B.CPP, HC05MM3C.CPP

Jeder einlangende Timer-Interrupt zählt die Speicherstelle 0040:006c um 1 hoch, beginnt dabei um Mitternacht mit 0. Daher hat man auf 6c und den folgenden drei Adressen eine Zahl vom C-typ `long`, die eine Maßzahl für die aktuelle Uhrzeit ist. Die Einheit ist 53ms. Das Programm HC05MM3.CPP gibt die Uhrzeit in Vielfachen von 53ms aus. Die entstehende Zahl ist nur für Differenzmessungen, nicht aber für die Auswertung in Stunden, Minuten, Sekunden geeignet.

```
/* HC05MM3.CPP */
/*
 * Lesen der Uhrzeit aus den BIOS-Daten
 * =====
 */
#include <dos.h>
#define P_STREAMS
#include <portable.h>
#include <mytypes.h>

VOID main(VOID)
{
    ULONG far *time_p;
    constream con;
    con.clrscr();
    con << "Uhrzeit aus dem BIOS-Datenbereich\n";
    time_p = (ULONG far *)MK_FP(0x0040, 0x006c);
    do
    {
        con << *time_p << ' ';
        delay(1000);
    }
    while (kbhit() == 0);
    cin.get();
}
```

Verzeichnis der BIOS-Variablen,

HC05MM5.CPP

Die BIOS-Variablen können gleichzeitig auf einem Bildschirm angezeigt werden. Um sie zu interpretieren, benötigt man noch eine Tabelle, wie etwa jene im selben Heft; noch besser wäre es, die Texte aus der Erklärung der BIOS-Daten mit den Daten selbst zu verknüpfen, wie das auch bei verschiedenen Systemanalyseprogrammen geschieht.

Beiden Programmen gemeinsam ist die Darstellung der Daten in Zeilen und Spalten. Die Konstante COLW legt die Spaltenbreite, COL die Spaltenzahl und ROW die Zeilenzahl. Die jeweilige Position wird in zwei Variablen row und col gehalten, die durch $col = (num / ROW) * COL$ und $row = num \% ROW$ berechnet werden. Die Kursorposition für den Schreibbeginn findet man mit `setxy(1+col*COLW,row+2)`, wobei 1 und 2 feste Abstände vom Rand darstellen.

```
/* HC05MM5.CPP */
/*
 * Anzeige der BIOS-Daten
 * =====
 */
#include <dos.h>
#include <mytypes.h>
#define P_STREAMS
#define P_POINTER
#include <portable.h>

#define COLW 4
#define COL 16
#define ROW 16

VOID main(VOID)
{
    INT off, row, col;
    UCHAR far * ip;

    constream con;
    con.clrscr();
    con << "BIOS-Daten\n";

    ip=(UCHAR far *)MK_FP(0x40,0);

    for (off=0; off<0x10; off++) // niederwertiges byte
    {
        con << setxy(1,off+4) << hex << setfill('0')
            << off << " |";
    }

    for (off=0; off<0x10; off++) // höherwertiges byte
    {
        con << setxy((off*COLW)+5,2)
            << hex << setfill('0') << off << " |";
        con << setxy((off*COLW)+5,3) << "----";
    }

    for (off=0; off<0x100; off++)
    {
        col=(off/ROW)*COL;
        row=off%ROW;
        con << setxy(5+col*COLW,row+4)
            << hex << setfill('0') << setw(2)
            << (UINT)*ip;
        ip++;
    }
    con << setxy(1,wherey()+1);
    con << "Taste";
    cin.get();
}
```

Nachdem wir jetzt die Namen der Interruptvektoren kennen, ihre Adresse, dann auch eine Liste der BIOS-Variablen kennen, und auch ein geeignetes Ausdruckprogramm dafür besitzen, wäre es eine lohnende Aufgabe, die Daten auch mnemonisch auszuwerten, ähnlich, wie es die bekannten Analyseprogramme, wie MFT (=Manifest/Quarterdek), SYSINFO (=System Information/Norton), MSD (=Microsoft Diagnostics/Microsoft) oder CHECKIT() tun. Gefragt ist also eine Liste der Interruptvektoren, wobei als Zusatzinformation eine leicht erkennbare Kurzbezeichnung mitangegeben wird. Genauso soll eine BIOS-Datenliste ausgegeben werden, wobei die Daten ihrer korrekten Länge und Gewichtung entsprechend gereiht werden.

Austausch der seriellen und parallelen Ports, HC05MM6.C

Interessanterweise werden sogar noch bei der DOS-Version 6.0 verschiedene kleine aber für den Betrieb eines Rechners doch wichtige Manipulationen nicht unterstützt. Es sind zwar drei LPTs vorhanden, die Standardausgabe erfolgt auf LPT1. Will man die Standardausgabe z.B. auf LPT2 ändern, muß man die entsprechenden Adressen im BIOS-Datenteil austauschen; etwas, das ohne hardwarenahe Programmierung nicht lösbar ist, da es weder durch das BIOS noch durch das Betriebssystem unterstützt wird.

Das Programm hieß `portex.c` und wurde für diese Serie in `HC05MM6.C` umbenannt. Der Hilfetext im Programm wurde nicht geändert.

Das Programm PORTEX erlaubt es die Adressen von seriellen oder parallelen Schnittstellen auszutauschen. Diese Möglichkeit ist im Betriebssystem nicht vorgesehen, daher muß man es mit geeigneten Utilities - wie diesem - ermöglichen.

Was in einem PC LPT1 (oder COM1) ist, bestimmt das BIOS beim Initialisieren des BIOS-Datenbereichs. Leider gibt es hier eine Unklarheit bei der Namensgebung: einerseits gibt es in einem System eine bis mehrere parallele und serielle Schnittstellen. Bezeichnen wir diese mit LPT1h...LPT3h, COM1h...COM4h. Andererseits gibt es für diese Kanäle je einen Speicherplatz im BIOS-Datenbereich.

Der MSDOS-Druckkanal PRN wird LPT1 zugewiesen. Jedes Mitprotokollieren mit ^P oder jede Druckerausgabe in C an `stdprn` wird an LPT1 gesendet. MSDOS erfährt über die vom BIOS initialisierten Adressen, was LPT1 ist. So wie DOS sollte es auch jedes andere 'saubere' Programm machen: nicht die Hardware selbst an 0x3bc ansprechen, sondern über die MSDOS-Druckausgabe (MSDOS-Funktion 5). Unglaublich wird dieses Verfahren, wenn man an einen anderen Kanal ausgeben will, z.B. LPT2. Dann muß man dafür sorgen, daß MSDOS die Adresse von LPT2 schreibt; aber wie, wenn es nur einen Kanal, PRN gibt?

PORTEX erwartet in der Kommandozeile die Angabe zweier Argumente, die Ports, die es auszutauschen gilt. Erlaubt sind LPT1, LPT2, LPT3, COM1, COM2, COM3, COM4. Jede unerlaubte Kombination wird durch die große `if..else if..`-Kombination am Eingang abgefangen und ein entsprechender Fehlertext ausgegeben. Für eine saubere Trennung von Text und Kode wird eingangs ein Array von Meldetexten in Form von `errtxt[]` angelegt. Zur Vereinfachung der Anzeige wird der Text nicht direkt über `printf()` angezeigt, sondern über die Funktion `error()`; damit wird erreicht, daß alle Anzeigen gleichartig erfolgen und jede Änderung des Ausgabeformats sich auf alle Ausgaben auswirkt.

Der Austauschvorgang selbst wird mit der Funktion `swap()` bewerkstelligt, die zwei Pointer auf die auszutauschenden Speicherstellen als Parameter übernimmt.

Bei einem Fehler in der Kommandozeile oder wenn die Kommandozeile keinen Parameter enthält, wird die Funktion `display_status()` aufgerufen, die die aktuellen Adressen anzeigt. `display_status()` wird auch vor und nach Austausch der Portadressen ausgeführt.


```

/* HC05MM6.C */
/*
 * Austausch serieller und paralleler Ports
 * =====
 */
#include <string.h>
#include <stdio.h>
#include <dos.h>
#include <mytypes.h>

VOID swap(UINT far *i, UINT far *j);
VOID display_status(VOID);
VOID error(CHAR *txt);

CHAR *errtxt[] = {
    "Invalid number of arguments",
    "Incorrect length in argument 1",
    "Incorrect length in argument 2",
    "Arguments must have equal type",
    "Port-number of arg1 must be in the Range of 1..4",
    "Port-number of arg2 must be in the Range of 1..4",
    "There is nothing to do",
    "Port-number of arg1 must be in the Range of 1..3",
    "Port-number of arg2 must be in the Range of 1..3",
    "There is nothing to do",
    "Only LPTx and COMx are allowed as arg1",
    "\nExchange of Ports\n",
    "=====\n",
    "Usage:\n",
    "portex                               Status-Display\n",
    "portex ?                             This display\n",
    "portex LPTx LPTy  x,y={1,2,3}      Exchange LPTs \n",
    "portex COMx COMy  x,y={1,2,3,4}    Exchange COMs \n";
};

VOID main(INT argc, CHAR * argv[])
{
    UINT far *fp;
    UINT far *fq;

    if (argc==1) // Status-Anzeige der Ports
    {
        display_status();
        return;
    }
    else if ((argc==2) && (argv[1][0]!='?'))
    {
    }
    else if (argc!=3)
    {
        error(errtxt[0]);
    }
    else if (strlen(argv[1])!=4)
    {
        error(errtxt[1]);
    }
    else if (strlen(argv[2])!=4)
    {
        error(errtxt[2]);
    }
    else if (strncmp(argv[1],argv[2],3)!=0)
    {
        error(errtxt[3]);
    }
    else if (strncmp(argv[1],"COM",3)==0) // change COM-Ports
    {
        if ((argv[1][3]<'1') || (argv[1][3]>'4'))
        {
            error(errtxt[4]);
        }
        else if ((argv[2][3]<'1') || (argv[2][3]>'4'))
        {
            error(errtxt[5]);
        }
        else if (argv[1][3]==argv[2][3])
        {
            error(errtxt[6]);
        }
        else
        {
            fp=(UINT far *)MK_FP(0x0040,(argv[1][3]-'1')*2);
            fq=(UINT far *)MK_FP(0x0040,(argv[2][3]-'1')*2);
            printf("Before: ");
            display_status();
            swap(fp,fq);
            printf("After: ");
            display_status();
            return;
        }
    }
    else if (strncmp(argv[1],"LPT",3)==0) // change LPT-Ports
    {
        if ((argv[1][3]<'1') || (argv[1][3]>'3'))

```

```

        {
            error(errtxt[7]);
        }
        else if ((argv[2][3]<'1') || (argv[2][3]>'3'))
        {
            error(errtxt[8]);
        }
        else if (argv[1][3]==argv[2][3])
        {
            error(errtxt[9]);
        }
        else
        {
            fp=(UINT far *)MK_FP(0x0040,(argv[1][3]-'1')*2 + 8);
            fq=(UINT far *)MK_FP(0x0040,(argv[2][3]-'1')*2 + 8);
            printf("Before: ");
            display_status();
            swap(fp,fq);
            printf("After: ");
            display_status();
            return;
        }
    }
    else
    {
        error(errtxt[10]);
    }
    error(errtxt[11]);
}

VOID swap(UINT far *i, UINT far *j)
{
    UINT temp;
    temp=*j; *j=*i; *i=temp;
}

VOID display_status(VOID)
{
    UINT far *fp;
    CHAR * txt[] = { "COM1","COM2","COM3","COM4",
                    "LPT1","LPT2","LPT3" };
    INT i;

    for (i=0; i<7; i++)
    {
        fp=(UINT far *)MK_FP(0x0040,i*2);
        if (*fp)
            printf("%s:%3x ",txt[i],*fp);
        else
            printf("%s:--- ",txt[i],*fp);
    }
    printf("\n");
}

VOID error(CHAR *txt)
{
    printf("%s!\n\n",txt);
}

```

Die Liste der nützlichen Programme, die die absoluten Speicheradressen benutzen, könnte noch weiter fortgesetzt werden. Wir wollen uns aber jetzt einen allgemeineren Standpunkt für den Speicherzugriff überlegen.

Klassen M, MEM, BIOS, IBIOS

Der Speicherzugriff erfolgt immer auf folgende Weise:

Pointer generieren: `TYP far * t = (TYP far *)MK_FP(seg, off);`

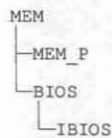
Wert bearbeiten: `*t=5;`

Nachteile:

- (1) man muß mit den dereferenzierten Werten arbeiten (*t), es gibt keine Variable für diesen Speicherplatz.
- (2) der Pointer ist gegen Veränderungen nicht abgesichert.

C++ bietet die Möglichkeit, in einer eigenen Memory-Klasse absolute Speichervariablen zu verwalten. Wie, entnehmen Sie aus der folgenden Headerdatei `memabs.h`, die auch den gesamten Code innerhalb der Klasse enthält.

Klassenbaum



Die Inhalte der auf absoluten Speicheradressen abgelegten Variablen haben verschiedene Länge und verschiedene Bedeutung. Im Terminus von C sind es die Typen `char`, `int`, `long` sowie pointer auf diese Typen oder Arrays davon. Eine Klasse im herkömmlichen Sinn, die einen solchen Speicherplatz beschreibt, hätte den Nachteil, daß sie für die anderen Typen noch einmal formuliert werden müßte, obwohl die Inhalte der Klasse im Prinzip dieselben wären. Die neue 'Ausbaustufe' von C++, Version 2.1 von ATT, bietet dafür Abhilfe in Form sogenannter 'templates', das sind Klassen oder Funktionen, bei denen ein oder mehrere Variablentypen noch nicht feststehen aber der Code typenunabhängig verfaßt werden kann. Diese Konzeption bietet sich hier an. Die Basisklasse M

Vorbemerkung zu MEM...

Der schnellste Zugriff auf absolute Speicheradressen ist nach wie vor der zuvor beschriebene. Kommt es aber weniger auf die Schnelligkeit, sondern eher auf einen verständlichen Code an, kann man die folgenden Klassen MEM... benutzen.

Der Phantasie sind bei der Konstruktion von Klassen keine Grenzen gesetzt, man kann sie, wie diese Beispiele zeigen, auch erfolgreich für hardwarenahes Programmieren einsetzen. Ob sich diese Konstruktionen bewähren, ob man sie in dieser Form einsetzen kann, wird die Kompiler-entwicklung zeigen. Derzeit bietet nur der BORLAND-Kompiler Templates an. Die Kompiler für Mikrokontroller kennen noch nicht einmal C++. Ob sie es je lernen werden? Wir werden sehen!

Die Klasse MEM hat die Aufgabe eine absolute Speicheradresse zu verwalten. Nicht einfach nur ein Byte, sondern auch 2 oder 4 Bytes, je nach Bedarf. Dazu wird die Klasse MEM, wie auch alle folgenden Klassen, um einen fiktiven Typ T konstruiert, der erst bei der Definition eines Objekts dieser Klassen eingesetzt wird, je nachdem, welchen Typ die absolute Speicherstelle enthält.

MEM besitzt zwei Variablen: Val und p. p ist ein far-Pointer auf die gewünschte absolute Speicherstelle und Val ist der aus dieser Speicherstelle zuletzt gelesene Wert. Bei den später verfaßten Klasse MEM_P hat es sich als praktisch erwiesen, sich den letzten Wert innerhalb der Klasse, Old zu merken.

Es gibt vier Konstruktoren für MEM:

```

MEM(T far *tp) // Mit einem Pointer
MEM(UINT s, UINT o) // Mit Segment und Offset
MEM(T far *tp, T t) // Mit einem Pointer und
// einem Initialisierungswert
MEM(UINT s, UINT o) // Mit Segment und Offset und
// einem Initialisierungswert
  
```

Vier Meldefunktionen geben über den Zustand der Klasse Auskunft: `val()` liefert den zuletzt gelesenen Wert, `addr()` liefert den Wert des Pointers, `seg()` liefert das Segment und `off()` den Offset.

Die Funktion `operator=()` dient zum Initialisieren der absoluten Adresse, die Funktion `operator()` liefert den aktuellen Wert der absoluten Adresse.

Die absolute Adresse kann mit den Operatorfunktionen

```
++, --, <<, >>, ~, +, -, *, /, %, &, |, ^=
```

so wie jede andere Variable auch bearbeitet werden.

Ein Objekt der Klasse MEM wird etwa wie folgt angewendet:

Die absolute Adresse `bildanfang` zeigt auf das Zeichen links oben am Bildschirm.

```
MEM<UINT> bildanfang(0xb800, 0);
```

Die Operatorfunktion `operator()` liefert den aktuellen Wert.

```
cout << "Links oben steht: " << hex << bildanfang() << endl;
```

Die absolute Adresse `bildanfang` kann wie alle anderen eingebauten Typen bearbeitet werden:

```
bildanfang++;
```

Jetzt steht links oben ein anderes Zeichen:

```
cout << "Links oben steht: " << hex << bildanfang() << endl;
```

Die Klasse MEM_P ist von MEM abgeleitet. Die adressierte absolute Speicherstelle enthält einen far-Pointer. MEM_P verwaltet nicht den Zahlenwert auf einer absoluten Adresse, sondern den Wert jener Adresse auf die der Pointer zeigt, der auf MEM_P gespeichert ist. Neben dem indirekten Zugriff ist auch eingebaut, daß der jeweils zuletzt gelesene Wert in Old gespeichert wird. Damit im Zusammenhang ist auch die Funktion `changed()`, die meldet, ob eine Zustandsänderung seit der letzten Befragung der Adresse stattgefunden hat.

Es hat sich als erforderlich erwiesen, zusätzlich zu den Elementfunktionen `seg()`, `off()` und `addr()` auch noch `pointseg()`, `pointoff()` und `pointaddr()` einzusetzen, die den Wert des Pointers zurückmelden.

MEMABS.H

```

#ifndef __MEMABS_HPP
#define __MEMABS_HPP

#define P_POINTER
#include <portable.h>
#include <mytypes.h>

#define GETMEM() Val=operator()()
#define SETMEM() operator=(Val);return Val

template <class T>
class MEM
{
protected:
    T Val;
    T far *p;

public:
    MEM(T far *tp)
    { p=tp; operator()(); }
    MEM(UINT s, UINT o)
    { p=(T far *)MK_FP(s,o); operator()(); }
    MEM(T far *tp, T t)
    { p=tp; operator=(t); }
    MEM(UINT s, UINT o, T t)
    { p=(T far *)MK_FP(s,o); operator=(t); }

    virtual VOID operator = (T t)
    { Val = *p = t; }
    virtual T operator () ()
    { Val = *p; return Val; }
  
```



```
// Ueberladene Operatoren
T operator ++ ()
{ GETMEM(); Val++; SETMEM(); }
T operator -- ()
{ GETMEM(); Val--; SETMEM(); }
T operator << (UINT i)
{ GETMEM(); Val<<=i; SETMEM(); }
T operator >> (UINT i)
{ GETMEM(); Val>>=i; SETMEM(); }
T operator ~ ()
{ GETMEM(); Val=~Val; SETMEM(); }
T operator += (T t)
{ GETMEM(); Val+=t; SETMEM(); }
T operator -= (T t)
{ GETMEM(); Val-=t; SETMEM(); }
T operator *= (T t)
{ GETMEM(); Val*=t; SETMEM(); }
T operator /= (T t)
{ GETMEM(); Val/=t; SETMEM(); }
T operator %= (T t)
{ GETMEM(); Val%=t; SETMEM(); }
T operator &= (T t)
{ GETMEM(); Val&=t; SETMEM(); }
T operator |= (T t)
{ GETMEM(); Val|=t; SETMEM(); }
T operator ^= (T t)
{ GETMEM(); Val^=t; SETMEM(); }

T val() { return Val; }
T far *addr() { return p; }

UINT seg() { return FP_SEG(p); }
UINT off() { return FP_OFF(p); }
};

template <class T>
class MEM_P : public MEM<T>
{
public:
    MEM_P(UINT s, UINT o) : MEM<T>(s,o)
    { Old = MEM<T>::operator()(); }
    MEM_P(UINT seg, UINT off, T t) : MEM<T>(seg,off,t)
    { MEM<T>::operator=(t); Old = t; }

    VOID operator = (T t)
    { Old = Val;
      **((T far **)p) = t;
      Val = t;
    }

    T operator () ()
    { Old = Val;
      Val = **((T far **)p);
      return Val;
    }

    BOOL changed()
    { return (Old==Val) ? FALSE : TRUE; }
    UINT pointoff()
    { return FP_OFF((T far *) (p)); }
    UINT pointseg()
    { return FP_SEG((T far *) (p)); }
    T far *pointaddr()
    { return (T far *) (p); }

protected:
    T Old;
};

#endif // __cplusplus

typedef enum BIOSA
{
    rs232_port_1 = 0x4100, // UINT
    rs232_port_2 = 0x4102, // UINT
    rs232_port_3 = 0x4104, // UINT
    rs232_port_4 = 0x4106, // UINT
    prn_port_1 = 0x4108, // UINT
    prn_port_2 = 0x410A, // UINT
    prn_port_3 = 0x410C, // UINT
    BIOS_data_seg = 0x410E, // UINT
    equip_bits = 0x4110, // UINT
    init_test_flag = 0x4212, // UCHAR
    main_ram_size = 0x4113, // UINT
    chan_io_size = 0x4115, // UINT
    keybd_flags_1 = 0x4217, // UCHAR
    keybd_flags_2 = 0x4218, // UCHAR
    keybd_alt_num = 0x4219, // UCHAR
    keybd_q_head = 0x411A, // UINT
    keybd_q_tail = 0x411C, // UINT

```

```
keybd_queue = 0x491E, // UINT
dsk_recal_stat = 0x423E, // UCHAR
dsk_motor_stat = 0x423F, // UCHAR
dsk_motor_tmr = 0x4240, // UCHAR
dsk_ret_code = 0x4241, // UCHAR
dsk_status_1 = 0x4242, // UCHAR
dsk_status_2 = 0x4243, // UCHAR
dsk_status_3 = 0x4244, // UCHAR
dsk_status_4 = 0x4245, // UCHAR
dsk_status_5 = 0x4246, // UCHAR
dsk_status_6 = 0x4247, // UCHAR
dsk_status_7 = 0x4248, // UCHAR
video_mode = 0x4249, // UCHAR
video_columns = 0x414A, // UINT
video_buf_siz = 0x414C, // UINT
video_segment = 0x414E, // UINT
vid_curs_pos0 = 0x4150, // UINT
vid_curs_pos1 = 0x4152, // UINT
vid_curs_pos2 = 0x4154, // UINT
vid_curs_pos3 = 0x4156, // UINT
vid_curs_pos4 = 0x4158, // UINT
vid_curs_pos5 = 0x415A, // UINT
vid_curs_pos6 = 0x415C, // UINT
vid_curs_pos7 = 0x415E, // UINT
vid_curs_mode = 0x4160, // UINT
video_page = 0x4262, // UCHAR
video_port = 0x4163, // UINT
video_mode_reg = 0x4265, // UCHAR
video_color = 0x4266, // UCHAR
gen_io_ptr = 0x4167, // UINT
gen_io_seg = 0x4169, // UINT
gen_int_occured = 0x426B, // UCHAR
timer_low = 0x416C, // UINT
timer_hi = 0x416E, // UINT
timer_rolled = 0x4270, // UCHAR
keybd_break = 0x4271, // UCHAR
warm_boot_flag = 0x4172, // UINT
hdsks_status_1 = 0x4274, // UCHAR
hdsks_count = 0x4275, // UCHAR
hdsks_head_ctrl = 0x4276, // UCHAR
hdsks_ctrl_port = 0x4277, // UCHAR
prn_timeout_1 = 0x4278, // UCHAR
prn_timeout_2 = 0x4279, // UCHAR
prn_timeout_3 = 0x427A, // UCHAR
prn_timeout_4 = 0x427B, // UCHAR
rs232_timeout_1 = 0x427C, // UCHAR
rs232_timeout_2 = 0x427D, // UCHAR
rs232_timeout_3 = 0x427E, // UCHAR
rs232_timeout_4 = 0x427F, // UCHAR
keybd_begin = 0x4180, // UINT
keybd_end = 0x4182, // UINT
video_rows = 0x4284, // UCHAR
video_pixels = 0x4185, // UINT
video_options = 0x4287, // UCHAR
video_switches = 0x4288, // UCHAR
video_1_reservd = 0x4289, // UCHAR
video_2_reservd = 0x428A, // UCHAR
dsk_data_rate = 0x428B, // UCHAR
hdsks_status_2 = 0x428C, // UCHAR
hdsks_error = 0x428D, // UCHAR
hdsks_int_flags = 0x428E, // UCHAR
hdsks_options = 0x428F, // UCHAR
hdsks0_media_st = 0x4290, // UCHAR
hdsks1_media_st = 0x4291, // UCHAR
hdsks0_start_st = 0x4292, // UCHAR
hdsks1_start_st = 0x4293, // UCHAR
hdsks0_cylinder = 0x4294, // UCHAR
hdsks1_cylinder = 0x4295, // UCHAR
keybd_flags_3 = 0x4296, // UCHAR
keybd_flags_4 = 0x4297, // UCHAR
timer_wait_off = 0x4198, // UINT
timer_wait_seg = 0x419A, // UINT
timer_clk_low = 0x419C, // UINT
timer_clk_hi = 0x419E, // UINT
timer_clk_flag = 0x42A0, // UCHAR
lan_1 = 0x42A1, // UCHAR
lan_2 = 0x42A2, // UCHAR
lan_3 = 0x42A3, // UCHAR
lan_4 = 0x42A4, // UCHAR
lan_5 = 0x42A5, // UCHAR
lan_6 = 0x42A6, // UCHAR
lan_7 = 0x42A7, // UCHAR
video_sav_tbls = 0x44A8, // ULONG
days_since_1_80 = 0x41CE, // UINT
prn_scrn_stat = 0x5200 // UCHAR
};
```



```

/* BIOS_FP
 * =====
 * creates VOID-far-Pointer to BIOS-Data-area
 * using enum BIOSA
 */
#define BIOSA_FP(1) MK_FP(((1)&0xf000)>>8,(1)&0x00ff)

/* MKB_FP
 * =====
 * creates VOID-far-Pointer to BIOS-Data-area
 * using integer-argument
 */
#define MKB_FP(1) MK_FP(0x40,(1)&0x00ff)

/* BIOS_OFF
 * =====
 * returns Offset of BIOS data
 * using enum BIOSA
 */
#define BIOS_OFF(1) ((1)&0x00ff)

#ifdef __cplusplus

/* class BIOS
 * =====
 * holds Pointer and Val of BIOS-Data-address
 * usable for all non-Pointer types
 */

template <class T>
class BIOS : public MEM<T>
{
public:
    BIOS(UINT offset)
    : MEM<T>(0x40,offset)
    { Old=Val; }
    BIOS(BIOSA offset)
    : MEM<T>((offset>>8)&0xf0,offset&0x00ff)
    { Old=Val; }
    BIOS(UINT offset,T t)
    : MEM<T>(0x40,offset,t)
    { Old=Val; }
    BIOS(BIOSA offset,T t)
    : MEM<T>((offset>>8)&0xf0,offset&0x00ff,t)
    { Old=Val; }

    virtual T operator () ()
    { Old=Val; return MEM<T>::operator()(); }
    virtual VOID operator = (T t)
    { Old=Val; MEM<T>::operator=(t); }
    BOOL changed()
    { return (Old==Val) ? FALSE : TRUE; }

private:
    T Old;
};

#endif // __MEMABS_HPP

```

Die Klassen MEM und MEM_P werden durch die Programme HC05MM7, HC05MM8, HC05MM9, HC05MM10 getestet. Die Programme HC05MM7, HC05MM8 und HC05MM10 können auch gleichzeitig zum Testen von der später besprochenen Klasse BIOS verwendet werden, die ein Spezialfall der Klasse MEM für den Bereich der BIOS-Daten darstellt.

Anzeige der Adressen der COM-Ports HC05MM7.CPP

Das einfachste Beispiel zeigt, wie man den Inhalt absoluter Speicherstellen ausliest. Es sollen die Adressen der seriellen Schnittstelle angezeigt werden. Man definiert ebensoviele MEM-Objekte als man benötigt, im Beispiel die Schnittstellen COM1..COM4. Die Funktion operator()() liefert den aktuellen Wert der Speicherstelle als Typ UINT.

```

/* HC05MM7.CPP */
/*
 * Anzeige der Adressen der COM-Ports
 * =====
 */

#include <membios.h>
#include <memabs.h>

VOID main(VOID)
{
    cout << endl << "Adressen der COM-Ports\n";
    // BIOS<UINT> COM1(rs232_port_1);
    MEM<UINT> COM1(0x0040,0x0000);
    cout << "COM1:" << hex << COM1() << endl;
    // BIOS<UINT> COM2(rs232_port_2);
    MEM<UINT> COM2(0x0040,0x0002);
    cout << "COM2:" << hex << COM2() << endl;
    // BIOS<UINT> COM3(rs232_port_3);
    MEM<UINT> COM3(0x0040,0x0004);
    cout << "COM3:" << hex << COM3() << endl;
    // BIOS<UINT> COM4(rs232_port_4);
    MEM<UINT> COM4(0x0040,0x0006);
    cout << "COM4:" << hex << COM4() << endl;
}

```

Operatorentest HC05MM8.CPP

Bei der Konstruktion von Klassen ist es, genauso wie bei allen anderen Programmprojekten, ratsam, diese durch geeignete Testsituationen einer Bewährungsprobe zu unterziehen und die Testprogramme bei jeder Änderung der Klasse wieder auszuführen und gegebenenfalls zu erweitern.

Im Beispiel HC05MM8.CPP werden alle definierten Operatoren ausgeführt. In den Beispielen wird das MEM-Typ-Objekt a auf die selten benutzte Adresse des seriellen Ports 4 gelegt. In jeder Anweisungsgruppe wird die Adresse mit dem Zuweisungsoperator operator=() initialisiert, der Wert mit der Operator-Funktion operator() zurückgelesen. Danach wird die Operation ausgeführt und das Resultat ausgegeben. Lediglich bei den Operatoren << und >> ergibt sich eine Zweideutigkeit mit den ebenso überladenen Operatoren des Ausgabestreams cout, daher werden die Ausdrücke a<<2 geklammert.

```

/* HC05MM8.CPP */
/*
 * Operatorentest
 * =====
 */

#include <membios.h>
#include <memabs.h>

VOID main(VOID)
{
    INT i;
    // BIOS<UINT> a (rs232_port_4);
    MEM<UINT> a (0x0040,0x0006);

    cout << "Using BIOS-Data-Address 40:6 "
          << "(=rs232_port_4)\n";
    cout << "for testing Operators\n\n";

    cout << "Testing operators = [abcd] and ()\n";
    a = 0xabcd;
    cout << hex << a() << ' ';
    cout << endl;

    cout << "Testing operators ++,-- starting with 3\n";
    a=3;
    cout << hex << a() << ' ';
    for (i=0; i<4; i++)
        cout << hex << ++a << ' ';
    cout << hex << a() << ' ';
    for (i=0; i<4; i++)
        cout << hex << --a << ' ';
    cout << endl;

    cout << "Testing operators <<,>> starting with 1\n";
    a = 1;
    cout << hex << a() << ' ';
    for (i=0; i<4; i++)
        cout << hex << (a<<2) << ' ';
    cout << hex << a() << ' ';
}

```

```

for (i=0; i<4; i++)
    cout << hex << (a>>2) << ' ';
cout << endl;

cout << "Testing operator ~ \n";
a = 0x5aa5;
cout << hex << a() << ' ';
~a;
cout << hex << a() << ' ';
cout << endl;

cout << "Testing operator += [1234+4321] "
        "and -= [5555-4321]\n";
a = 0x1234;
cout << hex << a() << ' ';
a += 0x4321;
cout << hex << a() << ' ';
a = 0x5555;
cout << hex << a() << ' ';
a -= 0x4321;
cout << hex << a() << ' ';
cout << endl;

cout << "Testing operator *= [10*123] "
        "and /= [1230/10]\n";
a = 0x10;
cout << hex << a() << ' ';
a *= 0x123;
cout << hex << a() << ' ';
a = 0x1230;
cout << hex << a() << ' ';
a /= 0x10;
cout << hex << a() << ' ';
cout << endl;

cout << "Testing operator %= [13%10] "
        "and &= [1234&00ff]\n";
a = 0x13;
cout << hex << a() << ' ';
a %= 0x10;
cout << hex << a() << ' ';
a = 0x1234;
cout << hex << a() << ' ';
a &= 0x00ff;
cout << hex << a() << ' ';
cout << endl;

cout << "Testing operator |= [1234|00ff] "
        "and ^= [1234^00ff]\n";
a = 0x1234;
cout << hex << a() << ' ';
a |= 0x00ff;
cout << hex << a() << ' ';
a = 0x1234;
cout << hex << a() << ' ';
a ^= 0x00ff;
cout << hex << a() << ' ';
cout << endl;

// BIOS<UINT> *b = new BIOS<UINT>(rs232_port_4);
MEM<UINT> *b = new MEM<UINT>(0x0040,0x0006);

cout << "Using BIOS-Data-Address 40:6 "
        "(=rs232_port_4)\n";
cout << "for testing Operators\n\n";

cout << "Testing operators = [abcd] and ()\n";
(*b) = 0xabcd;
cout << hex << (*b)() << ' ';
cout << endl;
}

```

Lesen der Uhrzeit aus dem Speicher

Das ursprünglich mit Pointern gelöste Problem wird jetzt mit der Klasse MEM angegangen.

```

/* HC05MM3B.CPP */
/*
 * Lesen der Uhrzeit aus den BIOS-Daten, mit MEM-Klasse
 * =====
 */

#include <dos.h>
#define P_STREAMS
#include <portable.h>
#include <mytypes.h>
#include <memabs.h>
// #include <membios.h>

VOID main(VOID)
{
    MEM<long> time(0x0040,0x006c);
    // BIOS<long> time(0x6c);
    constream con;
    con.clrscr();
    cout << "Uhrzeit aus dem BIOS-Datenbereich\n";
    do
    {
        cout << time() << ' ';
        delay (1000);
    }
    while (kbhit()==0);
    cin.get();
}

```

Eigene Klasse für Uhrzeit

Hat man oft mit Zeitmessungen zu tun, ist es nützlich die oft wiederkehrenden Umwandlungen zwischen den einzelnen Zeitkomponenten einer eigenen Zeit-Klasse zu übertragen (HC05MM3C.CPP).

```

/* HC05MM3C.CPP */
/*
 * Lesen der Uhrzeit aus den BIOS-Daten mit Klasse TIME
 * =====
 */

#include <dos.h>
#define P_STREAMS
#include <portable.h>
#include <mytypes.h>
#include <membios.h>

class TIME : public BIOS<long>
{
    long t;
public:
    long operator()() { t=BIOS<long>::operator()()*0.053; return t; }
    int h() { return (int)(t/3600); }
    int m() { return (int)(t/60 - h()*60); }
    int s() { return (int)(t - h()*3600 - m()*60); }
    TIME() : BIOS<long>(0x6c) {}
};

VOID main(VOID)
{
    TIME time;

    constream con;
    con.clrscr();
    cout << "Uhrzeit aus dem BIOS-Datenbereich\n";
    do
    {
        time();
        cout << time.h() << ':' << time.m() << ':' << time.s() << ' ';
        delay (1000);
    }
    while (kbhit()==0);
    cin.get();
}

```


Speicherzugriff über Pointer HC05MM9.CPP

```
#ifndef __TINY__
#error must use TINY model
#else
```

Dieses Beispiel kompiliert nur im Speichermodell TINY, daher muß dieses Modell in Optionen-Compiler-MemoryModel eingestellt werden.

```
UINT *mem = new UINT[80];
```

Erzeugt einen Speicherbereich mit 160 Bytes; der Pointer mem zeigt auf den Anfang.

```
cout << "at " << hex << mem << endl;
cout << "contents (read by pointers):\n";
```

Der Speicher wird wie folgt belegt, um damit die Memory-Klassen MEM und MEM_P testen zu können.

```
+-----+
mem      | mem+2 | \
+-----+ far-Pointer auf Adresse mem+2
mem+1    | DS   | /
+-----+
mem+2    | 0xab |
+-----+
```

```
*mem = (UINT)(mem+2);
*(mem+1) = _DS;
*(mem+2) = 0xab;
```

Diese Speicherbelegung wird zuerst durch Verendung des Pointer mem angezeigt.

