ATARI ST COMPUTER

Die Fachzeitschrift für ATARI ST- und TT-Anwender

Februar 91

DM 8,- Ös. 64,- Sfr. 8,-

2



Tempus Word
Endlich da!

Steve
Ein Programm für alle Fälle

ProFile T150
Neuer Streamer

Videodat
Datenempfang via TV

Button-Designer
Universeller Konstruktor

Neue Rubrik
PD-NEWS

Axel Weigand '91

PHOENIX

Besser kann man zwei Milliarden nicht anlegen. Zwei Milliarden Daten. Und deswegen kann die Bank, pardon die Datenbank Ihres Vertrauens eigentlich nur noch Phoenix heißen. Zumal diese zwei Milliarden für jede der Datenbanken gelten, von denen Sie bei Phoenix bis zu acht gleichzeitig eröffnen können. Mausmäßig einfach und saumäßig schnell. Denn ein eigener Cache-Puffer sorgt für Geschwindigkeiten, die man auf ST und TT bisweilen schmerzlich vermißte. Was ganz nebenbei verdeutlicht, daß Phoenix sowohl auf dem ST als auch auf dem TT läuft. Und das wahlweise in s/w oder schön bunt.

Kann man mit Phoenix nur Adressen verwalten ? Könnte man. Man kann aber noch viel mehr. Bereits einsatzfähig vorprogrammiert, verwöhnt Phoenix mit einer Adressverwaltung, einer Audio-Videoverwaltung und einem Literaturverzeichnis.

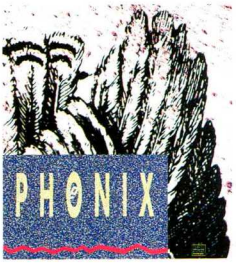
Darüber hinaus lassen sich aber auch die Mitglieder von FKK-Vereinen oder unbezahlte Rechnungen, die Playmates von 1958-1963 oder seltene Seevogelarten verwalten. In Form von Bildern, Formularen oder Tabellen. Das bringt uns ziemlich unvermittelt zu der Frage: Wie macht man das?

Man bedient sich einfach des integrierten Maskengenerators und legt dann schlankweg mit dem Mausmeister fest, in welcher Form man seine Daten geordnet haben möchte. Sollten tatsächlich Schwierigkeiten auftauchen (kaum unvorstellbar), hilft Phoenix sofort. Mit einem sogenannten kontext-sensitiven Hilfesystem. Was nichts anderes heißt, als daß Phoenix zu jeder gerade stattfindenden Tätigkeit einige äußerst nutzvolle Tips bereithält.

Zwei oder drei Worte (so zwischendurch) zum Begriff der relationalen Datenbank. Schließlich handelt es sich bei Phoenix um eine solche. Relational bedeutet, daß Sie aus purer Lust und Laune zwei völlig unterschiedliche Dateien miteinander verknüpfen können. Die Adressen aus der Freundinnen-Datei mit einer Telefonrechnung aus der Rechnungs-Datei.

Zum Beispiel. Um anschließend mit dem eingebauten Rechner (!) die durchschnittlichen Pro-Kopf-Gebühren präzise zu ermitteln. Nur so zum Beispiel.

Milliarden klitzekleiner Bits (ja, so viele) halten sich während Ihrer vergnüglichen Arbeit mit Phoenix sehr bescheiden im hintersten Hintergrund einsatzbereit. Damit sie auf Ihren leichthin geäußerten Wunsch solch mühselige Pflichten wie Importieren/Exportieren von Daten, Reporte



erstellen, Drucken etc. abarbeiten. Wovon Sie gar nichts merken werden, denn Sie können gleichzeitig weiterhin Ihrer Arbeit mit Phoenix nachgehen. Mit tollen Datentypen, die jedem Anwendungsnutzen gerecht werden. Genannt werden müssen da insbesondere Text, Zahl, Datum, Zeit und Grafik. Und Blob. Ein echt extremer Datentyp mit Zukunft.

Bei ihm sind die beliebigsten und unstrukturiertesten Daten ablegbar. Und aufrufbar. Und ablegbar. Und...

Nicht jeder sollte an Ihr Eingemachtes (datenmässig zumindest) herandürfen. Finden wir. Und deshalb bietet Phoenix einen unsäglichen Bankräuber-Verzweifelungs-Paßwort-Schutz und codiert damit auch gern die kleinste Ihrer Datenbanken. Da werden Computer-Hacker zu Computer-Hockern.

Anlegen oder nicht? Das dürfte jetzt wohl keine Frage mehr für Sie sein. Schließlich hat Phoenix genau das, was Sie brauchen. Und leistet dies mit unvergleichlicher Perfektion bereits bei bescheidenen 1 MB Arbeitsspeicher. Es wartet auf Sie eine zeitlos-elegant gestaltete Diskette, ein dickes Handbuch und ein wunderwunderschöner Aufkleber. Für nur 398,- DM - unser letztes Wort - wird Phoenix mit größter Freude die Datenbank an Ihrer Seite.

APPLICATION SYSTEMS HEIDELBERG

Application Systems Heidelberg Software GmbH, Englerstraße 3, Postfach 10 26 46, D-6900 Heidelberg 1, Telefon (0 62 21) 30 00 02, Fax (0 62 21) 30 03 89 **in Österreich:** Reinhart Temmel GesmbH. & Co. KG., St.Julienstraße 4a, A-5020 Salzburg, Telefon (06 62) 71 81 64, Fax 8 82 66 93 **in der Schweiz:** DTZ DataTrade AG, Landstraße 1, CH-5415 Rieden/Baden, Telefon (0 56) 82 18 80, Fax 82 18 84.

EDITORIAL

Narrhallamarsch!

Normalerweise steuert in den nächsten Tagen das Narrenschiff in den rheinischen Karnevalshochburgen dem absoluten Trara-Trara entgegen. Schnellebig wie unsere Zeit nun mal ist, kann das auch schon alles wieder ganz anders kommen. Als ich mir diese Worte durch den Kopf gehen ließ, stand noch nicht fest, ob nicht doch ölwirtschaftlich politisches Gezänke im fernen „Nahen Osten" unseren alljährlichen Narrhallamarsch verstummen lassen würde.

Da wir nun aber mal alle unverbesserliche Optimisten sind, gehen wir davon aus, daß sich unsere Welt zwischen Gestaltungs- und Erscheinungstermin dieser Zeitschrift hoffentlich nicht grundlegend verändert haben wird. Dann gibt es also die Rosenmontagsumzüge, die Kamellen und das „Wolle-mer-ne-reu-losse" in altgewohnter Manier. Für viele „gediente Närrinnen und Narrhallesen" stellt dann karnevalistisches Blech in Form phantasievoller Orden und Ehrenzeichen, das dann massenweise um wohlverdiente Hälse gelegt wird, ein bleibendes Symbol erfolgreicher Belustigungsaktivitäten dar – Narrhallamarsch!

Jede Branche kennt solche Auszeichnungen, mit denen herausragende Leistungen kenntlich gemacht werden sollen. Als bekanntester Vertreter solcher Symbolik wäre wohl der „Oscar" der amerikanischen Filmwelt zu nennen; aber auch unsere Computerbranche kennt ähnliche Metaphern zur Genüge. Gerade im angelsächsischen Sprachraum hat fast jede Computerzeitschrift (die etwas auf sich hält) eine Auszeichnung für Hard- und Software parat. Ähnlichkeiten im deutschen Blätterwald sind nahezu zwangsläufig (Narrhallamarsch?).

Warum gibt es eigentlich kein Analogon für den Atari ST? Ich schlage vor, auch für unsere Hard- und Software-Umgebung einen ebensolchen Oscar zu konstruieren. Als Namen würde ich in Anlehnung an den Ziehvater des Atari ST Jack Tramiel, einfach den Namen „Jack" vorschlagen. Und um die Posse vielleicht ganz auf die Spitze zu treiben, schwebt mir etwas ähnliches wie „Jack in the Box" vor - also ein Kästchen, dessen wirklichen Inhalt man vielleicht nie zu Gesicht bekommt.

Gerade weil ein Jahr kürzlich seine 365 Tage vollendet hat, könnte man doch am Anfang eines neuen die herausragenden Leistungen rund um den ATARI ST/TT-Computer durch einen „Jack" würdigen. Weil es zeitlich gerade so schön paßt, lassen Sie mich doch gleich damit beginnen:

Den „goldenen Jack" verleihe ich hiermit Herrn Alwin Stumpf, weil er als Kapitän des deutschen ATARI-Schiffs unermüdlich und allen Stürmen zum Trotz nun auch den großen Teich überqueren will, um auch das amerikanische ATARI-Schiff um Klippen mitherumzulotsen.

Einen „silbernen Jack" möchte ich Michaela Beckers und Uwe Wirth überreichen. Diese zwei Redakteure des ST-Magazins haben letztes Jahr allen Personalproblemen zum Trotz massiv zur Gestaltung unseres Konkurrenzmagazins beigetragen.

Den „Jack in Bronze" für die genialste Vermarktungsidee könnte man Meinolf Schneider mit seinem Spiel OXYD zuerkennen. Das Beispiel, ein voll funktionsfähiges Programm quasi als Shareware zu verteilen, sollte in der ATARI-Welt Schule machen.

Dann gibt es noch den „gläsernen Jack" für den größten Durchblick des Jahres. Den hatte zweifelsohne die Mannschaft des Entwicklungslabors von ATARI, als Sie, angesichts der Konkurrenz, das Flaggschiff TT nun doch behelfsmäßig mit 32 Bit versorgte.

Den „eisernen" Jack möchte ich auf zwei alte Atarianer aufteilen, die immer und in jeder Lage zu Atari gestanden haben: Julian Reschke und Arnd Beissner.

Den „Black Jack" (quasi das pechrabenschwarze Gegenstück zum güldenen) wird Sam Tramiel für die am glaubhaftesten gespielte Ahnungslosigkeit gewidmet. Er hatte letztes Jahr in einem Interview mit einem deutschen ATARI-Magazin noch die Existenz eines MEGA STE mit den Worten ad absurdum geführt: „... für wen und wofür soll denn ein solches Gerät gut sein?" (Zitatende).

Dieter Kühner

I N H A L T

SOFTWARE

1st_Lock
- Der Schlüsselknecht 50

Fast Filemover
- Turbo-KJopierer a la carte 28

PKS-Shell
- Ein Schritt in Richtung UNIX 42

Relax
- Aktuelle Spiele 156

Steve
- Ein Programm für alle Fälle? 31

Tempus Word 1.0 ist da!
- Na endlich 19

HARDWARE

Etwas mehr Ruhe, bitte
- Geräuschreduzierung SH205 ohne Geld 62

Paint it Black
- Mega ST im neuen Gewand 48

ProFile T150-Streamer
- Hau´weg, die Daten 14

Videodat
- ...und der Computer sieht fern? 52

Selbst-Assemblierung (Tetra ST)
- Hast Du keinen, hol´ Dir einen... 58

PROGRAMMIERPRAXIS

Ganz kurzes Accessory - Die zweite 90

Korrekturhilfe selbstgemacht 78

Schnelle Strings 74

Speicherverwaltungen noch effizienter 93

ST-REPORT

Computer-Einsatz an einer Schule für Körperbehinderte 12

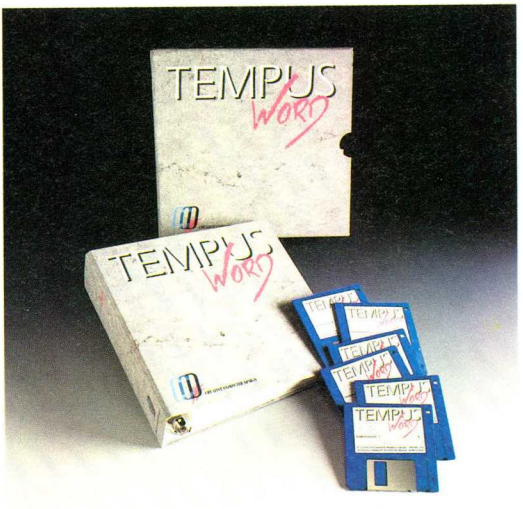


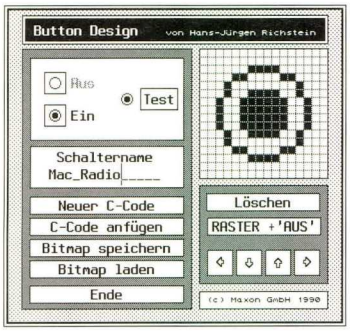
Tempus Word 1.0

Endlich da!

Das wird so mancher ausgerufen haben, als er in der ST-Computer letzten Monats die CCD-Anzeige las. In der Juni-Ausgabe '90 haben Sie deshalb bereits gelesen, was uns das Programm aus Eltville bringen wird. Seitdem verfolgen Sie bangen Herzens alle Pressemeldungen und fragen nun, ob Tempus Word der Stein der Weisen ist. Ob es **alle** Erwartungen erfüllt?

Seite 19





Button-Designer

Mittlerweile gehört es schon zum guten Ton, daß man seiner Benutzeroberfläche auch auf dem ST ein gewisses Macintosh-Flair verpaßt. Kleine Kästchen, die beim Selektieren nur vorsichtig mit einer filigranen Linie durchkreuzt werden, Radio-Buttons, die nicht - wie beim ST sonst üblich - einem Volksempfänger entlehnt scheinen, sondern eher den Miniaturbedienungselementen moderner HiFi-Anlagen entsprechen - alle Bemühungen Ataris bzw. DRIs, dem Anwaltsheer von Apple durch klobige Entstellungen etwaiger Parallelen im Userinterface zu entrinnen, scheinen nun von findigen Programmierern unterlaufen zu werden.

Seite 100

PD NEWS

Public Domain ganz groß

Vor etwa vier Jahren begann die Redaktion der Zeitschrift ST Computer, Public Domain-Programme zu sammeln und in einer eigenen Serie der Öffentlichkeit allgemein zugänglich zu machen. Inzwischen ist die Sammlung auf fast 400 Disketten angewachsen und gehört zu den meistkopierten auf dem Atari ST. Mit Beginn des neuen Jahres wollen wir in der ST Computer nun auch mehr auf Public Domain-Programme eingehen. Da in der kurzen Übersicht am Heftende viele Programme nur wenig oder gar nicht berücksichtigt werden konnten, sollen an dieser Stelle künftig die interessantesten PD-Programme näher vorgestellt werden. Dabei liegt der Schwerpunkt sicherlich auf den Neuerscheinungen, aber auch ältere Programme sollen Berücksichtigung finden. Ferner wollen wir über Updates und Tricks informieren. Schauen Sie rein ab

Seite 171

GRUNDLAGEN

Compiler-Bau - Teil 2 132

Datenstrukturen in Omikron.BASIC und Modula-2 - Teil 2 141

Den eigenen Knopf durchsetzen - Button-Design-Editor 100

Modula Marzipan Nº 4 - Felder fast grenzenlos 125

Programmer's Toolbox-Dateien - Teil 8: Die Shell oder das Kommando SSH 115

Quicktips 162

AKTUELLES

Demodisks 18

Immer up to date 180

NEWS 6

Neue Bücher 164

Sonderdisks 181

Vorschau 186

PUBLIC DOMAIN

Setz ab - Ausgeblockt 174

The Box - Die Kiste 174

Grafikzauber 172

Neue Public Domain-Disketten 178

Spooler - Drucken im Hintergrund 171

TreeShell 173

RUBRIKEN

Editorial 3

Einkaufsführer 65

Kleinanzeigen 72

Inserentenverzeichnis 184

Impressum 186

Leserbriefe 166

Rockus 76, 151, 163

NEWS

Syntex, Dulek, Augur und Imagic Wizard

Mit dem Syntex-Tool kann man jetzt Bibliotheken nachbearbeiten wie bei Augur. Schriften lassen sich aus verschiedenen Bibliotheken zusammensetzen. Schriftbibliotheken können von falschen Buchstaben befreit werden. Durch das Syntex Tool lassen sich einwandfreie, schnell arbeitende Bibliotheken erzeugen.

Vorbei ist die Zeit der falsch erkannten Zeichen! Durch zwei verschiedene Wörterbücher gleichzeitig können Sie die Wörterbücher individuell anlegen und benutzen. Dulek V2.0, so der Name des Programms, kostet DM 59,-; eine Demo ist ab DM 10,- erhältlich.

Augur, eine professionelle OCR, ist im Februar zum Sonderpreis von DM 1234,56 erhältlich. Die Aktion ist allerdings beschränkt auf den Februar.

Richter bietet auch ein neues Programm an. Imagic Wizard ist ein Filehandler, der jetzt auch für den ST angeboten wird. Mit 100 Batch-Befehle lassen sich Sicherheitskopien erzeugen, Dateien umbenennen, kopieren, sortieren, anzeigen, von, in und aus Ordnern kopieren, und all das auch noch von einem Rechner zum anderen! Die Geschwindigkeit ist höchstmöglich programmiert, der Preis: DM 89,- (Demo: DM 10,-). Ein Test folgt in einer der nächsten Ausgaben.

H. Richter
Hagener Straße 65
W-5820 Gevelsberg
Tel. (02332) 2706

Neue Designer-Fonts

Schach und Musik, zwei neue Fonts, bereichern das vielfältige Schriftenangebot im Calamus-Format von DMC. Musiker finden im neuen Designer-Font ´Musiknoten´ alle Zeichen, die für professionellen Notensatz erforderlich sind. Umfangreiche Tastaturbelegungen und Makros werden auf den beiden Disketten mitgeliefert und gehen bei der Arbeit hilfreich zur Hand. Längere Zeichen wie Legatobögen, Crescendi, Decrescendi und Triolen liegen als Vektorgrafiken vor und können beliebig positioniert werden. Damit stehen nun auch allen Notensetzern die professionellen Möglichkeiten von Calamus zur Verfügung.

Auch die Schachfreunde können ihre Spielzüge, Turnierberichte und Denkaufgaben jetzt auf Calamus der Nachwelt erhalten. Im Designer-Font ´Schach´ sind als Zeichensatz und Vektorgrafik alle Zeichen vorhanden, um Spielfelder, Figuren und Notationen zu setzen. Der Noten-Font kostet DM 198,-, der Schach-Font ist für DM 98,- zu haben. Beide Zeichensätze können direkt bei DMC bestellt werden.

DMC
Postfach 89
6229 Walluf
Tel. (06123) 71250

Arabesque Professional

Das neue Programm mit dem Namen Arabesque Professional erweitert das alte Arabesque um zahlreiche neue Funktionen, die teilweise auch in die normale Version eingeflossen sind. Einige Änderungen: Beliebig große Rastergrafiken können um 90 Grad gedreht, gespiegelt und gekippt werden. Das Laden von IMG-Dateien wurde nochmals verbessert. Die B-Spline-Funktion wurde durch eine Bézier-Funktion ersetzt. Auch im Vektorgrafikteil hat sich einiges getan. Zeichnen mit Bézier-Polygonen, die aus Bézier-Kurven, Linien und versteckten Linien bestehen - diese Funktion ist mächtig und ermöglicht neue Formen in hervorragender Wiedergabequalität. Calamus CVG-Grafiken lassen sich laden und speichern. Grafiken im GEM/3-Format ebenfalls. Objekte können automatisch mehrfach kopiert werden und dabei ihre Position und ihr Füllmuster ändern (z.B. für Verlaufsmuster und Schatten) und vieles mehr. Arabesque wird weiterhin zum Preis von DM 278,- vertrieben, während die neue Professional-Version DM 378,- kostet. Updates und Upgrades können direkt bei Shift vorgenommen werden.

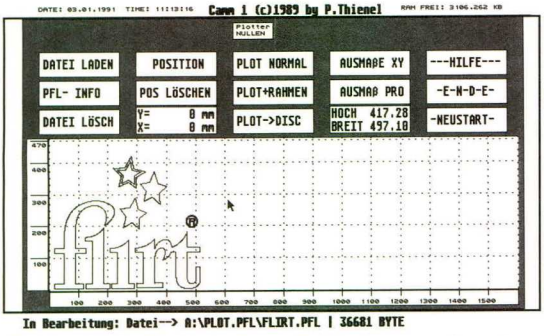
SHIFT
Unterer Lautrupweg 8
W-2390 Flensburg
Tel. (0461) 22828

Write On

Ab sofort liefert Comp auch die Textverarbeitung Write On. Installationsprogramm, Mausbedienung und umfangreiches Handbuch mit Einführung in die Textverarbeitung machen den Computer für Einsteiger einfach. Nach und nach kann man das Programm aber auch vollständig über durchdachte Tastaturkommandos bedienen. Als Clou ist es möglich, Befehle und Texte nach eigenem Bedarf auf Tasten zu legen. Bildschirmdarstellung und Drucktechnik wurden von That's Write übernommen. Der Bildschirm zeigt ständig das, was auch gedruckt wird. Write On ist ebenfalls in der Lage, druckerinterne Schriftarten und grafische Schriften in einem Dokument zu mischen. Ebenso können Signum!-Fonts verwendet werden. 10 Fonts werden mitgeliefert. Weitere Sonderschriften für NEC, EPSON, STAR, HP usw. sind erhältlich. Einbindung hochaufgelöster Grafiken, Serienbriefe, Clipboard sowie Layout-Funktionen, die man sonst nur in Programmen höherer Preisklasse findet, sind Standard. Das alles gibt's für DM 148,-. Bei wachsenden Ansprüchen ist ein Upgrade auf That's Write mit all seinen Optionen für Vielschreiber jederzeit möglich.

COMPO Software GmbH
Rotzstraße 13
W-5540 Prüm
Tel. (06551) 6266

Schneideplotter



Die Firma Rimkus bietet unter dem Namen Cutplot einen Schneide-Plotter-Treiber für Atari ST/TT an. Dieser Treiber verarbeitet alle HPGL-Dateien [auch fremder Systeme (MS-DOS)]. Geschaffen wurde der Treiber, um insbesondere die Dateien von Calamus SL und dessen HPGL-Treiber sowie Dynacadd-PFL-Dateien komfortabel bearbeiten zu können. So gehören beliebige Umbildungen, proportionale Größenänderungen nach Höhe oder Breite und freie Positionierung auch mehrerer Nutzen zum Umfang der Standardfunktionen. Die Einstelldaten lassen sich für jede Datei sichern und mit 5 Zeilen à 60 Zeichen kommentieren. Die Größe der Dateien ist nur durch den vorhandenen Speicher begrenzt. Das Programm kostet DM 99,-.

DTP-Center D. Rimkus
Ludwigstraße 38
W-8510 Fürth
Tel. (0911) 775735
Calligrapher

TT im 19"-Rack

Die Firma B. Dickers Meß- und Sicherungstechnik aus Vechelde bei Braunschweig bietet TT-Rechner im robusten und formschönen Schroff Comptec 19"-4HE-Gehäuse an. Die Vorteile liegen auf der Hand: 66 TE (von 84) für Erweiterungen (VME-Bus, Festplatten etc.), VME-IO-Bus von vornn zugänglich, geräuscharm, alle Schnittstellen an der Rückwand, Einschalten und Reset durch Schlüsselschalter an der Front, servicefreundlich, der Bildschirm kann auf dem Gehäuse stehen. Der Basispreis für den TT030/4 ohne Monitor im 19"-Gehäuse liegt bei DM 4800,-. VME-Bus-Rückwand, stärkere Netzteile etc. werden gesondert in Rechnung gestellt.