Danach werden drei MEM-Objekte off, seg und dat gebildet und die im Speicher befindlichen Daten angezeigt:

```
MEM<UINT> off(_DS, (UINT)mem);
MEM<UINT> seg(_DS, (UINT)(mem+1));
MEM<UINT> dat(_DS, (UINT)(mem+2));
cout << hex << off.off() << ':' << hex << off() << endl;
cout << hex << seg.off() << ':' << hex << seg() << endl;
cout << hex << dat.off() << ':' << hex << dat() << endl;
```

Dasselbe leistet auch ein MEM_P-Objekt m, das auf die Adresse des Pointers initialisiert wird. Das MEM_P-Objekt verwaltet daher zwei Pointer, die auch angezeigt werden: einerseits der in der Ausgabe als 'Primary-Pointer' angegebene, der mit seg() und off() ausgegeben wird, andererseits der Pointer auf den er zeigt, dessen Wert mit pointseg() und pointoff() ausgegeben wird. Die Daten, auf die der Pointer zeigt, werden mit den überladenen Operatoren probenhalber manipuliert.

```
cout << "creating MEM_P object, reading back data\n";
MEM_P<UINT> m(_DS, (UINT)mem);
cout << "Primary-Pointer:";
cout << hex << m.seg() << ':' << hex << m.off() << endl;
cout << "points to ";
cout << hex << m.pointseg() << ':' << hex << m.pointoff() << endl;
cout << "containing data: " << hex << m() << endl;
cout << "data ++, --, &0x0f, |0xf0" << endl;
++m; cout << m() << ' ';
--m; cout << m() << ' ';
m&=0x0f; cout << m() << ' ';
m|=0xf0; cout << m() << ' ';
cout << endl;
```

Dieselben Tests können auch mit einem dynamischen MEM_P-Objekt durchgeführt werden, wie die folgenden Zeilen zeigen:

```
cout << "creating MEM_P object, reading back data\n";
MEM_P<UINT> *mp = new MEM_P<UINT> (_DS, (UINT)mem);
cout << "Primary-Pointer:";
cout << hex << mp->seg() << ':' << hex << mp->off() << endl;
cout << "points to ";
cout << hex << mp->pointseg() << ':' << hex << mp->pointoff() << endl;
cout << "containing data: " << hex << (*mp)() << endl;
cout << "data ++, --, &0x0f, |0xf0" << endl;
++(*mp); cout << (*mp)() << ' ';
--(*mp); cout << (*mp)() << ' ';
(*mp)&=0x0f; cout << (*mp)() << ' ';
(*mp)|=0xf0; cout << (*mp)() << ' ';
cout << endl;
}
#endif
```

Das MEM_P-Objekt ist bei Behandlung von Datenstrukturen von DOS vorteilhaft anwendbar und wird in einer späteren Folge, die sich mit diesen Zusammenhängen beschäftigt, weiter verwendet.

```
/* HC05MM9.CPP */
/*
 * Pointer-controlled memory access
 * =====
 */
```

```
#ifndef __TINY__
#error must use TINY model
#else
#include <memabs.h>
```

```
VOID main(VOID)
```

```
{
    cout << "Pointer controlled memory access\n";
    UINT *mem = new UINT[80];
    cout << "at " << hex << mem << endl;
```

```
    cout << "contents (read by pointers):\n";
    *mem = (UINT)(mem+2);
    *(mem+1) = _DS;
    *(mem+2) = 0xab;
    cout << hex << mem << ':' << hex << *mem << endl;
    cout << hex << (mem+1) << ':' << hex << *(mem+1) << endl;
    cout << hex << (mem+2) << ':' << hex << *(mem+2) << endl;
```

```
    cout << "contents: (read by MEM-objects)\n";
    MEM<UINT> off(_DS, (UINT)mem);
    MEM<UINT> seg(_DS, (UINT)(mem+1));
    MEM<UINT> dat(_DS, (UINT)(mem+2));
    cout << hex << off.off() << ':' << hex << off() << endl;
    cout << hex << seg.off() << ':' << hex << seg() << endl;
    cout << hex << dat.off() << ':' << hex << dat() << endl;
```

```
    cout << "creating MEM_P object, reading back data\n";
    MEM_P<UINT> m(_DS, (UINT)mem);
    cout << "Primary-Pointer:";
    cout << hex << m.seg() << ':' << hex << m.off() << endl;
    cout << "points to ";
    cout << hex << m.pointseg() << ':' << hex << m.pointoff() << endl;
    cout << "containing data: " << hex << m() << endl;
    cout << "data ++, --, &0x0f, |0xf0" << endl;
    ++m; cout << m() << ' ';
    --m; cout << m() << ' ';
    m&=0x0f; cout << m() << ' ';
    m|=0xf0; cout << m() << ' ';
    cout << endl;
```

```
    cout << "creating MEM_P object, reading back data\n";
    MEM_P<UINT> *mp = new MEM_P<UINT> (_DS, (UINT)mem);
    cout << "Primary-Pointer:";
    cout << hex << mp->seg() << ':' << hex << mp->off() << endl;
    cout << "points to ";
    cout << hex << mp->pointseg() << ':' << hex << mp->pointoff() << endl;
    cout << "containing data: " << hex << (*mp)() << endl;
    cout << "data ++, --, &0x0f, |0xf0" << endl;
    ++(*mp); cout << (*mp)() << ' ';
    --(*mp); cout << (*mp)() << ' ';
    (*mp)&=0x0f; cout << (*mp)() << ' ';
    (*mp)|=0xf0; cout << (*mp)() << ' ';
    cout << endl;
```

```
}
#endif
```


Beginn und Ende der Tastaturwarteschlange

HC05MM10.CPP

Abschließend werden der Beginn und das Ende der Tastaturwarteschlange ausgelesen.

MEMBIOS.HPP hat zwei Aufgaben: Einerseits stellt die Klasse BIOS eine Spezialisierung der Klasse MEM dar, da für das Segment der Wert 0x0040 eingesetzt wird. Andererseits erspart die Klasse dem Programmierer, das Wälzen von Tabellen, da der Aufzählungstyp BIOSA ein Verzeichnis aller BIOS-Adressen darstellt. BIOSA hat folgenden Aufbau:

```
typedef enum BIOSA
{
    rs232_port_1 = 0x4100, // UINT
        ^^^...Offset der Variablen
        ^.....Typ der Variablen
        1...UINT
        2...UCHAR
        ^.....Segment (4 oder 5)
    ....
    prn_scrn_stat = 0x5200 // UCHAR
};
```

Dieser Aufzählungstyp ist nicht nur gemeinsam mit der Klasse BIOS - was nur in C++-Programmen ginge - sondern auch unabhängig davon mit den Makros BIOSA_FP, MKB_FP und BIOS_OFF anwendbar.

BIOS_FP erzeugt einen VOID-far-Pointer und erhält als Argument einen der BIOSA-Elemente. Beispiel:

```
UINT far *Timer_lo = (UINT far *)BIOSA_FP(timer_lo);
```

BIOS_OFF liefert die Offset-Adresse einer BIOSA-Variablen.

```
UINT timer_lo_offset = BIOS_OFF(timer_lo);
```

Die Klasse BIOS wurde gegenüber der Klasse MEM um die Möglichkeit erweitert, ein Objekt mit einem Offset oder mit einer BIOSA-Variablen zu initialisieren. Die Funktion changed() stellt fest, ob sich der Speicherwert zwischen zwei Ablesungen verändert hat.

Die Klasse IBIOS geht davon aus, daß der Inhalt der BIOS-Adresse ein Pointer ist, ähnlich wie bei der Klasse MEM_P. Die Klasse IBIOS wird in erster Linie bei der Handhabung des Tastaturbuffers verwendet.

```
/* HC05MM10.CPP */
/*
 * Beginn und Ende der Tastaturwarteschlange
 * =====
 */
```

```
//#include <membios.h>
#include <memabs.h>
```

```
VOID main(VOID)
{
    cout << endl;
    // BIOS<UINT> begin(keybd_begin);
    MEM<UINT> begin(0x0040,0x0080);
    cout << endl << "Begin of kbd-queue : ";
    cout << hex << begin() << endl;
    // BIOS<UINT> end(keybd_end);
    MEM<UINT> end(0x0040,0x0082);
    cout << endl << "End of kbd-queue : ";
    cout << hex << end() << endl;
}
```

```
/* HC05MM11.CPP */
/*
 * Zugriff auf Interruptvektoren
 * =====
 */
```

```
#include <mytypes.h>
#include <memabs.h>
```

```
VOID main(VOID)
{
    MEM<ULONG> harddisktable((UINT)0x0000,(UINT)0x118);
    cout << hex << harddisktable() << endl;
}
```

□

C Users Group Library

DSK-372\CUG.LZH

Im allgemeinen ist eine Nummer dieser Sammlung eine Diskette. Die niederen Nummern enthalten oft mehrere Programme, alle anderen aber ein ganzes Programmpaket.

Diese Sammlung wird durch die jeweiligen Programmautoren gewartet. Es kann daher vorkommen, daß eine einzelne Zeile auch mehr als eine Diskette umfaßt, da im Laufe der Zeit Ergänzungen und Verbesserungen durch den Autor vorgenommen wurden.

Bestellbeispiel: CUG203 YAM for MSDOS

Für eine präzisere Suche können Sie einen 4-bändigen Gesamtindex um \$ 28,- (+45% Versand) mit der Bestellbezeichnung CAT24 bei The C Users Journal, 1601 W.23rd St, Suite 200, Lawrence KS 66046-2743 bestellen. Am besten bestellt man mit einer VISA/MASTER/EURO-Card. Zur Vereinfachung gibt es auch einen Bestellschein als PCX-Datei CUG_BEST.PCX in der ZIP-Datei CUG.LZH auf der Diskette zu diesem Heft. In diesem Index kann auch systematisch gesucht werden.

CUG101 Games I	CUG150 Extractions from PC-SIG #50 and #142	CUG198 Micro Emacs v 3.9, Source	CUG242 Still More Cross Assemblers
CUG102 Games II	CUG151 Ed Ream's Screen Editor for MS-DOS	CUG199 GED	CUG243 DECUS C Preprocessor
CUG103 Just Like Mom's	CUG152 C Utilities for MS-DOS	CUG200 Small C Interpreter	CUG244 deBruijn
CUG104 Small C	CUG153 C Utilities for MS-DOS	CUG201 MSDOS System Support	CUG245 Linear Cellular Automata
CUG105 Functions I	CUG154 PC Tools	CUG202 KAREL for MSDOS	CUG246 Cycles, Mandelbrot Graphics
CUG106 Functions II	CUG155 B-Trees, FFT, ONED & Util I	CUG203 YAM for MSDOS	CUG247 MIRACL
CUG107 Functions III	CUG156 Small C w/Floats	CUG204 68000 C Compiler	CUG248 MicroSpell
CUG108 Utilities I	CUG157 OE for MS-DOS, CMODEM, FWT, TP	CUG205 Utilities VII	CUG249 C Spot Run (CSR) Library
CUG109 Utilities II	CUG158 OE for BDSC	CUG206 Checkbook Register, Doc. and Exec.	CUG250 68K FP Library & Mandelbrot Graphics
CUG110 Utilities III	CUG159 Adventure for MS-DOS	CUG207 Checkbook Register, Doc. and Source	CUG251 Pull Down Menu and Adventure System
CUG111 Utilities IV	CUG160 Programs from "Learning to Program in C"	CUG208 'e' for CP/M 68K	CUG252 C Tutor Doc
CUG112 Utilities V	CUG161 Programs from "Efficient C"	CUG209 Simplex Curve Fitting	CUG253 C Tutor Source
CUG113 Cross Assemblers, 6800 v 2.5, 1802 v 1.1	CUG162 Mchip80	CUG210 Simulation and Loadable BIOS for CP/M	CUG254 Withdrawn from the CUG Library
CUG114 Ed Ream's Screen Editor, BDS C Version	CUG163 Small C for MS-DOS	CUG211 Search, Sort and Merge	CUG255 EGA Graphics Library
CUG115 Ed Ream's Screen Editor, Small C Version	CUG164 Windows for C	CUG212 Simulation for BDOS	CUG256 DeSmet Carry Routines
CUG116 Adventure	CUG165 Programs from "Reliable Data Structures"	CUG213 Van Nuys Enhancements	CUG257 C Tutor Doc (Turbo C)
CUG117 Software Tools - Documentation	CUG166 CUG Directory	CUG214 File Display Utilities	CUG258 C Tutor Source (Turbo C)
CUG118 Software Tools - Primitives	CUG167 Windows, Unix-like Util.	CUG215 BBS for BDS C	CUG259 Console I/O & Withers Tools
CUG119 Mince/Scribble	CUG168 Simple Database	CUG216 ZMODEM & SAVERAM	CUG260 Zmodem, CU and tty Library
CUG120 Compiler Utilities	CUG169 Miscellany IV	CUG217 Spell and Dictionary, Part I	CUG261 68K Cross Assembler, MSDOS
CUG121 Miscellany I	CUG170 Miscellany V	CUG218 Spell and Dictionary, Part II	CUG262 Ramey Tools
CUG122 Pistol v 2.0	CUG171 Miscellany VI	CUG219 6502 Cross Assembler	CUG263 C_wndw Toolkit
CUG123 Tools I	CUG172 LEX, Part I	CUG220 Window Boss	CUG264 NRO & Other PC Tools
CUG124 TeX v 1a	CUG173 LEX, Part II	CUG221 6809 C for FLEX	CUG265 cpio
CUG125 Video Terminal Libraries	CUG174 Withdrawn from the CUG Library	CUG222 Small C v 2.7 Exec. and Doc.	CUG266 microPLOX
CUG126 Martz Library	CUG175 Withdrawn from the CUG Library	CUG223 Small C v 2.7 Source	CUG267 8085, 2650 & S6 Cross Assemblers
CUG127 RAP (A Text Formatter) v 2.21	CUG176 XLISP 1.6	CUG224 UTILIX	CUG268 Unicorn Library (Turbo C)
CUG128 ROFF4 ver 1.61	CUG177 Dr. Shaw's DOS Shell	CUG225 Utilities and ZMATH	CUG269 Unicorn Library (MSC 4)
CUG129 CITADEL v 2.10a	CUG178 TVX Source	CUG226 ART-CEE	CUG270 Miscellany X
CUG130 Bios Utilities	CUG179 TVX Doc and Executable	CUG227 Portable Graphics	CUG271 Steven's Library (Turbo C)
CUG131 ACRL	CUG180 Withdrawn from the CUG Library	CUG228 Miscellany IX	CUG272 Steven's Library (Datalight)
CUG132 6809 Tools, 6809 C compiler & Graphics Driver	CUG181 CFORUM and LZW	CUG229 Little Smalltalk, Part I, with bundled files	CUG273 Turbo C Utilities
CUG133 'e' Screen Editor v 4.0	CUG182 UNIX BBS	CUG230 Little Smalltalk, Part II, with bundled files	CUG274 Arrays for C
CUG134 Compiler Utilities	CUG183 PC Tool II	CUG231 Little Smalltalk, Part I, with unbundled files	CUG275 Linear Cellular Automata II
CUG135 VLI - Very Long Integer	CUG184 RUNAND	CUG232 Little Smalltalk, Part II, with unbundled files	CUG276 Z80 and 6804 Cross Assemblers
CUG136 Games III	CUG185 SORTS	CUG233 Make and other Utilities	CUG277 HP Plotter Library
CUG137 DDJ #1	CUG186 MAKE and AIM	CUG234 XDIR	CUG278 CXL v 5.1
CUG138 DDJ #2	CUG187 Functions IV	CUG235 Overview	CUG279 CROBOTS
CUG139 KAREL the Robot v 1.00	CUG188 Utilities VI	CUG236 Highly Portable Utilities	CUG280 Software Tools Source
CUG140 Miscellany II	CUG189 PC Tools III	CUG237 GRAD Graphics Doc and Lib	CUG281 Unicorn Library v 5.1
CUG141 Miscellany III	CUG190 Steve Passe's 68K Assembler	CUG238 GRAD Graphics Source and Font	CUG282 Quip and Graphics
CUG142 Boulton Catalog Volume	CUG191 Miscellany VII	CUG239 P.C. Gammon for MSC4	CUG283 FAFNIR
CUG143 Van Nuys Tools	CUG192 Miscellany VIII	CUG240 P.C. Gammon for Turbo C	CUG284 Portable 8080 System
CUG144 Van Nuys Tools Doc.	CUG193 Crypto Toolbox	CUG241 Inference Engine and Rule Based Compiler	CUG285 Bison for MSDOS
CUG145 ROFF4 v 1.6P	CUG194 JUGPDS 17		
CUG146 Small C for 6800, for FLEX Operating System	CUG195 JUGPDS 18		
CUG147 RBSB4	CUG196 JUGPDS 19		
CUG148 TMS9900/99000 Cross Assemblers	CUG197 Micro Emacs v 3.9, Exec. and Doc.		
CUG149 6800/1802 Cross Assemblers (Portable Version)			

CUG286	GRAD for MSC	CUG307	ADU and COMX	CUG328	WTWG	CUG348	8048 Disassembler/Z80 Asm
CUG287	GRAD for Turbo C	CUG308	MSU, REMZ & List	CUG329	Unix Tools for PC	CUG349	Simulation Subroutine Set
CUG288	Taveling Salesman, SD and Master Env. Access	CUG309	6809 C Compiler for MSDOS	CUG330	CTask	CUG350	PCX Graphics Library
CUG289	Othello	CUG310	Little Smalltalk for MSDOS	CUG331	SE Editor	CUG351	UltraWin
CUG290	FLEX for MSDOS	CUG311	DB Package	CUG332	PC Curses	CUG352	String and Vlist
CUG291	JJB for QC and TC	CUG312	Make-Maker	CUG333	GAWK	CUG353	Withdrawn from the CUG Library
CUG292	ASxxxxx Cross Assemblers	CUG313	Stevie	CUG334	GNUPLOT	CUG354	CES Mouse Tools Library
CUG293	3D Medical Imaging, Source	CUG314	MNP C Library	CUG335	Frankenstein Cross Assembler	CUG355	Sherlock for MSDOS
CUG294	3D Medical Imaging, Sample Imaging	CUG315	FTGRAPH	CUG336	EGAPAL/EDIPAL	CUG356	Withdrawn from the CUG Library
CUG295	blcio Library	CUG316	AS8 Cross Assembler	CUG337	Designing Screen Interfaces in C	CUG357	CSTAR
CUG296	C to C++ Migrator	CUG317	Group3 Image Processing	CUG338	68000 C Compiler and Assembler	CUG358	cbase
CUG297	Small Prolog	CUG318	RED	CUG339	CTRLCLIB	CUG359	GNU C/C++ for 386
CUG298	PC Curses	CUG319	CPP	CUG340	C-Window	CUG360	uspell
CUG299	MEL and BP	CUG320	Convolution Image Process	CUG341	Orbit Propagation	CUG361	Gadgets and Term
CUG300	MAT_LIB	CUG321	Mouse Trap Library	CUG342	I8255 Interface Library	CUG362	RMAXTask
CUG301	BGI Applications	CUG322	Doctor's Tools	CUG343	C Image Processing System	CUG363	68020 Cross Assembler
CUG302	3D Transforms	CUG323	Fireworks and Adventure	CUG344	C Grab-Bag #1	CUG364	C-Across
CUG303	MC68K Disassembler	CUG324	WGCONIO	CUG345	TLC/TLP		□
CUG304	ROFF5	CUG325	VGA Graphics Library	CUG346	ASxxxxx Cross Assembler, Part 2		
CUG305	HGA Mandelbrot Explorer	CUG326	SoftC Database Library	CUG347	TAVL Tree		
CUG306	Thread and Synapsys	CUG327	Panels for C				

Source Code CDRom, März 1992

DSK-372\SCC.LZH

Nur etwa ein Viertel des Inhalts der Source-Code CD-ROM ist MSDOS-Software. Der große Rest verteilt sich auf UNIX/AMIGA/ATARI. Allein das Verzeichnis aller Dateien ist 1MB groß. Um diese Mengen zu erforschen, bieten wir folgendes an:

1. Da sich der Großteil der Leser der **PC-NEWS** mit MSDOS/Windows-Software beschäftigt, beschränken wir die Darstellung auf dieses Viertel dieser CD-ROM. Wer alle Verzeichnisse und Dateien kennenlernen möchte, soll über den Diskettendienst die Datei SCC.ZIP bestellen oder von **His Master's Voice** downloaden.

2. Einen kurzen Einblick in die Vielfalt liefert die nachfolgende Darstellung. Sie können im Prinzip jedes Unterverzeichnis oder auch nur jede archivierte Datei (ZIP oder ARC) über den Diskettendienst beziehen. Geben Sie die **Kennbuchstaben für die CD-ROM** an, hier **SCC**, den **Titel des Unterverzeichnisses** an, und wenn Sie nur eine einzelne Datei wollen auch den Namen des **ZIP-Archivs** und zur Kontrolle die **Datellänge**. Wenn ein Unterverzeichnis den Rahmen einer Diskette sprengt, werden mehrere Disketten verwendet.

Bestellbeispiel: SCC\ASMUTL\D86.ZIP 53563

Verzeichnis der Verzeichnisse

MSDOS/ADA
MSDOS/AIJOURNL
MSDOS/ALGOL
MSDOS/ARC_LBR
MSDOS/ASM
MSDOS/ASMUTL
MSDOS/AT
MSDOS/AUTOCAD
MSDOS/AWK
MSDOS/BASIC
MSDOS/BATUTL
MSDOS/BBS
MSDOS/BBSDOORS
MSDOS/BORLAND
MSDOS/BSS
MSDOS/C
MSDOS/CAD
MSDOS/CALC
MSDOS/CLIPPER
MSDOS/COMPLINGM
MSDOS/COMPATBL
MSDOS/COMPRESS
MSDOS/CPLUSPLS
MSDOS/CROSSASM
MSDOS/CRYPT
MSDOS/DATABASE
MSDOS/DESKACCS
MSDOS/DESQVIEW

MSDOS/DIRUTL
MSDOS/DISASM
MSDOS/DJGPP
MSDOS/DSKUTL
MSDOS/EDITOR
MSDOS/EDUCATON
MSDOS/EEL
MSDOS/EGA
MSDOS/EMULATRS
MSDOS/FIDO
MSDOS/FILEDOCS
MSDOS/FILUTL
MSDOS/FORTH
MSDOS/FORTAN
MSDOS/FREEMACS
MSDOS/GIF
MSDOS/GRAPH
MSDOS/GRAPHICS
MSDOS/HAMRADIO
MSDOS/HYPERTXT
MSDOS/ICON
MSDOS/INFO
MSDOS/IRIT
MSDOS/KA9QTCPP
MSDOS/KERMIT
MSDOS/KEYBOARD
MSDOS/LAN
MSDOS/LASER

MSDOS/LISP
MSDOS/LOGO
MSDOS/MAC
MSDOS/MATH
MSDOS/MENU
MSDOS/MICROCRN
MSDOS/MICROSOFT
MSDOS/MISC
MSDOS/MODEM
MSDOS/MODULA2
MSDOS/MOUSE
MSDOS/MUSIC
MSDOS/NCATLNL
MSDOS/NETWORK
MSDOS/NEURLNET
MSDOS/OPS
MSDOS/PACKET
MSDOS/PASCAL
MSDOS/PCMAC
MSDOS/PCRESAC
MSDOS/PCTECH
MSDOS/PCMTL
MSDOS/PBTTERM
MSDOS/PLI
MSDOS/PLOT
MSDOS/PRINTER
MSDOS/PROGRN
MSDOS/PTSCRIPT

MSDOS/QBASIC
MSDOS/RBBS_PC
MSDOS/SCREEN
MSDOS/SNOBOL
MSDOS/SOUND
MSDOS/SPRINT
MSDOS/STARTER
MSDOS/SURFMOOL
MSDOS/SYSUTL
MSDOS/TEX
MSDOS/TIFF
MSDOS/TROJAN_P
MSDOS/TURBO_C
MSDOS/TURBOBAS
MSDOS/TURBOPAS
MSDOS/TXTUTL
MSDOS/UEMACS
MSDOS/UCPC
MSDOS/VGA
MSDOS/WAFFLE
MSDOS/WINDOWS3
MSDOS/XLISP
MSDOS/ZIP
MSDOS/ZMODEM
MSDOS/ZOO

Tabellenaufbau

<Dateiname> <Länge> <Datum> <Beschreibung>

ADA

ADA_TUTR.ZIP 178063 890630 The Interactive ADA Tutor with ADA Source Code.
NAPP.ZIP 224000 890829 NASA Ada source code reformatter

AIJOURNL

AI_APR88.ARC 6581 890513 Source code - Artificial Intelligence Journal
AI_AUG88.ARC 49413 890513 Source code - Artificial Intelligence Journal
AI_DEC86.ARC 24725 890513 Source code - Artificial Intelligence Journal
AI_DEC88.ARC 6913 890513 Source code - Artificial Intelligence Journal
AI_FEB87.ARC 8736 890513 Source code - Artificial Intelligence Journal
AI_JAN87.ARC 23189 890513 Source code - Artificial Intelligence Journal
AI_MAR87.ARC 14278 890513 Source code - Artificial Intelligence Journal
AI_MAY89.ARC 60952 890827 Source code - Artificial Intelligence Journal
AI_NOV86.ARC 51593 890513 Source code - Artificial Intelligence Journal
AI_NOV88.ARC 26199 890513 Source code - Artificial Intelligence Journal
AI_OCT86.ARC 69085 890513 Source code - Artificial Intelligence Journal
AIFRST.ARC 28756 890513 Source code - Artificial Intelligence Journal

ALGOL

ALGOL_M.ZIP 44297 891031 ALGOL-M language and compiler
ALGOLM.ZIP 40257 890630 ALGOL-M Modeled After ALGOL-60

ARC_LBR

AR002.ZIP 27732 901203 Source for Okumura's new pedagogical archiver
LHA211SR.ZIP 59459 910305 LHA 2.11 file compression pgm ASM & C src code
LHARCRC.ARC 56714 890628 TurboC/OPTASM source code for LHarc & its SFX
LSTARC.ARC 11248 871123 List filenames in ARCs, with Turbo C source
LZ_COMP2.ARC 25244 890410 Discussion of various compression algs. w/src
LZHUFF.ARC 27393 890505 LZHUFF encoding file compression w/Pascal src
LZHUFFPS.ARC 14448 890506 LZHUFF encoding file compression w/Pascal src
MODLHARC.ARC 3658 890630 Modify LHarc source code for MASM/TASM
REPAK2.ARC 16274 881005 Pascal pgm to repack archives w/source
SQUASH.ARC 35653 880714 'C' source for the PKPAK squashing compression
XONE23.ARC 10472 880829 V. Buerg's ARCHive split utility v2.3 (w/ASM)

ASM

8087SMPL.ZIP 56861 891031 8087 sample programs
A86A.ZIP 53832 891025 Assembler version A
A86B.ZIP 59675 891025 Assembler version B
A86V319A.ZIP 74754 890613 Eric Isaacson's Macro Assembler Ver 3.19 1 of 3.
A86V319B.ZIP 57453 890613 Eric Isaacson's Macro Assembler Ver 3.19 2 of 3.
A86V319C.ZIP 75701 890613 Eric Isaacson's Macro Assembler Ver 3.19 3 of 3.
ASM4QB.ZIP 12290 890607 Assembly Language Routines for QuickBASIC.
BMAC.ARC 6795 880728 ASM IF/THEN program
CASM.ZIP 8064 800101 Calls Macro-Assem from MS C language
CASML.ZIP 9682 880727 Lattice assembler interface
CR110.ZIP 30974 890624 Graphics Routines in Assembler for Turbo Pascal.
LBCLS.ZIP 34331 850508 asm functions for CRT and KB for C
LCFUNCS.ZIP 17544 871205 Access Lattice functions from ASM programs
M8087.ZIP 8970 880808 Assembler macro for 8087 coprocessor
QWIK41.ZIP 43463 890612 ASM Screen Writes Version 4.1 for TP 4.0
SNAP.ZIP 2805 880808 Debugger for ASM pgms w/ doc, ASM src
VDSK.ZIP 3682 891031 Sample device driver

ASMUTL

8088D.ZIP 72468 890613 Another Debugger, only Works on *.COM Files.
A86_D86.ZIP 237518 891030 Isaacson's assembler/debugger version 3.19
A86CNVRT.ARC 50213 880801 How to convert MASM files to A86 format
AFTXV100.ZIP 22912 900324 Conversions for 61 PC Mag MASM files to A86
ARROWASM.ARC 70784 890626 Public Domain MASM 3.0-compatible assembler
ASM001.ZIP 96416 891027 Assembler tools and utilities, 1 of 3
ASM002.ZIP 109618 891027 Assembler tools and utilities, 2 of 3
ASM003.ZIP 111558 891027 Assembler tools and utilities, 3 of 3
ASM1_01.ZIP 38599 890613 A Tiny Two-Pass 8086 Assembler with Docs.

ASMCOLOR.ZIP 4427 891031 Misc monitor code
ASMED_1.ARC 115214 891209 Editor making integrated env for TASM/MASM
ASMCN201.ZIP 21558 890613 Generate Cross-Referenced Assembly Code.
ASMHELP.ARC 23299 871102 MASM Assembler help files w/docs & examples
ASMTUTOR.ARC 35091 880606 A well done ASM tutorial by Joshua Auerbach
BLUEBOOK.ARC 93713 891206 Waite Groups ASM language Bluebook source code
CHASM4.ARC 146325 880808 Cheap ASSEMBLER (freeware)
CHEAPASM.ZIP 51502 891031 "Cheap" assembler and code
CONV_A11.ZIP 35170 891107 Case conversion util for Assembly language src
D86.ZIP 53563 891025 Debugger for use with A86
EXECOM14.ARC 21504 880726 Exe2Com v1.04 by Chris Dunford (w/MSC source)
EXTND_IQC 4480 851208 MASM macros for NEC V20/V30 instructions
LOADEMS.ARC 4024 900308 ASM example of load & execute from EMS memory
LOOP.ASM 2580 870815 Demonstrates block transfers of memory
MAC_ASM.ZIP 49594 891030 MACRO assembler with tutorial
MASTRENV.ARC 1732 880829 ASM source - find the DOS master environment
NOR_ASMB.ZIP 94619 910521 Norton ASM language rtms for disk file patcher
PC370.ZIP 209787 891030 IBM 370 assembly language emulator
PRENV.ARC 2975 880925 Read a program's parent's environment w/ASM
PRIMAC.ARC 6626 880927 Displays IAC area at 40:00F0 w/ASM
PRIMER2.ZIP 16061 910521 Whitman's ASM language primer for beginners
READOBJ.ARC 22739 890313 Dump Microsoft .OBJ & .LIB files, w/'C' source
SALLY.ARC 2730 850728 Structured Control Macros for MASM
SOUNDEXA.ZIP 5769 900519 Assembly language source for soundex routine
STDLIB.ZIP 146218 910324 Standard Library for 8086 ASM programmers
TESTNMI.ASM 2526 850106 Fiddle with nonMaskable Interrupts
TRACE.ARC 57573 871027 Interrupt tracer w/ASM/com files
TSRDEMO.ARC 21007 880829 Sample demo on a safe approach to TSRs (w/ASM)
TSRDEMO2.ARC 23021 880829 Sample demo on a safe approach to TSRs (w/ASM)
TSTKBD.ASM 316 831117 Return what type of system keyboard installed
USOFTPD.ARC 46016 871105 Assembler macros for use with MASM 5.*.0
WASM202.ARC 139776 870602 Wolf's Shareware assembler
X2B11.ARC 9931 891016 EXE2BIN replacement (tweaked ASM src) v1.1
XLT86.ARC 30119 880808 8080 to 8086 ASM translator, w/ASM source
Z80_22.ZIP 107819 891031 Z80 emulation under PC-DOS

AT

ATFMT.ASM 4567 871224 Disk format program for AT (MASM source)
KBMAP.ASM 4891 890217 Remap AT keyboard keys
SETTD.C 5223 880118 Set DOS time/date from realtime clock
VIEWCMOS.ARC 15374 900225 Display contents of AT CMOS RAM, w/C source
XKB.ARC 8570 900116 Set AT keyboard repeat rate & delay, w/C src

AUTOCAD

AUTOCAD.C 4392 880920 Faster print AUTOCAD files to PostScript Prntr

AWK

AWK320SR.ZIP 71366 910713 AWK v3.20 source from author's BBS
BANK2.ZIP 31493 880715 Public domain AWK

BASIC

ACCEPT.ZIP 8736 890607 QuickBASIC Subprograms for Prompts and Replies.
ADVBS341.ZIP 118238 890607 Advanced Quick BASIC Utilities, Version 3.41.
AGETST.ZIP 2461 890607 A Program to See How Long You Will Live.
AREACALC.ZIP 1294 890607 Calculates Area of An Irregularly Shaped Plane
BALLOON.ZIP 3873 890607 Draws Balloons on the Screen.
BANNER.ZIP 1189 890613 Create a Banner to the Screen or a File.
BARMENU.ZIP 3406 890607 Bar Menu Routines for LOTUS 1-2-3 in QuickBASIC.
BAS_SUB.ZIP 59763 891030 Basic language sub-routines
BASDLX14.ZIP 37229 890607 Deluxe Library of Functions, Comp. BASIC V 1.4.
BASDLX16.ZIP 66997 890607 Deluxe Library of Functions, Comp. BASIC V 1.6.
BASICO.ZIP 72578 891031 Misc BASIC utilities and files
BASMAN1.ARC 109072 880130 BASIC manual -- part 1 of 3
BASMAN2.ARC 119144 880130 BASIC manual -- part 2 of 3
BASMAN3.ARC 41629 880130 BASIC manual -- part 3 of 3
BASMENU.ARC 2245 880728 Menu subroutine for Basic programs
BASXREF.ZIP 3204 890607 Cross-Reference Program for BASIC Programs.
COM2DATA.ZIP 3528 890607 Convert BIN Files to BASIC Data Statements w/doc.