B. Dickers
Meß- und Sicherungstechnik
Wacholderweg 4
W-3303 Vechelde
Tel. (05302) 5692

Steuerlotse

Das Lohn- und Einkommensteuerberechnungsprogramm „Steuerlotse" liegt in der Version 11/90 für das Jahr 1990 vor. Es verfügt über eine GEM-ähnliche Benutzerführung und komfortable Eingabemasken mit integriertem Rechner. Die umfangreichen steuerlichen Hilfstexte lassen sich einlesen, im Programm anzeigen und ganz oder teilweise ausdrucken. Der Erwerbspreis beträgt DM 30,-.

Die doppelte FiBu „Fibumat" (V7/90) verfügt nun über eine integrierte Kassenbuchfunktion. Die Ermittlung der monatlich oder vierteljährlich anzumeldenden Umsatzsteuer gehört zu den ebenso selbstverständlichen Funktionen wie die jederzeitige Ausgabe einer Gewinn- und Verlustrechnung, Bilanz mit automatischer Kapitalentwicklung, Debitoren-, Kreditorenlisten etc. Der Erwerbspreis beträgt DM 70,-.

Beide Programme existieren auch als PC/XT/AT-Versionen und sind zur ST-Version 100% datenkompatibel.

V+G-Team GbR
Himmlingstraße 19
W-7080 Aalen
Tel. (07366) 6773

Meta-Bit

Mit „Meta-Bit" werden Vektorgrafiken im GEM-Metafile-Format in Rastergrafiken umgewandelt, die man dann in Malprogrammen punktgenau bearbeiten kann. Auch die erst in GEM/3 implementierten Graustufen werden unterstützt. Nach dem Zeichnen ist die Grafik noch manipulierbar, z.B. in ihrer Größe. Die fertige Grafik kann auf einem Epson-LQ-kompatiblen Drucker ausgedruckt oder in einem von fünf Rastergrafikformaten abgespeichert werden (Screen, Degas, IMG, PSC, Block). Meta-Bit kostet DM 34,-.

Softdesign Computer Software
Koppelweg 16a
W-3201 Diekholzen
Tel. (05121) 262007

Low-Cost-Scanner

Aufgrund der großen Nachfrage hat sich die Firma Technical Tools entschlossen, die Produktion ihrer Low-Cost-Scanner wieder aufzunehmen. Der Scanner für den ST ermöglicht es, auf einfachste Weise mit Hilfe eines gewöhnlichen Druckers beliebige Bildvorlagen in den Speicher des Rechners einzulesen, abzuspeichern und mit einem Mal- oder Zeichenprogramm (z.B. STAD) weiterzubearbeiten. Der Scanner arbeitet mit einem hochauflösenden Reflexgeber (Punktauflösung 0,19 mm) von HP. Der Preis für das gesamte Paket beträgt DM 279,-.

Technical Tools
Kobellstraße 13
W-6800 Mannheim 1
Tel. (0621) 335000

NVDI

Berichtigung zu Ausgabe 1/91

Beim NVDI-Test in Ausgabe 1/91 ist uns leider ein kleiner Fehler in einer Tabelle unterlaufen. Die ersten beiden Spalten enthielten durch einen verrutschten Tabulator falsche Werte. Deshalb drucken wir hier noch einmal die richtige Tabelle ab.

	TOS 1.4	TOS 1.4 NVDI	TOS 1.4 Turbo ST	Atari TT*	Atari TT* NVDI
Textausgabe:	182%	781%	601%	404%	2409%
Linien:	192%	398%	407%	390%	812%
Rechtecke:	518%	768%	792%	498%	1176%
Polygone:	113%	208%	173%	352%	742%
Kreise/Ellipsen:	104%	346%	119%	435%	1212%
Rasteroperationen:	726%	795%	725%	455%	881%
Attributfunktionen:	100%	335%	321%	195%	961%
Auskunftsfunktionen:	100%	280%	239%	213%	829%
Escape-Funktionen:	124%	196%	212%	272%	571%
BIOS-Ausgabe:	118%	201%	228%	234%	434%
GEMDOS-Ausgabe:	111%	631%	731%	222%	1704%
AES-Objekt-Ausgabe:	132%	295%	308%	256%	688%

Alle Messungen wurden, sofern möglich, mit eingeschaltetem Blitter durchgeführt.

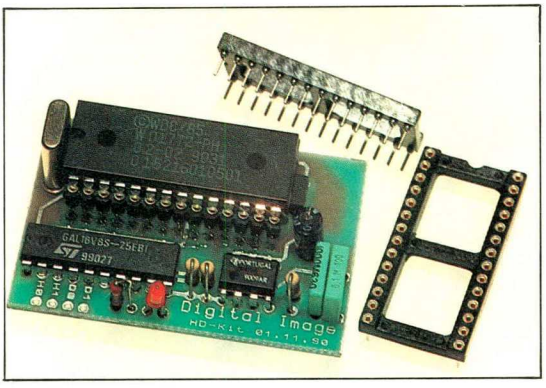
* Die Messungen auf dem Atari TT wurden alle in der hohen ST-Auflösung durchgeführt.

GTI erweitert IEC-Bus-Palette

Für alle ST-Computer (und natürlich auch für den TT) liefert GTI jetzt das IEC-Bus-Interface ACSI488. Dieses Gerät ist in einem flachen Metallgehäuse mit eigenem Netzteil untergebracht. Es wird über den ACSI-Port angeschlossen. Der ACSI-Port ist durchgeschleift, so daß Festplatten, Laserdrucker usw. gleichzeitig betrieben werden können. Zum IEC-Bus-Interface hin erfüllt ACSI488 ebenfalls die IEEE488-Norm als Controller, Talker und Listener.

Alle IEC-Bus-Interfaces von GTI haben jetzt eine funktionell einheitliche Treiber-Software erhalten, die mit dem Namen GPIB Phase 2 bezeichnet wird und gegenüber den Vorversionen komplett überarbeitet wurde. GPIB ist auch als Update zu beziehen.

GTI GmbH
Unter den Eichen 108a
W-1000 Berlin 45
Tel. (030) 8315021



HD-Kit

Ab sofort gibt es eine nochmals verbesserte Version des „HD-Kit"s von Digital Image. Neu ist das HD-Controllerboard mit eigenem Quarz-Taktgenerator und Steprate-Timer, das jetzt die Möglichkeit bietet, auch zwei HD-Laufwerke mit automatischer Density-Erkennung anzusteuern. Des weiteren wurde die Software weiter verbessert (Copy, Format, Diskmon und CookieJar) und ergänzt.

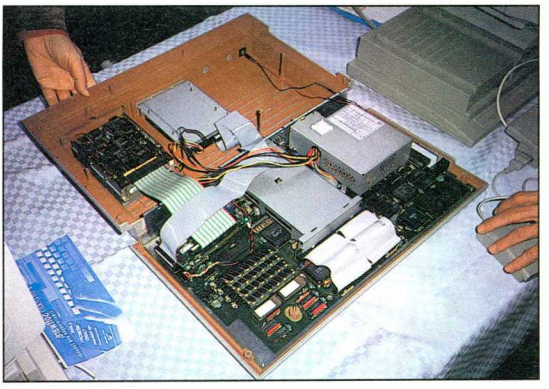
Das HD-Controllerboard wird mit selektiertem Original-WD-Controller ausgeliefert und kostet einzeln mit Software DM 198,-. Es ist natürlich auch mit anschlußfertigem Laufwerk 3,5" oder 5,25" lieferbar.

Digital Image
Postfach 1206
W-6096 Raunheim
Tel. (06134) 51706

Upgrade Megamax Modula

Ab sofort ist bei Application Systems ein Upgrade auf das Megamax Modula-2-System erhältlich. Der Compiler erzeugt jetzt wesentlich schnelleren und kompakteren Code. Die Shell bietet fast alle Funktionen des Desktops. Ein Make-Utility verwaltet die korrekte Übersetzung der Module. Ein Arithmetik-Coprozessor wird sowohl bei ST als auch bei TT unterstützt. Das System enthält einen neuen Editor, der auch Großbildschirme unterstützt. Der Sprachumfang beinhaltet die wesentlichen ISO-Erweiterungen und erfüllt vollständig die Sprachdefinition von PIM3. Megamax Modula kostet DM 398,-. Der Upgrade-Preis beträgt DM 150,- .

Application Systems
Postfach 102646
W-6900 Heidelberg
Tel. (06221) 300002



Atari Pressekonferenz

Am 12. Dezember letzten Jahres lud Atari zur Pressekonferenz nach München ein. Wie erwartet, bekam man den neuen Mega STE zu sehen. Ergänzend zu den bereits in der letzten Ausgabe genannten Features kann eigentlich nicht mehr viel genannt werden.
Den Mega STE wird es mit 1, 2 und 4 MB RAM geben. Vorerst kann allerdings nur die 4 MB-Version geliefert werden. Der Ladenpreis beträgt DM 2798,-. Der Mega STE verfügt über ein TOS 2.04 (256 k-ROMs) mit eingebautem TT-Desktop (nur der Eintrag Cache wurde beim Mega STE durch Blitter ersetzt) und neuem Kontrollfeld. TOS 2.04 wird es nicht als Software-Version geben (im Gegensatz zu einer weit verbreiteten Raubkopie). Das eingebaute 16 kB-Cache und eine softwaremäßige 16/8-MHz-Umschaltung erfolgen über ein mitgeliefertes Accessory. Der Mega STE läuft leider nur teilweise mit 16 MHz, da einige Bauteile vom bisherigen ST übernommen wurden und mit 8 MHz getaktet werden müssen. Intern verfügt der Mega STE über einen VME-Bus und eine 48 MB/28 ms-SCSI-Festplatte. Allerdings ist nur ein ACSI-Anschluß extern verfügbar, kein SCSI. Ein Sockel für einen optionalen Arithmetikprozessor ist vorhanden. Wie bereits vermutet, finden für den RAM-Speicher SIMM-Module Verwendung, so daß man ihn auch leicht selbst aufrüsten kann. Erfreulich ist auch, daß der Blechverhau des TT beim Mega STE wegfällt. Stattdessen wurde das ganze Gehäuse mit einer Abschirmpaste ausgestrichen.
Interessant auf der Pressekonferenz war auch die Bekanntgabe, daß Normen Kowalewski, einer der Programmierer des KAOS-TOS, seit 1.November bei Atari für die Systembetreuung zuständig ist. Er vertrat übrigens auf der Konferenz die Meinung, daß es unsinnig sei, ein verbessertes TOS zu programmieren, daß von Atari nicht unterstützt wird. Im Zweifelsfalle solle man sich immer an Atari wenden, die Verbesserungsvorschläge oder Fehler nach Sunnyvale zum Atari-Hauptsitz weiterleiten. Ob das in Zukunft besser klappt, bleibt abzuwarten.
Laut Alwin Stumpf, Geschäftsführer diverser europäischer Atari-Vertretungen, sind einige Fehler mit Absicht nicht behoben worden, da sich bestehende Programme auf diese Fehler stützen und sonst nicht mehr laufen würden. Ob dies der richtige Weg ist, möchten wir an dieser Stelle einmal bezweifeln.
Ab etwa der nächsten CES-Messe in den USA wird Alwin Stumpf übrigens teilweise auch bei Atari USA tätig werden. Er behält alle seine europäischen Aufgaben und wird ständig hin- und herreisen.
Angesprochen auf neue Atari-Produkte, versicherte er, daß zur CeBIT im März zwei Neuigkeiten und zur Atari-Messe Ende August eine präsentiert würden. Man darf also gespannt sein.



LIGHTHOUSE
A&G SEXTON GMBH.

PROFESSIONELL & PREISWERT

ZUBEHÖR und SOFTWARE
für Ihren ATARI™
im BÜRO oder ZUHAUSE

* ATARI ist ein eingetragenes Warenzeichen der Atari-Computer GmbH

KOSTENLOSEN KATALOG ANFORDERN

Riedstr. 2 – 7100 Heilbronn – Tel. 0 71 31/7 84 80

CeBIT '91

Für viele *das* Messe-Ereignis des Jahres und auf alle Fälle eine Reise wert. Damit Sie sich in Hannover auf der Messe ein wenig besser zurechtfinden, geben wir Ihnen hier ein paar wichtige Informationen vorab.

Messedauer:

Die CeBIT ist von Mittwoch, den 13. März bis einschließlich Mittwoch, dem 20. März täglich, auch am Wochenende, von 9 bis 18 Uhr geöffnet.

Eintrittskarten:

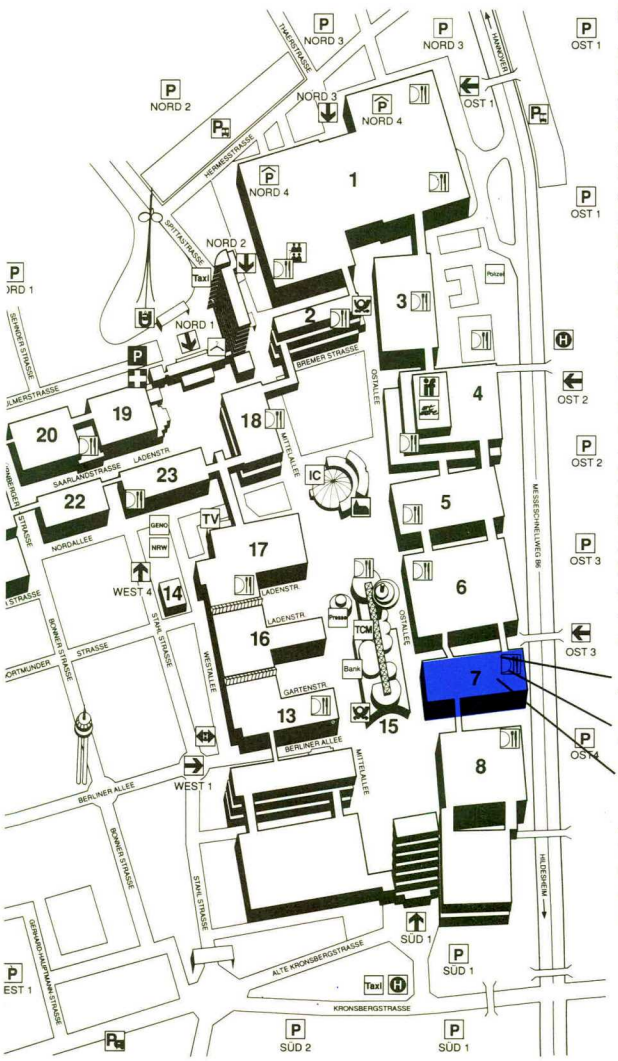
Die Eintrittspreise haben sich im Vergleich zum Vorjahr nicht erhöht. Im Vorverkauf bei Industrie- und Handels- sowie den Handwerkskammern kostet die Dauerkarte (für alle 8 Tage): DM 52,-
die Tageskarte: DM 21,-
An den Kassen des Messegeländes kostet die Dauerkarte
(für alle 8 Tage): DM 62,-
die Tageskarte: DM 26,-

SchülerInnen und StudentInnen erhalten die Tageskarte gegen Vorlage einer Immatrikulationsbescheinigung für DM 13,- an der Tageskasse. Jugendlichen unter 14 Jahren ist der Eintritt nicht gestattet.

Halle 7

Wie auch im letzten Jahr dürfte das Eldorado für Atari-Besitzer und -Anwender Halle 7 sein, denn hier finden sich die meisten Aussteller aus diesem Bereich. Auf dem Atari-Hauptstand werden, wie es schon gute Gewohnheit ist, die unterschiedlichsten Anbieter ihre Produkte ausstellen. Hier wird der europäischen Anwenderschaft zum ersten Mal der Mega STE vorgestellt.
Weiterhin finden sich in dieser Halle natürlich auch noch die Redaktion der ST-Computer und der Heim-Verlag. Eine gesonderte Liste der wichtigsten Aussteller entnehmen Sie bitte unserer Tabelle. Die Belegung des Atari-Stands war bis Redaktionsschluß leider noch nicht zu erfahren.

Aussteller	Halle/Standort
ADI Software	3 D23
Atari	7 D46
Bavaria Soft	4, 1.OG, D48
Biodata	7 D59
Brother	6 G30
C. Itoh	6 A50
Citizen	6 G06
DBP Telekom	16 B50
Dr. Neuhaus	3 C57
Elsa	23 A25
Epson	6 A20
Fujitsu	6 D48
GFA Systemtechnik	3 E39
Heim-Verlag	7 E46
Hewlett Packard	1APP, 1
Kommunikations Elektronik	16 B36
Markt & Technik	7 D34
Marvin AG	4, 1.OG, G50
MAXON	7 C42
NEC	6 H32
OKI	6 F20
Panasonic	5 A26
Seikosha	6 G16
Siemens	1 APP, 930
Star	6 C49
TKR	17 D71
Technobox	19 C50/1
TommySoftware	3 D20



Einladung

Und wenn Sie schon in Halle 7 sind, sollten Sie unbedingt den Stand von **MAXON Computer** (Stand C42) besuchen. Dort können Sie Fragen an die Redaktion der ST-Computer richten, Kritik, Anregungen oder evtl. Komplimente loswerden. Wir würden uns auch freuen, mit zukünftigen Autoren Kontakt aufzunehmen. Weiterhin werden wir natürlich unsere Soft- und Hardware vorstellen, etwa Harlekin, die MGE, den SCSI-Adapter, den GAL- und Junior-Prommer, MAXON Pascal und vieles andere. Natürlich sind auch alle Zeitschriften und Bücher vorrätig.

ATARI

MAXON computer gmbh

HeimVerlag

Auch der **Heim-Verlag** ist natürlich auf der CeBIT vertreten. Sie finden ihn ebenfalls in Halle 7 am Stand E46.

Nikolaistraße 2
8000 München 40

PRINT TECHNIK

Tel.: 089/368197
Fax: 089/399770

HANDY SCANNER PRECISION 400



Ein neuer, besonders gut zu führender Handy Scanner, der

GRAU und LINE-ART

hervorragend darstellt.

200 / 300 / 400 dpi / 32 grau

Dieses Paket kommt mit dem Malprogramm „Roger Paint" und mit Druckertreibern bis zum ST Laser.

HANDY SCANNER 105 mm **DM 498,-**

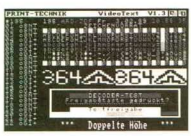
METEO-SAT-EMPFANGSANLAGE



von der Antenne bis zum Computer incl. Programm. Sie empfangen wie im Fernsehen Bilder vom Satelliten. Ideal für alle, die ständig über das aktuelle Wetter informiert werden möchten.

METEO-SAT MIT FILM SOFT, komplett **DM 2.498,-**

Videodigitizer PRO 8900 für ATARI



Der Videodigitizer PRO 8805 liefert die höchste Auflösung, die bei Verwendung einer normalen Videokamera möglich ist: 1024 Punkte in 512 Zeilen. Gleichzeitig digitalisiert er mit einer Genauigkeit von 7 bit, was einer Anzahl von 128 Graustufen entspricht. Technische Daten des PRO 8900: Bildformate: Neochrome, IMG, Doodle, Spat. Ausdruck auf: NEC P6/P7. ATARI Laser. Auflösung: 320 x 200, 640 x 200, 640 x 400, 512 x 512, 1024 x 512. Graustufen: 128 (7 bit). Anschluß: ROM-Port des ATARI ST. Eingangssignal: BAS oder FBAS. S/W und Farbmonitor

DM 498,-

Neue Colorsoft von Imagic
16 Farben aus 4096/Zusatzsoft zum PRO 8900 **DM 98,-**

PRO 8900 mit RGB-Filter + Imagic Soft
Der „Farb-Digitizer" **DM 698,-**

Realtizer für ATARI ST

Der REALTIZER ist ein in den ROM-Port einsteckbares Modul zur rasanten Digitalisierung von Videobildern aller Art. Die Auflösung beträgt 320 x 200 Punkte, wobei der Farb- und Monochrom-Modus (640 x 400) des ATARI ST unterstützt wird. Die Auflösung: 16 Graustufen. Pro Graustufe beträgt die Digitalisierungszeit 1/25 Sekunde.
Automatische Helligkeits- und Kontrastregelung

DM 148,-

RGB-Splitter

Der RGB-SW-Splitter zerlegt jedes Farb-Videosignal in seine Grundfarben Rot, Grün und Blau. Mittels Drehschalter kann jede Grundfarbe mit Schwarz/Weiß an einen Videoausgang geschaltet werden. Passend für alle Videodigitizer mit Farbdigitalisierungs-Software (z.B. PRO 8805).
Noch nie erreichte Farbbildqualität.

DM 198,-

Videotext-Decoder

Neue Generation **DM 198,-**

Zum Anschluß an den ROM-Port. Kann mit jedem Videosignal betrieben werden. Läuft auf Farb- oder S/W-Monitor. Seitenweises Aufrufen – automatisches Blättern – Seiten halten – Speichern und Laden der empfangenen Seiten im Text- oder Bildschirmformat – Textausdruck-Möglichkeit über beliebige Drucker.

OMR = OPTICAL MUSIC RECOGNITION

Paketpreis mit Scanner und Software incl. Manual **DM 2.498,-**

Ein bis zu A4 großes, bedrucktes Notenblatt wird mittels des Print Technik-Universalscanners in den Computer eingelesen. Der Computer verarbeitet das Bild und erkennt die Noten, Pausen, Zeichen etc. Das Musikstück läßt sich über ein MIDI-Keyboard sofort abspielen oder aber abspeichern und mit anderen Programmen weiterverarbeiten, z.B. C-Lab, Notator etc. Erkannte Symbole: Notensystem, Taktstriche, Taktbezeichnungen, G-Schlüssel, F-Schlüssel, Vorzeichen, alle Arten von Noten + Pausen, ganze bis 1/16 Noten, jeder Akkord, Kreuze, BE, Normal, Punkte, Doppelpunkte; Stakkato + Triolen etc.

Wir zeigen diese Neuentwicklung in Düsseldorf!

UNIVERSAL SCANNER II

FAX-SCANNER, KOPIERER, PRINTER:

Ein NEUER Universal Scanner löst die alte Generation ab. Endlich mit einem zweiten Motor versehen, stellt er das Gerät wieder in der Ausgangsposition automatisch ab. Mit SuperSoftPaket!

SCAN SOFT / FAXFUNKTION / MALPAKET / OCR 200 dpi / 16 Grau.

Eine Preis-Sensation:

FAX-PRINT-COPIER-SCANNER DM 1.898,-

NUR FÜR EXPORT oder intern-private Haustelefonanlagen. Ein Anschluß an das öffentliche Telefonnetz der Deutschen Bundespost ist in der BRD und West-Berlin nach § 15 Fernmeldeanlagen-Gesetz strafbar.