CROSSBAS.ZIP	79874	891030	A cross-reference for BASIC
DATETIME.ZIP	1284	890623	Gets the Date and Time from MS-DOS/PC-DOS.
DAYS.ZIP	913	890607	BASIC Program Gives the # of Days Between Dates.
DOSCALL.ZIP	1173	890607	CALL DOS Commands From Within BASIC.
EBASIC.ZIP	45291	890630	EBASIC Compiler, Runtime Interpreter, Like CBASIC.
EZLIST.ZIP	8326	890607	Pretty Printer for BASIC Aids Debugging.
EZSCREEN.ZIP	104782	890607	Attention Programmers! Make Screens Easily.
FCL3.ARC	4608	870625	Compare 2 BAS sources, produce MRG update file
FRIENDLY.ZIP	173451	890607	FriendlyWare, 30 Misc. BASIC Programs
FRONTEND.ZIP	80320	891031	Cross-reference and front-end utility
FUNCTION.ZIP	9219	890607	A BASIC Program to Help Plot a Math Function.
GLIB14.ZIP	122103	890607	Graphics Library for MS Quick BASIC Ver 4.0
MASK.ZIP	46777	890607	Microsoft Quick BASIC Input Routines.
MDUTIL.ZIP	42673	891031	BASIC utilities from Micro Development
MISBASIC.ZIP	9063	890607	Five Miscellaneous BASIC Programs.
MISCBAS.ZIP	185167	891031	78 MS Basic source code files
MTRK.ZIP	93309	891031	The Monstrous Trek source code and docs
NEATLIST.BAS	11884	851210	'Neat lister' for BASIC source code
PATTERNS.ZIP	691	890607	Draws Useless Patterns on Your Screen.
PBCLON16.ZIP	283053	911104	PBClone: ASM/BASIC lib. Replaces ProBas/ADVBAS
PC_PROF.ZIP	72607	890607	BASIC Professor. A Beginning BASICA Tutorial.
PCGRAF.ZIP	1833	890607	Example of a Three-Dimensional Graph in BASIC.
QB4GRAF.ZIP	20030	890607	Ten Miscellaneous Graphics Routines for QBASIC
QBINPUT.ZIP	4236	890607	QBASIC Field Input Routines with Source Code.
QBPAEL.ZIP	5763	890607	Quick BASIC Input Subroutines.
QBXL30.ZIP	40158	890607	Bueg's Quick BASIC Cross-Referencing Program.
QFPA.ZIP	97118	890607	Flight Planning Fuel-Time-Distance, Much More.
REREF.ZIP	43085	890607	Auto-Change References When Lines are Moved.
SCRN_DO.ZIP	107107	891101	Screen-DO BASIC screen layout generator
SCRNCODE.ZIP	89786	890607	Help with Input Fields.
SN0B01.ZIP	89400	891026	Snocrest BASIC language, 1 of 2
SN0B02.ZIP	42475	891026	Snocrest BASIC language, 2 of 2
SPLTSCRN.ZIP	1893	890607	Split-Screen Function in BASIC.
SQULSH.ZIP	7518	890607	Removes Blanks, REMs, Combines Lines of Pgms
SQULSH33.ARC	7680	870625	V.Bueg's compress & unnumber Basic source
SUBINFO.ZIP	3459	890607	Finds Sub-Directories for Use With Quick BASIC.
SWISS.ZIP	135225	890607	BASIC pgming tils, alias, strip, list, boxer, xref.
TXBM10.ZIP	7500	890607	The Turbo BASIC with Xmodem Source Code.
TYPER.ZIP	62828	890607	Nov PC Resource Typewriter Simulator in BASIC
ZBASIC14.ZIP	21579	890630	ZBASIC Version 1.4 - Like North Star BASIC.

BATUL

ASK.ASM	13651	860422	Query from within .BAT file
ASK.DOC	7595	860422	Doc for ASK.ASM
BAT204C.ZIP	42294	890630	Extended Batch Language, Version 2.04C.
BATCD.ASM	3595	840403	Change dir within .BAT file
BATCD.DOC	1809	840103	Doc for BATCD.ASM
BATMAKR2.ZIP	20231	901229	Multi-option .BAT maker, with TPas 3.0 source
BATMENU4.ZIP	96430	891101	BATCH menu maker version 4.0
BATQUES.ASM	2990	840102	Query within .BAT file
BATQUES.DOC	2187	840102	Doc for BATQUES.ASM
EBAT20.ZIP	33453	891031	Extended BATCH language version 2.0
EBL306B.ZIP	53742	890630	Ver 3.06B of Seaware's Extended Batch Language
EL.ZIP	3293	901017	Set errorlevel to value on cmd line, w/C src
ELF11.ZIP	7437	881028	Set errorlevel in .BAT files w/ASM
EXT.BAT.ZIP	51544	891030	Extensions for BATCH files
HOWEVLV.ARC	1224	880724	Returns hardware byte in errorlevel (w/ASM)
IFEXIT.ARC	10970	871015	Powerful additions to any .BAT (w/ASM)
TOADBOOT.ZIP	3461	910521	Warm boot after n minutes delay, w/ASM source

BBS

BBSC.ZIP	20796	880808	Simple BBS in C for UNIX
BBSM_250.ZIP	167758	911004	Binkleyterm mail/term pgm v2.50 Borland C src
BSRC_250.ZIP	463372	911004	Binkleyterm mail/term pgm v2.50 source
CMPPB510.ARC	53449	900308	Fast compare of two BBS lists, w/QBASIC source
HSRC_117.ZIP	135418	910510	Headedit v1.17 Sysop/Point reader source code
LISTCALL.ARC	12416	850911	List RBBS callers file (source in C)
LMK530C.ARC	14592	850928	Useful routines from Mike's C BBS
PUP.ZIP	140207	880301	T. Jennings's small BBS with C source
SRCH1325.ZIP	45696	900903	Search captured BBS msgs for up to 39 strings
WATCHKIT.ARC	17105	900117	Reboot crashed BBS after four rings, w/ASM src
WINDOW.ARC	60672	851120	FIDO-10, Simply window mgmt library
XDEV_117.ZIP	150999	910510	Developer Kit (util source) for XBBS v1.17
XSRC_117.ZIP	226480	910510	Complete C source for XBBS v1.17

BBSDOORS

BVOTE.ARC	51686	891208	Voting booth door with TP 5.0 source, QuickBBS
-----------	-------	--------	--

BORLAND

BORPATCH.ZIP	9580	890630	Borland's Languages Patching Utility Program.
JNFB88.ARC	253252	881126	Borland's Magazine Jan/Feb 88 Sources
TD1PATCH.ZIP	2205	890630	Borland's Patch to Turbo Debugger, Version 1.0.

BSS

PUP.ARC	140928	880302	T. Jennings's small BBS with C source
---------	--------	--------	---------------------------------------

C

64COLORS.ZIP	12382	900506	How to display all 64 EGA colors on VGA, w/Csrc
ABMAKEL4.ARC	62897	890725	An improved 'make' utility w/source
ABOUT.ZIP	4709	850706	make your own on line help in C
ACE.ZIP	57697	880721	A Comprehensive File Editor-V. 1.24
ADJRAM.ZIP	58093	890628	Adjustable RAMdisk, Docs, Desmet C Source
ADVCL10.ZIP	14121	880715	Advanced C routines
ADVCL11.ZIP	14735	870601	C subroutine library
ANALYZE2.ZIP	30278	871205	C source file s analyzer - function dir
ANSI.ZIP	2828	871205	Test ANSI driver functions
AREACODE.ZIP	5846	880809	Returns phone areacode for state or vice-vers
ARGTEST.C	1442	850916	Tests function, getargs()
ATOL.ZIP	512	880811	ATOL functions
BACKLOG.C	3797	880815	Get latest add to BACKUP.LOG from DOS backup/L
BAL.ZIP	1648	880809	Balance parens, comments, etc...
BATQUEST.ZIP	12857	890628	BATCH File Questionnaire Help File Written in C
BEYOND.ZIP	31037	880623	ASCII file comparison utility
BIGSORT.ARC	19451	871206	Big sort with C source code
BISICC.ZIP	17251	850626	Unix like CC command
BITMANIP.ZIP	3361	880721	Bit manipulation functions in C
BTUUDC.C	7191	871224	VMS UUdecode fixes lost trailing spaces
BPLUS11.ZIP	25330	880717	B+ file routines for C
BRACE.ZIP	9033	850212	Check C source code to verify paired braces
BTOA.ARC	6182	890126	"C" source for binary-to-ascii convert pgm
CL1.ZIP	5602	880101	Source for C printf, sprintf, etc.
C80TCOG.ZIP	43125	880808	CPM C80 C funcs (mostly f/software tools)
CANALYZ.ZIP	30278	880720	Analyze C source and header files
C_COMP.ZIP	38633	890628	Small C Compiler for the IBM PC
C_RESIDE.ZIP	5317	890628	Create Memory Resident Software. Written in C
C_WINDOW.ZIP	123103	890612	C-Window Toolkit TM for Turbo C and Quick C
CALLS.ZIP	12278	860125	Finds subroutine calls in C source
CALLS2.ZIP	12278	880101	PD C function call analyzer w/source
CASYN.C	5731	880809	Function to call any interrupt from Lattice
CB.ZIP	3345	880725	Great checkbox program (color/mono)9/86
CB2.ZIP	13937	880321	Code beautifier for C
CC.ZIP	13555	850802	C-DEBUGGER with source code
CC1.ZIP	13451	880809	C source paren, bracket & comment checker
CC68.ZIP	119937	880505	C cross compiler for 68k family
CCHECK.ZIP	27621	850705	Checks for {} in "C" programs
CCOMP.ZIP	38633	840620	C compiler for the PC
CCOUNT.ZIP	1510	880721	Counts lines w/comment, w/code & total
CDOC.ZIP	13838	870518	Scan C source & extraction function calls
CDOC2.ZIP	80941	880219	Document C source code
CFLOW.ZIP	15126	850723	Reveal C code structure w/ C80 source [CP/M]
CFLOW1.ZIP	13932	850721	Chart C program flow (Lattice version)

CFLOWX.ZIP	27346	851128	cflow w/cross ref in C186
CGRAPH.ARC	14336	860922	PC compiler for C
CHASE.ZIP	6041	880727	Robot chase game in Lattice
CIPHER2.ZIP	17963	870214	Lucifer/NBS encrypt/de-crypt algorithm
CLESSIONS.ZIP	21044	871007	Good tutorial for C
CLFEB88.ZIP	34360	880210	Source for Terminate Stay Resident menus
CLIB1.ZIP	56873	880731	C library with source (part 1 of 2)
CLIB2.ZIP	70571	880731	C library with source (part 2 of 2)
CLMFEB85.ZIP	44097	890628	Computer Language February 1985 C Routines
CMENU.ZIP	2388	880809	Menu system with C source
COLOR.ZIP	5168	880809	Lets user install color in your TP4 programs
COMM.ZIP	3119	880809	Small communications programs (DeSmet)
COMMASM.ARC	11544	880716	Info for programming MS-DOS COMM ports
COMMUSUP.ZIP	26975	880721	Communications support for DeSmet C
CONIO.ZIP	2288	880808	Larry Jordan's console I/O functions
CPG.ZIP	29292	871205	Source listing formatter
CPM2.ZIP	3191	880809	Critical path method
CPMFILES.ZIP	4707	880808	File manipulation for CP/M in C
CRP.ZIP	1519	890613	Computes the CRC Using Turbo Pascal Inline Code.
CRCB85.ARC	78080	851120	CRC error checking
CREF.ZIP	21286	890628	C Xref, Crisp & Clean, w/ Source, MSC 5.1, Quick
CRRESIDE.ZIP	5317	880718	C interrupt handler
CROOT.ZIP	5036	880728	f/DDJ-mods to AZTEC (C80) for wildcard
CRT.ZIP	4215	890613	Demos Terminal Interface.
CSOURCE.ARC	53248	871208	Turbo-C source file manager
CSOURCE.ZIP	116862	880211	C language source files
CSR.ARC	188046	880803	C Spot Run for Turbo C
CSR20A.ARC	227456	870412	C Spot Run version 2.0
CSR30_1.ZIP	115563	890628	C Spot Run, User-Supported C Add-On Lib 1 of 3
CSR30_2.ZIP	159378	890628	C Spot Run, User-Supported C Add-On Lib 2 of 3
CSR30_3.ZIP	82558	890628	C Spot Run, User-Supported C Add-On Lib 3 of 3
CSYSINT.ZIP	3214	880809	Lattice 'C' Interface to the World.
CSX11.ZIP	184636	880712	Multi-task MS C or Turbo C programs
CSX22.ZIP	339134	901123	Multitasking kernel in C w/ASM & C source
CTEXT.ZIP	135483	880211	C language tutorial
CTHREAD.ZIP	41063	871027	C help and tutor
CTLBREK.ZIP	3084	880720	CTL/BREAK handler
CTOOLBLX.ZIP	155841	880107	C toolbox utility shareware w reg req
CTOPAS.ZIP	1789	880721	C to Pascal translator
CTRL.C	13316	880906	Intercept Ctrl C's
CTUT.ZIP	116970	870222	Tutorial for programming in C
CTUTOR1.ZIP	123218	880609	Learn about C (part 1 of 2)
CTUTOR2.ZIP	123399	880609	Learn about C (part 2 of 2)
CU236.ZIP	110352	880731	C utilities from the C Users Group
CURPOS.ASM	766	830915	Cursor positioning for C
CURSES.ZIP	66752	871113	C window development toolkit for ms c: demo
CUTILS.ZIP	25779	871205	C UTILITY FUNCTIONS
CVF111.ZIP	3679	870507	Fixes some bugs in MS codeview version 1.0
CWD.C	4697	880815	Change working directory for C
CXL.ZIP	116936	880726	C library with windows, menus, and more...
CXL41.ZIP	125756	890628	C Programmer's Extended Function Library Ver 4.1
CXL41_MS.ZIP	43768	890628	Small Memory Library for Microsoft C & Quick C
CXL_MS.ZIP	41525	880726	CXL 4.0 library for MS C and Quick C
CXL_TC.ARC	44406	880727	CXL 4.0 library for Turbo C
CXLHERR.ZIP	3134	900402	C source for critical error handler with CXL
CXREF.ZIP	23294	880615	Another C cross referencing utility
DAY.ZIP	1506	880809	TURBO JULIAN TYPE DATE CONVERSIONS
DB2TXX02.ZIP	11168	890628	List Structure of dBaseII Files/Packs, C Source
DBREAK.ZIP	4361	911021	Disable CTL-C/CTL-BREAK/CTL-ALT-DEL. C source
DEBUGC.ZIP	3286	880809	C lang debug aid w/ doc, C src code
DFIELD.ZIP	12390	851122	Lattice obj file for data field functions
DIRLIB.C	12481	890101	Unix 'dirlib' package ported to MS-C
DISP10.ZIP	23798	880727	BIOS video functions & applications
DLC_TSR.ZIP	8335	871205	TSR Library Function for Daylight 'C'.
DLBS.ZIP	70886	880107	Shareware library for DATALIGHT C
DLBSRSC.ZIP	90267	880107	Shareware library for DATALIGHT C
DLB.ZIP	150934	870627	C library for Datalight C Compiler
DMAKE375.ZIP	514590	910512	C src: 'make' util for MS-DOS & Unix SysV/BSO
DOS.ZIP	18535	871205	C code to manipulate DOS functions
DRIVERS.ZIP	13066	850702	Device driver interface for Lattice C
DSKTEST.ZIP	13818	850806	disk test prog in Lattice C
DUPS.ZIP	15876	870516	All duplicate file names to STD out
ELEM.C	104887	880609	Basic C prgms, useful for advanced prgms
ERAQ.ZIP	5525	880727	Delete function with verify and pathing
ERRMERGE.ZIP	25611	880809	Merge error messages with source
EXECNEW.ZIP	2980	880727	Load & execute programs from Lattice C
EXTMEM.ZIP	13602	880107	Access extended memory in C
FFT142.ZIP	97916	910609	Fast Fourier transforms, with C source
FIELD.ZIP	2550	880811	Fields in C
FILEIO.ZIP	1635	880727	File I/O functions
FRAME.ZIP	1033	880808	Draw a frame on the screen
FXOC.ZIP	2000	880811	Function extraction utility
GDIR.C	8951	860613	Enhanced DIR w/attributes, hidden/system files
GENSRC.ARC	122617	890114	Coronado's Generic 'C' tutor v2.0, (1 of 2)
GET_IT.ZIP	24660	880210	FIND, GOTO sub-directories using text files
GETARG.C	4035	850916	Get cmd line args
GETF.ARC	49152	881120	Locate 'C' source file w/particular function
GETSEG.ZIP	1123	880808	Get value for all segment registers
GLOB.ZIP	1040	890624	Plot Globular Star Clusters as Dots on Screen.
GOODIE.ZIP	1980	880811	This is Bazaar!!
GOTHIC.ZIP	28858	880307	Gothic banner printer with C source
GRAPH.ARC	11264	870712	Graphics library for CGA
GYMAKE12.ARC	50745	881218	Make utility for MS-DOS w/C source v1.2
HANOI.ZIP	1158	890624	Tower of Hanoi, Ctr Graphics, Exmpl of Recursion
HASH.ZIP	5904	890624	Phone Book Prgm Uses Hash Code to Store Names.
HSORT.C	5018	850806	General purpose heapsort for C
IBMCR.C	3236	840624	C & ASM fast screen write, save, and restore
IBMTTY.ZIP	6483	880721	L. Cuthbertson's TTY program in C
IMC9101.ZIP	5014	910608	Inside Microsoft C Journal source, Jan. 1991
IMC9102.ZIP	4642	910608	Inside Microsoft C Journal source, Feb. 1991
IMC9103.ZIP	6426	910608	Inside Microsoft C Journal source, March 1991
IMC9104.ZIP	6802	910608	Inside Microsoft C Journal source, April 1991
IMC9106.ZIP	9863	910608	Inside Microsoft C Journal source, June 1991
INDENT.ARC	37886	890226	Indents 'C' programs (EXE and DOC, no source)
INDENTSR.ARC	63175	900106	Indents 'C' programs (C source only, no EXE)
INDEX.ZIP	3447	880721	MSDOS and CPM versions MS file, word, phase indx
INFERNZ.ZIP	58508	871205	Inference engine with C source
INGRES.ZIP	5593	871205	Example of good C code style
INKEY.ZIP	1955	880808	Input from keyboard function
INPUT.C	5718	860504	Place batch input in environment
INPUTS.ZIP	298	880809	Inputs string from keyboard
INTERCEP.ASM	6269	870705	Prevent printing AC with signal() in MS-C
INTERRUPT.ZIP	2054	880809	Provides vector lists & locates ROMs
IOS.ZIP	3859	880808	Level 1 I/O functions
ISCHECK.ZIP	961	880808	Checks SOMETHING matrix
ISRLCX.ZIP	4492	880727	Trap timer interrupts
JAZLIB.ARC	176128	870829	C utility library with source (w/C)
JPLC1.ARC	87762	880722	C library from the Jet Propulsion Laboratory
JPLC2.ZIP	91717	880722	C library from the Jet Propulsion Laboratory
K.R.ZIP	26818	880324	The C Programming Language, Kernighan & Ritchie.
KERM.ZIP	15828	880808	Reliable file transfer program
KERNEL.ARC	24710	881127	Multitasking kernel from DD

LOCATE.ZIP	1242	880811	C routine to position cursor
LU.ZIP	7403	880809	Library utility in C source
LUCIFER.ZIP	7899	880722	Encrypt/De-encrypt bytes, LUCIFER algorithm
MAEK.C	17577	850825	Public domain MAKE
MAIN.ZIP	1119	880808	Shell program - includes command line
MAEK.C	26326	840628	Public domain MAKE
MATCH10.ZIP	19272	910325	Unix sh-style regex parser (wild cards), C src
MATH.C	12117	840323	Double precision math rtns for Lattice or MS-C
MATRIX.ZIP	13153	850113	Document on features of OPUS V-1.0
MC20.ZIP	190861	901118	Small C compiler, source code available
MCB.ZIP	28216	890628	Learn Memory Control Blocks Dr. Dobbs C Src/EXE
MEMCLEAN.ZIP	5146	880808	Write to memory anywhere...C source
MENCON.ZIP	4651	880726	L Cuthbertson's menu driver in C
MF1LE.ASM	2918	880129	Allows more than 20 files open at once in MS-C
MF1LE.C	1111	910719	Allows more than 20 files open at once in MS-C
MIKES30C.ZIP	13663	850927	Microsoft 3.0 C utility functions
MILLISEC.ZIP	37734	911205	MSC/asm src for millisecond resolution timing
MINILINT.ZIP	1629	880809	T Janning's C source checker
MISC.ZIP	1299	880811	Change cursor on terminal
MISC1.ZIP	9888	880726	C functions for DOS
MISC2.ZIP	2765	880726	Misc C functions for C86
MOT2INTL.C	1431	890609	Convert Motorola S-record hex to Intel hex
MOVEMEM.ZIP	1189	880808	C callable function to move memory
MSC.C	3712	880815	Program to pass arguments from a file to CL
MSC_WIND.ZIP	9015	880703	Text window routines for MS C version 5.0
MSCTOOLS.ZIP	11362	850910	Lynn Long's tools for MS 3.0 C compiler
MSWITCH.ZIP	1718	871025	Undoc switches in MS C 4.0
MULTITASK.ZIP	33634	861105	Multitasking routines for Lattice C
MV.C	71847	860908	UNIX-like, moves and renames files
MV_LINE.ZIP	10183	890628	Move lines from one file to another C src, EXE
NAMES.ARC	2414	880809	Name and address database in C
NEWOPT88.ZIP	12326	870315	Optimize DeSmet C compiler output
NOTE11.ZIP	17824	870207	Notebook program w/ MS-C src.
NUMBIT.ZIP	8490	871205	Find value of number in another base
ONE_ON_9.ZIP	15938	880308	Learn C pointers and data structures
PAREN_ZIP	13853	870527	Checks for parenthesis errors in C progs
PARSER.ZIP	9194	871205	Finite state parser
PC_8250.ZIP	43556	910207	Serial port rtns. & VT100 emul. (C src only)
PCPILOT.ZIP	55049	900307	C programmer's TSR utility, with source
PPC.ARC	15232	881002	'Pretty Printer' for 'C' language sources
PROBE.ZIP	44588	880107	Execute Semi.exe and click icon to run. SubOp.
PROFIL.ARC	18332	891218	C language EXE program profiler (source only)
PROGWC.ZIP	1956	890628	Syntaxes C Prgms. {}, (), '', "", Nested Comment
QSORT.C	4886	850806	Quick Sort in C
QSORTNET.ZIP	2220	880726	C source to one QSORT -- f/usenet
QUEUE.ZIP	2441	880726	FIFO queue in C
R_LA4_01.ZIP	27690	880112	Thirty-three C programs
R_LA4_02.ZIP	40267	880125	Thirty C programs
RANDOM.ZIP	1996	880808	Random number generator
READWRIT.ZIP	4918	880726	L. Cuthbertson's read/write functions in C
REBOOT.C	961	900218	How to do a warm or cold boot in 'C' language
REDIRECT.C	2699	890427	Feed line to shell w/stdin/stdout/stderr redir
RENAME.ZIP	343	880808	Functional equivalent of DOS REN command
RES.ZIP	9816	880226	Reply to DOS RESTORE command w/C source
RFILE.ZIP	1365	880726	Accessing RANDOM files in 'C'
ROBOT01.ZIP	254227	911101	A Robot Control C Library, 1 of 4
ROBOT02.ZIP	46065	911101	A Robot Control C Library, 2 of 4
ROBOT03.ZIP	49860	911101	A Robot Control C Library, 3 of 4
ROBOT04.ZIP	41381	911101	A Robot Control C Library, 4 of 4
ROMAN.ZIP	661	880811	Convert Numbers to Roman Numerals
ROOTPATH.C	3131	860726	Conv pathname arg to root-based conical form
SCBENCH.ZIP	148537	890624	Good benchmark in Small C
SCI.ZIP	40762	890628	Version 1.3 Small C Interpreter
SCSIDRV.ARC	14106	890307	'C' source for SCSI disk drivers
SECRET.ZIP	21907	890628	CPM Encode/Decode Routines in C w/ Documentation
SHAR.C	644	850707	Concatenate text files like Unix 'shar'
SIGNAL.ZIP	11424	850917	Trap interrupts in Lattice C
SINE.C	6044	860601	Sine functions in 'C'
SMALL_C.ZIP	83678	890628	A Small C Compiler with Library and Source
SNAKE01L.ZIP	7850	880728	More C functions
SNAP.ARC	2739	880809	Dump memory area in HEX/character format
SORT.ZIP	1166	880101	C source for ShellSort & Quicksort
SORT2.ZIP	3577	871205	Lattice sorting function
SORTC.ZIP	12039	880727	A good ascii sort routine from DECUS
SORTDIR.ARC	2815	880729	Sorts directories
SOUNDEX.ARC	11648	791232	Soundex algorithm in C
SOUNDEX1.ARC	2048	790223	Soundex algorithm in C
SPEAKER.C	488	830915	'C' language routines to output sound to spkr
SPLINE.C	15722	860518	Interpolate using splines under tension
SQ_USQ.ZIP	17372	860503	C source of USQ and SQ (ver 18)
SRT.ZIP	1834	880809	Antoher C sorting utility
STAYRESC.ZIP	27459	860117	terminate & stay resident in C
STRINDS.ZIP	5847	871205	Find a string in C
STRING.ARC	77657	791232	More different string functions for C
STRLIB.ZIP	75635	871218	C string routine
STRUCT.ZIP	2668	880728	Hint on using filler in C structures
SUPER_C.ZIP	77926	880305	Source code from "Supercharging C"
SWITCH.ZIP	724	880808	Select active display
TABX.C	3564	900224	Filter that expands TABs to SPACES
TAGS.C	74214	860712	TAGS file system for C and Epsilon
TCBUG03.ZIP	7979	870624	C library documentation bug list
TCDEBUG6.ARC	71680	871120	Experimental debugger
TCDEV.ARC	4096	870819	Device driver with source code
TCXL.ZIP	130386	880615	Window library from CIS
TCXREF.ARC	13312	870628	Cross reference utility
TEACHC.ZIP	116973	880430	Teaches You How to Program in the 'C' Language.
TIME.ZIP	413	880809	A world time clock
TIMESTMP.C	3066	860121	Time & date routines for C86 and Lattice-C
TIMEX.ZIP	14717	871027	Timed execution of DOS cmds w/ C source
TOOLKIT.ZIP	36862	890628	A Good ACC Toolkit Utility File
TOOLS.ZIP	69920	880122	UNIX like tools with C source
TPRINT.ARC	3032	880809	Print one C function per page
TR.ZIP	2482	880728	Transliteration program
TSR_ASM.ARC	25012	880805	C-callable assembler routines to go TSR
TSRC.ARC	14713	880405	TSRs in C, source code from Computer Language
TTY.ZIP	11468	871205	General purpose terminal program
TYPESQ.ZIP	3425	880808	Type a squeezed ASCII file
UI6PC.ARC	15147	881208	'C' language source for 'uncompress' for PC
UNIXCLIB.ASM	9954	851014	Unix/C functions in MASM for DeSmet C
VT100TST.ZIP	14101	880101	vt 100 emulator and test in C
VTYPE.ZIP	12283	871205	Manipulate DOS TYPE command
WART.ZIP	20027	850707	Unix like LEX in C
WEEKDAY.C	1492	890511	Computes weekday for date w/Zeller Congruence
WILDCARD.ZIP	5253	880727	Wildcard command line parsing
WILDF113.ZIP	25438	910508	Unix sh-style regex parser (wild cards), C src
WIND_C86.ZIP	20005	880721	Window functions for C86
WINDOW4C.ZIP	2635	890630	Window Routines, WINDOW.DOC & WINDOW.PAS.
WINDSWT.ZIP	48447	850208	ADD WINDOWS TO YOUR C PROGRAMS
WINDSWT.ZIP	18279	850804	Window functions for Desmet (portable)
WPUTCHAR.ZIP	1318	880811	Fixes bug in TWINDOW.C
WR_PROT.C	1175	900215	Check to see if a drive is write protected
XC.ZIP	15192	880726	C cross referencer
XCL.ZIP	23035	871205	Cross ref for C w/Lew Paper additions
XENIX.ZIP	4036	880809	Ready to link XENIX
XMSLIB7.ZIP	19399	911021	XMS library for Turbo/Borland C/C++, w/source
XREF.ARC	2547	880812	Cross reference algorithm (version 1.0)
XRF.ARC	40324	850205	C crossref lister w/Computer Innovations C src
YACCUNIX.ZIP	85999	880416	UNIX / XENIX version of YACC w/ source
YEARCAL.ZIP	61142	880422	Super calendar program in C
PCB.ARC	127508	890325	Small PC board CAD pgm w/'C' src, 286 only

CALC

CALC.ZIP	5567	890613	Calculator Program.
DO.ZIP	10323	890612	Simple Command Line Calculator with TP4 Source.
HDBCALC.ZIP	6262	891030	HEX/Decimal/Binary programmers calculator
PRO_CALC.ZIP	67154	891026	Programmer's calculator and conversion program
RPN30SRC.ZIP	41387	900703	TCv2.0 source code for rpn v3.0 by bobon

CLIPPER

ACC86.ARC	41909	890927	Clipper 86 library written in ASM for speed
CLIPWIND.ARC	110675	890925	Windows for Clipper, no source, but LIB & docs
CLIPWIND.ZIP	5143	900509	A small CLIPPER window library with source
CODE_1.ARC	23851	891027	Sample source (template-like) for Clipper
NANNWS12.ARC	21193	891023	Nantucket News v1n2, dBASE/Clipper source code
NANNWS13.ARC	12460	891023	Nantucket News v1n3, dBASE/Clipper source code
NANNWS14.ARC	10366	891023	Nantucket News v1n4, dBASE/Clipper source code
NANNWS22.ARC	25077	891023	Nantucket News v2n2, dBASE/Clipper source code
NANNWS23.ARC	3533	891023	Nantucket News v2n4, dBASE/Clipper source code
NANNWS24.ARC	11143	891023	Nantucket News v3n1, dBASE/Clipper source code
NANNWS31.ARC	20845	891023	Nantucket News v3n3, dBASE/Clipper source code
NANNWS33.ARC	20480	891023	Nantucket News v3n4, dBASE/Clipper source code
NANNWS34.ARC	13312	891023	Nantucket News v3n5, dBASE/Clipper source code
NANNWS35.ARC	19214	891023	Nantucket News v3n6, dBASE/Clipper source code
NANNWS36.ARC	15808	891023	Nantucket News v3n6, dBASE/Clipper source code
NFSRC.ZIP	342677	911108	Nanforum Toolkit v2.0 Source code and makefile
NFTROFF.ZIP	122063	911125	Nanforum Toolkit v2.0 Norton Guides TROFF src
RLB20.ZIP	122409	910109	Clipper menus/dialog/record-select Tib w/src

CMPLNGMG

CL89APR1.ARC	22617	890602	Sources from Computer Language Magazine
CL_JUN89.ARC	97419	890706	Sources from Computer Language Magazine

COMPATBL

CKCOMPAQ.ASM	1984	841224	Return if system is COMPAQ
SET_PCSAVE.ASM	10242	850712	Scrn blanker for Zenith PCs
SET_CLTK.ASM	3645	841224	Set COMPAQ ROM BIOS data area key click count
TANDY232.ASM	8735	851006	Replacement for Tandy 2000 RS232 I/O handler

COMPRESS

COMP4305.ZIP	70872	910224	C src: Unix-compat. 16-bit compress/uncompress
DECOMP2.ZIP	17384	900430	Unix-compatible 16 bit uncompress, w/C source
HUFF_SC.ZIP	40299	910713	Huffman compress encoder/decoder, C src & EXES
LZ13.ARC	34046	881020	Lempel-Ziv file compress/decompress w/ASM src
LZHSRC10.ZIP	8158	890628	LZHUF Compression/Decompression, Turbo C Source!
LZW.ZIP	19821	851128	lempel-ziv compression in Attec C (crunching)
RL16_SC.ZIP	17165	910713	RLE compression (16-bit header), C src & EXES
RL8_SC.ZIP	16143	910713	RLE compression (8-bit header), C src & EXES
SQ_ZIP	10900	880728	File Compression Routines.
SQ_USQ.ARC	17773	860504	CP/M compatible file squeezer utility
SQUPORT.ZIP	23509	850626	portable squeeze-unsqueeze in C
TYPESQ.ARC	3381	880809	Type a squeezed ASCII file