NEU

FAX-SCANNER **DM 1.798,-**

Neuer Superpreis/Neue Software **DM 2.298,-**

PROFESSIONAL SCANNER II

mit OCR-Junior inkl. Ganzseiten-Malprogramm ROGER PAINT OCR Junior, selbstlernende Schrifterkennung PEGASUS + ST 1 Raster vector Konvertierungsprogramm.
300 x 300, 300 x 600, 600 x 600 DPI-Auflösung und 64 Graustufen, einschl. Zeichenprogramm und OCR-Schrifterkennung.
Diese Scannereinheit für den Indurstrie- und DTP-Bereich stellt einen absoluten Preishit dar. Mit ihm lassen sich sowohl Halbton als auch binäre Vorlagen scannen und ablegen und mit allen auf dem Markt befindlichen Programmen (auch Calamus) weiterverarbeiten.
Das mitgelieferte Schrifterkennungsprogramm erlaubt das Umsetzen von Text in ASCII-Zeichensatz und ist durch seine Lernfähigkeit von hoher Effizienz



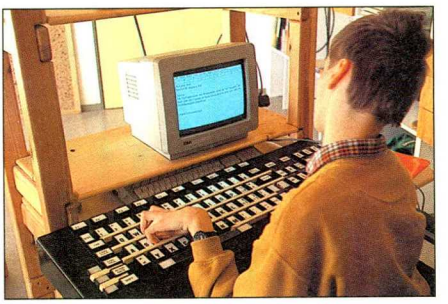
WIEDER IM PROGRAMM:

Genlock 90 für ATARI STE **DM 1.498,-**
Endlich ist es möglich, über die TV-Bilder Grafik und Schrift zu legen.

Schweiz MICROTRON 2542 Pieterlein Bahnhofstraße 2 Tel. 032/872429
Österreich 1060 Wien Stumpergasse 34 Tel. 0222/5973423 Telex 112996

Wir sind in Köln! **VISA / EUROCARD accepted**

Computer-einsatz an einer Schule für Körperbehinderte



Manfred bringt seinem Fischertechnik-Roboter gerade bei, wie er Kupferstücke umsortiert, Öztürk erstellt ein Formular für den Schülerbogen, Thomas und Christine fertigen eine technische Zeichnung mit CAD an, Sabine gestaltet mit DTP die Titelseite der neuen Schülerzeitung. Nicht unbedingt ein alltäglicher Schulbetrieb, der hier im Klassenraum der Berufsschulklasse der Schule für Körperbehinderte in Coburg abläuft - für die körperlich beeinträchtigten Schüler hat der Computer sowohl ihr Schulleben als auch ihre Zukunftsperspektive ganz wesentlich verändert.

Nach dem derzeit gültigen Lehrplan sind an der Oberstufe der Schulen für Behinderte und Kranke mindestens 40 Stunden Unterrichtszeit für die sogenannte ITG (Informationstechnische Grundbildung) vorgesehen. Schwerpunkt ist hier die Integration des Computers in die traditionellen Unterrichtsfächer.

Das ist zunächst nichts Außergewöhnliches; eine Schule für Körperbehinderte hat sich hier aber noch mit zusätzlichen Problemen und Ansprüchen auseinanderzusetzen.

So treten vielfach schon bei der Bedienung des Rechners Schwierigkeiten auf. Mehr oder weniger schwere motorische Beeinträchtigungen müssen mit speziellen Adaptionen so weit wie möglich kompensiert werden. Für den Atari ST steht hier eine Großtastatur zur Verfügung, an der Konstruktion einer Mini-Tastatur wird gerade gearbeitet. Der C64 ist über einen Fünf-Flächen-Joystick zu steuern.

Um Mobilität und optimale Nutzung der Rechner zu gewährleisten, sind sie auf fahrbaren Holztischen installiert, die im Werkunterricht von den Schülern auch selbst gebaut wurden.Der Computer wird direkt im Klassenzimmer eingesetzt - mit Blick zur Tafel beim Abschreiben.

Auch die eingesetzte Software muß unter besonderen Gesichtspunkten ausgewählt werden. Fachberater Helmut Gensler testet die immer zahlreicher auf dem Markt erscheinenden Programme auf ihre Verwendbarkeit im Körperbehindertenbereich.

Da die Lesbarkeit und das Erscheinungsbild der eigenen Handschrift häufig problematisch sind, liegt der Schwerpunkt zunächst auf Schreibprogrammen - möglichst mit klarer Benutzerführung, übersichtlicher Oberfläche und umfangreicher Rechtschreibkorrektur. Wissenschaftliche Untersuchungen bescheinigen dem Computer hier ganz wesentliche Erfolge.

Da der Leistungsstand innerhalb einer Klasse oft sehr unterschiedlich ist, bieten sich Lernprogramme zur individuellen Differenzierung an. Vor allem in den Kernfächern Deutsch und Mathematik, aber auch in einigen Sachfächern wie zum Beispiel Erdkunde liegen bereits Programme vor, die die Förderung jedes einzelnen Schülers nach seinem jeweiligen Leistungsstand ermöglichen.

Mit einem der Projekte unter Genslers Leitung, der Schülerzeitung, hat die Redaktion bereits zweimal den Wettbewerb gewonnen. Die Texte der *7 days* werden auf dem Schreibprogramm geschrieben, ins DTP-Programm *Timeworks* übertragen, wo auch über Videokamera digitalisierte Bilder eingeladen werden, und zum Schluß mit einer eigenen Druckmaschine fertiggestellt.

Die bisherigen Erfahrungen in den Klassen sind durchgehend positiv. Der Computer wirkt sehr motivierend, selbst sonst verhaltensauffällige Schüler arbeiten konzentrierter und ausdauernder, weil sie nicht „vom Lehrer gestört" werden.

Für manche Schüler - etwa mit schweren Sprachstörungen - bietet der Computer überhaupt erst den Einstieg in die „Kulturtechniken" bzw. ermöglicht Kommunikation - auch hier mit spezieller Software.

Besonders intensiv wird die Arbeit am Computer im Berufsschulbereich betrieben. Die beiden Klassenlehrer Bernhard Schneider und Helmut Franz gehen mit Blick auf die Zukunft der Schüler in den Werkstätten für Behinderte auch verstärkt auf den industriellen Einsatz von Computern ein.

Von der Programmierung und Steuerung von Fischertechnik-Modellen über CAD bis hin zu einfachen CNC-Programmierungen steht ein vielfältiges Angebot zur Verfügung, um jeden Schüler nach Möglichkeit seinen intellektuellen und körperlichen Fähigkeiten entsprechend auszubilden - natürlich auch ein gerne genutzter „Show-Effekt" für Besuchergruppen.

„Wir stehen hier aber noch am Anfang", meint Schneider, „wir haben im Moment einfach noch nicht die Ausstattung, die wir uns eigentlich wünschen; es ist aber von allen offiziellen Seiten Unterstützung signalisiert worden, auch die Schulleitung hat immer ein offenes - und gelegentlich auch notwendigerweise ein unbürokratisches - Ohr für uns." Dazu Schulleiter Horst Steinacker: „Unser wesentliches Anliegen ist das Wohl des Schülers. Dazu müssen unter Umständen auch unkonventionelle Wege beschritten werden."

Für die Lehrer nicht immer einfach: Neben der eigentlichen Unterrichtsarbeit müssen Konzeptionen und Pläne erstellt werden, die eigene Fortbildung muß in Abendkursen und - vom Schulamt immer großzügig geförderten - Betriebspraktika in Großfirmen erfolgen. „Ohne das Entgegenkommen der Firma Brose, speziell der Ausbilder in der Lehrwerkstatt, wären wir noch nicht so weit", gibt Franz zu, der auch eigene Programme in solchen Fällen mit Turbo-C erstellt, wo keine behindertengerechten erhältlich sind.

Als Fazit zieht Gensler:" Der Computer hat gerade den körperbehinderten Schülern den Weg zu einer neuen Lebensqualität eröffnet."

TEMPUS WORD

NICHT IRGEND-WANN, SONDERN JETZT!

Tempus Word® ist im Handel, wenn diese Anzeige erscheint! Ab sofort und ohne wenn und aber!

Tempus Word® steht für • innovative Absatzgestaltung • flexibles Layout • permanentes WYSIWYG • hohe Verarbeitungsgeschwindigkeit u.v.a.m.

Tempus Word® ist das Textverarbeitungsprogramm der 90er Jahre und

Tempus Word® können Sie tauschen, fragen Sie nach unseren Austauschkonditionen.

CCD · Burgstraße 9 · W-6228 Eltville · Germany
Tel. 06123/1638 od. 39 · Fax 06123/4389

DISTRIBUTOREN

Schweiz
DTZ Data Trade AG
Landstraße 1
CH-5415 Rieden/Baden

Österreich
Kneisz GmbH
Schönbrunnerstr. 110
A-1050 Wien

Niederlande
Marko Software
J.de Jong Straat 32-34
NL-3314 VB Dordrecht

United Kingdom
HiSoft
The old School Greenfield
Bedford MK 45 5DE

DEPOTHÄNDLER

COM-DATA GmbH
Am Schiffgraben 10
W-3000 Hannover

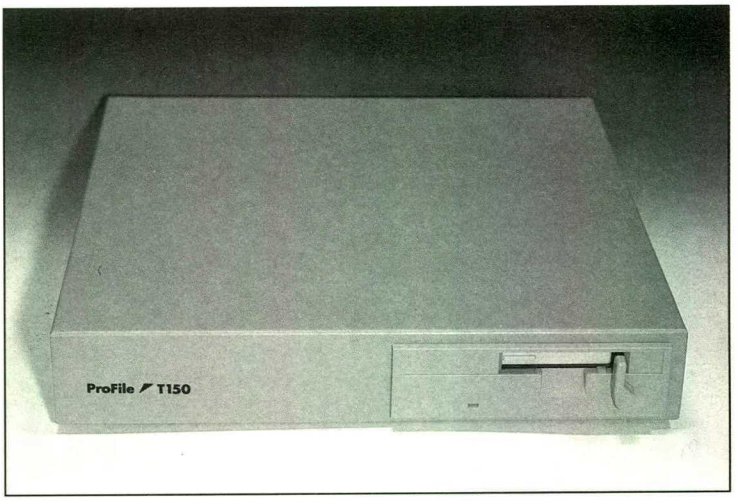
Eickmann Computer
In der Römerstadt 249
W-6000 Frankfurt/Main 90

Softhansa
Untersbergstr. 22
W-8000 München 90

CREATIVE COMPUTER DESIGN

ProFile T150

Hau weg die Daten



Vielen Anwendern stellt sich häufig das Problem, daß sie eine Sicherheitskopie einer großen Anzahl von Daten machen müssen. Da stellt sich natürlich sofort die Frage, wie diese Sicherung am besten auszusehen hat. Bei mehreren MB Daten wird die Angelegenheit allerdings schnell zum Geduldsspiel, wenn Disketten benutzt werden - ein Diskjockey ist nichts dagegen.

Deshalb gibt es Streamer. Mit solchen Cassettenspeichergeräten lassen sich Backups problemlos speichern. Diese Geräte besitzen gegenüber Disketten einen großen Vorteil. Zum einen passen wesentlich mehr Daten auf ein Streamer-Band, zum anderen ist die Speicherung der Daten um ein Vielfaches schneller. Das wird möglich durch die Art der Datensicherung und die Übertragungsgeschwindigkeit zum Streamer selbst. Unser Testgerät kommt von der Firma protar und wird in den Größen 60 und 155 MB ausgeliefert. Damit dürften selbst bei größeren Platten keine Probleme auftreten, ein vollständiges Backup auf die Cassette zu befördern. 155 MB Daten entsprechen dabei mehr als 216 Disketten und dürften auch für größere Platten vollkommen ausreichend sein.

Anschluß

Nach dem Auspacken des Geräts befinden wir uns im Besitz folgender Dinge: Streamer, Tape, Stromanschlußkabel, DMA-Anschlußkabel, Handbuch, Diskette, leerer Karton. Nach einer Untersuchung des Geräts finden wir auf der Vorderseite den Einschub für das Medium (also das Tape) und auf der Rückseite einen durchgeschleiften DMA-Bus sowie einen Druckschalter zur Einstellung der Geräteadresse. Letztere läßt sich justieren von 0 bis 7, wobei die Adresse in einem Sichtfenster angezeigt wird - sehr löblich. Das Gehäuse des Streamers ist im typischen Atari-Grau gehalten und paßt sich einer bereits vorhandenen Anlage somit bestens an. Hergestellt ist es aus Metall, wodurch die Plazierung unter einem schweren Monitor kein Problem ist.

Nach dem Anschluß geht's zum Programm, das zuerst die Registrierung durch Eingabe des Namens und der Adresse verlangt. Danach installiert es sich selbst auf der Festplatte, von wo aus es gestartet werden kann und sollte. Nachdem Letzteres ausgeführt ist, bietet sich dem Benutzer ein Bild wie in Bild 1. Hervorragend, möchte man sagen, ist hier doch ein Desktop nachprogrammiert worden! Es besitzt allerdings einen Unterschied: Die Icons sind neu, und ein Streamer befindet sich in der rechten oberen Ecke des Monitors.

Schürbel

Mein erster Gedanke („Damit kann man ja richtig arbeiten!") bewahrheitet sich schon nach einigen Sekunden, denn nach dem Anklicken eines normalen Laufwerks und darauffolgendem Selektieren einiger Dateien kann man bereits „Backup" anklikken. Bedienungsfreundlicher geht es nun wirklich nicht mehr. So werden denn meine ersten Testdaten auch auf das Band geschürbelt.

Beim ProFile T150 wird unterschieden zwischen einem File-Backup und einem Partition-Backup. Die letzte Variante ist leicht zu erklären. Hier werden alle Daten, die sich auf einer Partition befinden, einfach auf das Band geschrieben. Allerdings gibt es zwei unterschiedliche Arten, dies zu bewerkstelligen. Zum einen ist es möglich, nur die vorhandenen Daten auf den Streamer zu sichern. Arbeitet man mit einem alternativen Betriebssystem, ist das natürlich nicht möglich, da das TOS mit den Daten allein nicht zurechtkommt. Deshalb kann man auch ein „Image-Backup" machen, das alle Bits einer Partition so sichert, wie sie physikalisch auf der Platte gespeichert sind. Möchte man keine dieser Sicherungsarten benutzen, sondern es sollen beispielsweise nur bestimmte Dateien gesichert werden, wählt man das File-Backup, bei dem man die Dateien vorher selektieren kann. Die Auswahl der einzelnen Files geht dabei genauso wie beim TT vonstatten - einfach die Dateien

anklicken, fertig. Die Selektion verschwindet allerdings nicht, wenn man einen Scroll-Balken betätigt, sondern sie bleibt vorhanden - TT-like.

Schnell

Eins muß man dem Streamer lassen: Er ist schnell wie der Wind. Zum Test der Geschwindigkeit mußte meine C-Partition herhalten, in der sich 236 Dateien in 112 Ordnern befinden. Die Gesamtgröße der Dateien beträgt dabei 3.593.216 Bytes. Diese Dateien sollen also auf das Tape gesichert werden. Kein Problem, dazu sind lediglich einige Icons und ein Punkt in der Menüleiste anzuklicken. Per Partitions-Backup dauert die Aktion genau 4:15:85 Minuten. Das ergibt eine mittlere Übertragungsgeschwindigkeit von 14 kB pro Sekunde, was sich recht langsam anhört, effektiv jedoch ziemlich schnell ist. Selbiges merkt man eindeutig, sobald das soeben gemachte Backup auf die Platte zurückgespielt wird, denn dann dauert es nur noch 0:44:60 Sekunden. Das entspricht einer mittleren Übertragungsrate von 81,6 kB pro Sekunde, was sich schon besser anhört.

Nun gibt es noch die Möglichkeit, ein File-Backup (ebenfalls der gesamten Partition) zu machen, indem man alle Dateien per Hand selektiert (wofür ein Druck auf ALT-A genügt). Auch hier gibt es verblüffende Ergebnisse, denn die Zeiten sind umgedreht. Für das Backup brauchen wir nur noch 1:21:50 Minuten, einer mittleren Übertragungsrate von 44,4 kB pro Sekunde entsprechend, zum Zurückspielen jedoch wesentlich länger, nämlich genau 2:58:65 Minuten, was 20,2 kB pro Sekunde entspricht.

Die Übertragungsrate sollte man tatsächlich „mittel" nennen, denn nach dem Backup werden noch die Dateiattribute auf der Platte gesetzt, was natürlich auch seine Zeit dauert und mit dem eigentlichen Backup-Vorgang recht wenig zu tun hat. Somit kann sich die Übertragungsrate des ProFile T150 sehen lassen.

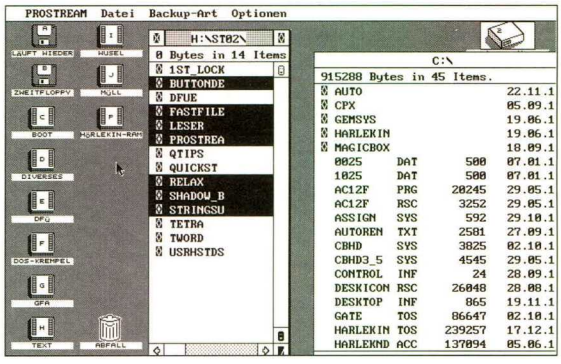


Bild 1: Die komfortable Oberfläche des ProFile-Programms.

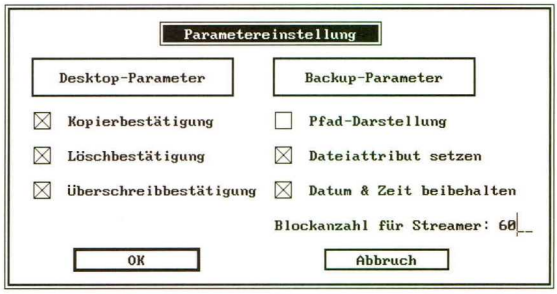


Bild 2: Hier lassen sich verschiedene Parameter einstellen.

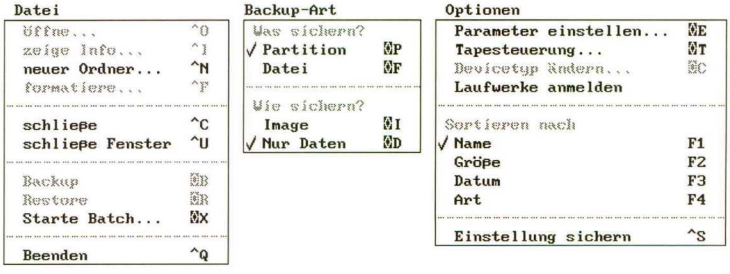


Bild 3: Eine Übersicht über die Funktionsvielfalt der Menüleisten

Batch

Batch as Batch can, so ähnlich hieß der Spruch. Die ProFile-Software kann Batch, und das nicht zu schlecht. Im Klartext heißt das: Wenn Sie beispielsweise allmorgendlich ein Backup immer der gleichen Daten machen müssen, steht dem nichts im Wege. Die Arbeit läßt sich nämlich durch sogenannte Batch-Files automatisieren. Dazu schreiben Sie einfach die durchzuführenden Aktionen in ein ASCII-Text-File und übergeben den Namen der Datei als Parameter, worauf automatisch alles abgearbeitet (und natürlich abgespeichert) wird, was Sie angegeben haben. Denkbar ist auf diese Art und Weise auch die Einbindung des Streamers in andere Programme, um automatische Backups zu machen, denn eine Batch-Datei könnte natürlich auch vom aufrufenden Programm generiert werden. Solche Anwendungen sind beispielsweise denkbar bei Abrechnungsprogrammen, großen Dateiverwaltungen, Mailboxen und vielen anderen Anwendungen mehr.

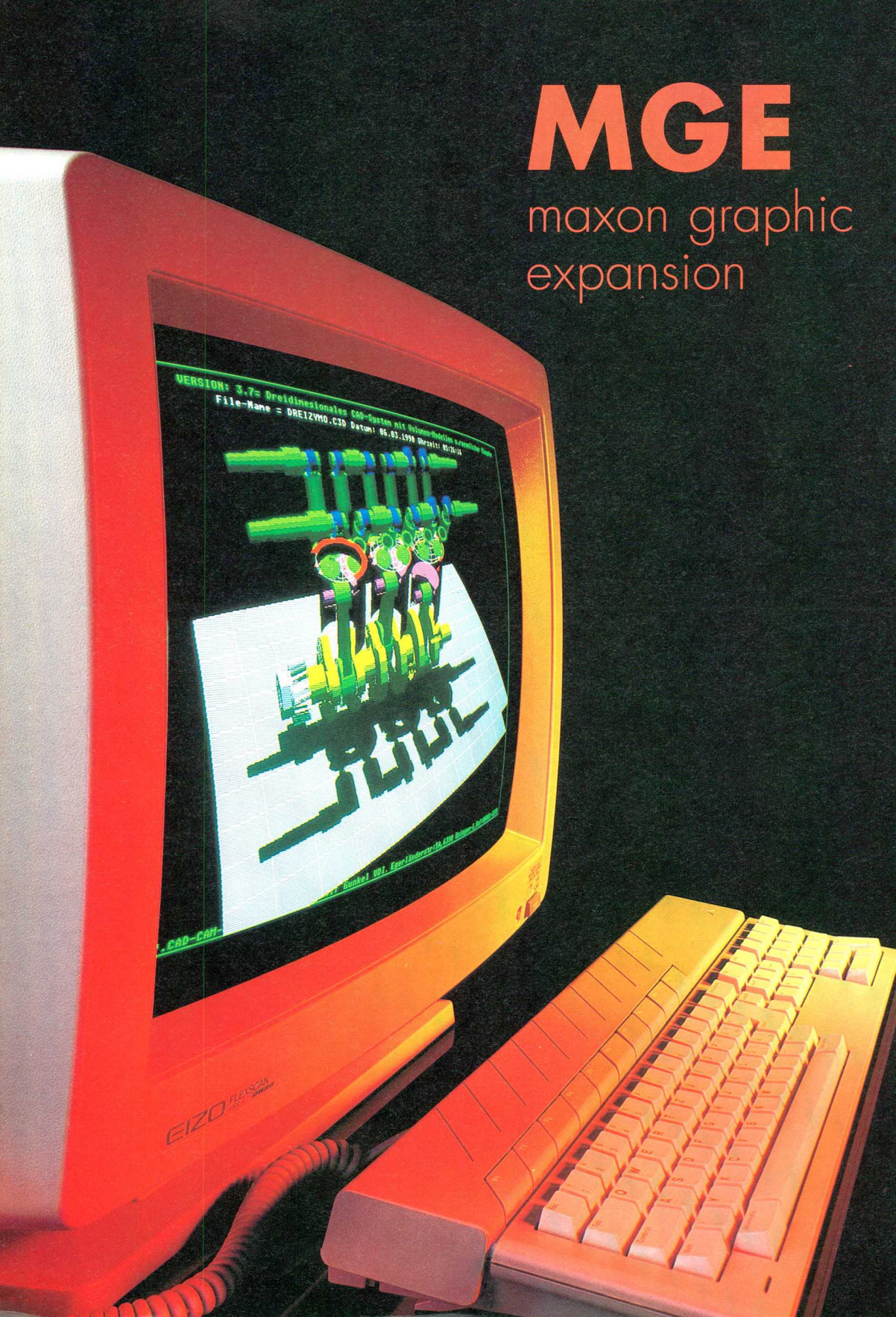
Für den Batch-Betrieb stehen die wichtigsten 14 Befehle zur Verfügung. Mit ihnen kann man die Nummer des Streamers auswählen (falls mehrere vorhanden sind), Laufwerke öffnen und schließen, Dateien nach den verschiedensten Kriterien selektieren und deselektieren, Selektionen laden und speichern und natürlich Backups machen. Einzig Restore (also das Zurückspielen eines Backups auf die Platte) läßt sich nicht im Batch-Betrieb durchführen, doch wo läge da auch der Sinn?