CPLUSPLS

LINKLIST.CPP	5997	911115	Linked lists in C++, from Dr. Dobbs
MENUS.CPP	17611	911115	Popup/pulldown menus in C++, from Dr. Dobbs
SERIAL_1.ZIP	5164	901028	C++ source code routines for serial ports
TASKING.CPP	17861	911115	A C++ Multitasking kernel, from Dr. Dobbs
WINDOWS.CPP	14489	911115	Window system in C++, from Dr. Dobbs

CROSSASM

1802.ZIP	43229	891101	RCA 1802 cross-assembler
6502.ZIP	44339	891101	MOSTek 6502 cross-assembler
6800.ZIP	47757	891101	Motorola 6800 cross-assembler
6805.ZIP	44974	891101	Motorola 6805 cross-assembler
6809.ZIP	46470	891101	Motorola 6809 cross-assembler
68ASMSM.ZIP	161140	910222	68000 cross-assembler/simulator with C source
8048.ZIP	46621	891101	Intel 8048 cross-assembler
8051.ZIP	49116	891101	Intel 8051 cross-assembler
8080.ZIP	55668	891101	Intel 8080 cross-assembler
8096.ZIP	46392	891101	Intel 8096 cross-assembler
AS68K.ZIP	34176	890613	A Cross-Assembler for the Motorola 68000 in 'C'.
ASM370.ZIP	95684	890613	IBM 370 Assembler, Linker, emulator w/ 370 ALC
ASM49.ZIP	60457	890613	Intel 8748 Cross Assembler, in Turbo BASIC.
ASREF.ZIP	22588	900314	Reference manual for MOTOASMS cross assemblers
COTG292.ZIP	260696	901222	Cross-ASMs: 6800/6801/6804/6805/6809/8085/280
MOTOASMS.ZIP	92103	900314	Motorola 6800/01/04/05/09/11 cross assemblers
SVASMO2.ARC	39519	880604	Cross assembler for 6502 and 65C02
TSVASM276.ARC	140231	900223	Table-driven cross assembler, for many CPUs
UASM.ARC	107392	870328	Cross assembler for 8051/6805/28, w/'C' src
Z80.ZIP	49083	891101	Zilog Z80 cross-assembler

CRYPT

SES152.ZIP	128404	891031	Super Encipherment System - MS-DOS version
SES153.ZIP	182519	891031	Super Encipherment System - PC-DOS version

DATABASE

NAMES.ZIP	2482	880808	Name and address database in C
QAD12.ZIP	47727	900412	Fixed length ASCII to CDF file cvt, w/QB4 src
SDB.ZIP	90039	850629	Simple DB lib and main pg w/doc and src
TRILLOGY.ARC	80466	880512	A notes retrieval system, with BASIC src & EXE

DESKACCS

CAL.ARC	20965	870125	Makes calander for any month & year, w/C src
REMIN.ASM	8931	860613	Tickle file/reminder

DESQVIEW

CALCDV.ARC	36865	880313	A calculator for DESQview (w/MSC source)
DVAREW.ZIP	2369	910329	DESQview program interface ASM code
DVQLEU10.ARC	203128	880902	TC interface & funcs for DESQview API src/exe
DVLOAD.ARC	14958	880427	Load Desqview apps from command line, w/C src
DVMACROS.ZIP	5703	910114	MSM macros for Topview/Desqview pgm develop.
DVPIF10.ZIP	14074	910428	PIF editor for DESQview, with C source

DIRUTL

ALLDIR.ARC	6686	880728	Directory function
ANYWHERE.ZIP	21495	871205	Find Files Across Multiple Drives
CHATT.ZIP	14069	880412	Change file attributes from command line
CHNGFILL.ASM	3052	860613	Change file attribute, read-only/hidden/system
DELETE.C	4371	870730	Delete a file
DEQ.ZIP	9321	850920	selective delete w/wild card(.C and .exe)
DI_UZ.ARC	15043	890727	Displays disk free space for all drives, w/src
DI.ZIP	51816	880413	Directory listing utility
DILL.ARC	53250	880403	Extended directory lister v1.1 w/C source
DULL11.ZIP	22575	911107	du: UNIXish Disk Usage. Many options, w/source
DUL_PV.ARC	7450	891030	Displays dir tree and disk usage, w/A86 src
DUPS.ARC	17408	870517	All duplicate file names to STD out
FL.ARC	29231	891105	Fast File locator with Turbo C v2 source
KS_LS.ARC	27457	881018	4.xBSD Unix-like 'ls' dir for DOS, w/TC1.5 src
LS.ZIP	2824	890624	List Files on Printer, Accepts Wildcard Cmds.
MDS_RM.ARC	19120	881109	Unix-like 'rm' clone with C source
OLDER.ASM	2797	840228	Return which files are older
PUSH_POP.ZIP	23887	900727	Directory stack manipulation programs, w/src
PUSHDIR.ASM	10447	860715	Save current drive and subdir
PWD_C.ZIP	9070	901001	Displays working dir on all drives, w/MSC src
RENAME.ARC	306	880809	Functional equivalent of DOS REN command
RM_HS.ARC	15637	890313	File/directory deletion program with source
RM_OM.ZIP	21778	911025	Unix-style file/directory deleter, w/C source
SORTDIR.ZIP	2952	880728	Sorts directories
TCSEARCH.ZIP	2253	870619	Directory search routine in Turbo C w src
TK_LS.ARC	21566	890114	Unix-like ls dir. lister for MS-DOS w/src v1.0
TREE.ARC	7632	880726	Graphic TREE directory (PD) with masm source

TREEDIR.ZIP	2667	890630	Routines to Access Directories.
UPDATE.C	17949	890826	Copy new/alters files, with confirmation
WHEREIS.ASM	10294	851024	Locate file on hard drive
WHEREIS.DOC	657	851026	Doc for WHEREIS.ASM

DISASM

ASMGEN.ZIP	17604	891031	8086/8087/8088 disassembler
ASMGEN3.ZIP	75203	901123	Generate assembly code from COM and EXE's
DASM48.ARC	23065	871017	Intel HEX disassembler for 8048 microprocessor
DASM80.ARC	21757	871017	8080 disassembler
DASM280.ARC	59387	871017	Z80 Intel HEX file disassembler
DEB2ASM.PQS	21454	870722	Pascal pgm changes debug 'u' captures to .ASM
DIS86PC.ZIP	57046	891026	Full screen disassembler: 86,87,88,186,286,386

DJGPP

MERGE.C	915	910704	Binary file merger source (for DJGPP)
SPLIT.C	1172	910704	Binary file splitter source (for DJGPP)

DSKUTL

800KS.ZIP	9355	910428	Format 360k disk to 800k on AT, w/MASM source
BOOTCODE.ASM	9467	860128	Create your own custom boot records
CPYDSD.ZIP	18914	900407	Similar to DOS's XCOPY, only better. w/C src
FDX12.ZIP	14081	910509	Finds and marks bad disk sectors, w/C source
DISKFL1.ASM	4358	900215	Corrects AT Floppy disk timing problems
DSKST.ASM	13733	850807	Disk test program in Lattice C
DSKWATCH.ASM	6686	851006	Report disk errors in screen window
DTSTZ0.ZIP	15214	900810	Scan disk weak sectors which may fail, w/C src
DU.C	7070	870515	Routine to display disk space usage
FIXDSK.ASM	3336	851225	Corrects AT Floppy disk timing problems
HDSENTRY.ARC	3584	870531	Rudimentary hard disk protection (w/ASM)
HOSENTRY.ARC	95713	870822	PD ver of IBM's Advanced Diags w/ASM source
LAN.ARC	75144	880630	Floppy-based LAN w/ Pascal source
LLFORMAT.ASM	2428	910115	Hard disk low-level format pgm. ASM source
PARK.ZIP	613	880809	Park that hard disk (source)
PASSWRD6.ARC	7832	900122	Install password on hard disk, w/ASM source
PAU50.ARC	3381	890511	Park one or two hard disks v5.0, w/ASM source
POLLIMPR.ARC	9269	870926	Public Domain auto HD park, with source
PROTECT.ASM	6045	880126	Toggle write-protect on hard disk
QFORMAT.ARC	7808	880422	Memory resident quick disk reformatter (w/ASM)
REFORMAT.ARC	24244	870523	Disk defragmenter [Turbo Pas src]
REFORM16.ARC	63973	900303	Disk defragmenter with TurboPascal v5.0 source
VACUUM.ARC	6144	870722	Recover data from damaged disks w/ASM
VARVDISK.ASM	12674	831024	Variable-size RAM drive
VERDISK.ASM	12751	850713	Check for diskette or hard disk for errors
VERDISK.DOC	953	850713	Documentation for VERDISK.ASM
ZERODISK.ARC	8423	890322	Erase unallocated disk blocks - with C source

EDITOR

AE14.ZIP	86777	910427	Multifile text editor w/cut & paste, w/TP src
DTES1.ZIP	118586	910206	PD text editor for MSDOS & mainframes. C src
E10_EDIT.ZIP	41804	900402	Small, fast, general purpose editor, w/ASM src
EDIT1.ZIP	7981	880721	Another editor in C
EDIT_C.ZIP	59635	880721	Wordstar like editor in C
EL0191.ZIP	24815	911011	Free ASCII editor for MS-DOS/Unix with source
ELVIS145.ZIP	162905	910629	Sources for ELVIS v1.4 vi-compatible. text editor
J414SRC.ARC	351831	891108	Source code in C for MSDOS JOVE text editor
ME39SRC1.ZIP	95149	880128	MicroEMACS ver 3.9 source part 1
ME39SRC2.ZIP	148567	880128	MicroEMACS ver 3.9 source part 2
MG2A.SRC.ZIP	394600	910322	MicroGnuEmacs: GNU emacs-compatible. editor, C src
PT20PC.ZIP	266052	910209	Point Editor v2.0 (PD). C src and executables
STEVIE695.ZIP	156384	900503	Source for STEVIE vi clone v3.69X (for TC2.0)
STEVIE.ZIP	156384	880729	UNIX V1 like editor
TDE120.ZIP	180680	911015	PD multi-window/file text editor w/C & asm src
TED2.ZIP	34752	900925	Fast editor for small text files, w/MASM src
TVX_EDIT.ARC	199906	860317	TVX editor 'C' source and executables for PC

EDUCATON

CATTST.ZIP	285219	910702	Computer Assisted (Tech) Testing w/TPAS srce
GEO_MATH.ZIP	129982	900903	Math tutor for grades 1-6. Includes BASIC src
INF_SRC.ARC	58205	871231	Information system, part 2 of 2
MAZE2A.ARC	78223	890712	Generate/solve mazes on your PC, w/TC2.0 src
POLYSYM.ARC	33973	890717	Polynomial simplex optimization, w/C source
TYPRD11.ZIP	17906	900912	TypeDrill: Improve typing skills, w/TP src
TYPEFAST.ZIP	91532	900912	TypeFast: Game stresses typing speed, w/C src

EEL

ASMTAG04.ZIP	10309	911017	Tag Assembly source in MASM/TASM syntax
CTAG05.ZIP	22470	911017	Comprehensive Tagging of C/C++ source code

EGA

CGRAPH.ZIP	13923	860921	EGA graphics primitives in C
EGAPRINT.ARC	8201	891029	Use PrtScr with EGA & Epson printer, w/asm src
EGAPRINT.ARC	124288	880527	Screen paint with source in Turbo C 1.5
EGASAVE.ZIP	26823	880109	SAVE RECALL EGA SCREEN IMAGES
EGASMALL.ASM	2160	850615	Load EGA 8x8 font for 25 or 43 line screens
EGASMALL.DOC	672	850615	Doc file for EGASMALL.ASM
HSA_TC.ZIP	57560	871215	C graphics lib for Turbo C CGA EGA HGC
SPEEDCRT.ARC	7780	881111	Resident video speedup for EGA, w/ASM source

EMULATRS

APL2EM.ZIP	216652	900808	Apple II emulator for 286+, w/ASM source code
V20BOOT.ZIP	81330	900522	Turbo Pascal source code for V20 CP/M emulator

FIDO

MSGD2SRC.ZIP	99726	901027	MSGED v2.05 Fido msg editor/reader source code
PUP_V2B.ZIP	127556	900904	Small BBS - like FIDO, C language, with source

FILEDOCS

IDXD2AT.C	7513	910312	Makes AutoFTP/BatchFTP scripts from SIMIBM.IDX
SIMCVT.C	2859	900427	Unix C program to print SIMIBM.IDX
SIMCVT2.C	6944	910104	Unix C program to search and print SIMIBM.IDX
SIMRAZ13.ARC	18855	891228	Shorten/rearrange SIMTEL20 indices, w/TP src
SIMSERCH.C	8219	910509	Search through SIMIBM.IDX. TC-2.0/VMS-C source

FILUTL

ADDLF.ZIP	539	880809	Add line feeds to files
ANYWHERE.ARC	21463	871206	File search across directories and drives
BEYOND.C	32768	880624	ASCII file comparison utility
BINCOMP.ZIP	12545	880717	The Telemachus File Comparison Utility in Turbo C
BRK2SRC.ARC	63549	890806	General-purpose CRC-32 pgm, v2.0, C source
CAT.ZIP	572	880811	Concatenate files
CHATT.ARC	14080	880413	Change file attributes from command line
CMP.ZIP	3675	880809	File compare in C source
CONCAT.ZIP	635	880809	Concatenate 2 files into a third
CPMFILES.ARC	4684	880809	File manipulation for CP/M in C
CRCBBS.ZIP	73412	851120	CRC error checking
DATE.ARC	8001	880809	Change date & time of file with source
DETAR.ZIP	19720	880315	DOS program to de-TAR UNIX .TAR files
DIFF.ARC	3627	880722	Source file comparer in C
DUMP.ZIP	3098	880808	Print parts of large files in DUMP format
DUMPS.ZIP	3908	800101	ascii/hex dump in C
FILEHEX.ASM	4182	830813	Hexify a binary file
FILEIO.ARC	1519	880728	File I/O functions
LDIFF12S.ARC	37310	890731	Source code for LDIFF12 compressed .DIF pgm
LUMPS.ZIP	1697	880808	LUMP / UNLUMP files
MLVINE.ARC	11392	870215	Move lines from one file to another
OCTDUMP.ZIP	7073	880726	Flexible file dump in C (OCT/HEX/ASCII/BIN)
OO.ZIP	3601	880726	Binary dump program, like UNIX utility.
P22V12.ZIP	30208	910808	Flexible & fast patching of any file, w/C src.
PRINT.ARC	1775	880809	Print TXT files w/page #, headers, etc...
REBUILD2.ZIP	14051	901108	Rebuild split uuencoded files, w/MSC 5.0 src

STRING.ZIP	69569	800101	Display ASCII strings in any file
STRINGS.ARC	11008	880312	Locate ASCII strings in binary files (w/C src)
TABALT.ZIP	16236	870826	Match tab spacing in C program to editor
TABS.ZIP	3178	871205	C source for tabber/de-tabber
TAIL.ZIP	1600	880728	View the end of an ASCII file
TOADSHR1.ARC	34760	890615	DOS Unix-compatible shar/unshar, w/TP 5.0 src
TOADU20.ZIP	25078	910312	Unix-compat. fast UUencode/UUdecode, w/ASM src
TPRINT1.ARC	10881	871206	Print an ASCII file
TR.ARC	2425	880729	Transliteration program
TSORT.ARC	3285	880729	Text file sort in C
UNSHAR.ZIP	19055	890630	Program to Decode Files Created by UNIX Shar Prg
UNTAB4.ARC	11436	880809	Remove tabs from C source listing
UUXFER20.ZIP	29996	901106	Comprehensive UU/XX en/decoder with C source
ZAPLOAD.ZIP	22600	880808	Converts binary files to Intel hex format w/ doc

FORTH

4THPROC.ZIP	115096	891030	A collection of FORTH language tools
4THUTOR.ZIP	37042	891101	A FORTH language tutor
EFORTH.ZIP	54528	900829	Ting's PD portable eForth, ROMable, w/asm src
F83HEL2.ZIP	12078	890630	Notes on the F83V2-80 FORTH for the Intel 8080.
F83V2_80.ZIP	140262	890630	Welcome to FORTH for the Intel 8080.
FIG_4TH.ZIP	130446	891031	Joe Smith's FIG-FORTH compiler
FORTH.ZIP	32959	890630	The Computer Language of FORTH.
MINI4TH25.ARC	17581	900120	MiniForth system v2.5, with A86 source
PAS386C.ZIP	66304	900307	386/386SX assembler v1.4 for Forth-PC v3.5
PCFORTH.ZIP	9322	890630	Version 1 Fig-Forth for the Intel 8086 and 8088.
PYGMY13.ZIP	67072	900921	Forth: src,editor,metacompiler,multi.open files
ZEN1_10.ZIP	68608	900220	Zen1.0 Forth w/src to match ANSI X3J14 BASIS 10

FORTRAN

CODE3_4.ZIP	156772	891031	PC-CODE 3 & 4 Fortran 77 encryption
D2PC.ZIP	70507	890612	Downwind Hazard Program D2PC.
D2SOURCE.ZIP	31583	890612	Downwind Hazard Program FORTRAN Source for D2PC.
F2C_SRC.ZIP	341585	910615	Source for f2c, Fortran 77-to-C converter
LINPACK.ZIP	127103	891031	LINPACK library of FORTRAN programs
LLSQ.ZIP	41754	890612	High-Quality Mathematical FORTRAN Subprograms.
MISCFCORN.ZIP	48318	891031	Misc FORTRAN programs
RATFOR.ZIP	55844	891031	RATFOR control constructs and text insertion
SUPERF87.ZIP	289405	891031	SuperTrek 8087 Fortran source code
TIDV621.ZIP	118474	900926	Clean up FORTRAN-77 programs w/FORTRAN src

FREEMACS

EMAC16AS.ZIP	102948	900329	FreeEMACS 1.6a text editor - .ASM sources
GIF			
GIF_CSRC.ZIP	5019	901218	C source code for a GIF decoder
GIF_PAS.ZIP	59023	901218	Pascal source code for a GIF viewer
GIFSRC.ZIP	41066	901218	Pascal and 8086 ASM source for a GIF viewer
PIXIT.ARC	16143	900203	Converts GIFs to COM and EXE files, w/ASM src

GRAPH

GRAPHSRC.ZIP	107847	900716	GRAPH v3.21 scientific graphics, Turbo C src
--------------	--------	--------	--

GRAPHICS

3DBB.ARC	57716	881130	3D black board - BASIC CAD with source
COLOR.ARC	4936	880809	Set screen color in text mode
DUMP_GRA.ZIP	1337	880721	Graphic dump in C
FRAME.ARC	1004	880809	Draw a frame on the screen
FRAS1611.ZIP	722922	910726	C & ASM src for FRACINT v16.11 fractal gen.
GRAPH.ZIP	10850	870712	Displays line graphs on Z-100
GRAPHICS.ZIP	6275	871205	Various graphics routines
HERCBIOS.ZIP	41750	871116	Hercules BIOS for CGA simulation
HPLVU10.ZIP	89457	910429	Viewer for HPLG files, with Turbo Pascal 5 src
HSATC.ZIP	59360	880107	Graphics control library
LFEE.ZIP	2884	871216	Graphics toolbox for Turbo C
MANDEL.ZIP	37181	890624	Graphics ver. of the board game life
MFRACT.ZIP	1502	880809	A Utility to Generate Mandel's Fractal Geometry.
PCX_C.ZIP	26890	910418	Fractals in C (needs GRAPHICS.H)
RCDSPLAY.ZIP	166208	910815	C source to handle PCX format graphics files
SHOGL10.ZIP	127243	910911	Display/analyze/edit X-Y graphical data. w/src
SIMGRAFC.ZIP	32381	850615	View HPLG/2 files on CGA/EGA/VGA w/MSC src
TCWIN10.ARC	74752	870723	simple graphics in C
TCXL.ARC	141312	880615	Evans' window routines
TTY.ARC	12195	871206	Window library from CIS
TURBOOGI.ARC	17408	870222	General purpose terminal program
VSXNC.ZIP	305	880809	Four frame .DGI picture
WIND_C86.ARC	19638	880722	Output to CRT during verticle blanking
WINDKW.ZIP	56341	851120	Window functions for C186
WINDOW.ARC	18432	870316	window functions for Aztec C -- nice!
WINDOW4C.ARC	48257	880728	A simple window management program
WINDOWNPR.ZIP	248469	890628	Windowing functions
WINDOMET.ARC	19898	850805	The Window Pro Version 1.2 Screen Control Lib
WINS1651.ZIP	311041	910815	Window functions for DesMet (portable)
WINDWOSS.ZIP	155442	880802	WINFRACT v16.51 C and ASM source
			Neat windowing in C with demo

HAMRADIO

GC103.ARC	23776	900127	Hams: Great circle distances & bearings, w/src
ICON.C	9014	871115	Hams: control your ICOM radio from PC
MININEC3.ARC	158720	870128	Radio antenna design aid, w/BAS src and EXEs
USAT90.ZIP	17670	900406	Real-time satellite tracking program w/BAS src
VEFC24.ARC	26774	900113	Hams: VE code test validator/timer, w/C source

HYPERTXT

EBKSR.C.ZIP	56013	911005	Borland C++ src for EBK.ZIP hypertext viewer
-------------	-------	--------	--

ICON

ICON.ZIP	159477	891026	MS-DOS Icon programming language
ICON5.ZIP	126979	891031	ICON programming language version 5.0

INFO

DHAMP.C	5830	880510	The dhampstone benchmark
DOS3SRC.ARC	10274	860921	ASMGEN .SEG files for DOS/BIOS disassembly
HELP_7N1.ARC	22673	871216	Capture and display HELP screens w/ASM
HRO91.ZIP	56152	910908	DRG's directory of human resources exes 09/91
IFP15147.ZIP	75594	910807	TP & MASM source code for InfoPlus v1.47
INTS.ARC	3606	880221	Displays page-zero interrupts w/asm
MEMLIST.ARC	8543	880923	Show MS-DOS memory allocation info, w/MASM src
STATUS21.ARC	9866	880811	List PC hardware and configuration, w/ASM src

IRIT

DRAWFN35.ZIP	72439	900511	Turbo C ver 2.0 sources for DRAWFN3D
DRAWFUNS.ARC	81101	890924	Turbo C ver 2.0 sources for DRAWFUNC
IRITLIBS.ZIP	15023	900511	Libraries for Irit sources
IRITS.ZIP	231209	900511	Turbo C ver 2.0 sources for Irit
POLY3DHS.ZIP	63819	900511	Turbo C ver 2.0 sources for POLY3D-H
POLY3DRS.ZIP	81448	900511	Turbo C ver 2.0 sources for POLY3D-R
POLY3DS.ZIP	94023	900511	Turbo C ver 2.0 sources for POLY3D

KEYBOARD

CAPCTRL.ASM	1940	880119	Swap left shift and control key on keyboard
CLICK10.ZIP	2763	910410	Free keyboard click program with asm source
CTRLABEL.ASM	7263	891217	Provides a means of disabling CTRL+ALT+DEL
DVORAK86.ASM	4668	830731	Remaps PC keyboard to DVORAK layout
DVORAKXT.ASM	4706	850613	Remaps PC-XT keyboard to DVORAK layout
FX101KY.ASM	5264	880924	AT-keyboard key redefiner. w/ASM
INKEY.ARC	1826	880809	Input from keyboard function
INPUTS.ARC	212	880809	Inputs string from keyboard
KBEDIT20.ASM	74522	871211	Expand kbd buffer, cmd line storage and edit
LOCK.ARC	4619	900129	Lock kbd to protect network connects, w/ASM src
NUM_OFF.ASM	413	880510	Turn numlock off
RECALL11.ZIP	23688	910810	Commandline editor/history utility, w/ src
REKEY.ZIP	7620	850626	key redefinition in C
SCNCODTP.ZIP	7107	901015	Display scancode for any key. (TP src & EXE)
SWAPKEY.ASM	4671	840912	TSR allows swapping keys on keyboard
SWPKEYAT.ASM	6922	850710	TSR allows swapping keys on keyboard
SWPKEYAT.DOC	2894	850715	Documentation for SWPKEYAT.ASM
TEMPL.ZIP	23356	880605	101 keyboard template generator

LAN

NBSAMP.ARC	34816	881021	NETBIOS send/rcv/test w/C source
------------	-------	--------	----------------------------------

LASER

HPMAZE02.ZIP	62759	910521	Generate 3D mazes on HP LaserJet, w/TP source
JPSRC11.ZIP	68091	910806	Source code for JETPAK11 utilities

LISP

ADLER.ZIP	26252	891031	A LISP interpreter from Marc Adler
LISP.ZIP	29249	890630	Nice LISP Computer Language with Documentation.
PDLISP.ZIP	191070	891026	Cybermetrics' PDLISP and UNKLISP
XLSP20.ARC	141293	880306	LISP program with C source

LOGO

LOGOPAT1.ZIP	9613	910127	Patch IBM LOGO for EGA/VGA/MCGA/386 w/TP55 src
--------------	------	--------	--

MAC

EDMAC.ARC	30520	880308	Edit READMAC/FMAC files. Source included
READMAC4.ARC	2327	880308	Read MAC pictures on CGA/Herc, with source

MATH

C-MATH.ZIP	21476	851029	math functions for C
MATH.ARC	115729	880810	Various math functions in C
MATRIX.ARC	11399	850114	Dornelle's matrix benchmark
PRAXIS.ARC	28241	870727	Minimize function values (source in Pascal&C)
SHUFFLE.ZIP	8336	910531	Number list order randomizer, with C source
SMATH.ZIP	8886	880728	Decimal math of #'s stored as strings

MENU

EXEMNU22.ZIP	21129	910516	Simple easy to use menu system, w/TurboPas src
TOADMENU.ARC	39374	871101	Menu shell with Turbo Pascal source

MICROCRN

ISSUE_30.ARC	4157	881126	Sources from Micro Cornucopia Magazine
ISSUE_31.ARC	10907	881126	Sources from Micro Cornucopia Magazine
ISSUE_32.ARC	16232	881126	Sources from Micro Cornucopia Magazine
ISSUE_33.ARC	21755	881126	Sources from Micro Cornucopia Magazine
ISSUE_34.ARC	12753	881126	Sources from Micro Cornucopia Magazine
ISSUE_36.ARC	37579	881126	Sources from Micro Cornucopia Magazine
ISSUE_37.ARC	15567	881126	Sources from Micro Cornucopia Magazine
ISSUE_38.ARC	19325	881126	Sources from Micro Cornucopia Magazine
ISSUE_39.ARC	66155	881126	Sources from Micro Cornucopia Magazine
ISSUE_40.ARC	138240	890520	Sources from Micro Cornucopia Magazine
ISSUE_41.ARC	118766	881126	Sources from Micro Cornucopia Magazine
ISSUE_42.ARC	51243	881126	Sources from Micro Cornucopia Magazine
ISSUE_43.ARC	33986	881126	Sources from Micro Cornucopia Magazine
ISSUE_44.ARC	101950	881126	Sources from Micro Cornucopia Magazine
ISSUE_45.ARC	52224	890520	Sources from Micro Cornucopia Magazine
ISSUE_46.ARC	52224	890520	Sources from Micro Cornucopia Magazine
ISSUE_48.ARC	27648	890607	Sources from Micro Cornucopia Magazine
ISSUE_49.ARC	61440	890802	Sources from Micro Cornucopia Magazine

MICROSFT

XMS20.ARC	127544	890331	Microsoft XMS distribution disk - sources/docs
-----------	--------	--------	--

MISC

CBT.ZIP	110777	890630	Language for Creation of Computer Based Trning.
IFP.ZIP	48787	890630	Interesting Experimental Language.
IFPOS.ZIP	108246	890630	Experimental Language. More Complete than IFP.
OLD_TEST.ZIP	1227962	891101	Programming comment testing utility
PCUNIX.ZIP	81550	890630	UNIX-like Utilities for IBM PC and Compatibles
PICNIX.ZIP	236174	890630	UNIX Commands for PC's. PC/MS-DOS 2.1+ Req.
RANDOMS.ZIP	5800	890612	Random Number Generators in 'C', FORTRAN, Pascal
REC.ZIP	334095	891031	Regular Expression Compiler
SPE.ZIP	114488	891031	Structogram and programming editor by Sass
TNH_PC.ZIP	138033	891031	Tips, Notes and Hints about the IBM-PC
UNSHAR.ZIP	19055	890630	Program to Decode Files Created by UNIX Shar Prg
V_TEACH.ZIP	110175	891031	Virtual teaching language interpreter
XLISP114.ZIP	35381	890630	An Experimental Object Oriented Language.
ZBASIC.ZIP	26403	890630	V3.01 Z100 Limited Public Ver from Zedcor Inc.

MODEM

ASM_TERM.ARC	6635	880829	Simple interrupt-driven terminal w/src/AS86asm
ASYNCL.ARC	16250	880808	COMM driver [.ASM]
ASYNCL.ARC	25454	880808	COMM driver [.ASM]
BAUD.ZIP	1236	880721	Determine baud rate in C
BYEPC300.ARC	205910	890409	BYE, remote operation pgm for PC, w/source
BYTECOMM.ZIP	76109	911114	Background COMM pgm w/TC2.0 source (BYTE Mag.)
CHKMODEM.ZIP	40331	900628	Finds a modem & checks its status, w/source
CKD11.ARC	9350	890915	Check modem CD status from cmd line, w/TC src
CLINK120.ARC	31077	880801	Ver. 1.20 SEALink file xfer program w/C source
CMODEM.ZIP	6359	880721	Modem program in C for Hayes compatible
COM_PKG1.ARC	5424	840420	COMM drivers (source only)
COM_PKG2.ARC	13914	851006	COMM drivers (source only)
COM_PKG3.ARC	18505	860601	COMM drivers (source only)
COMM.ARC	3012	880809	Small communications program (DeSmet)
COMM.C	4775	840406	COMM driver [.C]
COMMSUPP.ARC	26569	880722	Communications support for DeSmet C
COMRING1.ZIP	5778	850707	Lattice C lib of comm routines
GLASSMDM.ASM	18220	860715	COMM driver
GLASSTTY.ASM	4123	850911	Dumb terminal for GLASSMDM comm driver
JMOD309.ZIP	90576	910715	Jmodem file transfer protocol, with C source
KOMM.ASM	28588	850707	COMM driver
KOMM.INF	1198	850707	Doc for KOMM.ASM
MNPCI2.ZIP	48762	910603	MNP protocol - includes Turbo C source
MUFUS32R.ZIP	64031	911124	Microfusing MF30 terminal emulator w/TP5 src
OUTSTNG.ZIP	1117	880809	Output string to COMM port
RS232.ARC	4006	870805	Source for COMM drivers (.ASM)
RZ52.ARC	62288	880108	ZMODEM source for UNIX
RZ52.ZIP	64512	880725	Reformatted ZMODEM source for UNIX
SETHAYES.ZIP	1183	880809	Set up HAYES compatible modem
SETHAYES.ARC	13898	890124	Set up modems for BBS use, with Pascal source
TCOMM11.ARC	28672	870702	Communications library for Version 1.1
TSRCOMM.ARC	25786	880811	TSR serial device driver
TT_104.ARC	17852	880811	Tiny Terminal - Fossil front end (w/ASM)
UNPC201.ZIP	295632	911116	PC<->UNIX multi-window comm pgm, w/server src
WXMDEM.ARC	51906	890730	TP source for early version of WXMDEM module
WXTERM.ARC	29440	880217	Source code for WXTERM (WXMDEM) modem program
XMODEM.ZIP	26717	850615	Xmodem protocol to run under CrossTalk
XMODEM11.ZIP	5242	880728	Xmodem utilities

YAM.ZIP

63073	880728	Yet another Modem (C source BDS for CPM)
-------	--------	--

MODULA2

FM2CMP.ZIP	130421	890630	V1.1s Fitted Software Tools' Modula-2 Compiler.
M2_TUTOR.ZIP	194555	891101	A Modula-2 tutorial version 1.0
M2BOOKS.ZIP	1540	890630	List of Some Modula-2 Books and References.
MOD2SRC.ARC	98848	890114	Coronado's Modula2 tutor v2.0, (1 of 2)
MODULA2.ZIP	205184	891026	Fitted Software Tools' Modula-2 compiler
SORT_MD2.ZIP	3023	890630	Modula-2 QUICKSORT Customizing and Sorting Prgrms
TWOTREES.ZIP	2053	890630	Modula-2 Code Used to Search Two Trees at Once.

MOUSE

CMOUSE2.ZIP	26201	901121	Mouse management for MSC 5.1/MASM 5.0 or later
MENUNOUS.ARC	3000	890129	Map mouse to arrow keys, TSR, w/ASM source
MOUSE.ZIP	5801	850910	make joystick into mouse

MUSIC

POLYTC.ZIP	33152	910531	Three-voice music w/sample songs & TC/MASM src
------------	-------	--------	--

NCSATELN

TEL235RC.ZIP	719865	910822	Source for NCSA Telnet 2.3.03 and utilities
--------------	--------	--------	---

NETWORK

FIX36.ASM	4272	880228	Fix image FTPs from TOPS-20 systems (bit conv)
MONITOR.ASM	9310	860120	Run another PC console remotely on IBM PCNet
NETIO.ASM	16866	860120	Device driver for MONITOR.ASM
NETIO.DOC	2385	860119	Documentation for MONITOR.ASM and NETIO.ASM

NEURNET

GENALG.ZIP	34694	910815	A simple genetic algorithm demo program w/src
NASANETS.ZIP	171090	910815	A back prop neural net system W/ C src
NERVES.ZIP	294531	910815	Nervous System Construction Kit w/ C++ src

OPUS

FVSR620.ARC	55084	890607	View many compressed files on OPUS BBS's 2of2
-------------	-------	--------	---

PACKET

CBS6050.ARC	111104	890303	Source code for CBS Mailbox 6.0 from K3RLI
PMPSRC11.ZIP	65407	910914	Packet Radio term pgm for PCs without TNC. 2/2

PASCAL

CALCULAT.ZIP	5915	890613	MT Scientific Calculator w/ Trancendentals.
CCAT26.ZIP	21937	890613	Compares to Catalog Files, w/ Docs, Pascal Src.
CONSOLE.ZIP	15401	890613	MS Pascal Console Routines, Input Editing & Docs
CONVERT.ZIP	5049	890613	Conv TP 2.008 Chain Fls into Callable COM Files
CRFPNR.ZIP	889	890613	Print X-Ref Data Files from PCRF.PAS.
INTOOL.ZIP	2067	890624	MS Pascal Routines Avoids Data Format Error.
IOSTUFF.ZIP	48150	890624	PC Pascal Low Level thru ROM Disk Sec Mod.
KRTOOL.ZIP	55131	890624	V1.9.E Kernighan's Software Tools for Pascal
METRIC.ZIP	29317	891101	Measures complexity of PASCAL programs
PASC2.ZIP	196920	890628	Any Pascal to C Conversion Routine Version 1.51
PAS_ENG.ZIP	60301	891030	Common use PASCAL for engineers
PASCSRC.ARC	89450	890114	Coronado's Pascal tutor v2.0, (1 of 2)
PT01.ZIP	118013	891027	PASCAL programming tools, 1 of 3
PT02.ZIP	122632	891027	PASCAL programming tools, 2 of 3
PT03.ZIP	86682	891027	PASCAL programming tools, 3 of 3
PTOC.ZIP	99997	880812	Pascal to C conversion program
PP_PAINT.ARC	32677	890919	OOP paint program source in QuickPascal
TPAS2MSC.ARC	153600	880121	Convert Turbo Pascal to MS C
TPC14.ARC	39936	870620	Turbo Pascal to C translator
TPSPSPEECH.ZIP	15822	890630	Speech Sounding Routine in BIN File
VALZAH.ZIP	54600	891031	Bob Van Valzah's PASCAL compiler
VISPAS.ZIP	102319	890630	Public Domain Visible Pascal Compiler.

PCMAG

ASM1.ARC	50816	880221	Source for DOS utilities from PC Magazine
ASM2.ARC	49428	880221	Source for DOS utilities from PC Magazine
ASM3.ARC	44628	880221	Source for DOS utilities from PC Magazine
ASM4.ARC	45824	880221	Source for DOS utilities from PC Magazine
VOL6M17.ARC	42290	880228	PC MAG Utilities With Source Code - CARDFILE
VOL7M17.ARC	69402	880920	PC Mag Vol7#17: Hard disk DCACHE w/ASM etc.
VOL7M20.ARC	37542	881028	PC Mag Vol7#20: CONFIG etc. w/sources
VOL7M22.ARC	114248	881113	PC Mag Vol7#21: LOG, Write, Ansi etc w/source
VOL8M07.ARC	56889	881205	PC Mag Vol7#22: ENVELOPE etc w/sources
VOL8M08.ARC	41634	890315	PCMag align, LASERLIST, maxram, srchixf, stringl
VOL8M08.ARC	23588	890411	PCMag COLORS, loader, makehash, mousect1, srchash
VOL8M22.ARC	54829	891207	PCMag alias bckup jmp log netadd/num srchdel
VOL9M22.ZIP	43873	901208	PCMag: Ctrtext, Glass, Highlight, Winsrc, Winwhere

PCRESRC

PCR8807.ARC	72726	880829	Utilities from PC Resources magazine, 7/88
PCR8808.ARC	7197	880829	Utilities from PC Resources magazine, 8/88
PCR8905.ARC	19622	890430	Utilities from PC Resources magazine, 5/89

PCTECH

HLSRC.ARC	65078	881216	Source for High Level performance benchmarks
PTCJ1188.ARC	45885	881219	PC Tech Journal CRT utility pgms w/source

PGMUTL

DOSRCSEX.ZIP	136791	900829	Revision Control System for text files, 1of3
DOSRCSPM.ZIP	24412	900829	Revision Control System for text files, 2of3
OBORNM11.ZIP	165120	910429	Oboron-M compiler v1.1, w/sources and docs
PCTAGS1.ARC	249981	891023	MS-DOS & OS/2 source code retrieval system
SWITCH22.ZIP	38806	901230	Swap to disk or EMS from within ASM, C, Pascal
TAGS15.ZIP	195742	911017	Comprehensive Tags Generator for C/C++, ASM
VAL_LINK.ARC	214539	900217	Troendle's VAL experimental linker, w/C source

PIBTERM

PiBT4151.ARC	215741	880325	PibTerm 4.1 source (Turbo Pascal) 1 of 4
PiBT4152.ARC	233677	880325	PibTerm 4.1 source (Turbo Pascal) 2 of 4
PiBT4153.ARC	205056	880325	PibTerm 4.1 source (Turbo Pascal) 3 of 4
PiBT4154.ARC	233089	880325	PibTerm 4.1 source (Turbo Pascal) 4 of 4

PLI

RUNPLI1A.ARC	195020	890714	PL/I-like language Interpreter, w/MSC5.0 src
--------------	--------	--------	--

PLOT

3DF.ZIP	26096	910810	3D func. plot displays F(x,y) ortho proj,w/src
3DPLOT02.ARC	215445	890803	Standalone 3D plot interpreter, w/TurboC src
PLOT3D_A.ZIP	71272	910621	3D surfaces in any coord system, with TC src

PRINTER

CB1.ZIP	12867	871205	Printer formatter
GOTHIC.ARC	28672	880308	Gothic banner printer with C source
LPTX700.ARC	48801	880206	Updated LPT (redirect PRN to file) [.ASM src]
NOBS.C	4009	880913	Removes backspaces from text
PAGE.C	1944	840707	Print files with tab expansion and pagination
PCLCM128.ZIP	33989	910530	Compresses HP-PCL raster data. With C source
PRFORM17.ZIP	29440	880616	Multi-function printout formatter, w/C source
PRN2FILE.ARC	14451	871216	Redirect printer to disk (w/asm)
PRTFILE.ARC	35444	871121	Text file list utilities w source
SETPRTR.ZIP	2777	880808	Printer setup utility
SIDER.ARC	29074	880829	Print spreadsheets sideways, TP Source & .COM
SPOOL.ZIP	565	880809	A Memory-Resident Print Spool

PJ.5_4.ARC	6068	900110	Sources from Programmers Journal, vol 5 nr 4
PJ.5_5.ARC	13777	881126	Sources from Programmers Journal, vol 5 nr 5
PJ.5_6.ARC	154646	881126	Sources from Programmers Journal, vol 5 nr 6
PJ.6_1.ARC	323090	881207	Sources from Programmers Journal, vol 6 nr 1
PJ.6_2.ARC	18465	881126	Sources from Programmers Journal, vol 6 nr 2
PJ.6_3.ARC	24228	881126	Sources from Programmers Journal, vol 6 nr 3
PJ.6_4.ARC	65681	881126	Sources from Programmers Journal, vol 6 nr 4
PJ.6_5.ARC	32975	881126	Sources from Programmers Journal, vol 6 nr 5
PJ.6_6.ARC	163659	890602	Sources from Programmers Journal, vol 6 nr 6
PJ.7_1.ARC	53404	890602	Sources from Programmers Journal, vol 7 nr 1
PJ.7_2.ARC	127327	890602	Sources from Programmers Journal, vol 7 nr 2
PJ.7_3A.ARC	96329	890603	Sources from Programmers Journal, vol 7 nr 3A
PJ.7_3B.ARC	109357	890603	Sources from Programmers Journal, vol 7 nr 3B
PJ.7_5.ARC	98291	891204	Sources from Programmers Journal, vol 7 nr 5
PJ.7_6.ARC	157496	891209	Sources from Programmers Journal, vol 7 nr 6
PJ.9_3.ZIP	97349	910609	Sources from Programmers Journal, vol 9 nr 3
PJ.VGA.ARC	93682	900108	Programmer's Journal VGA ASM routines src code

PSTSCRIPT

ASCII_PS.ZIP	47816	900511	Converts ASCII-to-PostScript, w/QuickBasic src
DVTP554.ZIP	321884	910213	Convert a TeX DVI file to PostScript, w/C src
HPGL2PS.ARC	118499	890826	HPGL to Postscript trans, DOS/UNIX, src, EXE's
PS211.ARC	14457	900104	ASCII-to-postscript converters. utl (TP src only)

QBASIC

ASWIZ214.ZIP	85853	911004	Assembly Wizard: ASM routines for QuickBASIC
BSAQ0811.ZIP	53200	910124	Converts BASIC4/GWBASIC source to QuickBASIC
BASWIN08.ZIP	188909	910128	QB4.x window/menu/mouse routines with source
BASWZ15.ZIP	288707	910416	Library of ASM/BASIC routines for QBASIC 4.5
GETCPU.ZIP	2335	911021	QBASIC routine to determine CPU type w/ASM src
QBACV3.ARC	22996	890402	QuickBasic ARC/PAK/ZIP file dir util w/src
QBSR15.ZIP	306600	911104	QB screen builder and screen routines w/source
SMALLFNT.ZIP	5341	900427	Quick Basic source to create 106x33 CGA font
TOPMENU2.ARC	84097	891215	Quick Basic 4.x source: nice pulldown menu
VIDOBASIC.ZIP	120101	910103	MASM text video routines library for QBASIC

RBBS_PC

173_ASM.ARC	85013	900220	RBBS-PC v17.3 source code - Assembler + OBJs
173_BAS.ARC	378549	900220	RBBS-PC v17.3 source code - Basic code
CALLER.ZIP	3246	890613	RBBS Caller Upload, Download, Log Time Stats
LITCALL.ZIP	11251	850910	11st rbbs callers file (source in C)
USRCHK17.ZIP	72408	910122	User call back verification util. for RBBS-PC

SCREEN

BL_ASM	564	850626	Blank the screen to REAL top line. MASM src
BLANK.ARC	13159	890530	Screen blanking utility with MASM source
BORDER.ZIP	1660	890613	Set Border Color within Turbo Pascal Programs.
CLEAR.ARC	867	830825	Fast clear screen function, Assembler code
DCRNSAV.ARC	4438	900121	Screen saver, shifts screen. With ASM source
HSRPTSC.ZIP	18369	901012	Herc-Epson graphics screen printer (w/ASM)
GNTUPC.ZIP	22743	911206	GnuTermcap for MSC-src & small model lib
HERC_EPS.PAS	8346	880118	Print Hercules screen on Epson printer. TP src
HRPTSC.ARC	7592	880121	Herc-Epson graphics screen printer (w/ASM)
MOWED.ASM	717	840124	Set video modes, assembly language source
PRVSCR1.ZIP	6684	900713	PrtsC text & graphics (thru EGA), w/C,ASM src
SAVSCR.ASM	35331	860211	Blank screen after no activity
SAVSCR.DOC	3980	860211	Doc for SAVSCR.ASM
SCREEN.ASM	14856	831218	Set screen color text/background/mode
SCRNCOLR.ASM	9262	860716	Set screen color text/bkgnnd/mode from cmd line
SCRNCOLR.DOC	1026	840102	Documentation for SCRNCOLR.ASM
SCROLLK.ARC	7903	830712	TSR: Scroll lock key control for video display
SWAP.ASM	949	821024	Swap between B&W and color
VIDMODE.ZIP	4412	900829	Pgms to set 25/43/50 line video modes, w/src
WAS062.ZIP	52459	910713	Scroll screen backwards - uses EMM, w/ASM src
WB10S.ASM	21445	840617	TSR, add window functions to BIOS
WB10S.DOC	787	840612	Doc for WB10S.ASM
XFLASH1.ARC	12907	880829	Elim. CGA scroll flash w/o snow, w/ASM source
ZANSI_12.ARC	28933	880829	ANSI.SYS replacement, faster than NANSI, w/ASM

SNOBOL

SNOBOL_A.ZIP	285050	891031	Snobol 4+ language and utilities, 1 of 3
SNOBOL_B.ZIP	159198	891031	Snobol 4+ language and utilities, 2 of 3
SNOBOL_C.ZIP	284555	891031	Snobol 4+ language and utilities, 3 of 3

SOUND

SAMPLE20.ZIP	267019	901204	Digital sound playback/record (with TP5.0 src)
SOUNDEX.ZIP	10816	800101	Soundex procedure written in C w/ source
SOUNDEX1.ZIP	1400	870222	Soundex algorithm in C
TONE.ZIP	851	880809	Lattice C tone generator
TONI.ZIP	880	880811	make sounds in asm f/PCTech

SPRINT

SPFNTS4.ARC	22459	890704	WYSIWYG Sprint screen fonts, source code
STARTER	7002	860910	Source for DELBRIIA.EXE (see DELBRIIA.BUG)
DELBRIIA.C	5349	910418	C source for udeocoder
UUECODE.C	4212	910306	C source for uveocoder
XXINSTALL.BAT	87461	900316	BAT file to bootstrap XXENCODE/XXDECODE, w/Csrc

SURFMOVL

SURFM203.ARC	131327	880226	3D surface modelling 3 of 3 (TP4 source)
--------------	--------	--------	--

SYSUTL

16550S.ZIP	5943	900411	Toggle the NS16550A FIFO buffer, w/TPas source
16550S13.ZIP	10166	910329	Toggle the NS16550A FIFO buffer, w/TPas source
ANSI.ARC	2655	871206	Test ANSI driver functions
BIOS_ASM.ARC	64348	880118	Public domain generic PC BIOS (MASM source)
CHECK87.ASM	1540	850330	Check for 8087 processor
CHGPATH.ARC	6570	880629	Change path (add or delete), w/C source
CPUID.ASM	18840	860509	Identify your system's CPU
CTRLALT.ARC	68989	871231	Programmer's TSR sidekick w/asm source
DIAGERR.ARC	22210	880105	Memory res MSDOS error codes w/ASCII source
DTIME.ARC	11508	890519	Display time and date, with Turbo C source
ENVRPT.ARC	6516	850128	Examine env., give return code for BAT inquiry
ENVSEARC.C	1725	850812	Search environment for text
EXEC.DOC	1061	840512	Doc for EXEC 1/2/3.ASM
EXEC1.ASM	7160	850319	Callable DOS EXEC routines
EXEC2.ASM	5539	840512	Callable DOS EXEC routines
EXEC3.ASM	5614	840512	Callable DOS EXEC routines
EXECTEST.ASM	1785	840920	Test EXEC 1/2/3.ASM
HISTORY.ARC	17887	880321	Command history mechanism. COM & source
HOT_36.ARC	4336	890101	Replace Ctrl+Alt+Del (w/ASM src)
IDT.ARC	59430	890511	Internal DOS table viewer, w/C source
IRQR.ZIP	5221	910719	Report status of IRQ channels, w/TP4 & MS C src
MEMRES41.ARC	23381	880829	Load/release TSRS - with ASM source
MEMSIZ.ASM	2467	851001	Report free RAM
NEWELL.ASM	1558	850518	Replace DOS bell with shorter one
NS_16550.ZIP	39783	911104	Test 16450/16550/16550a serial chips w/source
OWMSWITCH.ARC	5739	891002	Change DOS switch character, with 'C' source
PARINT.ASM	6899	860119	Trap memory parity errors
PORT11.ZIP	28998	910516	I/O port debugging tool, with C source code
PROCTYPE.ARC	4873	881030	ASM routines to determine processor type
REDTERR.ARC	6295	880306	Redirect STDERR to a file, with src
REBOOT.ASM	1954	850320	Warm boot
REBOOT_18	840518	840518	Doc for REBOOT.ASM
REBOOT.DOC	319	840518	Doc for REBOOT.ASM
REBOOT_SEQ	65	840518	Doc for REBOOT.ASM
RTIME.ARC	12007	870727	Time a program's execution w/C SRC
SETENVJW.ZIP	21771	910708	Set environment variables, v1.6, w/ASM source
SETTIME.ASM	2015	860613	Set time via DOS call
SHOWDEVS.ARC	10143	890321	Show all active device drivers, w/'C' source

SLIKMODE.ZIP	8973	910416	New MODE: Set baudrate to any rate w/C src
SWITCHAR.ASM	5941	850708	DOS call to find '/'
SWITCHAR.DOC	1730	850712	Doc for SWITCHAR.ASM
TESS.A.ARC	22528	890628	TesSeReact TSR package for Assembler (MASM)
TIMER.ASM	13512	891217	Explains problems w/BIOS timer tick & fixes
TOAD8253.ARC	8192	880901	Tweak RAM refresh rate [ASM]
TOADDEXEC.ARC	5522	880630	EXEC programs [ASM]
TSRCOM31.ZIP	72569	911111	TSR memory management utils (MARK/RELEASE)
TSRSRC31.ZIP	68947	911114	Source code for TSRCOM v3.1 pkg. (TASM, TP6.0)
TSRRWK32.ZIP	79415	900809	Mark/Release TSR management pgms, with ASM src
U.ASM	17367	890103	Remove the TSR program, DOSEDIT, from memory
U.DOC	395	890104	Two line description of U.ASM

TEX

DVITG9.ARC	128512	890110	TeX DVI previewer source code (MSC)
WEBTP55.ARC	371096	891206	WEB system of structured doc for TP5.5, w/src

TIFF

DVITF1.ZIP	15932	900416	TIFF reader:PACKBITS compr/uncomp C src, ALDUS
TACS40.ZIP	126480	900425	TIFF Access/Create System - C source, TIFF 4.0
TFTOOL.ZIP	65975	900416	TIFF toolkit developer package C source, ALDUS

TROJAN_P

CHECKMEM.C	510	890609	Check computer memory for presence of viruses
CRCS013.ZIP	35367	910807	Self-scanning executable w/TP & MSC/TC source
FILETEST.ARC	64748	880314	Check for corrupted files. w/Pascal source
NOBRAIN.C	18750	890609	Source code: rid computer of 'Brain' virus
VKILL13.ZIP	15391	901225	Find and remove Jerusalem virus, with C source

TURBO_C

BLIT.ARC	21977	880323	PC Sprite graphics support 'C' source
BNSL1B.ZIP	41468	870914	Object code library manager for Turbo C
CXL_TC.ZIP	39351	880726	CXL 4.0 library for Turbo C
DATE.ZIP	8967	880808	turbo source+com easily change date/time
FFT.ARC	5492	880807	Fast Fourier Transform pgm w/C source
JOY.C	4421	880327	'C' language routines to deal with joy stick
LV_TC.ZIP	25938	910412	PC printer port bi-directional driver - TC src
MAPPER17.C	13949	881206	TCv1.5 src: Draws 3 types of map projections
MOVE.C	6605	890526	Fast file mover; uses rename funct. TC src
SETARGV.C	4991	880309	Expand wildcard arguments
TABALT1.ZIP	11549	890628	Improved TABALT, Changes Tabs in Turbo C Source
TC_BUG.ZIP	6719	870906	Shows a Bug in Turbo 'C' _Open Function.
TC_PAT03.ZIP	12014	871205	Six Patches to be Applied to Turbo C.
TC_TS10.ZIP	22978	910628	Turbo C source code examples of TSR programs
TCDBGUG.ZIP	75989	880724	Turbo C debugger version 1.0
TCDEV.ZIP	65256	871119	version 0.06 of turbo-c debugger
TCLIB.ZIP	4277	870613	Device Driver written in TURBO C W SOURC
TCMM11.ZIP	29552	870701	Lib utility w/source for Turbo C
TCPAT1.ZIP	1394	870624	Communications Library for TURBO C V1.1
TCPATCH.ZIP	1637	870527	Turbo C patches 1- 3 use TC_PATCH.ZIP
TCPATCHS.ZIP	16029	870715	Turbo C bug patch (1-3)
TCRES2.ZIP	26887	880502	Official Borland Turbo C patches 7 13 87
TCRND1.ARC	16454	890310	Make Turbo C memory resident program
TCUTOR1.ZIP	133017	871029	Updated doc and batch file to use TASM 1.0
TCUTOR2.ZIP	113765	871029	Turbo C tutor 2 of 2
TCUNI_A.ZIP	146365	890628	Turbo C Unicorn Library of 200 Functions 1 of 2
TCUNI_B.ZIP	94174	890628	Turbo C Unicorn Library of 200 Functions 2 of 2
TCWIND10.ZIP	75234	870722	Window Library for Turbo-c
TICKTOCK.C	1822	891014	Demo of IBM PC's high resolution clock, TCv2.0
TIMERTST.C	4084	890425	8253 timer test program for Turbo C 1.5/2.0
TROBCSRC.ARC	112555	890114	Coronado's Turbo C tutor v2.0, (1 of 2)
TURBCPAT.ZIP	11768	870625	Borland's patches to Turbo C (to date)
TURBOC.ZIP	4202	870221	Borlands new Turbo-C press release,01/26
TURBODGI.ZIP	15766	870221	Utility to Display Turbo *.DCI Files.
UCTURBO.ZIP	240517	880410	Library for Turbo C
WINDOW10.ZIP	101679	871205	Fast Windows for Turbo C

TURBOBAS

TBSUPLOW.ZIP	32307	901231	Turbo Basic source upper/lower case conversion
--------------	-------	--------	--

TURBOPAS

30TURUTL.ZIP	89389	890613	Execute DOS Commands from a Turbo Pascal.
3DGRAPH.ZIP	13740	890613	Graph Routines, Obj Rotate/Translate/Scale/Wndo
ACCESS.ZIP	5684	890613	Allow Access to More Than 64K Recs in Data File
ACTSK.ZIP	963	890613	Activate Side Kick from a Turbo Pascal Program.
ALMANAC.ZIP	27913	890612	Turbo Pascal Version 4.0 Almanac Program.
ALT_KEY.ZIP	13436	890613	Allow BATCh File Inp for Macros w/o Keybrd Input
AMOUSE55.ZIP	8192	900712	TurboPas5.5 Microsoft mouse unit (source only)
ANSIDRV.R	1452	890613	Sample TP Program to Use and Test ANSI driver.
APCLU.ZIP	12010	890613	Library Utility, LU, for NEC-APC with Source.
APPEND.ZIP	1679	890613	Init Variab in .COM Files by Appending Data.
APPPOINT1.ZIP	4365	890613	Displays Side Kick Appointments.
ARGLIB.ZIP	6268	890613	Get Arguments from the Command Line. UNIX-Like.
ARGLIST2.ZIP	1615	890613	Argc/Argv like in C.
ASC_WS.ZIP	1642	890613	ASCII Text to Wordstar Conversion Program.
ASYNCR.ZIP	5935	890613	Asynchronous Input/Output Routines
ASYNCR4.ZIP	5074	890612	Interrupt-Driven Communication I/O Routines.
ASYNCR_PC.ZIP	5228	890613	Asynchronous Input/Output Routines.
ASYNCR.ZIP	3119	890612	Asynch I/O Routines.
ATTRIB.ZIP	1916	890613	Change File Attributes or Flags.
BANNER.ZIP	1189	890613	Create a Banner to the Screen or a File.
BANNERC.ZIP	5104	890613	Make Banner to Screen/File.
BBSCHAT.ZIP	21735	890613	Online Eliza - Good, with TP Source for BBS.
BIGSORT.ZIP	7334	890613	Sort Large Files by Key. OK for Pipes.
BIGTUR.ZIP	9746	890613	TP Code for Large Code Model, Greater than 64K.
BLIST.ZIP	8572	890613	Count Comments, BEGIN-END Nesting in TP Programs
BOOSTDOC.ZIP	58081	890613	Documentation Demonstration w/ TP Source
BORDER.ZIP	1660	890613	Set Border Color within Turbo Pascal Programs.
BOX.ZIP	1422	890613	Draw Box Routine.
BOXMENU.ZIP	31197	890613	Routines to Draw Menus and to Select Items.
BOYMO4.ZIP	7483	890612	Boyer-Moore Search Routines.
BROWSE2.ZIP	16812	890613	Browse Two Files at Same Time.
BTREE4.ZIP	15841	890612	Pascal 4.0 BTREE Similar to Database Toolbox
CALC.ZIP	5567	890613	Calculator Program.
CHMOD.ZIP	4405	890613	Changes File Attributes or Flags.
CHNLOD.ZIP	11109	890613	Keeps Runtime Rtnes in Mem & Chains to TP Pgms.
CISBIN.ZIP	3110	890613	Cnvrts CompuServe ASCII Bin File to PCDOS & Back
CLA.ZIP	2591	890613	Command Line Argument Parsing Like VAX/VMS.
CLOCK.ZIP	1873	890613	A Simple Clock for Turbo Pascal Users.
CLRLINE.ZIP	883	890613	Clear any Line from One to Twenty-Five.
COLOR.ZIP	5168	880809	Lets user install color in your TP4 programs
COLORDEM.ZIP	5000	890613	Pset, Linedraw, Boxdraw, Screen, Color
COMMAND.ZIP	1102	890613	Patch V2.0 Turbo Leaves COMMAND.COM Resident
COMMCALL.ZIP	3323	890613	Communications Routines for COM1 Port.
COMP.ZIP	4412	890613	File Comparator that Searches for Same Lines.
COMPARE.ZIP	10690	890613	Line Compare of Two Files
COMPLEX.ZIP	1897	890613	Complex Num Routines from Dr. Dobbs Sept 1984
COMTOT.ZIP	2203	890613	Converts Binary Files into Inline Turbo Code.
CONCUR4.ZIP	17511	890612	Concurrent Programming Tools, Turbo Pascal 4.0.
COPIWYC6.ZIP	4205	890613	Copies HD Files onto Floppy, Prompt When Full.
CPMGR.ZIP	4206	890613	Critical Path Method Project Scheduler.
CPREADLN.ZIP	541	890612	Help CONCUR4 to READLN Without Holding
CRC.ZIP	1519	890613	Computes the CRC Using Turbo Pascal Inline Code.
CRC16.ZIP	1867	890613	Computes Cyclic Redundance Check for 16 Bits.
CRCASM.CRC	15100	900102	ASM CRC calculation code w/TurboPas interfaces
CRFONTS.ZIP	6508	890613	Create Font Tables Interactively.
CR7.ZIP	4215	890613	Demos Terminal Interface.
CRYPTO2.ZIP	12252	890613	Solves "Aristocrat" Encryption Methods.
CTRLC.ZIP	2106	890613	Trapping Ctrl-C in TP with Side Kick Resident.
CURSOR.ZIP	1979	890613	Position Cursor Using BIOS VIDEO_IO.

DANUTILS.ZIP	33375	890623	Model Programs.
DATETI.ZIP	1548	890623	TP Routines to Access the Date and Time.
DATETIME.ZIP	1284	890623	Gets the Date and Time from MS-DOS/PC-DOS.
DBASE2.ZIP	1743	890623	Convert dBase II Files to TP Source Code.
DBF3.ZIP	12183	890623	Read/Write dBase Files w/ TP Source Code.
DBFILES.ZIP	6886	890623	Compresses and Expands dBase II Files.
DBG23.ZIP	31385	890623	Turbo Pascal Code Debugger with Documentation.
DC.ZIP	11018	890623	CAD Program Draws and Prints Electric Circuits.
DEARC.ZIP	8001	890623	Version 1.01A, 10/19/85, De-ARCHIVE Program.
DEARCS.ZIP	28377	890623	De-ARCHIVE Utility with TP Source Code.
DECARCH.ZIP	9688	890612	ARCHIVE Extraction Routine with Buffered Input.
DECIP7.ZIP	5079	890623	Deciphers File in Place with Commented Source.
DETAB.ZIP	1287	890623	Replaces Tabs with Blanks. See EXPTABS.ZIP.
DETOKEENZ.ZIP	4561	890623	Convrt (de)tokenized IBM BASICA to ASCII BAS File.
DIFF.ZIP	3701	880721	Turbo Pascal V4.0 File Differences Program.
DIRECTRY.ZIP	10741	890623	Demo Pgm in TP to Display Files in Directories.
DIRLIST.ZIP	2822	890623	Disp All Files Incl System, Hidden, Read Only.
DIRPRINT.ZIP	1605	890623	CP/M DIR Lister, Documented, Uses BDOS Calls.
DIRSRCH.ZIP	1775	890623	Searches DIR for a File, Displays Attributes.
DIRTE.ZIP	5504	890623	Prints Cmt Lines of PAS Include Files.
DISKIO.ZIP	1913	890623	Disks Reads and Writes via BIOS Interrupts.
DISKOVER.ZIP	63455	890623	V1.17 Makes Disk DIR List Inc ARC Contents, Doc.
DISKSCNTR.ZIP	2827	890623	Dumps Disk Sectors.
DISKVR20.ZIP	76120	890623	V2.0 Prints Disc Cover Sheets w/ Doc, PAS Src.
DMA.ZIP	5886	890623	Replace TP Std Output by Pokes to Hardware Buff.
DOORPROC.ZIP	2726	890623	BBS Front Door INKEY, KEYCHAR, Serial In.
DOSBEL.ZIP	906	890623	Warning and Siren Sounds for IBM PC's or Clones.
DOSIO.ZIP	1780	890623	Disk File Block Read/Write, Faster than TP's I/O.
DPERF.ZIP	10447	890623	Disk Space Performance.
DUMBTERM.ZIP	6463	890623	Example Inline Code For Interrupt & Comm Port.
DWS.ZIP	1476	890623	Convrt WS Fmt Files to Normal ASCII, Req CLA.ZIP.
EASYBOX.ZIP	1117	890624	Draws Box For Menus, Etc.
EBMENU.ZIP	8114	890624	Routine to Present a Spreadsheet-Like Menu.
EDISK.ZIP	12202	890624	IBM/CPM Disk Sector Pgm with Docs & TP Source.
EDIT.ZIP	2411	890624	Input Data from User Using Wordstar-Like Cmds.
EMLINK.ZIP	6671	890624	Expanded Memory Routines in Turbo Pascal.
ENCIPH.ZIP	5112	890624	Enciphers a File Using Fractionating Algorithm.
ENCIPHER.ZIP	2290	890624	Encrypt/Decrypt, Transpo Chars from Key File.
ENG_PAS.ZIP	75776	890624	Engineering Routines in Turbo Pascal.
ENV12.ZIP	5450	890612	Get Environment Values.
ENVIA.ZIP	5210	890612	DOS Environment Access w/ Bug Fixed, TP4.
EPB13.ZIP	12484	900719	Ed's Pascal Beautifier, w/Turbo Pascal source.
EPSONMX.ZIP	1096	890624	Sets Epson MX Printer into Various Modes.
EQUIP.ZIP	13415	890624	Computer Stats with Turbo Pascal Source Code.
ERASCHK2.ZIP	12927	890624	Measure Effective Disk Erase for Sensitive Data.
ERRORPRF.ZIP	13478	890624	Detects When Critical Errors Occur TP3.
EXAMIN.ZIP	4322	890624	Examines Symbol Table for TP Vers. 3.00B.
EXDATE.ZIP	4699	890612	Julian/Gregorian Date Routines for TP4.
EXCEMENU.ZIP	34689	890624	A Great Custom-Built Menu System with Source.
EXP.ZIP	599	890624	Exponent Function Using the Taylor Series.
EXPTABS.ZIP	1923	890624	Expand Tabs. See DETAB.ZIP.
EXTEND.ZIP	12487	890612	TP4 Open Files <= 255 MS-DOS.
EXTKEY.ZIP	4656	890624	Turbo Pascal Unit for Accessing Extended Keys.
EXTTURBO.ZIP	10195	890624	Library: Integers, Strings, Link Lists, I/O.
FATSDA.ZIP	6093	890612	Fast Assembler Date Difference Routines for TP4.
FAT.ZIP	2218	890624	FAT Lister, Gives Unformatted Data.
FATINFO.ZIP	898	890624	File Allocation Table Lister. Short, but Works.
FCOPY4.ZIP	2709	890612	Filecopy Unit for Turbo Pascal 4.0.
FDUMP.ZIP	1938	890624	File HEX/ASCII Dump Using Windows.
FFT_PROC.ZIP	5553	890624	Fast Fourier Transforms.
FIELDIN.ZIP	2038	890624	Pst Cursor at Fld & Get Input.
FILEFND2.ZIP	3324	890624	Finds Data Files from Any Path Directory.
FILES.ZIP	8075	890624	Directory List.
FILEUT20.ZIP	7692	890624	Add LFS, Convert from WS 8 to 7 Bits, & More.
FLISIZ.ZIP	1504	890624	TP V3.00B for MS-DOS Fixes Filesize Function.
FIXFAN.ZIP	1976	890612	Fixes Fansi-Console When Running TP 4.0.
FLASH.ZIP	13652	890624	Shows Video Page or Saves Screen.
FLASHQ.ZIP	17308	890624	Shows type of Equip: Video Mode, RAM, Num Disks.
FLUSH.ZIP	7869	890624	Procs to Flush Data Files w/ Database Toolbox.
FRACAL.ZIP	5073	890624	Fractal Generator Plots Complex Nums w/ Lib.
FRACAL1.ZIP	1692	890624	Fractal Generator.
FRACALS.ZIP	884	890624	Another Fractal Generator.
FRAMES.ZIP	3736	890624	Runtime Stack Structures, Pass Procs as Params.
FST_SCR2.ZIP	7873	890624	Produce & Maintain Screen Image Files for TP.
FX_SET.ZIP	3286	890624	Full Featured Epson FX Printer Setup Utility.
GOEL.ZIP	6417	890624	Global Delete an Occurrence of a File in All DIR.
GENTINDEX.ZIP	5299	890624	PAS MT + Gen Index of AQ & AW Words in WS Text.
GETCHAR.ZIP	1480	890624	Bypasses Turbo's Conv of ESC Etc w/ Docs.
GETFLD.ZIP	15795	890612	Screen Field Input Routine in Turbo Pascal 4.0.
GETXP.ZIP	1438	890624	Gets Exprssn/String from Kbd w/ Bkspase Hndng.
GLOB.ZIP	1040	890624	Plot Globular Star Clusters as Dots on Screen.
GOMOKU.ZIP	7209	890624	Gomoku 5 in a Row Tic-Tac-Toe Like Game.
GRAFX.YIP	388	890624	Sets Graphics Cursor Position Using Inline Code.
GRAPHX.ZIP	7514	890624	Updates, Fixes to Graphix V1.03A Making V1.05A.
GRFMISC.ZIP	7443	890624	Miscellaneous Graphics Routines, Rotating Cube.
HANOI.ZIP	1158	890624	Tower of Hanoi, Ctr Graphics, Exmpl of Recursion.
HASH.ZIP	5904	890624	Phone Book Prgm Uses Hash Code to Store Names.
HEAP2SCR.ZIP	23059	890624	Copy Image in Heap to Video, Not Yet Complete.
HEAPTEST.ZIP	1127	890624	Demos TP Ver.2.0 Bug in Heap Management.
HELP.ZIP	11540	890624	Memory Resident Help Program, TP Syntax.
HEXCALC.ZIP	2077	890624	Hexadecimal to Decimal Calculator.
HOTM60.ZIP	5798	911021	Reassign the TP 6.0 IDE hot keys, w/TP 6.0 src.
HOTSTAT.ZIP	47373	890624	Statistical Spreadsheet.
ID_REF.ZIP	11353	890624	Cross Reference Turbo Pascal Program.
INDENT.ZIP	11193	890624	Pretty Printer Src Formatter Indents Code.
INDEX.ZIP	3447	880721	MSDOS and CPM versions WS file, word, phase index.
INLNE.ZIP	48080	890624	Assembler Produces Turbo Inline Src Code w/ Doc.
INLNE2.ZIP	20511	890624	Assembler Produces Turbo Inline Src Code w/ Doc.
INPUT2.ZIP	5091	890624	Formatted Data Entry Program.
INTX210.ZIP	50559	890624	V2.1 Menu Driven Patches to TP V3.00B & V3.01A.
INT25.ZIP	3407	890624	Call MS-DOS INT 25/26 Disk Read/Write w/ Docs.
INTRR.ZIP	8573	890624	Interrupt Routines Used with Comm Port.
INTRDMD.ZIP	85368	890624	Resident Interrupt Program in TP w/ Docs.
INTRFC62.ZIP	56398	910618	Turbo Pascal 6.0 TPU file dumper, with source.
INVTORY.ZIP	40356	890624	Inventory Package, Requires Database Toolbox.
IORESULT.ZIP	698	890624	Actual IORESULT Code for Turbo Pascal.
IPM.ZIP	21082	890624	Project Management/PERT Program.
JDATE.ZIP	2698	890624	Julian Day Algorithms.
JDTTPROG.ZIP	3610	890624	Get Time and Date, Phone Sound Routines.
JTLIB.ZIP	11774	890624	TP Pretty Formatter w/ Functions Library.
JULCAL.ZIP	2132	890612	Julian/Gregorian Date Routines for TP4.
JULIAN.ZIP	1816	890624	Julian Day Algorithms.
JUSTIFY.ZIP	1352	890624	Word Justify Program for Text Files.
KBDUIF.ZIP	458	890624	Small Routine Tests Keyboard Buffer.
KPGRAF.ZIP	1965	890624	Kaypro Scrn Graphics Lib, Circles, Others.
LASTCOM.ZIP	18867	890624	Recall the Last Ten Commands.
LASTCOM2.ZIP	9209	890624	Mem Res Pgm Saves the Last Ten DOS Cmds Typed.
LIFE4.ZIP	6637	890624	Game of Life Using Db1 Linked Lists w/Hdr Cells.
LIST.ZIP	1832	890624	Small Program to List File on Printer with Hdr.
LIST01.ZIP	4593	890624	Prints up to 20 Source Files on Printer w/ Hdr.
LISTWLD.ZIP	2881	890624	Directory List of Files from a Wildcard Command.
LONG.ZIP	8154	890624	Long Integer Math Routines w/ Inline Code.
LOWCASE.ZIP	1628	890624	Convert Lower Case to Upper Case, ReqS GON.ZIP.
LOWDSK.ZIP	4608	890624	Low Level Disk Sector Read and Info Program.
LS.ZIP	2824	890624	List Files on Printer, Accepts Wildcard Cmds.
LSPELL93.ZIP	29641	890624	Speller Based on Turbo Lightning, Builds Own Dic.
LTCOMM1.ZIP	41220	890612	LibComm-Turbo Pascal Toolbox for COM1 -COM4.
LUSG_2.ZIP	8327	890624	LiBary Maintenance Program.
MANDEL.ZIP	37181	890624	A Utility to Generate Mandel's Fractal Geometry.
MAPMEM.ZIP	3459	890624	Displays a Map of Memory Resident Programs.
MATCHCH.ZIP	1998	890624	Matches Characters in String.