Optionen

Natürlich kann man mit dem Programm auch noch etwas mehr machen. Schließlich wäre es ja langweilig, wenn die Streamer-Cassette immer mit einer Stereoanlage hin- und hergespult werden müßte. Deshalb gibt es dafür eine Steuer-Dialogbox, in der man das Tape spulen kann. Mit einleuchtenden Buttons kann man das Tape vor- und zurückspulen und Infos über die

MGE

maxon graphic expansion



Die Grafikerweiterung für Profis

Vorbei sind die Tage der Eintönigkeit, die MAXON Graphic Expansion bringt Farbe ins Spiel: Gleichzeitig bis zu 256 Farben oder echte Graustufen, Auflösungen von bis zu 1664*1200 Bildpunkten, blitzschneller Bildaufbau durch einen leistungsfähigen Grafikprozessor - dies sind die Merkmale, die die MGE zur idealen Grafikerweiterung für Profis machen. Speziell ausgerichtet auf die Anforderungen kreativer Anwendungen wie DTP, CAD oder elektronischer Bildverarbeitung eröffnet sie durch ihre enorme Flexibilität neue Dimensionen der Produktivität und verwandelt den Mega ST in eine Grafik-Workstation.

Intelligenz und Stärke:

INTELs Grafikprozessor 82786 treibt die MGE zu grafischen Höchstleistungen. Die serienmäßige CLUT stellt eine Palette von 16,7 Millionen Farben zur Verfügung. Die flexible Programmierbarkeit erlaubt beliebige Bildwechselfrequenzen. Ein freier Sockel lädt zum Einsatz eines mathematischen Coprozessors ein. Die Fähigkeit zur externen Synchronisation ermöglicht den Einsatz der MGE in Video- oder Genlock-Anwendungen. Und in ihrem Drang, bis zu 2 Millionen Bildpunkte gleichzeitig darzustellen, wird die MGE meist nur vom angeschlossenen Monitor zurückgehalten.

Einsetzen und loslegen:

Durch die GDOS-VDI-Treiber sind GEM-Programme auf der MGE lauffähig. Für nonkonforme Software wird eine S/W-Großbildschirmemulation mitgeliefert. VDI-Druckertreiber und Monitorbibliotheken sorgen sich um Routinejobs. Das Kontrollfeld ermöglicht die freie Einstellung von Auflösung und Farben. Und mit den mitgelieferten Bindings für Turbo-C, Assembler und GFA-BASIC wird jeder Programmierer zum Herrn der Formen, Farben und Frequenzen.

Geprüft und für gut befunden:

So urteilt die Fachpresse über die MGE. Etwa c't 5/90: "Die konzeptionelle und praktisch realisierte Vielseitigkeit dieser Karte ... basiert auf den exzellenten Eigenschaften des eingesetzten Grafikprozessors." Ebenso COMPUTER LIVE 8/90: "Für alle Atari-Besitzer, die sich auf professionellem Niveau mit DTP, CAD oder Desktop-Video beschäftigen wollen, bietet MAXON mit der MGE II eine schnelle, extrem vielseitige und leistungsstarke Lösung." Oder TOS 6/90: "Ein Eldorado für alle programmierenden Grafik-Enthusiasten..." Auch PAGE 6/90: "Hier hebt sich positiv die MGE II von MAXON Computer hervor..." Und schließlich ST Magazin 8/90: "Fazit: Grafikkarte für professionelle DTP- und CAD-Anwendungen." Da bleibt eigentlich nur die Frage offen, was die Tester wohl zur neuen Version mit der serienmäßigen Palette von 16,7 Millionen Farben gesagt hätten.

Daten und Features:

Grafikprozessor: Intel 82768
Bildspeicher: 1024 KByte
Farbpalette: 16,7 Millionen
gleichzeitig darstellbare Farben/Graustufen: 256
Pixelfrequenzen: 13.75, 27.5, 55 und 110 MHz
Betriebsarten: Interlace/Non-Interlace

Typische Auflösungen:
Interlaced (Halbbildfrequenz):
1664 x 1200 Pixel monochrom, 88 Hz
1280 x 1024 Pixel in 16 Farben, 65 Hz
896 x 684 Pixel in 256 Farben, 66 Hz
Non-Interlaced (Vollbildfrequenz):
1280 x 960 Pixel monochrom, 65 Hz
896 x 688 Pixel, in 16 Farben, 67 Hz
640 x 480 Pixel in 256 Farben, 66 Hz

Besonderheiten: Anschlußmöglichkeit für Genlock oder externe Synchronisation, Sockel für mathematischen Coprozessor 68881

Software: Installationsprogramm, Treiber für Hardware, GDOS-VDI und Drucker, Monitorbibliotheken, Kontrollfeld als Accessory, Diverse Demos und Bilder, Programmierbibliotheken.

Preise und Systeme:
MGE II ..... DM 2398.-
MGE II mit Eizo 9070S (16" Farbmonitor) ..... DM 5498.-
MGE II mit Eizo 6500 (21" Graustufengroßbildschirm) DM 6298.-
MGE II mit Eizo 9400 (20" Farbgroßbildschirm) ..... DM 8998.-

Sehen und staunen:

Gerne führen Ihnen die folgenden Händler die MGE vor.

Ernst Brinkmann KG
Spitalerstr. 10
2000 Hamburg 1
Tel. 040/30040

Sienknecht
Bürokommunikation
Heiligengeiststr. 20
2120 Lüneburg
Tel. 04131/46122

MCC
Holzkoppelweg 19a
2300 Kiel 1
Tel. 0431/54381

PS-DATA
Hard- und Software GmbH
Faulenstr. 48-52
2800 Bremen
Tel. 0421/170577

Kurt Neumann PC
Bürger 160
2850 Bremerhaven
Tel. 0471/42006

COM-DATA GmbH
Schiffgraben 19
3000 Hannover 1
Tel. 0511/326736

3K
Computerbild GmbH
Sassenfeld 71
4054 Nettetal 1
Tel. 02153/60001

CSA Computersysteme
Hüttenstr. 56
4650 Gelsenkirchen
Tel. 0209/203420

Data Pach
GmbH & Co KG
Neuhausstr. 4
5100 Aachen
Tel. 0241/4778537

Werbestudio Abakus
Römerstr. 24
5300 Bonn 1
Tel.: 0228/635712

Lehr Bürocenter GmbH
Güterstr. 82
5500 Trier
Tel. 0651/209710

Eickmann Computer
In der Römerstadt 249
6000 Frankfurt 90
Tel. 069/763409

Hard- & Softwareservice
Christine Veigl
Adalbertstr. 61-63
6000 Frankfurt 90
Tel. 069/774043

Pauly Büromaschinen
Salzgasse 6
6250 Limburg/Lahn
Tel. 06431/50040

WAVE
Computersysteme
Südanlage 20
6300 Gießen
Tel. 0641/72357

Schreiber Computer
Rotebühlplatz 10
7000 Stuttgart 1
Tel. 0711/221996

Walliser & Co KG
Marktstr. 48
7000 Stuttgart 50
Tel. 0711/567143

Comp & Phone
Alleenstr. 66
7312 Kirchheim-Teck
Tel. 07021/3949

GCS Röseler
Jahnstr. 1
7343 Kuchen
Tel. 07331/81925

Besch & Partner
Stuttgarter Str. 53
7432 Bad Urach
Tel. 07125/8199

Erhardt Bürotechnik
Am Ludwigplatz
7500 Karlsruhe 1
Tel. 0721/16080

Jöst Computer
Im Wendelrot 5
7520 Bruchsal
Tel. 07251/103091

Computer Ludwig
City Studio
Am Rindermarkt 6
8000 München 2
Tel. 089/2609801

JOBI'S Unternehmenssupport
DV & Marketing
Fliegenstr. 12
8000 München 2
Tel. 089/2608703

Computer Corner
Landshuter Str. 4
8313 Vilsbiburg
Tel. 08741/4211

HIB Computer
DTP-Center
Äußere
Bayreuther 57a-59
8500 Nürnberg 10
Tel. 0911/995140

CCN Computer Center
Nürnberg GmbH
Gibitzenhofstr. 86
8500 Nürnberg 70
Tel. 0911/421056

Sekui & Partner GmbH
Karmelitenstr. 26
8700 Würzburg
Tel. 0931/57555

V.Willgerodt
Bürobedarfshaus
GmbH & Co KG
Hanauer Str. 12
8750 Aschaffenburg
06021/21375

Adolf & Schmoll
Computer
Schwalbenstr. 1
8900 Augsburg
Tel. 0821/ 528533

Händleranfragen erwünscht!

MAXON Computer • Schwalbacher Str. 52 • 6236 Eschborn
Tel.: 0 61 96 / 481811 • FAX: 06196/41885

gespeicherten Daten erhalten. Nötigenfalls läßt sich das Band hier auch löschen, um ein neues Backup zu erstellen.

Besitzer einer Wechselplatte dürften ebenfalls keine Probleme mit dem Programm haben, denn immer, wenn eine Platte ausgewechselt wurde, kann man den Menüpunkt „Laufwerke anmelden" anwählen. Dadurch erkennt das Programm neu angeschlossene Laufwerke bzw. neu eingelegte Wechselplatten zweifelsfrei und problemlos.

Backups können nach bestimmten Selektionen bestimmt werden. Möchten Sie beispielsweise täglich ein Backup Ihrer neuesten Daten vornehmen, ist auch das kein Problem. Dateien lassen sichnach Datum, Maske oder Attribut selektieren. Dadurch wird es natürlich auch möglich, nur von den Dateien ein Backup zu machen, die sich seit der letzten Sicherung geändert haben.

Fazit

Der ProFile T150 ist ein zuverlässiger Streamer, der durch seine gute Anwender-Software um so mehr überzeugt. Besonders die Möglichkeit der Batch-Verarbeitung ist ausgeklügelt und verspricht weitreichende Anwendungsmöglichkeiten bei hohen Datenaufkommen. Der empfohlene Verkaufspreis von DM 2198,-, für die man jedoch auch Leistung erhält, reißt allerdings ein großes Loch in die Tasche.

MP

Bezugsadresse:
protar Elektronik GmbH
Alt-Moabit 91D
W-1000 Berlin 21
Tel. (030) 3912002

DEMO DISKS

Demo-Disketten

Damit Sie nicht immer die Katze im Sack kaufen müssen, haben wir ab sofort eine neue Rubrik für Sie eingeführt; es sind Demo-Disketten kommerzieller Software. Sie kosten lediglich DM 10,- pro Diskette und können über die Redaktion bezogen werden. So müssen Sie zum Vergleich verschiedener Programme nicht an verschiedene Hersteller schreiben, sondern können sich in aller Ruhe das Demonstrationsprogramm ansehen, bevor Sie das Original kaufen.

Bitte beachten Sie, daß die angebotenen Disketten nur Demonstrationsdisketten der Originalversionen sind und somit im Gegensatz zu den Originalen in Funktion eingeschränkt sind!

Folgende Demo-Disketten sind z.Zt. erhältlich:

D1: S.&P.-Charts
Chart-Analyseprogramm
(*S.P.S. Software*)

D2: SPC-Modula-2
Modula-2-Entwicklungssystem
(*Advanced Applications Viczena*)

D3: ST-Fibu
Finanzbuchhaltungsprogramm
(*GMa-Soft*)

D4: ST-Fibu-Fakt
Fakturierungsprogramm für ST-Fibu
(*GMa-Soft*)

D5: ST-Fibu-Text
Textverarbeitungsprogramm für ST-Fibu mit Serienbrieffunktion
(*GMa-Soft*)

D6: SciGraph
Programm zur Erstellung von Präsentationsgrafiken
(*SciLab GmbH*)

D7: ST-Statistik
Uni- und multivariates Statistikprogramm, Grafikeinbindung (*SciLab GmbH*)

D8: fibuSTAT
Finanzbuchhaltungs-/Statistikprogramm
(*novoPLAN Software GmbH*)

D9: Btx/Vtx-Manager
Programm zum Anschluß an Bildschirmtext
(*Drews Btx + EDV GmbH*)

D10: Edison
Editor für fast alle Gelegenheiten
(*Kniss Soft*)

D11 & D12: CADjA
CAD-Programm für hohe Ansprüche
(*Computer Technik Kieckbusch*).
Demo besteht aus zwei Disketten zu je DM 10,-!

D13: JAMES 2.0
Programm für Börsenspekulanten
(*IFA-Köln*)

D14: Soundmerlin
Sample-Editor-Programm mit vielen Modulen
(*TommySoftware*)

D15: Soundmachine II
Programm zur Erstellung und Wiedergabe von Sounds
(*TommySoftware*)

D16: ReProK
Büroorganisationsprogramm
(*Stage Microsystems*)

D17: Sherlook
Schrifterkennungs- und -verarbeitungsprogramm
(*H.Richter*)

D18: ST Matlab
Programmiersystem mit Schnittstelle zu Modula-2
(*Advanced Aplications Viczena*)

D19: Calamus
Desktop-Publishing-Programm
(*DMC*)

D20: GD-Fibu
Finanzbuchhaltungsprogramm
(*GDAT*)

D21: Omikron.Draw!
Zeichen- und Malprogramm
(*Omikron.Software*)

D22: Omikron.Libraries
Verschiedene Libraries für Omikron.BASIC
(*Omikron.Software*)

D23: Omikron.Compiler
Demo-Version des Omikron.BASIC-Compilers
(*Omikron.Software*)

D24: Mortimer
Multi-Programm für alle Gelegenheiten
(*Omikron.Software*)

D25: Script 1
Textverarbeitungsprogramm
(*Application Systems /// Heidelberg*)

D26: SuperScore
Sequencer- und Notendruckprogramm
(*BELA Computer GmbH*)

D28: STAD 1.3+
Zeichenprogramm mit 3D-Teil
(*Application Systems /// Heidelberg*)

D29: MegaFakt
Fakturierungsprogramm
(*MegaTeam*)

D30 & D31: MegaPaint II
Zeichenprogramm mit Vektorteil
(*TommySoftware*)

D32: Tempus Word
Textverarbeitung
(*CCD*)

D33: Creator
Zeichenprogramm mit Animationsteil
(*Application Systems /// Heidelberg*)

D34: Outline Art
Utility für Calamus
(*DMC*)

D35: compugraphic Schriften
für Calamus
(*DMC*)

D36: BTX-Börsen-Manager
Börsenprogramm
(*Thomas Bopp Softwarevertrieb*)

D37: Cashflow
Kassenbuch
(*C.A.$.H.*)

D38: TiM II
Finanzbuchhaltungsprogramm
(*C.A.$.H.*)

D40: Technobox Drafter
(Zeichenprogramm spez. f. Konstruktionen)
(*Technobox*)

D41: Platon
(Leiterplatten- CAD-System)
(*VHF-Computer*)

D42: Script 2
Textverarbeitungsprogramm
(*Application Systems /// Heidelberg*)

D43: Syntex
Texterkennungsprogramm (OCR)
(*H.Richter*)

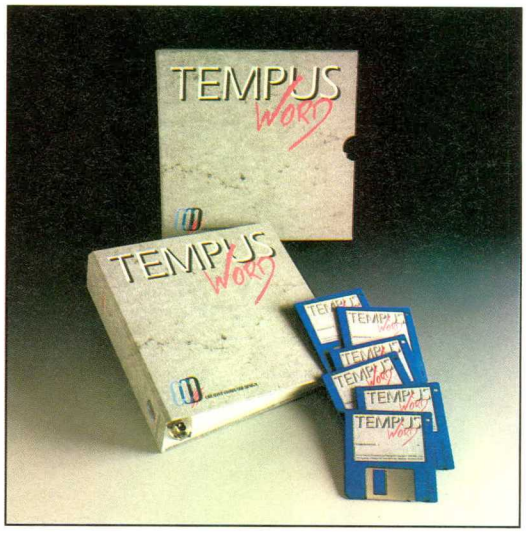
D44: Diskus 2.0
Disk-Utility
(*CCD*)

Es gelten die gleichen Vertriebsbedingungen wie für PD-Disketten (s. PD-Seiten am Ende dieser Ausgabe). Demo-Disketten können auch zusammen mit PD- und Sonder-Disketten bestellt werden.

Bitte vergessen Sie nicht die betreffende Bestellnummer (z.B. D1) anzugeben.

Tempus Word 1.0 ist da!

Na endlich!



Das wird so mancher ausgerufen haben, als er in der ST-Computer letzten Monats die CCD-Anzeige las. Selbstbewußt und im neuen Gewande präsentieren uns die Mannen aus Eltville ihre nunmehr erste endgültige Version der Textverarbeitung Tempus Word.

Ich weiß es: Sie sind ein freundlicher Mensch und haben diese Zeitschrift abonniert. In der Juni-Ausgabe '90 haben Sie deshalb bereits gelesen, was uns das Programm aus Eltville bringen wird. Seitdem verfolgen Sie bangen Herzens alle Pressemeldungen und fragen mich nun, ob Tempus Word der Stein der Weisen ist. Ob es **alle** Erwartungen erfüllt? Die Antwort lautet klar: Jein. Begeben Sie sich direkt zum letzten Absatz und ziehen Sie keine 4000 Bytes ein!

Rückblick

Noch einmal kurz zur Historie des Programms. Basierend auf dem außergewöhnlichen Erfolg von Tempus, des wohl langezeit schnellsten und umfangreichsten ASCII-Editors für den ST, begann man bei CCD zu überlegen, eine Textverarbeitung für eben diesen Rechner zu erstellen. Bedingung war natürlich wiederum maximale Performance in Geschwindigkeit und Ausstattung. Das liegt bereits sehr lange zurück. Auf der CeBIT '89, also vor nahezu zwei Jahren, konnte dann ein erster Eindruck gewonnen werden. Seitdem ist es mal ruhig, mal laut um das Produkt geworden, und die Mannen vom Rhein haben kräftig in die Tasten gehauen. Von der Version 0.91, einer Beta-Version, der noch Funktionen fehlten, haben sie 500 Stück zu einem Sonderpreis verkauft und von den Rückmeldungen der fleißigen Anwender profitiert. Aus vielen Ideen und Vorschlägen ist die nun auf den Markt gekommene V1.0 entstanden.

Neu an Tempus Word ist...

In der bereits erwähnten Juni-Ausgabe hatte ein kurzer Abriß gezeigt, daß wir es bei Tempus Word nicht mit einem 'normalen' Programm zu tun haben. Da ist zum einen die Detailwut, die in jeder Funktion steckt. Zum andern ist es die Überlegung, daß eine Textverarabeitung auch tatsächlich das Verarbeiten von Texten zulassen sollte. Denn landläufig versteht man darunter nur das Tippen - mal mehr, mal weniger komfortabel.

Drei Gedanken führen weg vom Schreibmaschinenemulator hin zur Textverarbeitung im Sinne von Tempus Word. Zum ersten ist es die vielfältige Gestaltbarkeit eines Dokuments. Dazu bietet uns das Programm:

- Seiten-Layout
- Absatz-Layout
- blockweise Formatierungen
- wechselbare Fonts
- Schriftattribute
- Grafikeinbindung
- Silbentrennung

Das sind teilweise Features, die bereits zur Überschneidung mit DTP-Programmen führen können. Zum zweiten bietet Tempus Word die Erkennung inhaltlicher Strukturen eines Textes:

- Kapitel- und Überschriftenkennung
- automatische Numerierung der Überschriften
- Gliederung
- Erstellung des Inhaltsverzeichnisses
- Erstellung des Stichwortverzeichnisses
- Fußnotenverwaltung
- Wörterbuch in deutscher Sprache

Als dritter Punkt bleibt, den fertigen Text als Basis weiterer Werke zu betrachten und nicht in den Tiefen des Massenspeichers zu vergessen. Um die Einbindung des so zu Papier und Platte gebrachten Wissens in die gesamte Arbeit am Rechner zu ermöglichen, bietet das Programm:

- Formularmodus
- Serienbrieferstellung
- Rechenfunktionen im Text
- 'versteckte' Anmerkungen im Text
- Modulschnittstelle zu Fremdprogrammen
- Stichwortverwaltung zu jedem Text

Verbunden mit weiteren Extras wie einer Dateiverwaltung, dem Wecker, dem Notizblock und dem Taschenrechner deckt Tempus Word Forderungen ab, die man an ein komplettes Textsystem stellen kann. Denkbar ist z.B. in Übersetzungsbüros der ausschließliche Einsatz dieses Programms. CCD-Geschäftsführer Beyelstein emp-

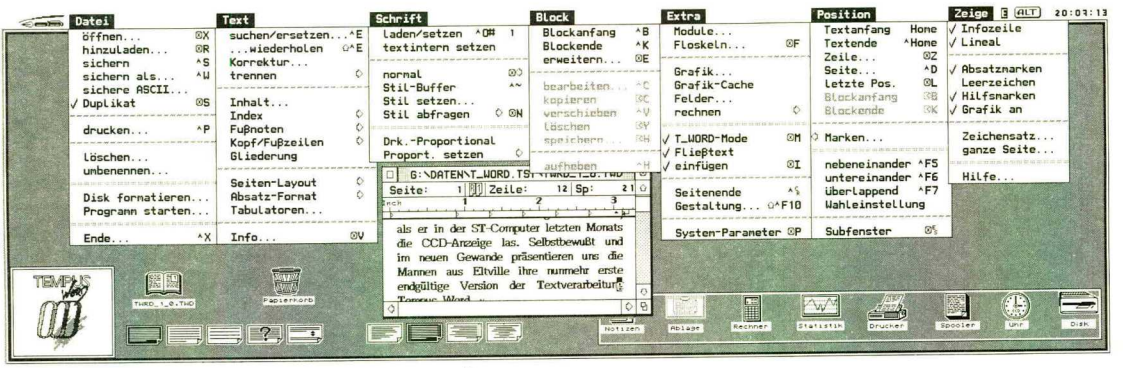


Bild 1: Tempus Word - Menüs und Desktop. Hier für die bessere Übersicht künstlich verbreitert.

fiehlt gar kleineren Betrieben, wie z.B. Handwerkern, Tempus Word als Komplettlösung für Angebote, Rechnungen und Kundenverwaltung einzusetzen.

Anatomie eines Textes

Lassen Sie mich einen Versuch wagen, die Philosophie des Programms zu schildern. Was versteht Tempus Word unter einem Text? Ein Text besteht aus Textzeichen (Buchstaben) und Grafiken sowie (Rechen-) Feldern. Geschriebener Text wird direkt im Programm getippt oder übernommen (ASCII-Format). Grafiken werden im GEM-Image Format (*.IMG) hinzugeladen. Felder werden vor dem Druck manuell oder durch eine externe Datei gefüllt.