MAZE.ZIP	3518	890624	Creates and Solves Mazes on the Tube.
MEMLOOK.ZIP	4979	890624	Dump Memory to the Screen.
MEMWRITE.ZIP	1153	890624	Sample Prgm to Write Directly to Video Memory.
MENU456.ZIP	6409	890624	Demo on How to Select Menu Items by Pointing.
MISC.ZIP	1299	880811	Change cursor on terminal.
MODIFY.PZ	2126	890612	Install Procedure for Turbo Pascal 4.0 Programs.
MOUSE.TP.ZIP	2886	890624	Turbo Pascal Routines for a Microsoft Mouse.
MOV2R2.ZIP	1246	890624	Some Turbo Pascal Source Code.
MRLGFL.ZIP	2063	890624	Sort and Merge Algorithms.
MULTIWPAP.ZIP	11044	890624	Display and Data Entry Screens w/ GCA Brd & Docs
MURUTL1.ZIP	226506	890624	Utilities by Murphy.
MURUTL4.ZIP	121942	890612	Harry Murphy's Utils, 22-Nov-88 Edition Ver 4.
NAMEPATH.ZIP	2942	890630	Parse Filename Paths.
NDR.ZIP	5490	890630	CP/M Directory List Program.
NEWEXEC.ZIP	4558	890630	V1.2 Run Other Prgms from Inside a TP Program.
NEWPC.TP.ZIP	51895	890630	Disk, Screen Entry, and LiBRary Routines.
NKTOOLS.ZIP	50076	910226	Source to some useful TurboPas 4/5/5.5/6 Units
NOHARD.ZIP	1848	890630	Move File from Directory into Another.
NOHARD.ZIP	12782	890630	Prevents Hard Disk Writes Until Warm Boot.
NONLIN.ZIP	11454	890630	Non-Linear Curve Fitting Routine w/ Docs.
NONMOW.ZIP	5483	890630	Eliminate Snow on CGA with Direct Screen Writes.
NOTEPA04.ZIP	6505	890630	Version 4 Resident Note Pad.
OAS500C.ZIP	146067	910530	TP4/5/5.5 Source Applications Generator, 1of3
OAS500PRO.ZIP	184959	910530	TP4/5/5.5 Source Applications Generator, 2of3
OAS50545.ZIP	56712	910530	TP4/5/5.5 Source Applications Generator, 3of3
OASDOC41.ZIP	138560	890612	Open Architecture Screen Interface Sys 4.1 1/3.
OASPRO41.ZIP	112852	890612	Open Architecture Screen Interface Sys 4.1 2/3.
OAS5AM41.ZIP	172439	890612	Open Architecture Screen Interface Sys 4.1 3/3.
OTHELLO.ZIP	4701	890630	Othello Game Uses IBM PC Graphics.
P1351.ZIP	3483	890630	Initializes Toshiba Printer.
P4MAT202.ZIP	61775	890630	V2.02 Pretty Formatter for Turbo Source Code
PAGER.ZIP	2020	890630	Spools Paginated Files from Given Page to Printer
PAGWRITE.ZIP	10158	890630	Handle Keyboard I/O by Redefining WRITE
PARK.ZIP	613	880809	Park that hard disk (source)
PAS_ENG.ZIP	60301	891030	Common use PASCAL for engineers
PASCAL.ZIP	107057	890630	Edit, Browse, Mapmem, Modem Prgms w/ Docs.
PASCLLESS.ZIP	45344	890630	Part 1 of a Turbo Pascal Tutor
PASFORMA.ZIP	18430	890630	Turbo Pascal Source Code Formatter - Excellent.
PASLIB.ARC	21504	890525	T.Pascal Source Library Routines
PASMAT.ZIP	33608	890630	V1 Pascal Formatter with In-Source Directives.
PASMSG.ZIP	3636	901127	Allows TPC(v6) or TPCX to run under TC++ IDE
PASOCT.ZIP	2952	890630	Octal Number Turbo Pascal Routines
PASTUTOR.ZIP	146198	890630	Pascal Tutor - Requires Turbo Pascal
PASUTL1.ZIP	3237	890630	File Append, Chk Key Locks & Display-Type Rtns.
PASUTL11.ZIP	61466	890630	File Copy, Dump, Printer Set, Xref, UNSQ, More.
PATMS.ZIP	3403	890630	Directory Routines
PBTURBO.ZIP	14291	890630	Pretty Printer Pascal Code Formatter by Crider.
PC_CHECK.ZIP	27181	890630	Checkbook & Savings Record Keeper w/ Docs
PC_DISK.ZIP	125174	890630	Disk Cat w/Sort, Accepts Filenames from AXQ/SDIR.
PCSELF.M.ZIP	2202	910124	TP source: Allow autoconfig prgs to be LEXED
PCLIFE.ZIP	33198	890612	Very Fast Conway's Game of Life in TP4.
PCRF.ZIP	6857	890630	Turbo Pascal Cross-Referencer Program.
PCSTATUS.ZIP	20254	890630	Show Mem, Equip, Video, Disk, & Envir Params.
PFIELD.ZIP	2970	890630	Routines for Protected Field Forms Input.
PGLAIDE.ZIP	55502	890630	Memory Resident Aid for Writing TP Code w/ Docs.
PH.ZIP	3134	890630	Filter - Prints Heirarchy Listing from PCRF.ZIP.
PHONE.ZIP	4975	890630	Maintains Telephone Records and Dials Telephone.
PIBASY.ZIP	41057	890612	Interrupt-Driven Comm I/O Unit from Phil Burns.
P1BCAT14.ZIP	39970	890612	Disk Catalog Program in TP 4.0.
P1BLZW.ZIP	10420	890612	LZW Compression in TP 4.0. Not ARCHIVE.
PLOTLIB.ZIP	11195	890612	Plotter Library in TP4, Drive a Tektronics 4662.
PP50.ARC	22909	891128	Pascal pretty printer/reformatter, w/TP5.0 src.
PULLSX.ZIP	149695	890612	Multi-Level Pull-Down Menus Version 5.Xa TP
PULLTP4.ZIP	20035	890612	Pull-Down and Pop-Up Menus in TP 4.0.
QWIK41A.ARC	53248	880523	Quick screen utilities v4.1a for TPas4 (w/src)
QWIK42.ARC	94763	881007	Quick screen utilities v4.2 for TPas4 (w/src)
QWIK42B.ZIP	66999	890612	LeMay's TP4 Qwik Screen Utils Reference Guide.
QWIK55.ARC	99885	891227	Quick screen utilities v5.5 for TP5.5 (w/src)
QWIK5X.ZIP	18753	890612	Qwik Screen Write Version 5.x for TP 5.0
SCRABBLE.ZIP	25299	890612	The Game of Scrabble Written in TP 4.0.
SEARCH4.ZIP	5805	890612	Inline String Search Unit for Turbo Pascal 4.0.
SFMSRC.ARC	93440	871124	Full Turbo Pascal source for SFM.ARC
SORTKIT1.ZIP	12049	911004	Sort objects sources for Turbo Pascal 6
SPRITES.ZIP	43010	910616	Fast sprites engine for TP 6.0, w/ASM source
STRNGTP4.ZIP	26165	890612	String Functions and Procedures for TP4
TACCESS.ZIP	1972	890612	Index File Fix for TP4 Toolbox Routine.
TDAPAT.ZIP	7065	890612	Corrected Patch for TDebug Use Only.
TLIST23.ZIP	30011	871224	Turbo-Pascal Source Code lister w/source
TOAD_IAC.PAS	5667	881217	Demo using InterApplication Comm area (TP src)
TOTSRC.ZIP	139928	910417	TechnoJock's Object Toolkit srccs (TP5.5+), 4/4
TP002.ZIP	222723	891030	Turbo Pascal programming tools, 2 of 3
TP003.ZIP	244151	891030	Turbo Pascal programming tools, 3 of 3
TP3GUIDE.ZIP	134456	890630	Norton Guide for TP3, Requires NG.ZIP.
TP40FX.ZIP	4495	890612	Turbo Pascal V.4/5 DataBase Toolbox Fixes
TP4I023.ZIP	55830	890612	Control Console Data Entry in TP 4.0
TP_TSR.ARC	35712	890429	Sample TSR pgm in TurboPas 4/5, with src/exe
TPAS2MSC.ZIP	149337	880120	Convert Turbo Pascal to MS C
TPAS_C13.ZIP	36124	890630	Convert TP Program to 'C'. Version 1.3.
TPASBOOK.ZIP	105925	890630	Programs from "The Complete Turbo Pascal" Book
TPATCH.ZIP	2472	890630	Patches TP *.COM Files from Data in TPATCH.DAT.
TPBASIC.ZIP	3513	890630	Change Bit, Beep, Ctr Line, String Rtns.
TPCL4.ZIP	38446	870619	Turbo Pascal to C translator
TPCV.ZIP	12267	890612	Use CodeView Debugger with Turbo Pascal 4.0
TPF.ZIP	65179	890630	Turbo Pascal Code Formatter with Documentation.
TPFAST30.ZIP	65625	901014	Fast ASM sources for Turbo Pascal functions
TPGET.ZIP	6247	890630	Get Ch/Input/Kbd/Date & Time, DIR Search Rtns.
TPINLINE.ZIP	30968	890630	Creates Inline Code for Turbo Pascal.
TPINST1.ZIP	2595	890630	Chg Start-Up Display Mode of Compiled TP Progs.
TPJOYSTK.ZIP	4896	910326	Turbo Pascal v4.x/5.x joystick TPU source
TPMSDOS.ZIP	2976	890630	Turbo Pascal Library Routines Get Date/Drive
TPONG1.ZIP	4569	890630	Ping Pong Game Written in Turbo Pascal.
TPPOP16.ARC	41941	890806	TPUS and ASM for popup and TSR programs
TPPOPUPS.ARC	21095	880829	Popup windows - TP source code and .EXE files
TPPROC19.ZIP	44282	890630	Appointment Calndr, Async, Begin-End Code Check.
TPRAT.ZIP	12587	890612	TP Rat Provides Mouse Support for TP 4.0.
TPRO.ZIP	14120	890630	Screen I/O, Joy-Stick, Add to ACCESS.BOX.
TPROUTN.ZIP	260250	891030	A collection of Turbo Pascal routines
TPSPool.ARC	5740	890503	Put printer output to a file (TP source only)
TPTC175C.ARC	64565	880829	Turbo Pascal to C - src for translator, 2of4
TPUTUTOR1.ZIP	45344	890630	Tutorial on TP w/ Sample Programs & Text 1 of 2
TPUTUTOR2.ZIP	32451	890630	Tutorial on TP w/ Sample Programs & Text 2 of 2
TPU2ASM.ZIP	38906	901118	Disassembler for Turbo Pascal 5 units
TPM2N.ZIP	3563	890630	V1.02 Test of POWERN functions with Docs.
TPZF.ZIP	32588	890526	Rudimentary ZMODEM send/rec, TurboPas source
TRANSCBD.C	1268	890630	Transcendental Functions in BCD Turbo Pascal
TREE.ZIP	7586	890630	Routines for Using Binary Trees w/ Docs.
TREEDIR.ZIP	2667	890630	Routines to Access Directories
TREES.ZIP	6277	890630	Telephone System Using Binary Trees.
TRSCRIPT4.ZIP	51964	890630	Version 4 Word Processor with Documentation.
TRPEECH.ARC	14930	881203	Turbo Pascal sources for PC voice routines
TTT5.ZIP	335438	890612	Turbo TechnoJock Toolkit Ver. 5.0 for TP V4 & 5
TTTDOC.ZIP	34929	890630	Documentation for TTTTPU - Turbo Pascal Unit
TTTPPU.ZIP	37733	890630	TP Toolkit Unit - Menus, Popups, Mouse, More!
TUFPINT.ZIP	2963	890630	Printer Error Control Routines
TUR6_101.ZIP	102778	890630	Enhancements to TP that Use External Asm Lang
TUR6_102.ZIP	160171	890630	TP Utils: File Compression, TSR, Memory Managmt
TURB_IN.C	11229	890630	Inkey,Power,String /Lcase/Ucase/Trim/Pad/Justfy
TURBARG.ZIP	2631	890630	Routines to Parse & Get Cmd Line Arguments
TURBCOMM.ZIP	54115	890630	Communications Routines for the DEC Rainbow.
TURBDRD.ZIP	2242	890630	File Directory Program.
TURBHELP.ZIP	24524	890630	On-Line Turbo Pascal Version 3.0 Help Files.

TURBO00.ZIP	161436	890630	Ninety-Three Turbo Pascal Programs of All Kinds.	CSPEL5RC.ARC	100733	871231	C Language source for CSPELLA.ARC
TURBO03.ZIP	3670	890630	Comments on V3.0 & How to Use an Intel 8087	DDJGREP.ARC	25365	871119	Find strings in files, with source
TURBO09.ZIP	96406	890630	70 Disk/Scr/World/10/Heap/Draw/Draw/Sort/Xref.	DOS2UNIX.ARC	28108	891005	MSDOS->Unix newline conversion pgm, w/C src
TURBO_SQ_ZIP	13364	890630	Squeeze/UnSqueeze Utility Written in TP.	DTSRCH11.ZIP	285800	910908	dtSearch: Indexed/unindexed text search w/edit
TURBO_UT_ZIP	28649	890630	Miscell! Graphics, I/O Routines with Docs.	FL1P5SRC.ARC	26708	890713	Convert text files MSDOS->UNIX format, 2of2
TURBOBUG.ZIP	3530	890630	Subtraction Bug in Turbo Version 2.00 & Fix.	GESTALT.ARC	28278	881207	New string similarity function. ASM,C,TurboPas
TURBOCLR.ZIP	3865	890630	Color Patch Correction for Turbo Version 2.00B.	INDEX.ARC	3375	880722	Create text file indexes in C
TURBOFIX.ZIP	1062	890630	Changes TP COM Files for: Ctr Scrn at Pgm Start	LEX_YACC.ARC	182821	900301	Creat/Yacc compiler compiler, w/Turbo Pascal src
TURBOHLP.ZIP	24524	890630	Set of Online Help Files While Using TP.	LST6OASM.ARC	13833	880724	ASM for early V Bueg LIST file viewer
TURBOINT.ZIP	11988	890630	Keyboard Read Routine for Keyboard Scan Codes.	LUCIFER.ARC	7827	880723	Encrypt/De-crypt bytes, LUCIFER algorithm
TURBOIO.ZIP	33651	890630	Turbo Pascal Input/Output routines.	MORE13.ZIP	45278	901223	Scroll forward/backward thru text file, w/src
TURBOJP.ZIP	50091	890630	Toolbox ACCESS >64K Records, Cmd Line Parse,More	NGCLON11.ZIP	33170	901216	Norton Guides clone, w/Turbo Pascal 5.0 src
TURBOPIC.ZIP	1382	890630	V.2 Cnvrts BASICA Bload Screen to Memory Images.	NOCT12.ARC	10243	891013	Strip Ctrl Z's from files, 3 versions, ASM src
TURBOPM.ZIP	50091	890630	Misc Procs - Xref,DB Access,Driver,Int24,Etc.	NOE111.ARC	19456	870208	Adds command line text to notebook file
TURBOPR2.ZIP	31467	890630	Pretty Printer Src Code Formatter & Xfer Prgrm.	NR0FF1.ZIP	34371	901210	Unix V7 nr0ff clone with source for MS C 5.1
TURBOPRT.ZIP	10576	890630	Print Pascal File, Expands Include Files	PCPATCH.ARC	70019	880209	Patch old source to current version
TURBOPW.ZIP	3749	890630	Signon Password Program.	PDSRT12.ZIP	23749	900713	PD disk-based external sort program, w/C src
TURBOSCR.ZIP	18198	890630	TP Code Screen Gen from Inbedded Special Chars.	PRINT.ZIP	1812	880808	Print TXT files w/pg# , headers, etc...
TURBOTUT.ZIP	168100	890630	Turbo Pascal Version 3.0 Tutorial Package.	PROFF.ZIP	56584	880420	Another version of ROFF with source code
TURBOTW2.ZIP	59589	890630	Screen, I/O, Box, Menu Handling Routines w/Docs.	QUOTE24.ZIP	99327	910615	Fortune cookie quotation generator, w/C source
TURBOWHL.ZIP	137151	890630	Eighty File and I/O Routines with Documentation.	SCRIP170.ZIP	5298	910702	Builds cursive writing from text, w/TPas src
TURBSC.ZIP	8417	890630	Creates Source Code for Input Screen with Docs.	SEDI5.ZIP	67052	910105	Unix-compatible streaming editor v1.5 TC src
TURBSERL.ZIP	3645	890630	Serial Routines to Access COM1 and COM2 Ports.	SPELL01.ZIP	98649	911114	TurboC++ or Unix C source for Unix Spell clone
TURBSTRG.ZIP	11172	890630	String Handling, Power, Inkey, with Docs.	TABX10.ARC	14125	900305	Filter that expands TABS to SPACES, w/TC src
TURBTUT1.ZIP	19543	890630	Window/Password/Screen/Encrypt/Serial/Time	TOADCR11.ARC	9598	891107	Unix <-> DOS text file EOL converter w/ASM src
TURMEN.ZIP	15117	890630	Menu of Disk Input/Output Routines.	TRAVESTY.ARC	4325	900119	Strip trailing spaces from text file, w/ASM src
TURPRT16.ZIP	14750	890630	Pretty printer for Turbo Pascal source code.	TRAVESTY.ARC	21854	881218	Text/music style simulator w/C source
TURUTI.ZIP	3237	890630	File Append/Check Kbnd Locks/Other General Routs	TSORT.ZIP	3429	880728	Text file sort in C
TURTOPAS.ARC	202811	890630	Jack Greshaw's compiler for Turbo Pascal src	UNDIRCT.C	5155	901127	Break down newsletter digests to individ. msgs
TURV3.5.ZIP	38416	911026	CROSSREF 1.02: TURBO VISION IN VGA/ECA: sources	UNTAB4.ZIP	11499	880808	Remove tabs from C source listing
TXREF1A.ZIP	6386	890630	Cross Reference and Pretty Printer List Program.	WISDOM2.ZIP	111085	910528	A fortune cookie program with Turbo C source
TXTRZND.ZIP	1321	890630	Conv Text File to 80 Char Rec File for Rndm Acc.				
U.ZIP	3781	890630	Convert TP's Reserved Words to Upper Case Chars				
UPCONV14.ZIP	16848	901205	Convert Pascal reserved word w/TP5.0 src				
USING.WIZ	2489	890630	Print Using Like BAS \$\$\$###,###,### Routine in TP.				
UTIL.ZIP	19331	890630	S2 Scrn, Pwr,Ser,Encrypt, Menu, Window Routines.				
UIDR.ZIP	7869	890630	Directory & Char Matching Routines w/Docs				
VALDFIL.ZIP	6388	890630	Check for Valid Input Filenames w/ Doc, TP Src.				
VDL.ZIP	5736	890630	Verify Before Deleting Files.				
VIDEO_10.ZIP	5287	890630	Direct Vid On/Off/Cursor/Attrib/Read/Wrt Routs.				
WAIT.ZIP	583	890630	Wait Until any Key is Pressed, Turbo Pascal.				
WCTUNITS.ZIP	22875	910811	Collection of Turbo Pascal units with sources				
WHATV2.ZIP	8893	890630	Lists to File Procedures/Functions in TPascal				
WTH.ZIP	15068	890630	Stay Resident & Window Programs with Docs				
WINDOWMG1.ZIP	91126	890630	V1 Window Manager Program with Docs & Src Code.				
WINDOWMG2.ZIP	56587	890630	V2 Window Manager Program with Docs & Src Code.				
WINDOWD21.ZIP	29543	890630	V2.1, Routines for Making a Window. Fast.				
WINDOWD.ZIP	2635	890630	Window Routines, WINDOW.DOC & WINDOW.PAS.				
WINDOWS.WIZ	1436	890630	Window Include File in Turbo Pascal.				
WRD52.ZIP	1925	890630	Tallies Occurrences of Each Word in a Text File.				
WRITDEMO.ZIP	1069	890630	Demo of Writelin & Writexy Fast Routines.				
WS2ASCII.ZIP	5548	890630	Converts Wordstar Formatted to ASCII Text Files.				
WSX.ZIP	3931	890630	Converts Wordstar Files to ASCII Text Files.				
XAFMT2.ZIP	2389	890630	Compresses CompuServe's Directory Format.				
XREF.ZIP	2627	880812	(turbo) Pascal Cross Reference (.pas/.com)				
XREFF3B.ZIP	9337	890630	Xref Using TP Reserved Words, from Grogono's Bk				
XREFP3B.ZIP	19122	890630	Xref from Wirth's Program 4.8, Algo Data-Pgms				
XREFP3AS.ZIP	22275	890630	Cross-Reference Program for Turbo Pascal Files.				
XREFTP.ZIP	10773	890630	Xref Using Faster Indexing. Also See XREFF3B.ZIP.				
YASYNC.VIC	14797	890612	"Yet Another" Async I/O Unit. Interrupt-Driven				
YESNO302.ZIP	5509	890630	Bypasses the Request: "Include Error Messages?".				
YESNO3B.ZIP	2943	890630	Latest Patches for Message File "Yes/No" Questn.				
YMODEM.ARC	3670	880523	YMODEM protocol Turbo Pascal source				
TXUTL							
ASC2EBC.ARC	2069	900203	Converts ASCII to EBCDIC, w/ASM source				
ASC2WORD.ARC	10560	900203	Converts ASCII to MS Word docs, w/C source				
BIGSORT.ZIP	7334	890613	Sort Large Files by Key. OK for Pipes.				
BREAKUP.C	5997	860426	Break up large text files into several smaller				
BYACC.ZIP	103026	910228	Berkeley YACC for MS-DOS, w/C source & docs				
CASER.ZIP	35042	901026	Text case changing pgms with TC++ v1.0 source				
CAWF2.ZIP	134442	911004	Formats docs intended for nr0ff -man, w/C src				
CSPELLA.ARC	109278	871231	Spelling checker (see CSPEL5RC for source)				

Schluß des Beitrags Harddiskmanagement

```
void KillFiles()
{ char FileType[3][6]={".EXE",".OBJ",".BAK"};
  char Pfad[MAXPATH];
  int Handle, Zaehler, RueckWert;
  struct ffb1k FFStruktur;
  struct ftime DateiZeit;
  char Delete[20];
  for (Zaehler=0; Zaehler<=2; Zaehler++)
  { RueckWert=findfirst (FileType[Zaehler],&FFStruktur,0);
    while (! RueckWert)
    { Handle=open (FFStruktur.ff_name,0_RDWR);
      gettimeofday(Handle,&DateiZeit);
      close (Handle);
      if ((Heute.da_year == DateiZeit.ft_year+1980)
          && (Heute.da_mon == DateiZeit.ft_month)
          && (Heute.da_day == DateiZeit.ft_day))
      { strcpy(Delete,"DEL ");
        strcat(Delete,FFStruktur.ff_name);
        system(Delete);
        getcwd(Pfad,MAXPATH);
        if (strlen(Pfad) > 3)
          strcat(Pfad,"\\");
        printf("\n%s gelöscht.",
              strcat(Pfad,FFStruktur.ff_name));
      }
      RueckWert=findnext (&FFStruktur);
    }
  }
  return;
}
```

```
void RootEinstallen(char *RootPfad)
{ char Pfad[MAXPATH]="";
  if (RootPfad[1] == ':')
  { setdisk(RootPfad[0]-'A');
    strcpy(Pfad,RootPfad);
  }
  else
  { Pfad[0]=getdisk()+'A';
    Pfad[1]=':';
    Pfad[2]=0;
    if (RootPfad[0] == '\\')
      strcat(Pfad,"\\");
    strcat(Pfad,RootPfad);
  }
  chdir (Pfad);
  strcat (Pfad,"\\");
  return;
}
```


Elektronik- Bauteile



Computerzubehör-Handels-GesmbH.
A-1070 Wien, Lindengasse 20
Tel. 93 51 97 FAX 52 666 95

Rüsten Sie auf!

● DRAM 41256	43,-
● DRAM 44256	93,-
● Cache RAM 64kBx1	88,-
● Cache RAM 64kBx4	198,-
● Cache RAM 32kBx8	198,-
● SRAM 6264 (8kBx8)	68,-
● SRAM 62256 (32kBx8)	128,-
● GAL 16V8	ab 35,-
● GAL 20V8	ab 38,-
● serielle EEPROMs, z.B. 93C46 (1kBit)	ab 43,-
● TTL-Oszillatoren, z.B. 50-66-80-100-132 MHz für Motherboards	ab 63,-

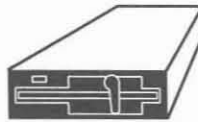
Wir führen das vollständige SMD-Programm!

● SMD-Widerstände, Bauform 1206	10 Stk. 6,-
● SMD-Widerstände, Bauform 805	10 Stk. 6,-
● SMD-Kondensatoren	10 Stk. 9,-
● SMD-Tantal-ELKOs	ab 6,-
● SMD-Dioden	ab 2,-
● SMD-Zener-Dioden	ab 2,-
● SMD-Transistoren	ab 2,-
● SMD-LEDs	ab 3,-
● SMD-Spulen	ab 8,-
● SMD Serie 74HCxxx, 74HCTxxx	
● SMD Serie CMOS 4xxx	
● SMD EPROMs	
● SMD RAMs	

Computer- Sonder- Aktion



Computerzubehör-Handels-GesmbH.
A-1070 Wien, Lindengasse 20
Tel. 93 51 97 FAX 52 666 95



5,25"-Floppy (360k/1,2M)
neuwertig/getestet

480,-

- Wechselrahmen für 3,5"-AT-Bus-Harddisks 480,-
- DataSwitch (CENTRONICS oder RS232) ab 420,-
- Ventilator 80x80 mm 128,-

Kabel u. Adapter

- SIMM/SIPP-Adapter - macht SIMM zu SIPP 44,-
- SCSI-Kabel ab 98,-
- Harddisk-Kabel f. 3,5"-AT-Bus ab 98,-
- Harddisk-Kabel f. 2,5"-AT-Bus 198,-
- Floppy-Kabel (auch für MFM-Platten) 128,-
- Netzwerk-Kabel RG58, per m 9,-
- BNC-Stecker ab 29,-
- BNC-Abschlußwid. 59,-
- Monitor-Adapter PS/2 auf VGA 108,-
- Keyboard-Adapter AT auf Mini-DIN 98,-
- Maus-Adapter 120,-

Alles zum Kabel-Selbst-Konfektionieren!

Flachbandkabel - Pfostenstecker - Edge Connectors



Computerzubehör-Handels-GesmbH.
A-1070 Wien, Lindengasse 20
Tel. 93 51 97 FAX 52 666 95



Computerzubehör-Handels-GesmbH.
A-1070 Wien, Lindengasse 20
Tel. 93 51 97 FAX 52 666 95

Wir halten über 7000 verschiedene Artikel für Sie bereit. In unserem Geschäftslokal erhalten Sie gegen einen Unkostenbeitrag von öS 10,- eine Diskette für PC (DOS) mit einem Katalog unseres gesamten Programms.

Alle Preise in öS incl. 20% MWST.
Preisänderungen, Irrtum und Druckfehler vorbehalten.

Speziell für PC-NEWS-Leser!



-10% - Gutschein

Mit diesem Gutschein erhalten Sie auf alle Elektronik-Bauteile (ausgenommen RAMs) der Fa. ASCOM Computerzubehör Handels-GesmbH. eine Ermäßigung von 10%. Der Gutschein ist nur gültig, wenn er zusammen mit der PCC-TGM- oder der MCCA-Clubkarte vorgewiesen oder gefaxt wird. Gültig bis 1. 5.

PC-NEWS Heft 35



EXPERT MOUSECLOCK



Atomuhr für PC

Mit Expert MouseClock empfängt Ihr PC das atomgenaue DCF77-Signal. Die mitgelieferte Software korrigiert die

- Backup ab bestimmtem Datum
 - Prozeßsteuerung
 - Zeitgesteuerte Meßwertaufnahme
 - Verteilung der Uhrzeit vom Netzwerk-Server an die Hosts
- Expert MouseClock ist anschließbar an XT, AT u. PS/2-Kompatible und wird mit einem 9-poligen Anschluß für die serielle Schnittstelle und Treibersoftware für DOS, Windows und OS/2 geliefert.

öS 1680,-

Technische Daten:

- .) Schnittstelle: RS232
- .) Empfangsanzeige: grün/rot blinkend
- .) Stromaufnahme: ca. 55mA
- .) Anschluß: 9-polige Buchse
- .) Kabellänge: 2m

SYWA EDV-Handelsges.mbh

Gentzgasse 9, 1180 Wien vis a vis vom WIFI, U6 - Station Volkstheater



Sprechstunden Montag - Freitag: 9:30 - 18:30

Samstag: 9:30 - 12:30

Telefon: 470 70 05

Umbau - Erweiterung - Reparatur - Zubehör

Festplattenpreise - solange Lager reicht

Seagate ST3243A	- 214MB, IDE, 16ms	3.480,--
Seagate ST3390A	- 342MB, IDE, 12ms	4.680,--
Seagate ST3550A	- 452MB, IDE, 12ms	6.690,--

PCI Boards lagernd - wir besorgen auch original AMI Mainboards

ZyXEL FaxModem U1496e plus Herbst/Winter-Aktion

- bekannt - beliebt - bewährt - bis 19.200bps nach ZyXEL-Norm
 - inklusive Software und deutschem Handbuch
 - nachträgliche Postgenehmigung gegen Aufpreis möglich (ca. 12/93)
- 7.980,--

Pocket FaxModem 1414P

- 14.400bps Modem + Fax mit neuem AT&T Chip
 - ideal für Notebooks oder als externes FaxModem
 - für Batterie- und Netzbetrieb (Netzteil inkludiert)
 - BitFax und BitCom Software für Windows 3.1 in deutsch
 - Dokumentation der Hayes AT-Kommando-Codes
 - CCITT / MNP-Normen + Datenkompression und Fehlerkorrektur
- V.23, V.32, V.32bis, V.42, V.42bis und MNP2-5
- 4.260,--

ASAP TF555 automatischer Faxumschalter

- arbeitet lautlos - automatisch - schnell
 - macht defacto aus einer Telefonleitung zwei
 - europäisches Läutsignal (BRD/Österreich) für den Anrufer
 - erkennt FAXANRUF automatisch oder durch CODEWAHL
 - erkennt MODEMANRUF durch CODEWAHL
 - funktioniert mit Pulswahl oder Tonwahl
 - problemloser Anschluß für Fax, Telefon, Anrufbeantworter
 - programmierbare Funktionen für individuelle Konfiguration
 - rufen Sie uns einfach an: Fax/Tel = Wien 470 70 06
 - Installationsbeschreibung in deutsch
- 2.970,--

Verlangen Sie unsere Fax-Preisliste = 4 Seiten - wöchentlich neu!