Wo steht dieser Text? Nun, zunächst steht jeder Text auf einem Blatt Papier. Doch innerhalb der Blattfläche sind ihm bestimmte Bereiche zugewiesen: Textspalten, Kopf- und Fußbereiche. Diese Bereiche werden durch das Seiten-Layout festgelegt. Dabei läßt Tempus Word bei jedem Layout getrennte Eingaben von linken und rechten Seiten zu. Das Spiegeln oder Kopieren der Layouts von links nach rechts ist ebenfalls drin. (Bild 2). Grafiken sind nicht an diese Aufteilung der Blätter gebunden, sie können aus den Spalten herausragen.

Nachdem definiert wurde, wo der Text stehen wird, gilt es, sein 'Gesicht' festzulegen. Da gibt es zunächst globale Einstellungen, die in ihrer allgemeinen Art dem entsprechen, was wir von anderen Textprogrammen gewohnt sind: Festlegen des Zeilenabstandes, des linken und rechten Randes, des Randausgleichs (linksbündig/Blocksatz etc.). Das geschieht mit den Icons, die Sie in Bild 1 auf dem Desktop sehen. Zusätzlich jedoch kann einem Absatz in Tempus Word ein Absatzformat zugewiesen werden. Dieses Absatzformat legt die bereits genannten Parameter und noch mehr fest. Einzüge, Initialen, Attribute, Tabulatorzeilen. Jedem dieser Formate (bis zu 64) wird ein Name gegeben, und ein Absatz kann ihm zugewiesen werden. Darüber hinaus wird dem Absatz mit der Formatfestlegung eine inhaltliche Bedeutung gegeben. Dadurch wird diese Option zum mächtigen Werkzeug. Doch dazu später mehr.

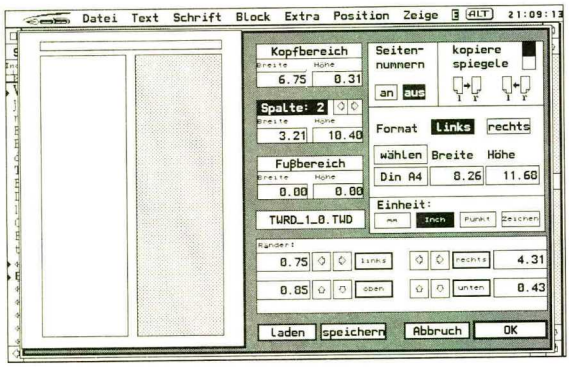


Bild 2: Festlegung des Seiten-Layouts

Weiterhin versieht man natürlich via Blockbefehle einzelne Textpassagen oder Wörter mit Attributen, Einrückungen etc. Sie vor erneutem Zeilenumbruch zu schützen, gehört ebenso dazu. Das ist bei Formeln wichtig, die sonst - mühselig gesetzt - jedesmal wieder zerstört würden.

Tempus Word benutzt Pixel-Fonts eines eigenen Formats. Jedem Font liegt ein 720-DPI-Master-Font zugrunde, den der Font-Editor in die Auflösungen der Drukker und des Monitors umrechnet. Aber auch Signum!-Fonts können direkt geladen und verwendet, allerdings nicht ediert werden. Damit stehen für Tempus Word eine große Menge bekannter und neuer Schriften zur Verfügung.

Gerade für die Optik des Block- und Mehrspaltensatzes ist eine gut funktionierende Silbentrennung nötig. Tempus Word trennt algorithmisch, verwaltet aber zusätzlich ein Ausnahmelexikon. Getrennte Passagen können nun auch wieder 'ungetrennt' werden, was für die Übernahme in andere Programme wichtig ist. Die Einstellung einer Trenntiefe habe ich vermißt.

Wie kann eine Software ohne Eigenintelligenz auf die inhaltlichen Elemente eines Textes zugreifen? Wie bereits angesprochen, werden für Absätze Gewichtungen verteilt: jeweils für Text oder Überschrift von 0 bis 9. Überschriften mit der Gewichtung 0 betrachtet Tempus Word als Kapitelüberschriften. Auf Wunsch werden sie immer oben auf rechte Seiten gesetzt, wenn nötig wird davor eine Leerseite eingefügt.

Hat man die Überschriften verschiedenen Hierarchien zugeordnet, wird es Tempus Word möglich, sie automatisch zu numerieren. Sie sehen den Effekt in Bild 3. Daß die Form der Numerierung wie Art der Zahlendarstellung, Trennzeichen etc. variiert werden können, versteht sich fast von selbst. Auch die Gliederung bedient sich der Hierarchiefestlegungen. Sie zeigt nämlich nur noch die Überschriften, die nun gegeneinander vertauscht werden können. Der an ihnen hängende Text wird mitverschoben - ähnlich dem Einklappen von Prozeduren bei GFA BASIC 3.0. Auch das Inhaltsverzeichnis, das Tempus Word auf Mausklick erstellt, hangelt sich an den so zu Überschriften erklärten Absätzen entlang. Das alles sind ungeheure Hilfen, wenn man bedenkt, wie schnell damit ein

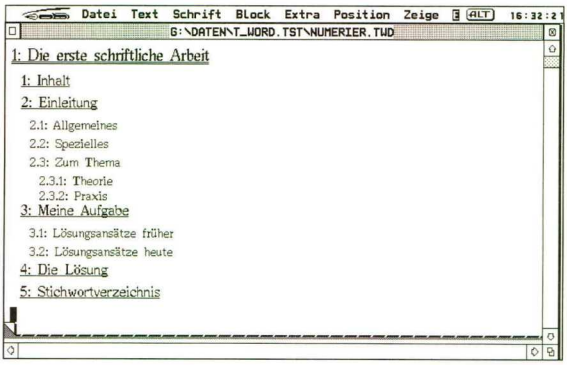


Bild 3: So sieht ein automatisch numerierter Text in der Gliederung aus.

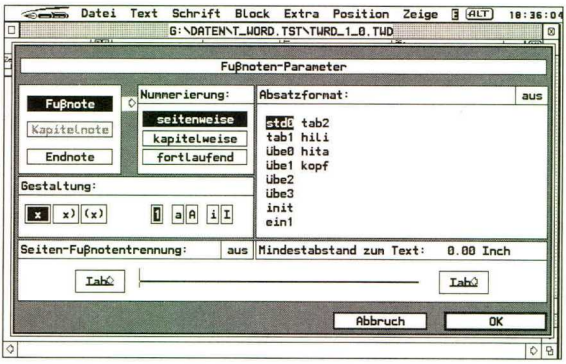


Bild 4: Lassen bei der Gestaltung keine Wünsche offen: die Fußnoten und ihre Parameter.

umfangreiches Werk umstrukturiert werden kann. Das mühselige Numerieren und Ändern des Inhaltes entfällt vollständig.

Die Vergabe von Gewichtungen für Text hat lediglich auf dessen Numerierung Auswirkung (auch normale Textabsätze lassen sich auf Wunsch numerieren). Beim Seitenumbruch macht Tempus Word keine besonderen Unterschiede zwischen den Absätzen. Hat man in Auftrag gegeben, ein bestimmtes Absatzformat zu schützen, werden die Absätze nicht durch Seitenenden getrennt. Grundsätzlich wird kontrolliert, ob nicht eine Überschrift ohne den zugehörigen Text oder nur mit einer bestimmten (natürlich einstellbaren) Zeilenanzahl auf einer Seite unten landet. Wie gesagt - die Parameter hierfür lassen nur eine globale Einstellung zu und keine für jedes Format einzeln.

Wie der Inhalt, so ist auch das Stichwortverzeichnis mit einigen wenigen Mausklicken erledigt. Es wird sauber formatiert an den Text angehängt. Allerdings ist hier Vorarbeit bereits beim Tippen nötig. Jede Passage, die aufgenommen werden soll, muß per Control-Sequenz als Stichwort indiziert werden. Die Sache hat einen Haken: Tempus Word sucht nicht im ganzen Text nach dieser Passage. Sie müssen zu Fuß auch gleiche Stellen immer wieder markieren. Logisch wäre es ebenfalls, könnte man die Indexmarkierungen der Reihe nach mittels eines Tastenbefehls anspringen. Mit Feldern und Grafiken ist das nämlich möglich. In meinen Augen unverständliche Einschränkungen.

Wie oft passiert es jedoch, daß Sie sich an einer Textstelle über 'Floppies' auslassen, aber nur das Wort 'Diskettenlaufwerk' benutzen. Wünschenswert wäre es, auch diese Passage unter 'Floppy' in den Index zu übernehmen. Das ist auch durchaus möglich. Sie schreiben 'Floppy' und markieren das Wort als verdeckt. Es wird sichtbar bleiben, jedoch nie gedruckt werden. Im Index wird dann ein Hinweis auf diese (unsichtbare) Textstelle stehen. Der unsichtbare Text ist ein nützliches Feature für alle, deren Texte am Monitor redigiert werden. Denn so finden Anmerkungen ihren Platz, ohne auf dem Papier aufzutauchen.

Inhalt

Verweise anderer Art sind Fuß-, Kapitel- und Endnoten. Wie bei nahezu allen Funktionen decken die möglichen Parameter für ihre Gestaltung auch ausgefallene Wünsche ab (Bild 4). In der vorliegenden Version 1.0 von Tempus Word sind die am Ende eines Kapitels noch nicht drin, was sicher zu verschmerzen ist. Leider zeigt sich gerade an diesem Punkt, daß die Programmierer bei CCD zum Testen ihrer zuletzt erstellten Funktionen nicht genügend Zeit hatten. So läßt der globale Umbruch des Textes die Fußnoten aus. Nach dem Reformatieren eines Absatzes hat die Fußnotenzahl ihr Attribut 'Superscript' verloren. Zwar springt das Programm aus den Fußnoten an die richtige Stelle im Haupttext. Doch von einer Fußnotenzahl zur anderen zu springen, erlaubt es nicht. Die Behebung bzw. Implementierung wurde allerdings von CCD bereits angekündigt.

Eine inhaltliche Kontrolle des Textes, wenn auch nur formaler Art, übt ein Textverarbeitungsprogramm mit Hilfe des Wörterbuches aus. Das Problem jedoch, besonders beim Einsatz in deutscher Sprache, ist die immense Größe oder mangelhafte Vollständigkeit eines solchen Wörterbuches. Da unsere Sprache Wörter zu jeweils neuen zusammenfügt und darüber hinaus durch Deklinationen und Konjugationen eine Unmenge verschiedener Endungen an die Wortstämme anfügt, stößt man leicht an die Grenzen des Rechners. Die Programmierer aus Eltville haben sich auch hier Gedanken gemacht. Das Ergebnis ist ein Hauptwörterbuch, das vom Umfang her dem Duden entspricht. Dabei ist es nur 366 kByte groß, und es wird darauf so schnell zugegriffen, daß es On-Line-Korrektur erlaubt. Sobald Sie ein unbekanntes Wort tippen, wird also gebimmelt oder gemeckert. Dann kann man das Wort nachschlagen oder sich Vorschläge geben lassen. Das dauert jedoch recht lange - 20 Sekunden muß man sich oft gedulden. Tempus Word versucht dabei, ähnlich klingende Wörter - gemäß den Regeln der Phonetik - vorzuschlagen, also Tippfehler zu erkennen. Mal abgesehen von der viel zu großen Wartezeit - daran muß noch gearbeitet werden: Zur 'Fantasie' wird die 'Phantasie' nicht gefunden.

Ein ganz anderes Problem ergibt sich dann, wenn man sich entschließt, das neue Wort ins Wörterbuch aufzunehmen. Denn im Brief an Onkel Herbert schwärmen Sie noch vom 'affenstarken' Urlaub. Das Wort landet im nun erweiterten Wörterbuch, und im Schreiben ans Finanzamt wird es folgerichtig nicht mehr angemahnt - welcher Fauxpas. Das Ende vom Lied: Sie haben ein Wörterbuch voll mit Wörtern, die nur in speziellen Zusammenhängen richtig sind, oder Sie benutzen die Korrektur gar nicht. Die Lösung nach CCD-Art: Es werden bis zu drei Wörterbücher gleichzeitig benutzt. Das erste entspricht dem Duden, das zweite soll themen- und das dritte textbezogen sein. Ein Spezialwörterbuch, das über die (auch in diesem

Blatt unvermeidliche) Computer-'Sprache' wacht, ist im Lieferumfang enthalten. Diese Lösung ist durchaus praktikabel, zumal der entsprechende Dialog eine komfortable Wörterbuchpflege ermöglicht (Bild 5).

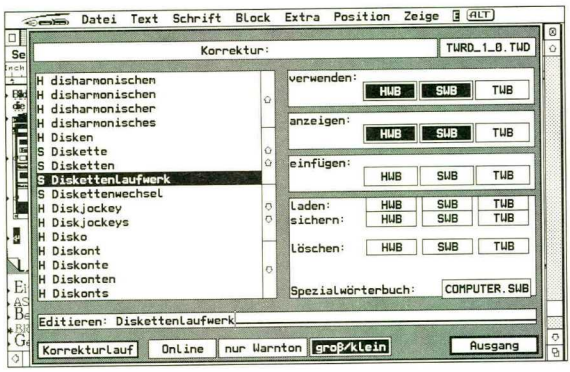


Bild 5: Hier werden Wörterbücher verwaltet und gepflegt.

Nachsorge

Neben all den Funktionen, die im weitesten Sinne der Erstellung und Verschönerung der Texte dienen, bietet Tempus Word eine Reihe Details, die der täglichen, oftmals sich wiederholenden Arbeit dienlich sind.

Da ist die Serienbrieffunktion: Im Text werden Felder definiert, die beim Druck aus einer ASCII-Datei oder dem in Tempus Word integrierten Stichwortlexikon gefüllt werden. Es ist aber auch möglich, die Felder manuell auszufüllen. Dabei sind leider nur Großbuchstaben erlaubt - eine unsinnige Einschränkung. Das Lexikon gehört zu den Helferchen, die das Programm zu einem Komplettsystem abrunden. Mit ihm lassen sich Adressen, Ideen etc. speichern und verwalten.

Definiert man ein Zahlenfeld, können dort auch Formeln eingegeben werden, die mit einfachen Rechenfunktionen andere Felder verknüpfen. Das ersetzt zwar fortgeschrittene Kalkulationsanwendungen nicht, doch sind 2574 dieser Felder erlaubt - für Rechnungen etc. wohl mehr als ausreichend.

Lädt man Tempus Word in einer Minimalkonfiguration, so verbraucht das etwas mehr als 600 kBytes des Speichers. Eine bequeme Arbeitsumgebung mit Spooler, Wörterbüchern, Fonts etc. nimmt rund 1,2 MB in Anspruch. Die Dateien, die wenigstens geladen werden müssen, sind zusammen rund 450 kBytes lang. Also ist eine schnelle Platte durchaus empfehlenswert. Logisch, daß wesentliche Erweiterungen des Programms selbst größere STs an die Grenze brächte. Daher haben die CCD-Programmierer eine Schnittstelle eingebaut, die es fremden Programmen ermöglicht, sich in Tempus Word einzuklinken. Sie haben Zugriff auf Text und Funktionen. Interessierte Software-Produzenten erhalten die lizenzfreien Unterlagen auf Wunsch. Somit wären Anschlüsse an Fakturierungen denkbar, Dokumentensysteme mit Verbindung zum CAD-Programm oder vielleicht ein FAX-Modul? Die Korrekturfunktion ist das momentan einzig verfügbare Modul. Durch einfaches Umbenennen wird es nicht mitgeladen und belastet auf diese Weise weder Zeit noch Speicher - ein Schritt hin zum 'Programm aus dem Baukasten'?

Zu einem Text gehören bei Tempus Word die Seiten- und Absatzformate sowie eine ganze Reihe von Einstellungen dazu. Auf Wunsch wird nach dem Laden auch gleich der Cursor an die letzte Position gesetzt. Bei aller Perfektion bliebe nur noch, auch die maximal 40 Such-/Ersetzeinstellungen zu speichern. Bei Formtexten geht man häufig so vor: Man definiert sie, lädt sie später, um sie entsprechend umformuliert unter neuem Namen zu sichern. Hier bietet sich das Standard-Dokument an. Beim Laden eines solchen Textes verlangt Tempus Word sofort einen neuen Namen. Verbunden mit den Druckvariablen, in die z.B. das Datum beim Drucken automatisch eingefügt wird, sind sie eine große Hilfe bei der täglichen Korrespondenz. Natürlich speichert das Programm alle Texte in einstellbaren Zeitabständen unter ihren aktuellen Namen.

Zu jedem Text verwaltet Tempus Word Informationen wie Autor, Paßwort, die Bearbeitungsdauer etc. Eine Liste von Stichwörtern gehört ebenfalls dazu. Diese sollte man als allererstes eingeben. Auf dem Desktop findet sich ein Floppy-Symbol, hinter dem sich der Dateimanager verbirgt. Hier kann nach diesen Stichwörtern oder Autoren gesucht, sortiert werden. Eine 40stellige Kurzbeschreibung wird mit angezeigt. Diese Hilfe läßt einen mit einer Platte voller Texte nicht ganz allein dastehen, sondern erleichtert die Suche nach ganz bestimmten Texten. Sicher ist an dieser Stelle noch nicht das letzte Wort gesprochen. Denn beim Laden eines Textes bietet Tempus Word die Tiefensuche. Das heißt, Sie geben einen Dateinamen ein, und ab der ausgewählten Ordnertiefe wird die Platte nach dem Text durchforstet. Doch hat das einen Haken: Sie sehen die Dateien aller darunterliegenden Ordnerebenen ja nicht. Das heißt, Sie müssen den Namen richtig eintippen. Mausklick ade! Hoffen wir, daß diese im Ansatz nützlichen Funktionen zur Datenverwaltung eines Tages zusammenwachsen. Ich könnte mir z.B. das Suchen nach Stichwörtern innerhalb aller Texte eines Ordners - ähnlich einem Hypertextarchiv - vorstellen.

Eingabe - Eingebung

Jeder gewöhnt sich nach und nach einen bestimmten Arbeitsstil an. Das gilt für jede Art von Tätigkeit, nicht nur für solche am Rechner. Dieser Stil ist aber immer von den Fähigkeiten der Werkzeuge abhängig, die man zur Verfügung hat. Wer Blöcke ausschließlich mit der Maus definieren und verschieben kann, wird einen Satz erst zu Ende formulieren, evtl. vorschreiben müssen, bevor er ihn tippt.

Die Blöcke und Blockfunktionen bieten zentrale Features bei Tempus Word. Daher möchte ich kurz darauf eingehen. Natürlich kann ein Block mit entsprechenden Menüpunkten, Control-Sequenzen oder der Maus definiert werden. Auf die so gekennzeichneten Passagen lassen sich die bekannten Funktionen wie Löschen, Verschieben, Kopieren, Speichern etc. anwenden. Attribute können gesetzt oder gelöscht werden. Hat man nur Anfang oder Ende gesetzt, läßt sich der Block auf Absatz, Kapitel, Text etc. ausdehnen. Das alles geht auch mit Tastenbefehlen.

Zusätzlich gibt es bei Tempus Word die temporären Blöcke. Diese verschwinden sofort, nachdem eine Funktion auf sie losgelassen wurde. Beispiel: Ein Wort soll gefettet werden. Doppelklick mit der Maus darauf, >F1< gedrückt, fertig. Einen Satz an eine andere Stelle schieben? >Shift< und Doppelklick, >Control V<, fertig. Wer sich daran gewöhnt hat, wird es bald nicht mehr missen wollen. Zusammen mit den Positionierbefehlen für den Cursor wie z.B: Springen wort-, absatz-, spalten-, seitenweise, ans Zeilenende, an zuvor gesetzte oder die letzte Position ermöglicht das den schnellen Umbau ganzer Texte. Wer den Überblick behalten möchte, öffnet einen Text in bis zu vier Fenstern.

Es gibt Software, die weitermacht wo andere aufhört.

tms

Studio
tms CRANACH® Studio

Das erste Programm für die echte Bildverarbeitung (EBV) in Grauton und in Farbe auf ATARI ST und TT. Lauffähig auf SM124, SM194 und allen GEM®-Farbgraphikkarten.
Über 16,77 Mio. Farbtöne und/oder 256 Graustufen. Beliebige Wandlung zwischen Grauton-, Bitmap-, Farbgraphiken; mit Vektor-Modul auch Vektorgraphiken. In der EDV-Welt einmalige Möglichkeiten. Kompatibel zu vielen Programmen (z.B. CALAMUS®, tms VEKTORST). Umtausch von tms CRANACH möglich. Weitere Informationen erhalten Sie nur bei Ihrem ATARI® EBV-Fachhändler oder direkt bei tms unter Angabe des Kennwortes 'BN113'.

8400 Regensburg
Cranachweg 4
Tel: 0941-95163
Fax: 0941-991236

CRANACH ist eingetragenes Warenzeichen der Firma tms
CALAMUS ist eingetragenes Warenzeichen der Firma DMC
ATARI ist eingetragenes Warenzeichen der Firma ATARI
GEM ist eingetragenes Warenzeichen der Firma Digital Research

MAGIC Print

Farbband-Power
von APPLE bis ZENITH

Druckertyp	*Transfer-Farbband*	*Normal-Farbband*
Citizen Swift 24	**28,70 DM**	**8,70 DM**
Star LC 10	**24,90 DM**	**7,50 DM**
Präsident 63xx	**19,90 DM**	**6,20 DM**
Mannesmann Tally	**19,90 DM**	**6,20 DM**
NEC P6/7/60/70/2/2200	**29,80 DM**	**9,90 DM**
Epson alle Typen	**28,50 DM**	**8,85 DM**

Super !!!Endlich problemloses Übertragen vom Normalpapier-Ausdruck auf Textil, Keramik, Metall, Glas und auf Kunststoff. Verschicken Sie doch ganz einfach Weihnachtsgrüße und Glückwünsche auf einem T-Shirt; originelles Eigen-Design in Top-Qualität, wasch- und kratzfest, lichtecht und lebensmitteltauglich !

AUSDRUCKEN - AUFBÜGELN - FERTIG

MAGIC-PRINT GARANTIE ALU-QUALITÄT

UNI■COMP Handels GmbH
Feldmarkring 233 D-5860 Iserlohn
Tel.: 02371/4 10 81 Fax: 4 10 83

COMTEX R.Enzesberger
Fuschl 94 A-5330 Fuschl
Tel.: 06226/616 Fax: 616

UNI■COMP Handels GmbH
Postfach1110 D-8225 Traunreut
Tel.: 08669/3 66 93 Fax: 1 26 00

Info-Telefon: Hier können Sie Info-Material und Preislisten anfordern. (Bitte Druckertyp angeben!)
Täglich (auch Sa/So) von 9 - 21 Uhr: 02371/41082 + 08669/12600 + Austria 06226/616

© fjM

Fürs schnelle Tippen bietet sich der sog. ASCII-Modus an. Zur Textdarstellung wird dann der System-Font des ST mit der ihm eigenen Übersichtlichkeit benutzt. Da Tempus Word die Absätze ständig formatiert, fängt es in der grafischen Darstellung bei kleinen Fonts und langen Absätzen schon mal an zu rucken. Zwar erhöht der variable Cursor, der immer so breit und so hoch ist wie das Zeichen unter ihm (notfalls auch kursiv), die Übersichtlichkeit, doch ist die Unproportionalität des ST-Fonts augenschonender.

Wer sich in die Software eingefuchst hat, den nennt man einen Power-User. Und der weiß ohnehin, was für Dialog-Boxen kommen werden, er wartet bei Tempus Word auf deren quälenden Aufbau. (Im Handbuch rühmen sich die Autoren, langsame Funktionen des ST durch eigene ersetzt zu haben, auch die wäre es wert gewesen.) Mag auch das Betriebssystem schuld sein, Bela NVDI wirkt hier Wunder. In den Dialogen sind die meisten Funktionen mit Alternate-Sequenzen erreichbar. Das schreit geradezu nach Makroprogrammierung - doch die fehlt noch.

Zwei ebenfalls herausragende Features sollen nicht unerwähnt bleiben: Die Seitenvorschau erlaubt in Verbindung mit einer Lupe, den Text zusätzlich manuell umzubrechen. Das gilt ebenfalls für den Umbruch von einer Spalte zur nächsten. Die Suchen und Ersetzen-Funktion sucht ihresgleichen. Mit ihr sind Suchen nach bestimmten Attributen, Fonts, Tabulatoren, ja sogar Trennungen möglich - genauso wie die Ersetzung durch ebensolche.

Gutenbergs Erben

Die Druckerei auf dem Schreibtisch; mit Tempus Word wird sie zum Dauerläufer. Im Programm integriert ist eine Druckerwarteschlange, ähnlich denen von Netzwerkprogrammen. Darin aufgereihte Dokumente werden über die drei möglichen Schnittstellen (parallel, seriell, DMA) ausgegeben. Der Druck an den drei Ports erfolgt gleichzeitig. Treiber, Spooler-Größen und Geschwindigkeiten für parallele und serielle Ausgabe sind variabel.

Erwarten Sie keine Wunder: Wer einen Rechner dermaßen auslastet, hat kaum noch Chancen, vernünftig daran zu arbeiten. Doch wenn man drauf warten kann und auf nur einen Drucker ausgibt, läßt sich passabel weitertippen. Auch beim Druck gibt es zwei grundsätzliche Modi: ASCII oder Tempus Word. Ersterer nutzt die Fonts, die die Drucker intern bieten - bei heutigen Druckern oft eine ansehnliche Zahl. Allerdings sind einige Attribute eingeschränkt: Doppelt unterstrichen werden Sie kaum beim Drucker finden. Das gilt auch für die Font-Größen. Doch lohnt sich der Druck allemal, die übertragenen Datenmengen sind wesentlich geringer als im grafischen Modus. Dabei ist aber der Druckertreiber gefordert. Eine Menge Angaben über den Drucker müssen dem Programm richtig mitgeteilt werden, damit alles klappt; vor allem die Breiten der einzelnen Zeichen. Denn damit steht und fällt die Formatierung im Blocksatz.

Entsprechend läßt sich die Bildschirmdarstellung auch von den Proportionalweiten der Tempus Word-Fonts umschalten auf die der jeweiligen Druckertreiber. Danach ist natürlich ein kompletter Umbruch des Dokumentes nötig. Mit dem beigelegten NEC P6plus-Treiber gelangen mir denn auch ansprechende Ergebnisse im ASCII-Modus. Ebenfalls dabei: Treiber für die gängigsten Brother-, EPSON-, STAR-, OKI- und HP-Drucker. Weitere Treiber sind in Vorbereitung.

Literatur

Ein solch komplexes Programm wie Tempus Word braucht eine vollständige und einfühlsame Anleitung. Obwohl man bei CCD durchaus auch Schulungen plant, wird das Handbuch in den meisten Fällen die Stütze des Anwenders bei seinen Gängen durch die Programmfunktionen sein.

Das mitgelieferte Werk richtet sich auch an den reinen Anwender - eine löbliche Einstellung. Mit ein wenig Geduld wird sich damit jeder in das neue Werkzeug Textverarbeitung einarbeiten können. Das heißt jedoch nicht, daß es nicht überarbeitenswert sei. Genervt haben mich die Unmengen an Kommafehlern.

Das Layout jedoch ist übersichtlich und gestattet schnelles Auffinden wichtiger Stellen, die meist auch entsprechend gekennzeichnet sind. Wie CCD-Chef Beyelstein versichert, ist es komplett mit Tempus Word erstellt. Die reine Textdatei allein sei 1,2 MB lang, und auf den fast 600 Seiten tummeln sich noch 500 Bilder! Das war sicher ein aussagekräftiger Beta-Test für die Software.

Doch meist steht das Manual weit weg im Schrank, und man hat gerade keine Lust, aufzustehen. Dann drücke man die HELP-Taste oder klicke über der gewünschten Funktion rechts - und in Kurzbeschreibung wird sie erläutert.

Lieferumfang

Auf den sechs Disketten, die die schick gestaltete Verpackung birgt, befinden sich nicht nur Programm, Wörterbücher, Beispieltexte und Druckertreiber. Zunächst aber gilt es, mittels Installationsprogramm die Software auf den eigenen Namen einzutragen. Eine faire Art, sich vor Raubkopien zu schützen, wie ich meine.

Auf den Disketten sind mehrere Schriftfamilien in verschiedenen Höhen und Schnitten anzutreffen. Bevor sie benutzt werden können, müssen die Fonts für Drucker und Monitor aber erst ausgepackt werden. Das erledigen kleine Programme sicher und narrensicher. Schaffen Sie vorher Platz - bei mir fehlten nach der Tempus Word-Installation 4 MB auf der Platte, davon entfielen 2 MB allein auf Fonts.

Werfen wir noch einen Blick auf die *.PRG-Dateien im Ordner 'Zusatz'. Da ist zum einen das Druckerprogramm. Die Treiber sind zwar auch mit einem ASCII-Editor zu edieren, doch müssen sie hier 'compiliert' werden, damit das Programm sie akzeptiert.

Treiber anderer Art sind die *.TAS-Dateien. Auch sie werden von einem eigenständigen Programm verdaut, bevor sie Tempus Word serviert werden. Sie legen fest, welche Control- und Alternate-Sequenzen welchen Befehlen zugeordnet werden. Sage also niemand, die Belegung sei unlogisch - sie ist zu ändern. Dummerweise schließt das die Attributbefehle aus, denn die liegen auf den Funktionstasten. Ebenfalls nicht zu erreichen: die Shift-Befehle. Im übrigen sind wirklich nur die Kommandos neu zu verteilen, die uns die Programmierer anbieten. Es gibt keine Chance, die Funktion 'Fußnote edieren' über die Tastatur aufzurufen. Der Font-Editor ist ebenfalls ein umfangreiches, externes Programm, genau wie das mitgelieferte Bildkonvertierungswerkzeug. Mit ihm können nicht nur die gängigsten Formate ins *.IMG-Format übertragen werden. Auch das Ausschneiden und Manipulieren der Bilder ist damit möglich. Obligatorisch ist das Snapshot-Programm zum Einfangen beliebiger Bildschirme.

Etwas schwerer tun sich die CCDler mit fremden Textformaten. Das mitgelieferte 'CONNY.PRG' konvertiert lediglich Wordplus-Dateien unter Beibehalt der wesentlichen Formatierungen und Attribute. Wer jedoch, wie die Tempus Word-Produzenten, andere Textverarbeitungen in Zahlung nimmt (je nach Kaufpreis gibt's bis zu 250 DM), sollte z.B. auch Signum!-Format lesen können. Schülern und Studenten werden übrigens noch andere Rabatte angeboten: Wer mindestens sechs Käufer bringt, der bekommt die Software zum halben Preis.

Wie schon erwähnt, ist Tempus Word auch auf Rechnern mit einem Megabyte Speicher lauffähig, dann allerdings nur mit gehörigen Einschränkungen. Das Doppelte verhilft zu angenehmer Arbeit. Eine Festplatte ist dringend anzuraten,

ernsthafte Anwender werden es nicht lange ohne sie aushalten. Ansonsten lautet die einzige Bedingung: Monochrombildschirm. Die Größe spielt dabei keine Rolle. Auch darf es der TT sein. Dann muß zwar ein kleines Auto-Programm (24-BIT.PRG) gestartet werden, doch die Zeichen stehen in Eltville in Richtung TT-Anpassung und Farbfähigkeit.

Vergleich

Bereits im Juni lautete eine der Kernfragen: Tempus Word = Signum!-Killer? Damals hatte das Programm aus Heidelberg noch mehr zu bieten. Mittlerweile jedoch verfügt Tempus Word über alle Fähigkeiten, die auch Signum! bietet. Wer umsteigt, wird sich sicher umgewöhnen müssen. Doch das ist normal und kein Anlaß zur Kritik. Besonders das Setzen von Formeln mit Mikroschritten haben die CCD-Leute etwas anders gelöst. Was fehlt, ist das Verschieben grafischer Ausschnitte. Bei nahezu allen anderen Funktionen bietet Tempus Word mehr oder ist schneller (wie bei Suchen/Ersetzen). Jetzt ist Applications am Zug: Wir sind gespannt auf die dritte Version von Signum!.

Ausblick

Wesentliches Argument professioneller Anwender bei der Entscheidung für oder wider ein bestimmtes Programm ist die Frage: Wird das Produkt gepflegt? Das kann man im relativ kleinen und unsicheren Atari-Markt nicht von allen Herstellern behaupten. Die niedrigen Software-Preise erlauben es den Firmen oft nicht, ständig Programmierer zu beschäftigen.

CCD ist eine der etablierten Firmen, die mit einigen Produkten seit Jahren am Markt sind. Man kann nur auf die Erfahrung und Routine der Männer und Frauen um Dirk Beyelstein setzen. In Eltville gibt man sich zuversichtlich, was Planungen für Tempus Word betrifft (TT-Version, neue Tastensteuerung ähnlich MS-Windows, Archivsystem für die Dateien). Mit diesem Programm unterstreicht CCD seine Tendenz hin zu Utility-Programmen und eben Text-Software. Ebenfalls wichtig: CCD bietet ein Sorgentelefon (deutsch: Hotline), an dem täglich ein kompetenter Mitarbeiter Auskunft erteilt.

Persönliches

Da Textverarbeiten nunmal Geschmackssache ist, erlaube ich mir, an dieser Stelle und abschließend mein ganz persönliches Bild von Tempus Word zu schildern. Früher habe ich mit Tempus getippt und mit Signum! gedruckt. Denn der Editor mit seiner Menge an Positionier- und Blockbefehlen ist für mich das richtige Werkzeug. Solange die Texte im ASCII-Format an die Redaktion gehen, ist ja alles kein Problem. Doch wenn gedruckt werden soll, die Abbildungen gar dazu sollen, dann wird's heikel. Zu deutsch: Sollte ein Text gut aussehen, wich ich auf Signum! aus.

Weil hier beides möglich ist, bin ich zum Tempus Word-Freund geworden. Im ASCII-Modus zeigt sich das Programm in nahezu identischem Gewand wie Tempus selbst. Die Geschwindigkeit ist (noch) völlig ausreichend, wenn auch Dinge wie der Dialogaufbau unbedingt beschleunigt werden müssen. Via Umschaltung in den 'Schönschriftmodus' kann ich dem Text die richtige Form geben. Gleiches gilt für den Druck: Der Probedruck kommt in hoher Geschwindigkeit aus dem P6, endgültige Qualität schafft der Laser im Grafikmodus.

Die temporären Blöcke sind eine echte Erleichterung beim schnellen Umbauen und Formatieren meiner Ideen, wenn auch hier und da noch Fehlfunktionen von mangelnder Standfestigkeit der Software zeugen. Die fehlenden Befehlsmakros und die UNDO-Funktion stehen ganz oben auf meiner Wunschliste, dicht gefolgt vom zentrierten Tabulator und einer Formelsprache. Es besteht bei CCD also durchaus noch Handlungsbedarf!

Manch einer wird einwenden, das Programm sei überfrachtet mit Funktionen. Eine gründliche Lektüre des Handbuchs vor dem ersten Programmstart ist natürlich unerläßlich. Dann kommt man mit den meisten Funktionen gut zurecht. Zumal, wenn man sich das eingangs erläuterte Konzept vor Augen hält. Für einen Brief alle 14 Tage bietet Tempus Word wahrlich zuviel. Wir haben es mit einem Werkzeug für Vielschreiber zu tun.

Viele Funktionen werde ich - außer für diesen Test - nur selten nutzen, wie z.B. das Stichwortlexikon. Andererseits konnte ich mich bisher nie mit Wörterbüchern anfreunden. Dank der schnellen Platte kann ich das Hauptwörterbuch jedoch schnell laden und hab' es immer parat. Die mittlere Zugriffszeit ist mit <ALT W> deutlich kürzer als die auf den Duden, der im Chaos meines Schreibtisches verschollen bleibt.

Seien wir nicht ungerecht: Jeder tritt mit einer Menge Erwartungen an eine Textverarbeitung heran. Eine Software geschrieben zu haben, die davon sehr viel erfüllt, trotzdem leicht zu erlernen und sicher ist und auch auf den kleinen STs läuft, das ist der Verdienst der CCD-Leute. Und dafür sind 650 DM mit Sicherheit nicht zuviel.

IB

Bezugsadresse:

CCD
Hochheimer Str. 5
6228 Eltville
Tel.: (06123) 1638

Public Domain Software für Ihren ATARI

PD Software ist in erster Hinsicht Vertrauenssache. Den nötigen Durchblick in Sachen PD vermittelt Ihnen DER Katalog, Sie erhalten ihn zusammen mit 3 prallvollen 2DD Disks mit ausgesuchter PD für nur 10.– Schein oder V–Scheck (Für 4.– erhalten Sie *nur* DEN Katalog!)

Und sonst...? liefere ich Ihnen PD zu Bedingungen, die auch Sie überzeugen werden:

- DER KATALOG ist thematisch geordnet, und enthält viele nützliche Programme, die Sie endlich auch finden können (s.o.)
- Und das ganze im lesbaren DIN A4 Format mit kartoniertem Umschlag...
- PD Disketten aus den großen Serien einzeln schon für 5.– DM, natürlich Staffelpreise
- Im Abo schon ab 3.– DM
- Schnelle Lieferung, alle Disketten virengetestet
- Thematisch geordnete PD – Pakete mit der besten PD, zu den verschiedensten Themen, z.B. Spiele, Anwendungen, Utilities.

Nicht die größte PD Sammlung, dafür aber eine der Besten ihrer Art. Lassen auch Sie sich den Katalog nicht entgehen, er wird Sie überzeugen...



Andreas Mielke
EDV Software und mehr...
Vinnhorster Weg 35
3000 Hannover 21
Tel. 05 11 / 79 41 42 (0–24h)

ADIMENS IM GESCHÄFT

ADIMENS ST plus 3.1 oder ADITALK 3.0 349.-

BUCHUNGEN 79.-
Einfache Buchführung mit Haben- und Sollkonten als Adimens Plus Anwendung. Freier Kontenrahmen und Konteninformation während der Eingabe. Ausführliche Anleitung mit Einführung in die doppelte Buchführung.

GESCHÄFT 399.-
Integrierte Geschäftsbuchhaltung für Auftragsbearbeitung und Finanzbuchhaltung als Adimens Plus Anwendung. Die Positionserfassung erlaubt freie Gestaltung, auch Einspielung aus Artikel- oder Personalstamm. Formulare (Angebot, AB, LS, Rechnung) auch mehrseitig. Automatische Buchung bei Faktura, Umsatz und Zahlungseingang. Mahnwesen. Journaldruck u.v.m.
(GESCHÄFT inkl. ADIMENS ST plus 699.-)

GESCHÄFT Demo-Disk 20.- Info anfordern!

Weitere Standardlösungen und Anfertigungen auf Anfrage.
Versand gegen Vorkasse (frei) oder per Nachnahme (plus DM 6.-)

GÜNTERBERG COMPUTERTECHNIK
FRIEDRICH-KARL-STR. 36 · 5000 KÖLN 60 · TEL. (0221) 7407194

* HCS *
macht Computerelectronic bezahlbar

Speicher - Atari - Speicher - Atari

Speichererweiterung für Atari ST 260/520/1040 steckbar

auf 2,5 Megabyte	DM 298.--
auf 4.0 Megabyte	DM 579.--
Einbaukosten	DM 50.--

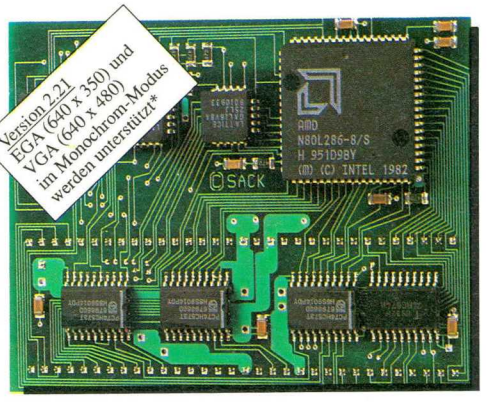
Speicherbausteine und Module

51000-70	1Mega * 1 Bit	Dil Geh.	DM 9.50
514400-80	1Mega * 4 Bit	Zip Geh.	DM 70.--
Speicher Modul 1Mega * 9 Bit Simm			DM 90.--

HCS electronic
Reichenberger Str. 15 7000 Stuttgart 80
Tel.: 0711 728 87 59 Fax: 0711 72 77 73

AT Speed

Der DOS-Emulator
mit
80 286-Prozessor



Was zählt:
ist
Leistung

EGA (640 x 350)

Grafik-Emulation im Monochrom-Modus

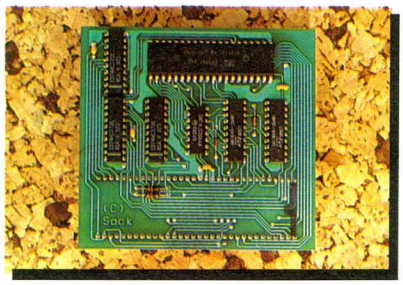
VGA (640 x 480)

Grafik-Emulation im Monochrom-Modus

Erschließen Sie sich die MS-DOS-Welt mit AT-Speed oder PC-Speed. Informieren Sie sich über die großartigen Gesamtleistungen bei Ihrem Händler. Rufen Sie uns an – wir nennen Ihnen gern Ihren Händler

PC-SPEED

Was zählt
ist
Leistung



Der DOS-Emulator
mit
NEC V30-Prozessor

UPDATE V1,5

Mit den herausragenden Leistungsdaten

1. **Nutzung als ACCESSORY bei ST's mit mehr als 1 MB-Speicher (Schneller Wechsel zwischen TOS und MS-DOS)**
2. **Unterstützung des SLM-LASERDRUCKERS**
3. **Komfortables INSTALLATIONSPROGRAMM mit GEM-Oberfläche**
4. **Verbesserte PARTITIONS-ZUORDNUNG**
5. **Die AUTOBOOT-FUNKTION wurde verbessert**

Besitzer des Update-Abo-Passes erhalten die Version V1,5 ab 15.12.90 automatisch zugesandt

MS-DOS ist eingetragenes Warenzeichen der Microsoft Corp./ IBM ist eingetragenes Warenzeichen der IBM Corp. / ATARI ST ist eingetragenes Warenzeichen der ATARI Corp.
Lotus ist eingetragenes Warenzeichen der Lotus Corp. / Alle anderen Firmen- und Produktnamen sind Warenzeichen der jeweiligen Inhaber.

Vertrieb weltweit:

Heim Verlag

Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

In Deutschland: Über 400 ATARI-Händler informieren, beraten, bauen ein, betreuen

Österreich: Darius
Inh. K. Hebein
Hartlebengasse 1-17/55
A-1220 Wien

Schweiz:Data Trade AG
Landstr. 1
CH-5415 Rieden-Baden

für alle
anderen Länder:

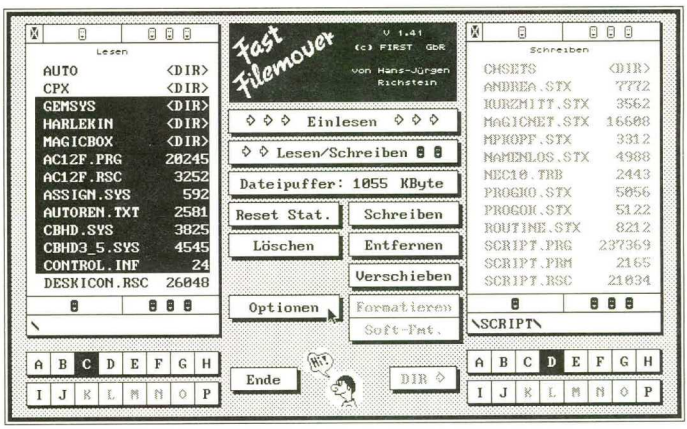
COMPO
SOFTWARE GmbH
Postfach 1051
D-5540 Prüm (FRG)
Tel.: 0 65 51 / 62 66

Händler:

**Rufen Sie an.
Wir nennen Ihnen
gern Ihren Händler**

Fast Filemover

Turbo-Kopierer à la carte



Da haben Sie den Salat: Sie sitzen vor Ihrer Festplatte und müssen 200 Signum!-Zeichensätze umkopieren. 2,6 MB von C nach I schieben. Danke schön. Was tun? Mit dem Desktop? Nein, ab sofort nehmen Sie besser Fast Filemover!

Wer tatsächlich schon megabyteweise die Daten auf seiner Festplatte herumgeschoben hat, kennt das Problem: Zwischendurch kann man sich einen Kaffee kochen oder vielleicht die neueste Folge der Lindenstraße ansehen. Das Desktop ist beim Kopieren so langsam, daß es keinen Spaß mehr macht. Wer ein Backup auf Disketten schreiben muß, hat noch mehr Probleme, denn hier dauert es noch länger!

Flupp, flupp

Von „flupp, flupp" ist also keine Spur zu sehen, eher von „schleich, kriech". Genau hier setzt der Fast Filemover der Firma First GbR an. Das Programm, das die Probleme in Wohlgefallen auflöst, läßt sich als Programm oder Accessory nutzen - das ist bereits der erste Vorteil. Als Accessory nimmt es weniger als 50 kB in Anspruch, was wohl auch bei 2MB-Rechnern noch im Bereich des Erträglichen liegt.

Machen wir doch einfach die Probe aufs Exempel und versuchen, 215 Dateien mit 2,6 MB von C nach I zu kopieren. Das Desktop, das wir ja schon zur Genüge kennen, braucht für diese Aktion üble 9 Minuten und 44 Sekunden. Fast Filemover braucht, und es stimmt wirklich, 22 Sekunden! Dieses Ergebnis ist dreifach geprüft, weil zuerst auch nicht geglaubt. Das ist über 26mal so schnell wie das Desktop bzw. eine Geschwindigkeitssteigerung um 2554 Prozent! Da kann man wirklich „flupp, flupp" sagen.

Wie geht's?