Computersysteme - nach Maß

KASSA-ABHOLPREISE - inklusive 20% MwSt.

Basis-System	Desk-Top Tischgehäuse, 200 Watt Netzteil, AT-Bus- und Diskettencontroller, 2 ser. + 1 par. Schnittstelle, Game Port, Kabel, Rahmen, große AT-Tastatur	Preis: 1.980,--
Assemblingtest (Kombinationstest) gewählter Komponenten ohne Zusammenbau		990,--
Zusammenbau und Test des Computers durch den 'Computer-Doktor'		1.500,--
Diskettenlaufwerk	<ul style="list-style-type: none"> - 5 1/4" = 1,2 MB Panasonic Laufwerk 780,-- - 3 1/2" = 1,44MB Panasonic Laufwerk 690,-- - 5 1/4" + 3 1/2" = 1,2 + 1,44MB Teac od. Canon Kombi-Laufwerk 1.980,-- - 3 1/2" IOMEGA Floptical = 21MB Cartridge + 1.44MB / 720KB Disketten (bootbar!) 4.980,-- 	
Speichererweiterung	- Sonderpreis zu Systemen - pro 1MB SIMM 70ns 3-Chip	690,--
Festplatten	<ul style="list-style-type: none"> - 215MB Seagate ST3243A 16ms AT-Bus, 1" hoch 3.480,-- - 340MB Seagate ST3390A 12ms AT-Bus, 1" hoch 4.680,-- - 452MB Seagate ST3550A 12ms AT-Bus, 1" hoch 6.690,-- 	
Controller VESA Local Bus	<ul style="list-style-type: none"> - IDE-Interface mit Schnittstellen 2s/1p 870,-- - IDE-Cache-Interface DC-680, 0K (bis 16MB Cache) 2.370,-- - IDE-Interface + 2s/1p/1G + Cirrus VGA 1MB 1.980,-- - SCSI-2 + IDE + FDC + 2s/1p, mit ASPI Treiber 4.680,-- 	
Mainboards mit AMI Bios	<ul style="list-style-type: none"> - 486DLC-33 Cyrix-CPU, 8K internem Cache, MX-Chipset 2.370,-- - 486DLC-33 Cyrix-CPU, 256K Cache, EFAR-Chipset, 2x VESA Local Bus 3.960,-- - 486 Mainboard (ohne CPU), 2x VESA Local Bus, 256K Cache 2.370,-- - 486DX-33 Mainboard mit Intel-CPU, 256K Cache 5.970,-- - 486DX-40 Mainboard mit AMD-CPU, 256K Cache 6.450,-- - 486DX-50 Mainboard mit Intel-CPU, 256K Cache 8.370,-- - 486DX2-66 Mainboard mit Intel-CPU, 256K Cache 9.480,-- 	
PENTIUM P24T 60MHz	- zusätzlich ist die Aufzählung zu PCI, EISA, UMC oder ALI VLB notwendig	22.560,--
Aufzählungen:		
VESA Local Bus (VLB)	- 2x VLB Slots, Contaq oder EFAR Chipset, 256K Cache, bis 32MB Ram	480,--
UMC Chipsatz + VLB	- 7x ISA Slots, davon 3x VLB, UMC Chipset, 256K 15ns Cache, bis 128MB Ram	1.560,--
ALI Chipsatz + VLB	- 7x ISA Slots, davon 3x VLB, ALI Chipset, 256K 15ns Cache, bis 128MB Ram	1.980,--
EISA + VLB	- 7x EISA Bus, davon 3x VLB, ALI Chipset, 512K 15ns Cache, bis 128MB Ram	2.280,--
PCI Bus	- 3x PCI Bus + 4x ISA + NCR-53C810 SCSI-2 F+2s1p +IDE, 256K Cache, -128MB Ram	4.470,--
PCI Bus + VLB	- 3x PCI Bus + 5x ISA mit 3x VLB, SIS Chipset, 512K 15ns Cache, bis 128MB Ram	4.470,--
PCI BUS + EISA + VLB	- 3x PCI Bus + 5x EISA mit 3x VLB, ALI Chipset, 512K 15ns Cache, bis 128MB Ram	5.360,--
BIG Tower	- Tower-Gehäuse mit 5x 5 1/4" Einschübe, 220 Watt Netzteil - anstatt Tischgehäuse	990,--
Space Tower	- wie Tower-Gehäuse, aber 5x 5 1/4" + 2x 3 1/2" (+ 1+1 interne) Einschübe, 230 Watt	1.500,--
iceFAN CPU-Kühlung	- unbedingt zu 486DX-40, DX-50, DX2-66 oder P24T !	990,--
Coprozessor	- Cyrix oder USLI für 386DX oder 486DLC - 33/40MHz	990,--
Microsoft Mouse	- die Originale Microsoft Maus - seriell	660,--
MS DOS 6.0	- zu unseren Systemen, deutsch mit Handbuch	990,--
MS DOS + Windows 3.1	- zu unseren Systemen, deutsch mit Handbüchern	1.800,--
250MB Floppy Streamer	- intern zum Anschluß am Floppy-Controller, inklusive Software + Kabel	2.790,--

Video- & Grafikkarten 640 x 480 in True Color, bis mind. 1024x768 non interlaced bei 70Hz VL = VESA Local Bus

ET4000	Tseng ET4000 Chip, 1MB	Wintach 1.2:	6,40	1.380,--
IIT AGX-014	IIT 014 Chip, 1MB VRAM (2MB) = ISA Bus 16bit !		34,06	3.480,--
VL-Cirrus	Cirrus GD5428 Chip, 1MB (2MB)		4.) 31,59	1.980,--
VL-S3	S3 86C805 Chip, 1MB (2MB)		5.) 25,22	2.970,--
VL-IIT AGX-015	IIT 015 Chip, 1MB VRAM (2MB)		3.) 44,54	3.990,--
VL-Star 2000	Weitek 9000 Chip, 2MB VRAM, Farben: 640x768 + 800x600 = 16.7M, 1024x768 = 32K		2.) 47,56	5.970,--
Diamond VIPER	von Diamond Systems USA, wie Star 2000 + mehr Treiber + 5 Jahre Garantie		1.) 49,58	6.960,--

Farbmonitore

alle mit einer Mindest-Auflösung von 1024x768 Bildpunkte

VGA-S	14" 0,28mm 640x480, 800x600, 1024 x 768	F I	3.480,--
VGA-LR	14" 0,28mm 640x480, 800x600, 1024 x 768	F S	3.990,--
TM 1510NLD	15" 0,28mm elektronisch geregelt, bis 1024x768	E M NI S	5.980,--
NOKIA 449a	15" 0,28mm 31,5-62KHz, 50-100Hz, bis 1024 x 768 - 76Hz, TCO-91	E M NI S	9.960,--
CPS 1760LR	17" 0,28mm Flat Screen entspiegelt, 30-65KHz, 50-90Hz, bis 1280 x 1024	E M NI S	10.480,--
NOKIA 447b *)	17" 0,26mm Flat Screen entspiegelt, bis 1280 x 1024 60Hz, 31,5-64KHz, 48-100Hz, TCO-91	E M NI S	16.980,--
NOKIA 445m	21" 0,26mm Flat Screen entspiegelt, bis 1280 x 1024 74Hz, 30-79KHz, 50-120Hz, TCO-92	E M NI S	37.960,--
NOKIA 445x	wie 445m aber mit Auflösung bis 1600 x 1200 mit 80Hz NI	E M NI S	42.480,--

*) mit Sony Trinitron Bildröhre E = elektron. geregelt S = Strahlungsarm NI = non interlaced I = interlaced F = Fixfrequenz M = Multisync

Fax - Modems - Fax-Umschalter (für Puls- und Tonwahl / ohne Postzulassung!)

ZyXEL U-1496	externes Fax-Modem bis 19200bps bei Modem, mit LCD-Anzeige, Software	11.970,--
ZyXEL U-1496E plus	wie oben mit LED-Anzeige	7.980,--
ZyXEL U-1496E	externes Fax-Modem bis 16800bps bei Modem, mit LED-Anzeige, Software	6.960,--
LD 1414 iX	internes Fax-Modem bis 14400bps bei Fax+Modem, MNP 2-5, deutsche Windows-Software	3.690,--
LD 1414 eX	gleiche technische Daten, aber externes Gerät	4.470,--
LD 1414 Pocket	Pocket Fax-Modem, 14400bps wie LD 1414eX, MNP2-5, deutsche Windows-Software	4.260,--
TF555	automatischer Umschalter für Fax, Telefon, Anrufbeantworter u. Modem (programmierbar!)	2.970,--
FaxBuffer	Fax-Puffer für bis 40 einkommende Faxe - Fax-Ausdruck auf Nadel- oder Laserdrucker	4.470,--
Adapterkabel	TDO-Stecker -> 1x RJ11 Stecker (= TDO intern gebrückt) 6m	300,--
RJ11 = mittlere 2pol/ von 6	TDO-Stecker -> 2x RJ11 Stecker (= LINE + PHONE) 6m	300,--
TDO = a/b/a1/b1	TDO-Buchse -> 1x RJ11 Stecker 10cm	150,--
	2x TDO-Buchse parallel -> 1x RJ11 Stecker 2m	210,--

Unsere Notebooks**SywaNote 486SLC 33** 33MHz i486SLC-Prozessor, 4MB RAM (bis 8MB)

2,9Kg inkl. Akku, 120MB Festplatte, 1,44MB Floppylaufwerk - MS DOS deutsch vorinstalliert, inkl. Handbuch
 LCD-Schirm: 10", 32 Graustufen, Stromsparmmodus, - Netzteil im Notebook integriert! 90-240 Volt
 Akku für 2,5 Stunden Betrieb, deutsche Tastatur, Anschluß - Optionen: internes FaxModem
 für externen VGA-Monitor, Tastatur, und Autobatterie 4 MB Speichererweiterung
komplett: 21.980,--

PEACOCK Voyager P40 25MHz i486SL-Prozessor (mit Coprozessor), 4MB RAM (- 20MB)

2,8/3,0Kg inkl. Akku, Abmess.: H:4,6(5,3) x B:29 x T:22cm - MS DOS und MS Windows 3.1 vorinstalliert
 200MB Festplatte, 1,44MB Floppylaufwerk, - 1 serielle + 1 parallele Schnittstelle
 LCD-Schirm 9,5": TSTN 64 Graust. oder Farb Aktiv Matrix TFT - 2 PCMCIA 2.0 Einschübe
 Stromspartechnologie, Akku für 3-4 bzw. 2,5-3 Stunden Betrieb - integrierter Trackball
 Netzteil (Ladegerät), deutsche Tastatur, deutsche Handbücher
 Anschluß für externen VGA-Monitor (simultan) und PS/2 Maus - Optionen: 4MB /16MB Modul, internes FaxModem

komplett monochrom: 29.970,-- mit Farb-LCD: 49.980,--**Wir führen auch Zenith Notebooks von Zenith Data Systems**

CPU's = Prozessoren (ohne Mainboard !)

Intel 80486DX - 33MHz	4.500,--
AMD 80486DX - 40MHz	4.980,--
Intel 80486DX - 50MHz	6.690,--
Intel 80486DX2 - 66MHz	7.980,--

Wechsel-Festplatten

Syquest	SQ 5110 88MB, internes SCSI-System	4.470,--
	SQ 5110C schreibt/liest 44+88MB Platten	5.490,--
	SQ 555 44MB, internes SCSI-System	1.770,--
	SQ 400/ SQ 800 44MB/ 88MB Wechselplatte	990,--/ 1.470,--
	SQ 3105A 105MB, internes 3 1/2" IDE-System	5.880,--
	SQ 3105S 105MB, int. 3 1/2" SCSI-System	6.870,--
	SQ 310 105MB Wechselplatte	1.590,--

IOmega Floptical intern, liest/schreibt Standard-Disketten!

3 1/2" SCSI-Laufwerk: 21MB Cartridge, 720KB + 1.44MB Disketten	4.980,--
SCSI-Hostadapter (damit ist Floptical bootfähig als A: Laufwerk)	570,--
21MB Floptical Cartridge	420,--

CD ROM Laufwerke

TOSHIBA XM3401B	intern, SCSI, multis., Photo CD, Audiokabel	6.570,--
TEAC	intern, SCSI, multis., Photo CD	5.490,--
Mitsumi	intern, mit eigenem Interface	2.970,--
Gehäuse für externes CD-ROM Laufwerk für SCSI oder AT-Bus		1.980,--

SCSI Streamer

Tandberg TDC 4220 2GB	16.590,--
NovaBack Software für DOS - deutsch	1.980,--
NovaNET NLM für Novell, 20 User - englisch	6.990,--
unbegrenzt - englisch	9.960,--

SCSI Hostadapter

IDE Interface + SCSI Hostadapter + Floppycontroller	1.470,--
Always AL1000 SCSI-Druckerport-Interface - " -	2.970,--
Adaptec 1542cf Hostadapter mit FDD-Controller	3.180,--
Adaptec 1542cf wie oben - inklusive Software-Kit	4.260,--
Adaptec 2742 EISA Hostadapter mit Software-Kit	5.970,--
Adaptec 2842 VESA Local Bus Hostadapter + S/W-Kit	4.980,--
Tekram DC880 VESA Local Bus Cache-Hostadapter + S/W-Kit	
CI2020 SCSI-2 Hostadapter + FDC + ASPI Software	2.490,--
CI2200 Local Bus SCSI-2 Fast Adapter+IDE+FDC+2s1p+ASPI	4.680,--
COREL SCSI - Treibersoftware (ASPI) für WORM, CD, Streamer	1.290,--

Wechselrahmen-System

für 3 1/2" HH Platten SCSI oder IDE - Rahmen + Schlitten	660,--
für 3 1/2" HH SCSI-Platten Rahmen mit Ventilator + Schlitten	870,--
für 3 1/2" 1" Platten SCSI oder IDE - Rahmen + Rack	690,--
Standard Rahmen für alle Systeme ohne Ventilator	480,--
SCSI-Rahmen mit Ventilator	590,--
Schlitten oder Rack für alle Systeme	390,--

WORM und MO-Laufwerke intern (ohne Interface und ohne S/W)

PIONEER WORM-Laufwerk 650MB CD's	28.590,--
650MB CD-Medium - leer, 1x Beschreiben, mehrfach Lesen	2.190,--
PIONEER MO-Laufwerk 1,3GB 2-seitig à 650MB	43.980,--
1,3GB MO-Medium - leer, mehrfach Beschreiben/Lesen	3.480,--

SCSI-Gehäuse

1x 5 1/4" oder 3 1/2" HH, Standgehäuse, Netz. 43 Watt	1.980,--
1x 5 1/4" HH Gehäuse mit Griff, 43 Watt Netzteil	2.490,--
2x 5 1/4" HH (= 1x 5 1/4" FH), Netzteil 100 Watt	2.670,--
2x 5 1/4" FH, (= 4x 5 1/4" HH), Netzteil 100 Watt	3.270,--
7x 5 1/4" HH Disk Array Subsystem	8.850,--

IDE - FDC - I/O

IDE (AT-Bus) Interface + Floppycontroller, 2ser.+ 1par.+1G I/O	270,--
VESA Local Bus IDE-Interface - sonst wie oben (ohne Gameport)	870,--
IDE Interface + FDC bis 2,88MB Floppy (für AMI-BIOS 92) +2s1p1G	1.470,--
IDE Cache-Spiegel-Interface + Floppycontroller	2.370,--
IDE Interface + SCSI Hostadapter + Floppycontroller	1.470,--
Floppy-Controller 360KB - 2.88MB für 4 Floppylaufwerke	990,--
I/O-Karte:	
2 seriell (2x 16450) + 1 parallel + Game Port	210,--
2 seriell (1x 16C550, 1x 16450) + 1par. + Game Port	450,--
2 seriell + 2 parallel + Game Port	480,--

Tastatur - Maus - Trackball

AT-Tastatur, groß, deutsch (auch andere Sprachen möglich)	480,--
Cherry AT-Tastatur (B.R.D.), deutsch	990,--
Logitech MouseMan - PS/2 + seriell	780,--
Logitech TrackMan - Trackball	1.200,--
Logitech TrackMan Portable - Trackball für Notebooks	1.290,--
Legend Ball - ergonomischer Trackball	690,--
Microsoft Mouse - seriell, 200dpi	660,--
Microsoft Mouse - ergonomisch, seriell + PS/2	1.290,--

Scanner (siehe auch Notebook-Scanner bei Notebooks)

Handy HS800	64 Graust., 100-800dpi, Software	1.980,--
Handy GS800	256 Graust., w.o.	2.970,--
Handy CG8000	24bit Farben, 100-800dpi, Software	5.490,--
A4 Flatbed Color	Flachbett-Farbscanner 24bit, 100-600dpi	12.990,--

Drucker

Brother HJ100	Tintenstrahldrucker	3.870,--
	Einzelblatteinzug für HJ100	990,--
Brother HJ-400	100 Blatt od. Kuv., 64 Düsen, 360x360dpi	5.490,--
Canon BJC600	neuer Farb-Tintenstrahldrucker 360x360dpi	10.480,--
Star SJ-144	neuer Farb-Thermotransferdrucker	9.960,--
Brother HL-6V	6 Seiten/Min., 300dpi, PCL5, 150Blatt	13.980,--
Brother HL-10H	10 Seiten/Min., 600dpi, 2MB Ram, 74 Fonts	
	mit PCL5e, Postscript + Laserjet 4 Emulation!	28.470,--

Tinten-Nachfüllung

HP DeskJet 500	Original-Druckkopf 1x gefüllt	390,--
	Original-Druckkopf mit Doppelfüllung	480,--
	Druckkopf gefüllt + 2 Nachfüllungen	597,--
	2 Nachfüllungen	297,--
	5 Nachfüllungen	597,--
HP DeskJet 500C	Original-Druckkopf mit 3 Farben 1x gefüllt	1.380,--
BubbleJet	Canon BC01 Original-Druckkopf 1x gefüllt	480,--
(Canon/Brother/Star)	Druckkopf gefüllt + 1 Nachfüllung	597,--
	2 Nachfüllungen	396,--

Auto-Switch

2 PC - 1 Drucker/ 2 Drucker - 1 PC	870,--
4 PC - 1 Drucker/ 4 Drucker - 1 PC	1.200,--

Drucker-Netz

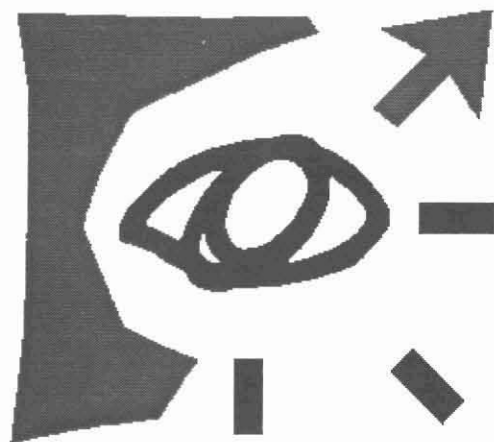
Sende-Modul am PC (bis 16 PC's)	870,--
Empfangs-Modul je (max. 3) Drucker	990,--

PC-Gehäuse

Baby AT-Gehäuse	3x 5 1/4" + 1x 3 1/2", 200W Netzteil	870,--
Baby Tower	2x 5 1/4" + 2x 3 1/2", 220W Netzteil	1.170,--
Middle Tower	3x 5 1/4" + 2x 3 1/2", 220W Netzteil	1.470,--
Big Tower	5x 5 1/4", 220W Netzteil	1.680,--
Space Tower	6(5)x 5 1/4"+3(2)x 3 1/2" 230W Netzteil	2.190,--

Techniker-Honorar:

Mindestsatz = 1/2 Stunde	480,--
An- und Rückfahrtszeit wird als Arbeitszeit gerechnet	



LOGITECH

1.990,--

Scanman 256 inkl. Fototouch

Schulversion Preis inkl. MwSt. zuzüglich Versandspeser

UPDATE

EDV Dienstleistungs- und Handelsges.m.b.H.

A-1030 Wien

Modecenterstraße 14

Tel 0222/7986893 Fax 0222/7986893-99

SYMANTEC.

C++

**SYMANTEC C++
PROFESSIONAL 6.0**

1.299,--

32 Bit Windows Compiler

Schulversion Preis inkl. MwSt. zuzüglich Versandkosten

UPDATE

EDV Dienstleistungs- und Handelsges.m.b.H.

A-1030 Wien Modecenterstraße 14

Tel 0222/7986893 Fax 0222/7986893-99

PREISLISTE-NOVEMBER 1993

Komplettsysteme

PC 386 DX 40 MHz 9.900,-

CPU AMD 386DX40, AMI-Bios, 128kB Cache,
 4MB Ram, 2ser, 1par, 1Game, 1,44MB FDD Teac
 170MB HDD, SVGA-Card 512kB, Deskcase TÜV
 LED, Tastatur

PC 486 DX 33 MHz 13.900,-

Konfig. wie PC 386 jedoch INTEL 486DX33

PC 486 DX 33 MHz VesaLB 15.900,-

VesaLB Mainboard mit INTEL 486DX33, AMI Bios
 256kB Cache, 4MB Ram, 2ser, 1par, 1Game,
 VesaLB IDE Contr., 1,44MB FDD, 170MB HDD,
 Vesa LB Bildschirmkarte S3-Chip 1MB(auf 2MB auf-
 rüstbar) Miditower, Tastatur

PC 486 DX 33 MHz VesaLB/EISA 16.900,-

Konfig wie PC 486 VesaLB jedoch zusätzlich
 3 EISA Slots

Mehrpreise

Mehrpreise ermitteln Sie selbst lt. Preisliste:

z.B.:	PC 486 DX 33 Mhz	13.900,-
	- CPU 486DX33	3.970,-
	+ CPU 486 DX 2-66	6.990,-
	ergibt	16.920,-

Mainboards

Mainboard 386 DX 40 auf 486 Upg. 1.990,-

AMD 386DX40CPU, 128kB Cache, AMI BIOS
 Sockel für Intel 486 bis DX2-66 CPU

Mainboard 486 VesaLB 1.890,-

2-3Vesa Slots, 256kB Cache ohne CPU

Mainboard 486 VesaLB/EISA 2.880,-

2Vesa, 3Eisa Slots, 256kB Cache ohne CPU

INTEL - 486DX CPU

INTEL 486 DX 33 MHz CPU 3.970,-

INTEL 486 DX 2-50 MHz CPU 4.920,-

INTEL 486 DX 50 MHz CPU 6.400,-

INTEL 486 DX 2-66 MHz CPU 6.990,-

Speichererweiterungen

Simm Modul 256kB 70/80nS 149,-

Simm Modul 1MB 70nS 648,-

Simm Modul 4MB 70nS 2.390,-

AT-BUS Festplatten

Western Digital 2170 14ms 170MB 2.690,-

Western Digital 2250 12ms 250MB 3.190,-

Western Digital 2340 12ms 340MB 4.490,-

Western Digital 2420 11ms 420MB 5.790,-

Alternativ liefern wir je nach Verfügbarkeit CONNER Festplatten

Größere Festplatten und SCSI Festplatten auf Anfrage

Diskettenlaufwerke

TEAC Diskettenlaufw. 3½" 1,44MB 590,-

TEAC Diskettenlaufw. 5¼" 1,2MB 690,-

Einbaurahmen für 3,½" LW 48,-

Kontroller und Bildschirmkarten

IDE Kontroller ISA 216,-

2FDD, 2HDD, 2ser, 1par, 1Game

IDE Kontroller VesaLB 490,-

DC 600 IDE Cache Kontr. ISA 1.320,-

DC 680 IDE Cache Kontr. VesaLB 2.240,-

DC 820 SCSI Cache Kontr. EISA 5.590,-

TRIDENT 8900CL VGACard 512kB 690,-

TRIDENT 8900CL VGACard 1MB 990,-

DIAMOND Speedstar Pro 1MB ISA 1.800,-

DIAMOND Stealth 24 1MB S3 ISA 2.580,-

andere DIAMOND Bildschirmkarten auf Anfrage

Vesa LB S3 Chipset 1MB 2.280,-

Vesa LB S3 Chipset 2MB 2.980,-

CD-Rom und Streamer Laufwerke

MITSUMI CD-Rom multisess. 2.490,-

MITSUMI CD-Rom multisess. db-sp. 2.790,-

TEAC TCD50 SCSI Db.Sp. multis. 5.800,-

TOSHIBA XM3401B SCSI-2 DbSp. 6.990,-

COLORADO DL-20 intern 250MB 2.890,-

COLORADO Trakker 250MB ext. par. 5.590,-

Streamerkassetten

VERBATIM Datacartridge DC 2120 240,-

VERBATIM Datacartr. DC 2120 form. 273,-

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,-

PREISLISTE-NOVEMBER 1993

LOGITECH-Eingabegeräte

MouseMan cord. Combo	690,-
MouseMan cordless	1.390,-
Mouse Man Bus	790,-
MouseMan large	690,-
Pilot Mouse seriell	390,-
Pilot Mouse PS/2	490,-
ScanMan 32 Win inkl. OMNI-PAGE	2.190,-
ScanMan 256 für Windows	2.590,-
ScanMan Color für Windows	6.590,-
Scan Man Easy Touch par. Interf.	4.290,-
Audio Man für par. Interface	1.490,-
Sound Man 16	1.680,-

16Bit-, 44KHz Stereo, Yamaha OPL-3, Midi-Schnittst.

Cyber Man	1.190,-
Foto Man Plus Digitalkamera	9.990,-

GENIUS-Eingabegeräte

GENIUS Mouse One	199,-
GENIUS Mouse Too	390,-
GENIUS Cordless Mouse	490,-
GENIUS Scanner 4500 / 256 Gray	1.790,-
GENIUS Scanner Color 256	2.990,-

Genius Scanner inkl. IPhoto 3.0 for Win, OCR for Dos

Gehäuse und Tastaturen

Desktopcase, TÜV PS	936,-
MidiTowercase, TÜV PS	1.080,-
CHICONY 102 Tasten	270,-
CHERRY G81-3000HAD	890,-
CHERRY G80-1600HAD	4.990,-

mit integrierten Barcodedecoder

Barcodelesestift für G80-1600HAD	1.990,-
CHERRY G81-3100SAD	7.490,-

mit integrierten Magnetkartendurchzugsleser

Computerzubehör

BIRCHTEC Lüfter für 486er CPU	190,-
Monitorfilter 14" Glas mit Erdklemme	190,-
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. 100 Stk. 3 1/2"	89,-

Monitore

14" TVM 1024x768 b/w LR	2.190,-
14" Goldstar 1024x768 color MPRII	3.890,-
14" Samsung 1024x768 nl LR MPRII	4.890,-
15" Tystar 1280x1024 nl LR MPRII	5.990,-
17" Targa 1280x1024 dig. LR MPRII	9.990,-

CANON-Drucker

CANON BJ10SX 360dpi A4	3.490,-
CANON Einzelblatteinzug BJ10SX	990,-
CANON BJ10SX + Einzelblatteinzug	3.990,-
CANON BJ230 360dpi A3	5.990,-
CANON LBP4 Lite 300dpi, 0,5MB	6.990,-

CANON-Zubehör

Tintenpatrone für BJ10SX/BJ230	390,-
Akku für BJ10SX	450,-
Tonercartridge für LPB4	1.290,-

TEXAS INSTRUMENTS Laserprinter

microWriter Basic, 0,5MB, 300DPI	8.990,-
microWriter PS 23, 2MB, 300DPI	13.990,-
microWriter PS 65, 2MB, 300DPI	17.990,-

TEXAS INSTRUMENTS Zubehör

1MB Upgradekit	1.490,-
RS232 Schnittstelle	790,-
microwriter Tonerkassette	690,-
microWriter Bildkassette	2.290,-

Software

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

VERBATIM Disketten

VERBATIM Datalife 3 1/2" 1,44MB	159,-
formatiert	
VERBATIM Datalife 3 1/2" 1,44MB	189,-
formatiert, Teflonbeschichtet.	

Ab mitte November halten wir in unserem Geschäft reichhaltig CD-Rom (Spiele, Grafik, WinTools, ...) für Sie bereit. Preise sind Kassaabholpreise inkl. MWSt.

Druckfehler, Irrtümer und Änderungen vorbehalten.

Stand per 6.11.1993

excon

Rögerg. 6-8, 1090 Wien
Tel.: (0222) 310 99 74-0
Fax: (0222) 317 10 77

SONDERANGEBOT

(Gültig bis max.22.12.93, solange der Vorrat reicht!)

excon 486DX40VL/350

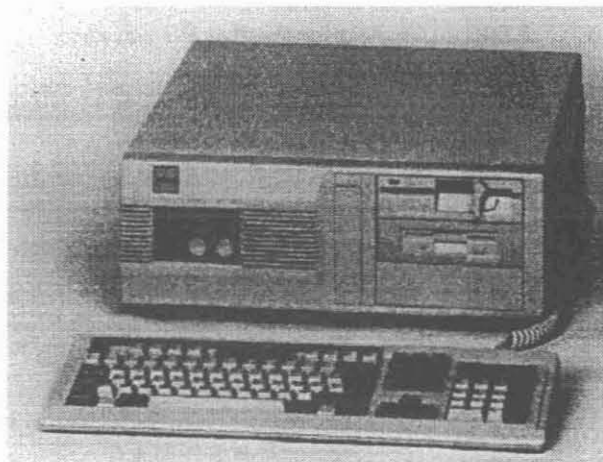
ÖS 18.990,-

486DX40 (AMD) cpu, 256k Cache, 4MB RAM, 350MB/12ms
Festplatte (Seagate), Cirrus Logic VL-LOCAL Bus VGA Karte, True
Color, IDE-VL-Bus Controller, 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/350 - SET14"

ÖS 22.980,-

wie oben, inkl. 14" SVGA non-interlaced Monitor, max.1024x768,
strahlungsarm,



excon 486DX2-66VL/350

ÖS 22.470,-

486DX2-66 Intel cpu, 256k Cache, 4MB RAM, 350MB/12ms
Festplatte (Seagate), Cirrus Logic VL-LOCAL Bus VGA Karte, True
Color, IDE-VL-Bus Controller, 1.44MB 3½" Floppy, Desktop
Gehäuse mit 200W Netzteil, 2 Seriell, 1 Parallel Port, MS-DOS und
Windows 3.1 vorinstalliert, Tastatur (Datacomp)

excon 486DX2-66VL/350 - SET15"

ÖS 27.990,-

wie oben, inkl. 15" SVGA Monitor AOC CMLB-536, max.1280x1024
strahlungsarm nach MPR II

Zahlung: Bar/Scheck bei Warenübernahme, Gültig solange der Vorrat reicht!
Abholung oder kostenpflichtige Zustellung, Preise inkl. 20% MWSt, 12 Monate Garantie

X

Firma
Excon Warenvertriebsges.m.b.H.
Rögergasse 6-8
1090 Wien

Ich bestelle lt. Sammelbestellung 11/93 wie
wie oben beschrieben:

Type: _____ ÖS _____ incl.MWSt.

Name: _____ Plz: _____ Ort: _____
Straße/Nr.: _____ Telefon: _____

Datum: _____ Unterschrift: _____

excon

Rögerg. 6-8, 1090 Wien
Tel.: (0222) 310 99 74-0
Fax: (0222) 317 10 77

SONDERANGEBOT

(Gültig bis 22.12.93, solange der Vorrat reicht!)

ZENITH Z-STAR 433VL MOD.80

21.990,-

486SLC33 cpu Cyrix, 4MB RAM, 80MB Festplatte, 1.44MB 3½" Floppy, 9,5" Mono LCD Display, 64 Graustufen, 1 Seriell, 1 Parallel, Mouse Port, Windows 3.1 u. DOS6.0 vorinstalliert, integrierte J-MOUSE, Simultanbetrieb von Display u. ext. Monitor, ca.3h Betriebsdauer, PCMCIA Type II Slot, Standby u. Resume Mode.

ZENITH Z-STAR 433VLP MOD.200

29.970,-

200MB Festplatte, 9,5" Color -LCD Display, 256Farben, ca. 2h Betriebsdauer, kein Simultanbetrieb, sonst wie Z-STAR 433VL Mod.80



Zahlung: Bar/Scheck bei Warenübernahme! Gültig, solange der Vorrat reicht.
Lieferung: ab Lager Wien, Preise inkl. 20% MWST, Garantie: 12 Monate

Firma
EXCON Warenvertriebsges.mbH.
Rögergasse
A-1090 Wien

Ich bestelle lt. o.a.Sonderangebot:
___ Stk Z-Star 433VL Mod.80/ 433VLP Mod 200
à ÖS 21.990,-/ 29.970,- incl.MWst.

Name: _____ Plz: _____ Ort: _____
Straße/Nr.: _____ Telefon: _____
Datum: _____ Unterschrift: _____

(bei Minderjährigen: Unterschrift des Erziehungsberechtigten!)