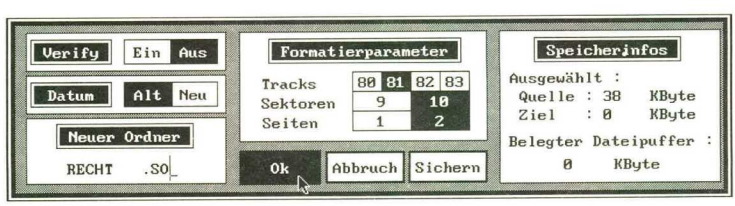
Das muß natürlich einen Grund haben. Da fragt man sich natürlich, wieso Atari das nicht gleich eingebaut hat. Nun, die Antwort auf diese Frage ist recht einfach, wenn man die Funktionsweise des Desktop-Kopierens kennt. Dabei werden nämlich alle Dateien einzeln eingelesen, in das Ziel-Directory geschrieben und die FAT (in der FAT ist vermerkt, welche Blöcke belegt und welche noch frei sind) aktualisiert - ebenfalls nach jeder Datei. Dadurch dauert es ziemlich lange, bis 215 Dateien geschrieben sind: Die FAT wird 215mal aktualisiert.

Fast Filemover macht es anders. Das Programm liest alle Dateien ein, die in den Speicher passen, und schreibt sie nacheinander auf die Platte. Erst danach wird die FAT aktualisiert. Den enormen Zeitvorteil dieser Methode können Sie an den gemessenen Zeiten erkennen. Das Verfahren hat natürlich auch einen kleinen Nachteil, denn Dateien, die nicht vollständig in den Speicher passen, können mit dem Programm nicht kopiert werden - schade, denn dies ist das einzige Manko, das ich während der Testzeit feststellen konnte. Fast Filemover ist ansonsten ein wirklich sehr gutes Programm, das ich auf keinen Fall mehr missen möchte!

Optimierung

Auch bei Disketten ergibt sich ein enormer Vorteil. Gerade bei diesen Medien kann es unheimlich oft vorkommen, daß sich die Dateien nur zerstückelt vorfinden lassen. Der Lesekopf muß von Track 1 zu Track 30, von dort nach Track 5 usw. springen. Das kommt daher, daß immer, wenn Dateien gelöscht werden, freie Bereiche entstehen, die dann wieder aufgefüllt werden müssen. Im schlimmsten Fall kann sich die Zugriffszeit auf Ihre Diskette dadurch fast verdoppeln!

Fast Filemover behebt auch dieses Manko, indem es die eingelesenen Dateien immer an einem Stück auf das Medium schreibt. Dadurch werden häßliche Lükken vermieden, und die Ladezeit für Programme verkürzt sich auf ein Minimum. Wem das Kopieren der Dateien dann im-



Hier lassen sich die Optionen einstellen.

mer noch zu lange dauert, der kann Verify ausschalten und das Fastload-Bit setzen, dadurch kann eine weitere Geschwindigkeitssteigerung erzielt werden.

Wie es sich für Disketten gehört, sind sie formatiert, wenn man sie in ein Diskettenlaufwerk einlegt. Natürlich sind Disketten genau dann nicht formatiert, wenn man sich in einem Programm befindet und eine wichtige Datei abspeichern muß. Was ist also zu tun? Natürlich: Computer über Nacht anlassen, Disk beim Händler formatieren und am nächsten Tag abspeichern. Das ist natürlich Unsinn, denn schließlich lassen sich mit Fast Filemover auch Disketten formatieren. Das aber leider nur mit einer kleinen Einschränkung, nämlich nur Disketten mit bis zu 10 Sektoren. Wieso 11 Sektoren ausgelassen wurden, ist unverständlich, aber verschmerzbar. In der Not frißt der Teufel Fliegen.

Datum von gestern

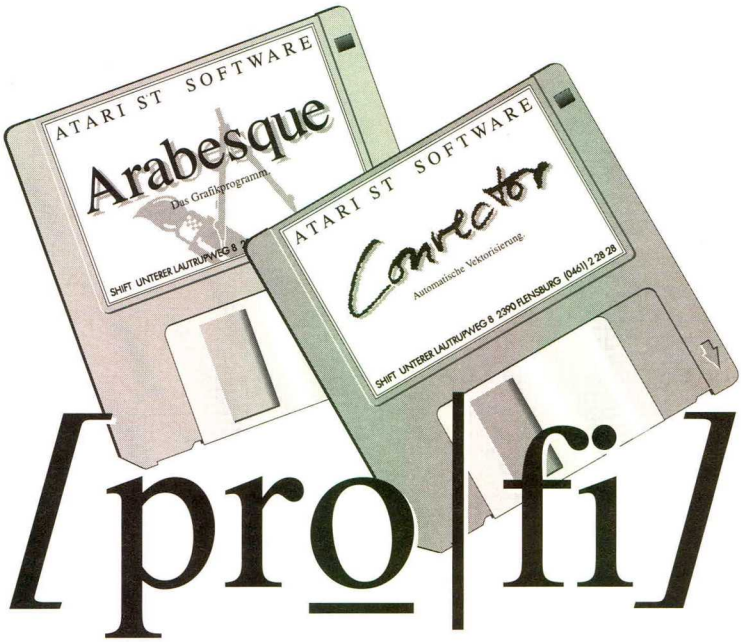
Bei älteren Desktop-Versionen war es zumeist so, daß die Dateien, die man von A nach B kopierte, ein unterschiedliches Datum trugen, nämlich das Datum des Kopiervorgangs. Dieses überaus lästige Manko läßt sich beim Fast Filemover durch einen Button abstellen - hier kann man also entweder das Originaldatum erhalten oder auch das neue einstellen, also das Datum des Kopiervorgangs.

Keine Frage, mit dem Fast Filemover macht das Arbeiten endlich wieder Spaß. Das Programm kann wirklich nur jedem empfohlen werden, der viel mit dem ST arbeitet. Dem „Gelegenheitskopierer" sollte auch die Desktop-Funktion ausreichen. Die zwei klitzekleinen Mankos (Disketten nicht mit 11 Sektoren und Speicherbegrenzung) sind leicht verschmerzbar. Auch der Preis läßt die Herzen der Anwender sicherlich höher schlagen, denn mit 59 Mark kann sich Fast Filemover sehen lassen.

MP

Bezugsadresse:

First GbR
Lorettostraße 22
4000 Düsseldorf 1
Tel. (0211) 370465



Arabesque ist durch die Tool-Box-Serie noch professioneller geworden. Ihr erstes Modul: ***Convector***, das Programm zur automatischen Vektorisierung.

Es ist durch spezielle Schnittstellen besonders für die Zusammenarbeit mit Arabesque ausgelegt und wandelt beliebige Grafiken oder Bildschirmausschnitte in Vektorgrafiken um, die dann (unter anderem) mit Arabesque nachbearbeitet werden können.

Auch von Arabesque gibt's Neuigkeiten. ***Arabesque Professional*** ist lieferbar. Die neue Pro-Version erweitert Arabesque um Bezier-Polygone und unterstützt sowohl das GEM/3 als auch das Calamus®- Format für Vektorgrafiken.

Arabesque und Convector sind die professionellen Lösungen für Atari ST und TT. Die richtige Software für Ihre Gestaltungsarbeiten. Zu einem fairen Preis.

SHIFT
UNTERER LAUTRUPWEG 8
2390 FLENSBURG
☎ (0461) 2 28 28 FAX 1 70 50

SCHWEIZ: EDV-DIENSTLEISTUNGEN
ERLENSTRASSE 73
8805 RICHTERSWIL
☎ (01) 784 89 47

ÖSTERREICH: AMV-BÜROMASCHINEN
MARIAHILFERSTRASSE 77-79
1060 WIEN
☎ (0222) 586 30 30

NIEDERLANDE: MOPRO
POSTBUS 2293
3500 GG UTRECHT
☎ (030) 31 62 47

SHIFT. Sachen gibt's...



Convector
Automatische Vektorisierung.



Arabesque
Die Grafikprogramme.



THEMADAT
Assoziative Datenbank.



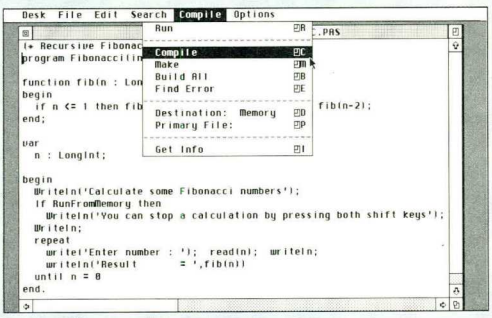
CyPress
Die Textverarbeitung.

MAXON PASCAL

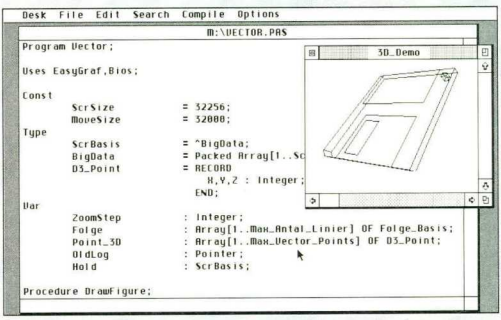
Das integrierte PASCAL-Entwicklungssystem für den ATARI ST/TT

Mit seinen Leistungsdaten und Funktionen definiert MAXON PASCAL neue Maßstäbe in der Software-Enwicklung. Herausragender Komfort, höchste Leistungsfähigkeit und Effizienz sowie größtmögliche Kompatibilität zu verschiedenen bestehenden Standards machen dieses integrierte Entwicklungssystem zu einem Meilenstein in der Software-Entwicklung.

- Der komfortable Multi-Window-Editor bietet unter anderem folgende Vorteile:
  Compiler und Linker werden vom Editor aus aufgerufen und benötigen keine Ladezeiten.
  Sämtliche Programme können wahlweise komplett im Speicher oder auch über Diskette bearbeitet werden.
  Interaktive Fehlererkennung (Compiler- und Programmfehler).
- Der extrem kompakte und schnelle Single Pass-Compiler belegt wenig Speicherplatz. Compilierungsgeschwindigkeit: mehr als 20.000 Zeilen/min. auf einem normalen ST.



- Extrem hohe Ausführungsgeschwindigkeit der fertigen Programme durch effektive Code-Optimierung.
- MAXON PASCAL arbeitet mit UNITs, wodurch extrem schnelle Durchlaufzeiten des Compilers realisiert werden können.



- Das UNIT-Prinzip ermöglicht den einfachen modularen Aufbau von Programmen. Oft benötigte UNITs können zu Libraries zusammengefaßt und resident im Speicher gehalten werden, wodurch beim Compilieren extreme Zeitvorteile entstehen.
- Integrierter Inline-Assembler
- Weitgehende Kompatibilität zu Turbo Pascal 5.0 auf dem PC.
- Die Übernahme von ST-Pascal-Programmen wird durch eine spezielle UNIT unterstützt.
- Schnelle, IEEE-kompatible Floating-Point-Library für Fließkommaarithmetik (Single, Double, Ext. Precision).
- Mitgeliefert wird eine Command-Line-Version, die das Compilieren über Batch-Dateien ermöglicht, sowie eine 68881-kompatible Floating-Point-Version.
- Eine komplexe und funktionsabhängige Hilfefunktion unterstützt den Anwender in Fragen auf die Pascal-Syntax, den Editor und die generelle Bedienung.

MAXON PASCAL ist zum unverbindlich empfohlenen Verkaufspreis von DM 259.- erhältlich.
Gerne nennen wir Ihnen einen Händler in Ihrer Nähe oder senden Ihnen weitere Informationen zu.

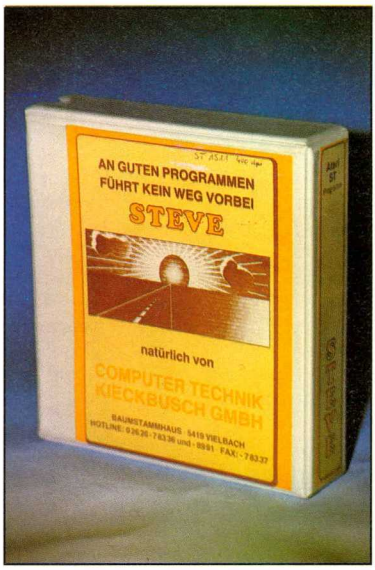
MAXON computer gmbh

MAXON Computer GmbH • Schwalbacher Straße 52 • D-6236 Eschborn
Tel. 06196/481811 • Fax 06196/41885

Erwähnte Computer- und Softwarebezeichnungen sind Handelsmarken und/oder Warenzeichen der betreffenden Hersteller.

STEVE

Ein Programm für alle Fälle?



Das Problem kennen alle Computerneulinge: Da hat man sich diverse Computerzeitschriften besorgt (allein im deutschsprachigen Raum gibt es derzeit 80 regelmäßig erscheinende Titel), einige Fachhändler konsultiert (die leider oft sehr herstellerabhängig argumentieren) und vielleicht auch mal eine Messe besucht (am besten die CeBIT, denn dort ist das Angebot geradezu unüberschaubar geworden), und das alles nur, weil man sich einen Computer kaufen wollte. Wenn nun das gute Stück Hardware zu Hause steht, hören damit die Anfängerprobleme nicht zwangsläufig auf - ganz im Gegenteil: Dann gilt es zu entscheiden, welche Software für welchen Zweck und mit welchen weiteren Peripheriegeräten anzuschaffen ist. Wie gut ist die Software, brauche ich das alles, was sie mir bietet, oder kann sie etwa nicht all das, was ich mir wünsche?

Gerade die Zeit vor dem Jahreswechsel ist prädestiniert für solche Fragen, denn für den Hobbyeinsteiger steht das Weihnachtsfest als magisches Datum im Raum, und der Gewerbetreibende hat triftige Gründe, sich in den letzten Wochen des alten Jahres (oder manchmal auch in den ersten des neuen) Zeit für Überlegungen bezüglich Computereinkauf zu nehmen. EDV-Händler haben dann Hochsaison, das berühmte Jahresendgeschäft bricht herein (im wahrsten Sinne des Wortes).

Professionelle EDV-Berater gehen bei den (weiter oben) genannten Fragen ihrer Klientel längst nicht mehr nach dem Schema vor: Sage mir, welche Hardware Du hast, und ich sage Dir, welches Programm darauf läuft. Für einen seriösen Berater steht das Kundenproblem im Vordergrund, das unmittelbar mit der Frage nach der passenden Software verknüpft ist. Die Suche nach der richtigen Hardware ist (glücklicherweise) zweitrangig geworden.

Dann gibt es auch einige Zeitgenossen, die gleich alles mit einem Computer lösen wollen, was ja nicht unbedingt falsch sein muß. Der Verfasser dieser Zeilen kennt Beispiele dieser Art aus seiner Zeit als EDV-Fachhändler zur Genüge. Da gibt es potentielle Kunden, die einen riesigen Wunschzettel in Sachen Software mit sich herumtragen: Also Textverarbeitung soll's auf jeden Fall sein, denn damit schafft man den leichtesten Einstieg in die Computerei (sagt man). Dann wäre eine Datenbank nicht schlecht (hört man), weil ja die Schallplattensammlung riesig geworden ist. Wenn ich dann noch das Haushaltsgeld kalkulieren könnte und auch mal Zeit für auflockernde Bildchen für die nächste Geburtstagseinladung hätte, das wäre toll (meint man). Vielleicht mache ich dann auch gleich noch die Mitgliederzeitung vom Taubenzüchterverband, das soll ja mit 'Desktop Publishing' einfach und leicht zu bewerkstelligen sein (glaubt man).

Fürwahr, den Anwendungen auf dem Computer sind kaum noch Grenzen gesetzt, wenn es da nicht die maximal mögliche Ausdehnung des lieben Geldbeutels, „DMmax" genannt, gäbe. Denn die Wunschvorstellungen bezüglich Software haben die unangenehme Eigenschaft, daß sie reichlich Geld kosten. Gleichsam muß oft den Wunschvorstellungen der Nutzer folgend, für jede Anwendung (siehe oben) ein eigenes Programm angeschafft werden.

Zahlreiche Programmentwickler hat diese Tatsache nicht ruhen lassen. So wollen sie möglichst viele Programmlösungen in ein Paket zusammenfassen und „DM max" dabei möglichst gering halten. Der Trend geht mittlerweile dahin, möglichst viele unabhängige Funktionselemente aus den Bereichen Textverarbeitung, Datenbank, Grafik, Tabellenkalkulation usw. zu einer „Allround-Lösung" zusammenzuschnüren, „Integrierte Pakete" genannt.

Hallo, ich bin STEVE!

Seit 1986 gibt es für den Atari ST ein Programmpaket, das völlig zu Unrecht etwas ins Abseits geraten ist (einige Gründe dafür werden wir gleich noch betrachten): STEVE, zusammengestellt unter Federführung von Primoz Jakopin, wohnhaft in der Gegend von Ljubljana, Jugoslawien, vertrieben von der Firma Kieckbusch. STEVE ist seinerzeit angetreten, für den Atari ST genau das nachzuvollziehen, was bekannte Namen aus der MS-DOS-Welt ebenfalls versucht haben: FRAMEWORK, Open Access, Works usw., allesamt integrierte Pakete.

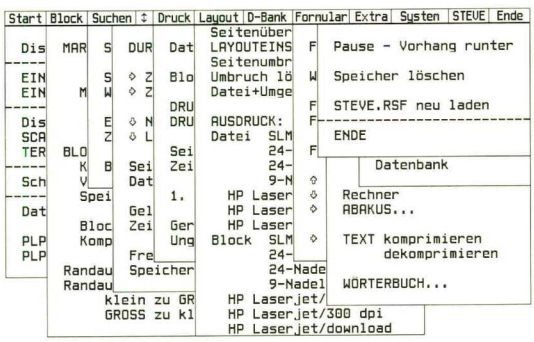
STEVE bewegte sich ständig im Schatten anderer Namen. Nie hat es die Popularität erreicht, die andere in der Szene genießen. Ein Grund dafür könnte sein, daß STEVE keine GEM-Oberfläche aufzuweisen hat. Gerade in einer Zeit, wo von grafischen Benutzeroberflächen überall

geschwärmt wird, fehlen die typischen Merkmale von GEM. Dies macht sich beim Einstieg in das Programm bemerkbar. Aber dennoch hat STEVE eine Pull-Down-Menüleiste. Sie wurde von den Entwicklern völlig losgelöst von den GEM-Konventionen aufgebaut, man hat hier das Rad (gemeint ist die GEM-Oberfläche) ein zweites Mal erfunden. Ungewöhnlich ist auch die Art und Weise, wie diese Menüleiste aktiviert werden muß. Drei Wege gibt es. 1.: Man fährt mit dem Cursor über den oberen Bildschirmrand hinaus bis er verschwindet und fährt viermal hin und her (merkwürdig). 2.: Man strebt mit dem Cursor die linke oder rechte obere Bildschirmecke an und fährt über diese Ecke hinaus. 3.: Einfaches Drücken der HELP-Taste.

gang weiterlaufen. Noch interessanter: der Menüpunkt *Zu Position* (siehe Bild). Es erscheinen am rechten Bildrand zwei Säulen, von denen die linke die gegenwärtige Auslastung des Arbeitsspeichers anzeigt. Das ist sehr wichtig, wenn ein Hinzuladen von weiterem Text beabsichtigt ist. Der rechte Balken stellt die relative Gesamtlänge des Textes dar, die gegenwärtige Cursor-Position im Text sowie eine Wahlmöglichkeit zum Anspringen einer neuen Cursor-Position. Meine Wertung: sehr nützlich! Darüber hinaus sind solche Steuerungen auch über CTRL-Sequenzen möglich (darüber später mehr).

STEVE ist also in erster Linie ein Textverarbeitungsprogramm. Zu den größten Vorzügen des Programms gehören einfache, schnelle Blockoperationen. Um Markierungen der Blockbegrenzung zu setzen, gibt es ein eigenes Menü: Block. Am einfachsten lassen sich diese Blockdefinitionen mit Hilfe von Befehlstasten im Ziffernblock durchführen. Innerhalb eines Blocks sind dann Randausgleich mit Trennvorschlägen und Flattersatz (quasi das Gegenteil vom Randausgleich) möglich. STEVE kennt zwei verschiedene Arten von Blockdefinitionen:

1. den Zeilenblock. Das bedeutet, zwischen Anfangs- und Endmarkierung werden alle Zeilen vollständig, also mit allen Zeichen, in die Blockfestlegung einbezogen. So arbeiten entsprechende Blockfunktionen aller bekannten Textverarbeitungsprogramme;
2. den Rahmenblock. Hier wird zwischen Anfangs- und Endmarke ein Rahmen gezogen. Das wiederum bedeutet, daß nicht zwangsläufig komplette Zeilen mit einbezogen sind, wenn die Anfangs- und Endpunkte beispielsweise mitten in den Textzeilen lagen. Der Rahmenblock trennt regelrecht einen Bildausschnitt aus dem Text heraus, und Zeilenanfänge bzw. -enden bleiben (wenn außerhalb) unberücksichtigt. Der so definierte Block erscheint invertiert auf dem Bildschirm.



Nicht gerade wenige und nicht gerade kleine Pull-Down-Menüs verbergen sich in der Desktop-Leiste des Textbereiches.

Einstellungssache

Von dem Textbereich aus zugänglich sind zwei Dialogboxen, die eigentlich kaum der Erwähnung wert sind. Banale Dinge verbergen sich dort, die wohl zu jeder Textverarbeitung gehören. Aber außerdem fallen einige Besonderheiten auf, die man sonst nicht kennt: Wird der Bildschirm invertiert gewünscht (Buchstaben weiß, Hintergrund schwarz - nicht unbedingt jedermanns Sache)? Text komprimieren beim Abspeichern? Soll der Ziffernblock normale Zahlen oder Befehle auslösen? nach wieviel Zeilen soll eine automatische Zwischenspeicherung erfolgen? Selbst die Schreibrichtung läßt sich von Rechts-nach-links auf Links-nach-rechts umschalten (die Araber wird's freuen).

Die Textverarbeitung

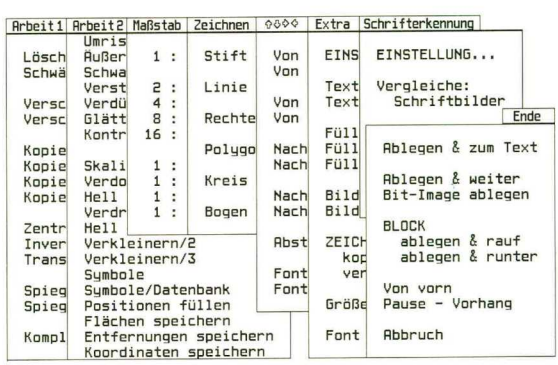
Wer das Nichtvorhandensein von GEM verschmerzt hat, wird bald feststellen, daß die Möglichkeiten der Tastaturbelegung und der Maus in ungewöhnlicher Art und Weise ausgeschöpft wurden.

Beispiele für die Maus:
Cursor im Text - linke Taste fügt Leerzeichen ein, rechte Taste löscht Zeichen.

Cursor am Zeilenende - linke Taste fügt Leerzeilen ein, rechte Taste löscht Zeilen.

Fährt man mit der Maus über den oberen bzw. unteren Bildschirmrand hinaus - linke Taste blättert um eine Zeile nach oben bzw. unten, rechte Taste um ganze Bildschirmgrößen.