Große Modem - Umtauschaktion

Modems gibt es viele, wir haben für Sie die Modems mit den überzeugenden Qualitätsmerkmalen ins Programm aufgenommen:

ACTIVE - MODEMS

- aufrüstbar auf 19.2 k, 24.0 k und 28.8 k (V.fast)
- V.32bis 14.400 BPS Full Duplex Datentransfer
- Datenkompression nach V.42bis, V.42 und MNP 5
- Hayes V-Series kompatibler Befehlssatz
- V.17 14.400 bps G3 Send & Receive Fax
- Fax Class 1 & 2 kompatibler Befehlssatz
- V.23 (BTX) - Modulation
- 12 Status-LED's und 8 einstellbare DIP-Switches bei externem Modell
- Eingebauter FIFO UART 16550 bei internem Modell
- COM 1 - 4 und IRQ 2 - 5 einstellbar
- BitFax for DOS und Bitcom - Software serienmäßig
- 2 Jahre Garantie

Bestell Nr.	Beschreibung	Verk.Preis inkl.MWSt
144UF	Externes Modell: 14400 bps V.17 G3 FAX S/R + 14.400 V.32bis Modem mit V.23 und MNP2-5 (max. 57.600 bps)	7.230.--
9624F	Externes Modell: 9.600 bps V.29 FAX S/R + 2.400 bps Modem mit V.42bis und MNP2-5 (max. 9.600 bps)	3.950.--
9624P	Pocket - Modem: 9.600 bps V.29 FAX S/R + 2.400 bps Modem mit V.42bis und MNP2-5 (max. 9.600 bps)	3.150.--
144UB	Internes Modell: 14400 bps V.17 G3 FAX S/R + 14.400 V.32bis Modem mit V.23 und MNP2-5 (max. 57.600 bps)	6.390.--
144PC	Internes Modell: 9.600 bps V.29 G3 FAX S/R + 14.400 V.32bis Modem mit V.23 und MNP2-5 (max. 57.600 bps)	5.650.--
9624B	Internes Modell: 9.600 bps V.29 FAX S/R + 2.400 bps Modem mit V.42bis und MNP2-5 (max. 9.600 bps)	2.695.--
BFW-d	Bitfax für WINDOWS, deutsch *)	990.--
BFW-e	Bitfax für WINDOWS, englisch *)	690.--

*) nur in Verbindung mit einem Modem - MAC Software und Kabel auf Anfrage

HINWEIS: Die Modems haben keine Postzulassung!

Wenn Sie sich für ACTIVE - Modems entscheiden, profitieren Sie gleich dreifach:

Vorteil 1:

Hochwertige, ausgereifte, zukunftssichere Technik zu einem erstaunlich günstigen Preis!

Vorteil 2:

Wenn Sie ein Modell 144 UF oder 144 UB kaufen, vergüten wir Ihnen für Ihr altes Gerät **S 1.000.--!**

Die Abwicklung ist unbürokratisch: Wir senden Ihnen Ihr neues Modem und Sie schicken uns spätestens 14 Tage nach Erhalt Ihr altes Gerät einschließlich Software und Handbuch. Ihre Vergütung geht sofort nach Eintreffen des alten Gerätes per Überweisung oder Postbaranweisung an Sie ab!

Vorteil 3:

Wenn ein Kollege, Freund oder Bekannter mit Ihnen zusammen ein Modem der 144er Serie bestellt, erhalten Sie:

bei einem Mitbesteller: 1 Bitfax für WINDOWS - englische Version **gratis**

ab zwei Mitbestellern: 1 Bitfax für WINDOWS - deutsche Version **gratis**

Voraussetzung ist, daß Sie uns den/die Namen der Mitbesteller bekanntgeben und, daß der Versand in einer Sendung erfolgt! Das Umtauschangebot gilt natürlich auch für Mitbesteller!

Diese Aktion ist bis 31. 12. 1993 befristet!

Klaus Gloner

Hard- und Software

Dr. Hainstrasse 8

5620 Schwarzach

Tel. 06415/7215 - Fax 06415/7218

Für Schnellentschlüssene....

Corel Draw! 3.0

2.490.--

Deutsche Version, Update - fähig, einschließlich Handbücher, Diskettensatz und CD - ROM

in Verbindung mit einem anderen Artikel aus unserem Angebot

1.990.--

FoxPro 2.5

3.490.--

die aktuelle Version für MS - Windows jetzt bei uns zum Kennenlernpreis!

Pagemaker 5.0

8.200.--

MS - Office

9.100.--

bestehend aus: WinWord 2.0, Excel 4.0, Powerpoint, 1 Lizenz Mail

MS - Office Professionell

12.750.--

wie vor, zusätzlich MS - Access

WINDOWS 3.1

990.--

Das aktuelle Release: Stabiler - Standfester - Schneller!

DOS 6.0

990.--

Die letzte Version des Betriebssystems mit einigen Erweiterungen:
MEMMAKER - DOUBLESPEACE - VSAFE - BACKUP etc.

in Verbindung mit einem anderen Artikel aus unserem Angebot

890.--

Mitsumi CD-ROM Laufwerk

2.990.--

Multisessionfähig, Foto-CD st.

KGS - Truetype Schriften

990.--

Serifen - Sans Serif - Antiqua - Script - Monospaced - Fremdsprachen - Symbolfonts

BONUS für PC-NEWS-Leser: Inklusive Schriftenkatalog und TTFMan-Schriftenverwaltungssoftware!

Geld sparen mit unseren Bündelangeboten:

Die Bundles bestehen jeweils aus 1 Stück MS-DOS 6.0 Update , 1 MS-kompatiblen Maus und einem der folgenden Produkte:

- Stacker 3.0 für Windows/DOS deutsch
- Microsoft Flugsimulator 5.0
- Microsoft Golf
- Microsoft Money
- Microsoft Cinemania (CD)
- Microsoft Multimedia (CD)

Preis pro Bundle

2.790.--

Superqueue 3.0

1.220.--

mehr als ein Druckmanager - Ersatz!

Global Explorer

2.437.--

Die Welt auf einer CD: Über 120000 Orte, Staaten, Flugverbindungen, Informationen etc.

Lieferung solange Vorrat reicht!

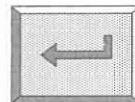
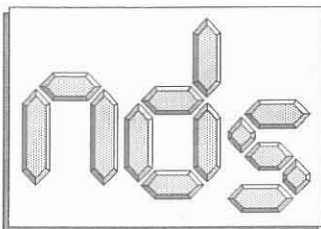
Klaus Gloner

Hard- und Software

Dr. Hainstrasse 8

5620 Schwarzach

Tel. 06415/7215 - Fax 06415/7218

**Zahlungskonditionen:**

Bei Warenübernahme Bar oder Scheck

Preise:

ab 1.11.1993, in ATS inklusive 20% MWSt.

Lieferung:

ab Lager Wien, solange der Vorrat reicht

Garantie:

12 Monate auf alle Komplettgeräte u. Noteb./24 Monate=Quantum, WD/60M=HP

Versand:

Mit Transoflex per Nachnahme unfrei in ganz Österreich innerhalb von 24 Stunden Werktags!

Änderungen und Irrtümer in Druck und Satz Vorbehalten

NDS 386DX/40MHz**12.775,-**

Babytower Gehäuse YY+200W Schaltnetzteil TÜV mit Thermocontoll
Motherboard MSI AT386DX40 128KB externes Cache
4MB Hauptspeicher (erweiterbar bis 32MB on Board)
Controller IDE 2x HDD / 2x FDD inkl. At Multi I/O / 2ser/1par
Floppy Laufwerk 3,5"/1,44MB
Grafikkarte 1MB Cyrus Logic Chipsatz / 1024x768-70Hz
deutsche Tastatur 102 Tasten Cherry GQ81-3000
DOS 6.0 deutsch
Festplatte Quantum ELS170 IDE 170MB

NDS 486DX/40MHz Vesa Local Bus**16.990,-**

Baby Tower Gehäuse YY+220W Schaltnetzteil TÜV mit Thermocontoll
Motherboard MSI AT486DX33 Vesa Local Bus 256KB externes Cache
4MB Hauptspeicher (erweiterbar bis 32MB on Board)
Controller IDE 2x HDD / 2x FDD inkl. At Multi I/O / 2ser/1par
Floppy Laufwerk 3,5"/1,44MB
Grafikkarte Cirrus Logic VLB / 1024x768-70Hz und Truecolor
deutsche Tastatur 102 Tasten Cherry GQ81-3000
DOS 6.0 deutsch
Festplatte Quantum ELS170 IDE 170MB

NDS 486SX/25MHz**14.545,-**

wie NDS 486DX/40 jedoch mit SX25MHz CPU

NDS 486DXII/66MHz Vesa Local Bus**19.990,-**

wie NDS 486DX/40 jedoch mit DXII66 CPU

Aufzahlung auf Big Tower+230W**700,-****Notebooks****Texas Instruments TM4000 WinSX Mono****22.990,-**

486SX/25MHz mit 4MB RAM
10" LCD Display/64Graustufen 640x480
120MBHDD/2 ser/1 par/externe VGA bis 800x600
Dos 6.0 + Windows 3.1 deutsch

Texas Instruments TM4000 WinSX Color**30.990,-**

9" STN Color Display sonst wie Mono

Festplatten IDE

Western Digital AC2170 170MB 3,5" 14ms	2.990,-
Western Digital AC1210 210MB 3,5" 14ms	3.290,-
Western Digital AC2250 250MB 3,5" 12ms	3.990,-
Western Digital AC2340 340MB 3,5" 14ms	4.490,-
Western Digital AC2420 425MB 3,5" 14ms	5.990,-
Seagate 3145A 127MB 3,5" 16ms	2.390,-
Seagate 3550A 450MB 3,5" 12ms	5.990,-
WDAC 2540 540MB 3,5" 12ms	7.990,-
Conner 540MB 3,5"/12ms	8.590,-

Festplatten SCSI

Quantum ETS170 SCSI-II 170MB 15ms 3,5"	2.990,-
Maxtor SCSI-II 340MB 12ms 3,5"	5.590,-
Saegate/Quantum/Maxtor 540MB	9.990,-
Seagate/HP/DEC 1050MB	15.990,-
Seagate/DEC 1400MB	17.990,-
Seagate/DEC 1600MB	19.990,-
Seagate/HP 2000MB	29.990,-
Seagate 3400MB	44.690,-

Floppylaufwerke

Teac 5,25" 1,2MB	830,-
Teac 3,5" 1,44MB	640,-

Controller IDE

IDE/2xHDD/2xFDD + I/O (2 ser/1 par/ 1 Gameport)	295,-
Promise VLB IDE 2xHDD/2xFDD/2xser/1xpar/1xGame	550,-
DC600B ISA Cache (bis 16MB) IDE/2xHDD/2xFDD	1.790,-
DC680B Cache (bis 16MB) VESA IDE/4xHDD/2xFDD	2.590,-

Controller SCSI

DC800 Cache (bis 16MB) 7xHDD/2xFDD	3.590,-
DC320 Non Cache EISA 7xHDD/2xFDD	4.390,-
DC820 Cache (bis 16MB) 7xHDD/2xFDD EISA	5.990,-
DC880 Cache (bis 16MB) 7xHDD/2xFDD VLB	5.590,-
Adaptec 1522 ISA 7xHDD/2xFDD	1.840,-
Adaptec 1542CF 7xHDD/2xHDD	2.990,-
Adaptec 1542CF Kit/7xHDD/2xHDD	4.390,-
Adaptec 2842VL-Risc Kit (Vesa)/7xHDD/2xFDD	5.590,-
Adaptec 2742 EISA KIT /14xHDD/2xFDD	7.490,-

RAM's

1MB SIMM-70ns	675,-
4MB SIMM-70ns	2.390,-

Modem-Fax

Boca 1496i V42 Bis (14400Bd/9600Bd)	3.590,-
Pro Link 1496e V42Bis (14400Bd/9600Bd)	3.990,-
Pro Link 1496p V42Bis (14400Bd/9600Bd)	4.390,-

Alle Modems in Österreich nicht Postgenehmigt. i= intern/e= extern/p= Pocket
Alle Fax-Modem beinhalten Win-Faxsoftware und Modemsoftware f. Dos.

Motherboards inklusive Prozessoren

386DX-40/AMD/128KB Cache	1.990,-
486SX-25/Intel/256KB Cache/3xVESA LB	3.690,-
486DX-40/AMD/256KB Cache/3xVESA LB	6.090,-
486DX-2-50/Intel/256KB Cache/3xVESA LB	7.290,-
486DX-50/Intel/256KB Cache/3x VESA LB	8.590,-
486DX-2-66/Intel/256KB Cache/3x VESA LB	9.590,-
486DX-50/Intel/256KB Cache/3x VESA LB/5x EISA	12.990,-
486DX-2-66/Intel/256KB Cache/3x VESA LB/5x EISA	13.990,-
486DX-40/AMD/256KB Cache/5xISA 4x32Bit PCI	7.990,-
486DXII-66/Intel/256KB Cache/5xISA 4x32Bit PCI	1.490,-
Pentium Board inkl CPU/60MHz/512KB Cache VLB	33.990,-
Eisa-PCI Board in Vorbereitung	
Alle 486er Boards haben den Pentium Zif-Sockel	
Coprozessor 80387DX/40	990,-

Multimedia & Sound

Sound Blaster 2 Deluxe Edition	1.290,-
Sound Blaster Pro 2 Deluxe Edition	1.990,-
Sound Blaster Pro 16	2.490,-
Sound Blaster Pro 16 ASP	3.490,-
Mulimedia Kit CD 16 Discovery Pack	5.990,-
-Sound Pro2, CD ROM CR-563 Double Speed, Stereoboxen, CD Secret o. Monkey Island, CD Loom, CD Just Grandma&Me, CD Animals, CD Where in the world is carmen Sandiego	
Mulimedia Kit CD 16 Performance Pack	7.990,-
Soundblaster 16 + Mikrophon CR-563 CD ROM Laufwerk/Doublespeed/Multisession/280ms ansonsten wie Discovery Pack	

Video

Video Blaster	5.990,-
Video Micro	5.290,-
Video Micro Converter (von PC=>VIDEO)	3.990,-
Miro Movie PRO VL oder ISA	15.990,-

GRAFIKKARTEN

Type	Chip/ Prozessor	Speicher	max. ** Auflösung Non Interlaced	max. Farben	Garantie	Preis
Trident 9000	Trident	512KB	1024x768 i	256	6M	650.-
Cirrus Logic 5422 ISA	Cirrus L. 5422	1MB	1024x768-70Hz	16,7 Mio	6M	1.090.-
Cirrus Logic 5426 VLB	Cirrus L. 5426	1MB (bis 2MB)	1024x768-70Hz	16,7 Mio	6M	1.490.-
Tseng Labs W32/ISA	Tseng W32	1MB	1024x768-70Hz	16,7 Mio	12M	1.990.-
Tseng Labs W32/VLB	Tseng W32	1MB	1024x768-70Hz	16,7 Mio	12M	2.290.-
Tseng Labs W32/VLB/2MB	Tseng W32	2MB	1280x1024-60Hz	16,7 Mio	12M	2.990.-
Diamond VIPER VLB	Weitek 9000	2MB VRAM	1280x1024-74Hz*	16,7 Mio	60M	6.490.-
ATI Mach 8/8514A	ATI Mach 8	1,5MB	1024x768-74Hz	256	12M	2.590.-
ATI Mach 32 VLB/8514A	ATI Mach 32	2MB VRAM	1280x1024-76Hz	16,7 Mio	12M	6.990.-
Miro Crystal 8S ISA/VLB	S386C801	1MB VRAM	1280x1024-60Hz*	16,7 Mio	24M	2.990.-
Miro Crystal 24S VLB/EISA/ISA	S3/928	3MB VRAM	1280x1024-75Hz	16,7 Mio	24M	10.590.-
Miro Crystal 32S VLB	S3/928	4MB VRAM	1280x1024-75-Hz	16,7 Mio	24M	14.990.-
Miro 16 SI PCI-Bus					24M	6.290.-
Miro RAINBOW EISA /ISA/VLB	S3/928	4MB VRAM	1408x1024-72Hz	16,7Mio	24M	21.990.-
Matrox Impression MGA VLB 3Z	Matrox 64-Bit Chip	3MB VRAM+ 2MB DRAM	1600x1280-76Hz	16,7 Mio	24M	25.990.-

** 1MB = 640x480 mit 16,7 Mio Farben/800x600 mit 65.000 Farben/1024x768 mit 256 Farben/2MB = 640x480 bis 800x600 mit 16,7 Mio Farben/800x600 und 1024x768 mit 256 Farben/3MB = 640x480 bis 1024x768 mit 16,7 Mio Farben/4MB = bis 1280x1024 mit 16,7 Mio Farben

MONITORE

Type	Zoll	Loch- maske	MPRII	Auflösung maximal	Garantie	Preis
Philips 7BM749 VGA Mono	14"	0,28		640x480	12M	1.380.-
Philips 4BM2797	14"	0,28	ja	640x480	12M	2.690.-
Philips 1409 (7CM5209)	14"	0,28		1024x768i	12M	3.990.-
Philips 1410 (7CM5279)	14"	0,28	ja	1024x768i	12M	4.590.-
Philips 4CM4270	14"	0,28	ja	1024x768/70Hz	12M	5.390.-
Philips 1520	15"	0,28	ja	1024x768/70Hz	12M	6.990.-
Philips 4CM6088 Trinitron	17"	0,26	ja	1280x1024/60Hz	12M	17.990.-
TARGA	17"	0,26	ja	1280x1024/60Hz	6M	9.990.-
Philips 1720 Invar	17"	0,26	ja	1280x1024/76Hz	12M	20.990.-
Philips 2010 (4CM2799)	20"	0,31	ja	1280x1024/60Hz	12M	19.990.-
Philips 2120DAS Invar	21"	0,28	ja	1280x1024/80Hz	12M	36.990.-
Philips C2082DAS Trinitron	20"	0,31	ja	1280x1024/80Hz	12M	33.990.-
EIZO Monitore						auf Anfrage
TARGA	15"	0,28	ja	1024x768-72Hz	6M	5.990.-

CD ROM

Bezeichnung	max. MB	Datentransfer permanent	Zugriff	KB Buffer	Photo CD/Multi- session	Schnitt stelle	Preis
Philips CCD521 CD-Recorder 1xBeschreibbar; ohne Software	680MB	350KB/s 100KB/s-Copy	280ms 1000ms	256KB	ja	SCSI/2	74.990.-
Philips CM206 KIT inkl. Kabel, Adapter, Photo CD Software, Treiber	635MB	352KB/s	350ms	64KB	ja		4.990.-
Toshiba XM3401-B	680MB	350KB/s	260ms	64KB	ja	SCSI1/2	6.290.-
NEC CDR38 Multispin portable->über Parallele extern	680MB	300KB/s	400ms	32KB	ja	SCSI1/2	6.390.-
NEC CDR84-1 Multispin	680MB	300KB/s	280ms	256KB	ja	SCSI1/2	5.990.-
NEC CDR74-1 Multispin extern	680MB	300KB/s	280ms	256KB	ja	SCSI1/2	7.990.-
Creative Labs CR-563+ Photostyler auf CD	680MB	303KB/s	280ms	64KB	ja	Sound Bl. Pro/16	4.690.-

Gehäuse und Netzteile

Bezeichnung	Netzteil TÜV+TC	5,25" FDD	3,5" FDD	3,5" HDD	Abmessungen B x T x H (mm)	Preis
Slimline Gehäuse YY7020	200W	2	1	1	430x400x105	1.390.-
Desktop Gehäuse YY6020	200W	3	1	1	420x413x155	1.390.-
Baby Tower Gehäuse YY3020	200W	2	2	1	180x405x330	1.390.-
Tower Gehäuse YY1020	230W	5	1	1	190x425x620	2.090.-

Mäuse

Microsoft Maus seriell /Bus Version 2.0 ERGO	12M	1.590.-
Logitech Pilot seriell	12M	490.-
Digitus 2 Tasten (mit Daumentaste) MS Kompatibel	12M	690.-

Tastatur

TAST001	Tastatur Cherry GQ-3000 HAD 102 Tasten
TAST002	Tastatur Mitsumi deutsch

12M
6M

790.-
390.-

HP Laserdrucker

Type/Mod.	Laserjet 4 L	Laserjet 4LM	Laserjet 4P	Laserjet 4MP	Laserjet 4	Laserjet 4M	Laserjet 4Si
Druckvol/Monat	8.000	8.000	8.000	8.000	20.000	20.000	75.000
Max Geschwind.	4 Seiten/Min.	4 Seiten/Min.	4 Seiten/Min.	4 Seiten/Min.	8 Seiten/Min.	8 Seiten/Min.	16 Seiten/Min.
Auflösung	300x300	300x300	600x600	600x600	600x600	600x600	600x600
skal. Schriften	26	10 TrueType	26	10 TrueType	45	80	45
Postscript		35 Schriften		35 Schriften	Optional	35 Schriften	Optional
Steckplätze f. Schriftenk.	1	1	1	1	1	1	2
RAM	1MB MET*	4MB	2MB	6MB	2MB(=>34MB)	6MB(=>22MB)	4MB(=>34MB)
1 Papierkassette	100 Blatt	100 Blatt	100 Blatt	100 Blatt	250 Blatt	250 Blatt	500 Blatt
2. Papierkassette					100 Blatt	250 Blatt	500 Blatt
3. Papierkassette					Option 500 Bl.	Option 500 Bl.	Option 1500Bl.
Duplexdruck							Optional
autom. Kuverteinzug	nur manuell	nur manuell			Option 20 K	Option 20 K	Option 100K
Jet Direct Card					Optional	Optional	Optional
Apple Talk		ja		ja	Optional	Standard	Optional
HP PCL5/ HPGL2	Standard	Standard	Standart	Standart	Standard	Standard	Standard
Epson/IBM Proprinter					Option	Option	Option
Preis	10.990.-	16.990.-	17.990.-	21.990.-	23.990.-	32.990.-	53.990.-

TEXAS INSTRUMENTS/Compaq A3 LASERDRUCKER

Type/Mod.	Microwriter Basic	Microwriter PS23	Microwriter PS65	Microlaser PS23	Microlaser PS65	Compaq Pagemarq 15
Druckvol/Monat	5000	5000	5000	20.000	20.000	50.000
Max Geschwind.	5 Seiten/Min	5 Seiten/Min	5 Seiten/Min	8 Seiten/min.	8 Seiten/Min.	15 Seiten/Min
Auflösung	300x300	300x300	300x300	600x600	600x600	bis 800x400
skal. Schriften	5	23	65	23	65	50
Postscript	Option	Standard	Standard	Standard	Standard	Standard
Steckplätze f. Schriftenk.	1	1	1	1	1	2
RAM	512KB=>4,5MB	2MB=>4,5MB	2MB=>4,5MB	6MB=>22MB	6MB=>22MB	4MB=>18
1 Papierkassette	100 Blatt	100 Blatt	100 Blatt	250 Blatt	250 Blatt	250 Blatt
2. Papierkassette	Option	Option	Option	250 Blatt	250 Blatt	250 Blatt
3. Papierkassette				Option 500Blatt	Option 500Blatt	100 Blatt A3 Opt.
Duplexdruck						Option
autom. Kuverteinzug	Option	Option	Option	Option 20K	Option 20K	Option
Novell Card				Option	Option	Option
Apple Talk				Standard	Standard	Option
HP PCL5/ HPGL2	HP II	HP II	HP II	Standard	Standard	HP PCL5
Epson/IBM Proprinter	Option	Option	Option	Option	Option	
HEK	8.190.-	13.990.-	16.990.-	23.990.-	26.990.-	57.990.-

HP/CANON Tintenstrahldrucker

Type/Mod.	Deskjet 310 Color	Deskjet 510	Deskjet 500C	Deskjet 550C	BJ600 Color
Drucktechnik	Tintenstrahl Farbe	Tintenstrahl monochrom	Tintenstrahl Farbe	Tintenstrahl Farbe	Tintenstrahl Farbe
Druckvol/Monat	1.000	1.000	1.000	1.000	1.000
Entwurfmodus	245cps bei 10cpi	240cps bei 10cpi	240cps bei 10cps	240cps bei 10 cpi	4 Seiten/min.
Qualitätsmodus	167cps bei 10 cpi	120cps bei 10cpi	167cps bei 10cpi	240 bei 10 cpi	2 Seiten/min.
Geschwindigkeit	2-3 Seiten/min.	2-3 Seiten/min.	2-3 Seiten/min.	2-3 Seiten/min.	
Auflösung:	300x300	300x300	300x300	300x300	360x360Color
skal. Schriften	6	3	3	3	3
RAM	48kb	16KB (max. 512Kb)	48KB (max.512KB)	48KB (max.512KB)	
1 Papierkassette	Option 50 Blatt	Standard	Standard	Standard	
Kuverteinzug		Manuell	Manuell	Standard/20Bl.	
parallele Schnittstelle	Standard	Standard	Standard	Standard	ja
serielle Schnittstelle	Standard	Standard	Standard	Standard	ja
Emulation	HP PCL3	HP PCL3	HP PCL3	HP PCL3+	CANON Capsl
Epson/IBM Proprinter		Option			
Preis	4.690.-	4.690.-	5.790.-	9.590.-	11.990.-

FLACHBETTSCANNER Color

ArtiScan 3000C 300dpi (interpoliert 600x600dpi) mit Twain Schnittstelle (ASPI); 24BIT Farbtiefe;

Treiber für Photostyler, Photoshop, Corel 4.0 usw.

ArtiScan 6000C, 8000C, 12000C auf Anfrage

11.990.-

PC-NEWS *Bestellkarte*

☐ *Gesamtverzeichnis = PC-NEWS-34* ☐ 2te Ausgabe für Freunde (gratis)

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

Datum:

Unterschrift:

Disketten *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.

Datum:

Unterschrift:

PC-NEWS *Bestellkarte*

☐ *Gesamtverzeichnis = PC-NEWS-34* ☐ 2te Ausgabe für Freunde (gratis)

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

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,-
frankieren

An den

PC-NEWS -Eigenverlag

Siccardsburggasse 4/1/22
1100 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,-
frankieren

An

His Master's Voice Mailbox
Werner Illsinger

Flurschützstraße 36/12/5
1120 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,-
frankieren

An den

PC-NEWS -Eigenverlag

Siccardsburggasse 4/1/22
1100 Wien

Offenlegung gemäß Mediengesetz 1981

Grundlegende Richtung: Auf Anwendungen im Unterricht bezogene Informationen über Personal-Computer-Systeme, Berichte über Veranstaltungen von **PCC-TGM**, **MCCA**, **ADIM** und **FIDO**. Die PC-NEWS sind ein Forum für Beiträge aus dem Leserkreis.

Impressum

Medieninhaber: **PC-NEWS-Eigenverlag**
 Herausgeber: **PCC-TGM, MCCA, ADIM** und **His Master's Voice**
 Druck: Zlinské tiskárny a.s., POBOX 79, CZ-76097 Zlin-Kudlov,
 TEL: 0042-67-27239, FAX: 0042-67-28066
 Versand: CONCEPT, Baumgasse 52, 1030 Wien,
 TEL: 713-59-41, FAX: 713-87-72

PCC-TGM Personal-Computer-Club-Technologisches Gewerbemuseum

Anschrift: Wexstraße 21, Postfach 59, 1202 Wien.
 Telefon/Telefax: TEL: 332-23-98, FAX: 332-23-98-2
 Mailbox: **His Master's Voice**, Mitteilungen PCCTGM
 BTX: *56458# 912-222-584
 Bürozeiten: Mo: 19.00-20.30 (Frau Jedleňová)
 Konto: EÖSPC, Blz 20111, Kto. 053-32338, PCC-TGM
 DVR-Nr.: 0596299
 Jahresbeitrag: Schüler: 200,- Student: 300,- sonst: 400,-
 Vorstand: KÖNIG(OBmann), SYROVATKA(OBmannstv),
 THUMFARTH(Schriftführer), WEISSENBOCK(Schriftstv),
 ZEHETNER(Kassier), PAY(Kassierstv),
 OSTERMAIER(BTX), ZELINKA(PCAD), REITER(Scanner),
 BERTHOLD(Rechnungsprüfer), NITSCHKE(Rechnungspr.)

ADIM Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik

Anschrift ADIM-Wien: Martin Weissenböck, Postfach 23, 1191 Wien
 Telefon/Telefax/Telex: TEL: 369-88-59-8, FAX: 369-88-59-7, TX: 75210388=weim a
 BTX: *56458# 912-218-106
 Anschrift ADIM-Graz: Klaus Scheiber, Postfach 37, 8028 Graz.
 Telex: 75210659=sber a
 Mailbox: **His Master's Voice**, 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 Bräun, Konto 37283
 DVR-Nr.: 0547328
 Telefon Sprechstunde: 369-88-59-8 Montag ab 20.00
 Themen: Technik, Modems, Skripten, Unterricht.

MCCA BTX und Micro Computer Club Austria

Anschrift: MCCA, Traungasse 12, Postfach 143, 1033 Wien
 Clublokal: Am Heumarkt 4, 1030 Wien
 Telefon/Telefax/Telex: TEL: 585-13-93, FAX: 712-01-05, TX: 75210079=mcca a
 BTX: *2550# 912-222-064
 DVR-Nr.: 0536229
 Jahresbeitrag: Firma: 1200,- Einzel: 600,- Ermäßigt: 300,-

His Master's Voice, Mailbox unterstützt vom PCC-TGM

Anschrift: Werner Illsinger, Flurschützstraße 36/12/5, 1120 Wien
 BBS: 815-48-71 (8-N-1), 14400-300 Blls
 FIDO: 2.310/1.0@fidonet.org
 Spenden-Konto: PSK, Blz. 60000, Kto. 7.918.896, Werner Illsinger

PCC-TGM-Preise, Stand November 1993

Literatur, Sonderdruck pro Seite	6S	,-80
Tabellen A5/A4, verschweißt	6S	15,-/25,-
Kop.Disketten 360k/1.2M/720k/1.44M	6S	40,-/50,-/50,-/60,-
Leerdisketten 360k/1.2M/720k/1.44M	6S	10,-/20,-/20,-/30,-
PC-SIG-Disketten 360k/720k (PC-SIG)	6S	15,-/50,-
3-fach Verteiler für PC-Netzteil	6S	100,-
Bausatz µPROFI-51, incl. Handbuch	6S	950,-
EPROM für µPROFI-51 (PC-NEWS-2/92)	6S	55,-
Scannerdienst erste Seite (+Disk)	6S	10,-
Scannerdienst Folgeseite, pro Format	6S	2,-
Verpackungskostenanteil	6S	23,-/29,-

PC-NEWS-Eigenverlag, PC-NEWS-Redaktion

Anschrift: Franz FIALA, Siccardsburggasse 47/22, 1100 Wien
 Telefon/Telefax/Telex: TEL: 604-50-70, FAX: 604-50-70-2
 Mailbox: **His Master's Voice**, Dateien PCNDISK
 BTX: *56452# 912-218-242
 EMAIL: TELEBOX: FRANZ-FIALA, BIX: fiala
 CompuServe: 100024.1325, FIDO: 2.310/1.36@fidonet.org
 Internet: 100024.1325@compuserve.com
 Konto: PSK, Blz. 60000, Kto. 7486.555, PC-NEWS-Eigenverlag
 ISSN der PC-NEWS: ISSN 1022-1811
 DVR-Nr.: 0735485
 Bezugsbedingungen: Mitglieder des **PCC-TGM** kostenlos
 Mitglieder des **MCCA** Jahresabo (5 Hefte) S 50,-
 Einzelbezugspreis S 40,-
 Jahresabo Inland, inkl. Versand (5 Hefte) S 150,-
 Jahresabo Ausland, inkl. Versand (5 Hefte) S 220,-
 Bezahlte Werbung: S 1.000,-/Seite+10% (Inseratensteuer)+20% (MWST)
 Auflage: 2500 Stück.
 Kopien: Mit Quellenangabe gestattet. Zwei Belegexemplare erbeten.
 Layout, Belichtung: WinWord 2.0B, FOLEX
 Erscheinungsort: Wien, November 1993
 Programme: PCN-DSK 372
 Beitragskennzeichnung: Name, [Firma], [Zusatzinformation/Programm]
 Nicht gekennzeichnete Beiträge stammen von der Redaktion.
 Kopien: Mit Quellenangabe gerne gestattet. Zwei Belegexemplare erbeten.

PC-NEWS-Vorschau	Nr.	Red.	schluß	Erscheint	Thema
	36	Mitte Jan	Mitte Feb	Windows-Programme	
	37	Mitte Mär	Mitte Apr	LAN + ISDN	
	38	Mitte Mai	Mitte Jun	DFÜ	
	39	Anf Aug	Anf Sep	Mikrocontroller	
	40	Ende Okt	Ende Nov	Windows-Programme	

Autoren

Autorenliste siehe Seite 5.

Statistik

Quelle	Seitenzahl
Zu dieser Ausgabe	5 (inklusive Titelseite)
Clubs	4
Redaktion	11 (CD-ROM) + Berichte
Autoren	38
Eigen	17
Inserate	21
Seitenzahl gesamt	96
Beilagen	1 A4-Seite, Post

Inserenten

Firma	Seite
ADIM *)	45,50 (Bestellschein)
ASCOM	79
CompDelphin	3
exxon	88,89,96
GLONER	90,91
His Master's Voice *)	Bestellkarte beigeheftet
MANZ(Riemer) *)	51
nds	92,93,94
PC-NEWS-Eigenverlag *)	Bestellkarte beigeheftet
PESACO	86,87
POST	Beilage nach Seite 2
SYWA	80,81,82,83
UPDATE	84,85

*) unentgeltlich

Druckfehler und Irrtümer vorbehalten.

Alle Fernsprechnummern ab Ortsnetz Wien.
 Vorwahl Wien aus Österreich: 0222, (aus dem Ausland: ++431)

excon SERVICE -U. REPARATUR CENTER
SCHNELL - ZUVERLÄSSIG
- KOSTENGÜNSTIG



Wir reparieren in unserer eigenen Werkstatt :

- *- Monitore
- *- Personalcomputer
- *- Schaltnetzteile
- *- Peripherie

Preise für Reparaturen bzw.Service:

- | | |
|--|----------|
| *- Reparaturpauschale für MONITORE inkl. Ersatzteile
(ausgenommen Zeilentrafo, Bildröhre und Gehäuse) | ÖS 800,- |
| *- Reparaturpauschale für PC-Netzteile inkl. Ersatzteile | ÖS 350,- |
| *- Reparaturkosten für Personalcomputer und Peripherie
Peripherie excl. Ersatzteile per Stunde | ÖS 700,- |

Zahlung: Bar/Scheck bei Warenübernahme, keine weiteren Rabatte auf Dienstleistung möglich !
 Preise incl.20% MWSt, **excl.** Abholung und Zustellung der zu reparierenden Geräte.

Lesen

Lehren

Bieten

Sie mit! Als

Abonnent

Autor

Inserent