Das Wandern in einem großen Text kann wahlweise auch über Steuertasten im Ziffernblock oder über Menüpunkte bewerkstelligt werden. Besonderheit: Im Menü *Durch die Datei* startet man zeilen- bzw. seitenweises Blättern, das so lange automatisch weiterläuft, bis eine Shift-Taste dies unterbricht. Ein kurzes Bewegen mit der Maus oder das Anschlagen einer beliebigen Taste läßt diesen Vor-



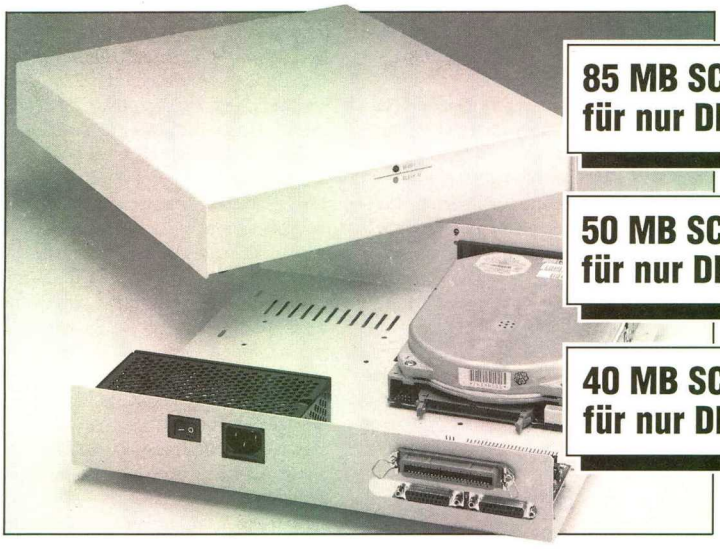
Auch das Grafikfenster spart nicht mit Menüeinträgen.

Die Systemtabellen

Zu den erstaunlichsten Einrichtungen dieses Programms gehören die sogenannten Systemtabellen. Es handelt sich dabei ausschließlich um Textdateien, die wichtige Systemparameter aufnehmen:

Textbausteine sind oft verwendete Standardformulierungen, wie Briefkopf,

SCSI-Festplatten zu »Schotten-Preisen«!



Zum Beispiel:

85 MB SCSI-Festplatte (28 ms) für nur DM 1.198,–

50 MB SCSI-Festplatte (28 ms) für nur DM 1.098,–

40 MB SCSI-Festplatte (19 ms) für nur DM 1.198,–

Unsere SCSI-Festplatten werden komplett anschlußfertig incl. Software und Kabel ausgeliefert.

Ausstattung und Leistungsmerkmale unserer Festplatten:

- Datentransferraten >600 KByte/s (mit CDC- und Maxtorlaufwerken bis zu 850 KByte/s erzielbar), mittlere Zugriffszeiten bis zu 14 ms
- Spitzensoftware: 255 Partitionen installierbar, Passwortfunktion, jede Partition autobootfähig, Interleave 1:1 einstellbar, Cache, Backup, Optimizer in der Software enthalten
- 100% Atari-kompatibel, sämtliche Fremdbetriebssysteme (PC-Speed, PC-Ditto, Spectre, Aladin, Minix, OS-9, RTOS) sind voll lauffähig
- Superleise (3,5″-Festplatten ohne Lüfter, 5,25″-Festplatten mit thermogeregeltem Lüfter)
- Durchgeschleifter gepufferter DMA-Bus, Autoparkfunktion hardwaremäßig
- Herausgeführter SCSI-Bus (50-poliger Centronics-Anschluß, Apple MacIntosh und PC's anschließbar)
- Zweite SCSI-Festplatte im Gehäuse nachrüstbar (SCSI-Hostadapter und Gehäuse für interne zweite Festplatte vorbereitet)
- Unsere SCSI-Festplatten werden komplett anschlußfertig im Gehäuse incl. Netz-, DMA-Kabel, Software und Handbuch geliefert

Preise:

32 MB, 40 ms, ST138N-0	DM 998,–
40 MB, 19 ms, Quantum	DM 1.198,–
49 MB, 28 ms, ST157N-1	DM 1.098,–
85 MB, 28 ms, ST296N	DM 1.198,–
80 MB, 24 ms, ST1096N	DM 1.298,–
80 MB, 19 ms, Quantum	DM 1.798,–
170 MB, 28 ms, 2x ST296N	DM 2.498,–
280 MB, 17 ms, Maxtor	DM 3.498,–
380 MB, 17 ms, Maxtor	DM 3.998,–
702 MB, 14 ms, CDC	DM 5.998,–
1200 MB, 14 ms, CDC	DM11.998,–
44 MB, 25 ms, SQ 555	DM 1.598,–

SCSI-Kits (Festplatte und SCSI-Hostadapter für ST):

32 MB Kit (ST138N-0)	DM 798,–
40 MB Kit (P40S)	DM 998,–
49 MB Kit (ST157N-1)	DM 898,–
85 MB Kit (ST296N)	DM 998,–
80 MB Kit (ST1096N)	DM 1.098,–
80 MB Kit (P80S)	DM 1.598,–
SCSI-Hostadapter (incl. Software und DMA-Kabel)	DM 198,–
DMA-Kabel	DM 39,–
SCSI-Kabel	DM 39,–
Netzteil 50 W	DM 99,–
Gehäuse	DM 99,–
Cartridge für SQ555	DM 239,–

Weitere Modelle sowie sonstige Soft- und Hardware auf Anfrage!

CALTEC.
Datensysteme

Eugenstraße 28
7302 Ostfildern 4
Telefon 0711/457 96 23
Telefax 0711/456 95 66

Schlußfloskeln oder feststehende Absätze. Sie können sogar Grafik enthalten. Eine andere Systemtabelle beinhaltet die Abkürzungen. Dabei handelt es sich um kürzere Texte wie etwa Anrede, Namenszüge oder markante Überschriften. Sie sind zur späteren Ausführung über eine kurze Tastenfolge erreichbar, während die Textbausteine (etwas umständlicher) mit ihrem vollen Namen über ein Menü anwählbar sind.

Kopf- und Fußzeilen sind feststehende Texte, die am Anfang und am Ende eines ausgedruckten Blattes stehen sollen. Am bekanntesten dürfte die Seitenzahl sein, aber es könnten beispielsweise auch eine Kapitelüberschrift, das Datum, vielleicht ein Autorenname oder gar ein Copyright-Hinweis dort wiedergegeben werden. Selbst eine Unterscheidung zwischen Seiten mit gerader und ungerader Seitennummer ist möglich.

Die Weckerdatei kann ebenfalls ein beliebig gestalteter Text mit Grafikelementen sein, der erst an einem bestimmten Datum zu einer vorgegebenen Uhrzeit auf dem Bildschirm erscheint.

Wieder interessanter erscheint mir die Systemtabelle mit der Tastaturbelegung. Alle Zeichen und Symbole, die in der Zeichensatztabelle vorhanden sind, können Sie hier auf die Tasten legen. STEVE unterscheidet sogar zwischen zwei Normalbelegungen, die während des Schreibens alternativ eingeschaltet sein können. Desweiteren sind andere Zeichensätze mit der (gleichzeitig gedrückten) Control-, der Alternate-Taste, sowie der Kombination Alternate und Shift erreichbar.

Funktionstastenbelegung kennt man von anderen Programmen sicher zur Genüge. Außer den bekannten zehn sind hier sogar die linke und rechte Maustaste als Funktionstasten zu benutzen. Allerdings fallen sie dann für ihre üblichen Tätigkeiten im Textfenster aus, für die Menüleisten sind sie aber dennoch wirksam. Die einzige Beschränkung für eine Funktionstastenbestimmung ist die maximale Länge von 1796 Zeichen, aber das ist praktisch schon eine volle DIN-A4-Seite. Sinnvoll wäre es, nicht unbedingt nur Text dort zu deklarieren, sondern ganze Befehlsfolgen, die zudem noch Textbausteine oder andere Befehle aufrufen.

Gegenüber anderen Computersystemen hat der Atari ST den großen Vorteil, daß er seine Bildschirm-Fonts nicht unveränderbar im Festwertspeicher ROM hält, sondern von außen (üblicherweise beim Systemstart) hinzulädt. Das bedeutet, daß der Anwender einen direkten Zugriff auf diesen Systemzeichensatz hat. STEVE selbst bringt schon in der Grundausstattung 10 verschiedene Zeichensätze mit, die ebenfalls in einer Systemtabelle vorliegen. Unter dem Namen „weitere Zeichensätze" können unabhängig davon beliebig viele selbst erschaffene Zeichensätze als weitere Systemtabelle erstellt werden. Auch das Ändern von Zeichen läßt sich spielend leicht bewerkstelligen, weil die Systemtabellen ja Textdateien sind.

Es können bis zu 21 Zeichensätze gleichzeitig geladen sein (wer braucht überhaupt so viele in einem Dokument?), und Sie müssen nicht zwangsläufig auch alle 256 Definitionsplätze belegen. (Definitionsplätze bedeutet, so viele verschiedene Tastaturkombinationen sind erlaubt.) Man könnte sogar eine Art Geheimschrift damit konstruieren, weil zwar der Tastencode beim Drücken einer Tastaturkombination abgefragt und im Dokument festgehalten wird, aber ein Bildschirmzeichen vielleicht nicht dafür definiert ist - der Phantasie sind hier kaum noch Grenzen gesetzt.

Meldung bitte

Eine weitere Systemtabelle stellt die Meldungsdatei dar. Dort ist alles aufbewahrt, was irgendwann einmal innerhalb des Programmablaufs auf dem Bildschirm zu sehen sein wird: alle Menüs, alle Dialogboxen, alle Warn- und Hinweismeldungen. Auch diese Datei steht dem Benutzer zur freien Gestaltung offen. Weil diese Sache so überaus interessant ist, hier ein kleines Beispiel für eine solche Meldung:

```
000001640 m 1
Start Block Suchen Druck Layout D-
Bank
00
1..... 2.....A 3.... 4.....C.
1=m181 2=m3 3=m4 4=m5
A=m207 C=m248
..
```

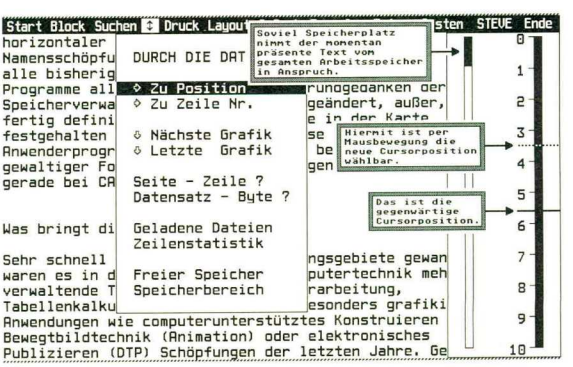
Jede Meldung dieser Datei - und es gibt in der derzeitigen STEVE-Version davon 280 - beginnt mit einer Ziffernfolge, d.h. mit fünf Ziffernpaaren. Sie geben die Koordinaten an, wo die Meldung auf dem Bildschirm erscheinen soll (Anfangszeile, -spalte, Höhe und Breite), sowie die fortlaufende Nummer (hier *m1*). Als nächstes folgt eine Textzeile, die genauso auf dem Bildschirm sichtbar sein wird. Darunter finden wir eine Positionsangabe für den Text, also mit *00* beginnend in der obersten Zeile des Bildschirms. Die nächste Reihe mit den Zahlen und Punkten heißt Invertierungszeile. Die Zahlen fungieren darüber hinaus aber auch noch als

Quellenangabe für die auszuführende Tätigkeit. Die Nummer gibt also an, was nach dem Anfahren dieses Menübereiches mit der Maus weiter geschehen soll. Das führt uns zu einer weiteren Zeile, und zwar jener mit den Befehlsnummern. Dort bedeutet beispielsweise, daß beim Aktivieren des Menübereiches mit der Befehlsnummer *1*, identisch mit dem invertierten Text *Start*, in die Meldung *m181* verzweigt werden soll. Diese weitere Meldung kann entweder eine Textausgabe per Dialogbox, ein Untermenü oder eine Befehlsfolge (Batch) sein.

Zugegeben, das eben beschriebene Verfahren, in STEVE seine eigenen Menüleisten in einer einfachen Textdatei zu konstruieren, erscheint für den Anfänger ungemein kompliziert. Mit etwas Geduld und Übung ist es aber ein leichtes, dies zu bewerkstelligen, ebenso wie das völlig freie Umgestalten von Druckertabellen, Formularen oder Bild-Bibliotheken, um nur einige zu nennen. Was das für ein als offen und flexibel bezeichnetes Programm bedeutet, ist klar: Man kann individuell das gesamte Aussehen der Programmoberfläche damit gestalten. Wertung: Geschickte Lösung für alle, die nicht programmieren können/wollen/sollen.

Das Wörterbuch

Ausgangspunkt für zwei nützliche Abteilungen in der Textverarbeitung ist das integrierte Wörterbuch. Es ist eigentlich nichts anderes als eine besonders aufgebaute Datenbankdatei, die parallel zum



Sehr übersichtlich sind Positionsanzeige (im Bild rechts) und -wahl (per Maus), relativ bezogen auf den gesamten Text.

Text hinzugeladen wird. Sie dient uns zunächst als Rechtschreibhilfe.

Über ein entsprechendes Menü eingeschaltet, lauert sie als „Online-Prüfung" während des Schreibens im Hintergrund und macht sich nach jedem unbekannten Wort mit einem akustischen Signal („Ping!") bemerkbar. Jetzt kann man aber nicht gleich weiterschreiben. STEVE wünscht beharrlich die Korrektur dieses Wortes (wenn falsch) oder die unmittelbare, automatische Übernahme in die Wörterbuchdatei mittels Return-Taste (wenn unbekannt). Nachteil dieses Verfahrens: Es hemmt den Schreibfluß der Tastenakrobaten (roter Faden und so ...) und zwingt ständig zum Drücken der Return-Taste.

Eine andere Möglichkeit stellt die nachträgliche Überprüfung des Textes nach dessen Fertigstellung dar. Dabei rast der Prüfalgorithmus durch das gesamte Dokument und kennzeichnet jedes unbekannte Wort mit einem Sternchen. Dieses Symbol kann man übrigens in einer Systemtabelle frei definieren. Großer Nachteil dieser Vorgehensweise ist, daß die unbekannten Wörter nun nicht mehr automatisch in die Wörterbuchdatenbank übernommen werden. Meine Empfehlung für beide Verfahren: durchaus lobenswerter Grundgedanke, aber die Handhabung und Funktionsweise sind überarbeitungsbedürftig. Übrigens, auf Diskette wird eine Wörterbuchdatei mit 35000 deutschen Begriffen mitgeliefert.

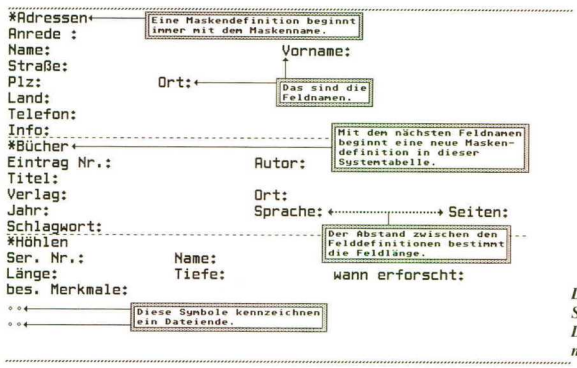
Ein etwas anders gestaltetes Wörterbuch versteckt sich hinter dem Menüeintrag *Übersetzung*. Wo das Rechtschreibwörterbuch lediglich auf unbekannte Wörter gelauert hat, folgt jetzt etwas, das wie *Suchen und Ersetzen* funktioniert. Mit der Übersetzungsfunktion kann nämlich von einer Sprache in eine andere übersetzt werden. Eine kleine Beispieldatei Englisch/Deutsch befindet sich ebenfalls auf Diskette. Man darf sich bei dieser Einrichtung aber keiner Illusion hingeben, es wird eben nur Wort-für-Wort übersetzt; ob dann der stilistische Satzbau, die orthografische Konstruktion und die Wortbedeutung im Sinnzusammenhang überhaupt richtig sind, bleibt sehr fraglich. STEVE möchte hier auch nur eine kleine Hilfe sein und keine ausgefeilte Übersetzungroutine liefern (das wäre wohl wirklich etwas zuviel verlangt). Wertung: So lobenswert die Idee einer solchen Übersetzungshilfe auch sein mag, es bleibt dahingestellt ob sie wirklich sinnvoll einsetzbar ist.

Die Datenbank

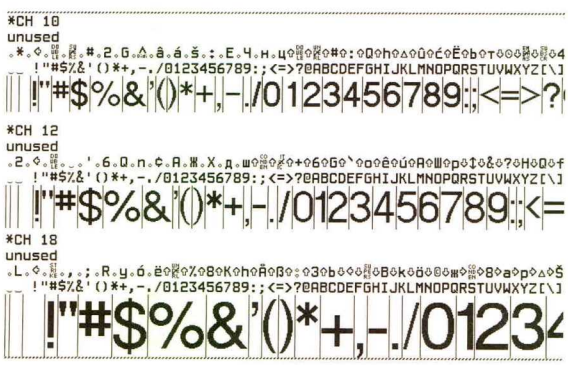
Wenn in der Überschrift die Frage gestellt wurde, ob STEVE wirklich ein Programm für alle Fälle sei, dann darf eine Programmgattung wirklich nicht fehlen: die Datenbank. Eine Textverarbeitung macht doch sicherlich in Kombination mit Datenbankfunktionen den meisten Sinn. Man denke doch nur an Serienbriefe. Gibt es denn noch einen Anwenderkreis, der Briefe zwar mit einer Textverarbeitung erstellt, hernach aber die Anschriften mittels Adreßaufkleber anheftet? Welche Textverarbeitung kann denn noch keine Serienbriefe verarbeiten? Auf den Müll damit!

Die Vorgehensweise ist hinlänglich bekannt. Eine Datenbank besteht in aller Regel aus einer Bildschirmmaske, die alle Feldnamen zeigt. Hinter den Feldnamen ist dann immer etwas Platz freigehalten, um die Daten (Feldinhalte bzw. Feldeinträge) aufnehmen zu können. STEVE bleibt auch hier seinem Prinzip treu: der Konstruktion von Bildschirmmasken über das Anlegen einer Systemtabelle als normaler Text.

Wer nun mit dem Zusammenbau der Maske zufrieden ist, wird üblicherweise die Freiräume mit Daten füllen wollen. Jetzt macht sich eine Besonderheit von STEVE bemerkbar. Die Eingabemenge, d.h. die Feldgröße, ist nicht von dem vorher definierten Abstand der einzelnen Feldnamen abhängig. Wenn ein Datenfeld mehr Information als vorgesehen aufnehmen muß, fügt das Programm einfach weitere Leerzeilen ein, und die anderen Felddefinitionen verschieben sich nach unten. Dieses Prinzip der flexiblen Datenfeldgröße ist bei den bekannten Datenbankprogrammen (meist Vertreter des relationalen Modells) nicht üblich und wird erst seit jüngster Zeit von neueren Schöpfungen verwandt. Ein Wort zum Fassungsvermögen: Ein Datensatz kann bis zu 500 Felder enthalten, die bis maximal 15000 Zeichen umfassen dürfen. Ansonsten ist die Kapazität nur durch die Größe des Speichermediums begrenzt (wer arbeitet noch ohne Festplatte?).



Dies ist eine Systemtabelle für Datenbankmasken.



Eine Zeichensatzübersicht, sie ist im Grafikteil beliebig manipulierbar.

Interessant ist auch, wie STEVE die Daten abspeichert. Alle Leerzeichen zwischen dem Eintrag und dem Beginn des nächsten Feldes verschwinden. Es wird also wichtige Speicherkapazität eingespart. Natürlich genügt das Eingeben und Speichern von Datenfeldinhalten bei weitem nicht, man möchte doch gerne auf geschickte Art und Weise die Datensätze verwaltet wissen und besonders schnell gesuchte ausfindig machen. Jetzt wird der Ziffernblock mit Leben erfüllt. Wichtige

Bewegungen in der Datenbankdatei sind über eine Zweitbelegung des Ziffernblocks ausführbar.

Ein Datenbanksystem kommt selbstverständlich nicht ohne Sortierfunktion aus. Zur Festlegung des Sortierkriteriums lassen sich bis zu 5 Feldnamen angeben, die unabhängig steigend oder fallend durchschritten werden. Dieses Verfahren nennt man „verschachtelte Sortierung", d.h. es werden erst alle gleichen Inhalte von Feld 1 (z.B. Postleitzahl) entsprechend der Sortierrichtung (fallend oder steigend) zusammengetragen, dann startet innerhalb dieser gleichen Inhalte ein zweiter Sortierdurchgang (z.B. Familienname) usw.

STEVE hat eine Datenbankdatei mit 3 Bytes langen Indikatoren angelegt, jeder davon enthält die Startadresse eines Datensatzes. Zum Sortieren müssen nun mindestens 32 kByte Speicherplatz frei sein, wenn es bis zu 10666 Datensätze gibt. Bei mehr Datensätzen muß immer das Dreifache in Bytes an freiem Speicher zur Verfügung stehen. Dieser Platz wird immer für die Indikatoren gebraucht, da die Datensätze variabel sind und das Programm entsprechend schnell sein soll. So gruppiert es nicht die ganzen Datensätze um, sondern führt dies mit den Indikatorlisten durch. Ein Wort an die Datenbankspezialisten: Es wurde hier der Mengensortieralgorithmus von Knuth, 1973, angewandt. Die Sortiergeschwindigkeit hängt sehr stark von der Menge an Datensätzen ab und auch davon, wieviele unterschiedliche Zeichen oder Werte das erste zu sortierende Feld aufweist. Im Durchschnitt sind folgende Zeiten gemessen worden: 10000 Datensätze (jeder aus einem Wort bestehend) dauern 2 Minuten, 75000 brauchen dagegen 20 Minuten.

Da alle Datenbankdateien - gemeint sind jene, welche die Feldeinträge aufnehmen - wie normale Textdateien angelegt sind, läßt sich eine äußerst nützliche Einrichtung von STEVE auch hier zur Anwendung bringen: das Komprimieren von Text. So benötigte eine Beispieldatei mit insgesamt 75207 Einträgen 522652 Bytes in komprimierter Form. Sehr schön ist die Angabe des benötigten Speicherplatzes in einer Dialogbox. So kann man immer prüfen, ob die Kapazität einer doppelseitigen Diskette noch ausreicht, um den gesamten Datenbestand komprimiert aufzunehmen. Welcher Komprimieralgorithmus Anwendung findet und wie er arbeitet, konnte ich leider nicht in Erfahrung bringen.

Jetzt kann es natürlich passieren, daß unsere Schallplattensammlung unermeßlich groß angewachsen ist. Die Kapazität einer Diskette reicht nicht mehr aus (Sie wollen noch immer keine Festplatte anschaffen?). Dann gibt es eine recht nützliche Einrichtung, der wir uns fortan bedienen wollen: Kettendateien. Sie können maximal die Gesamtkapazität der Festplatte ausschöpfen. Die einzelnen Dateien in der Kette dürfen logischerweise nur so groß sein, daß sie gerade noch in den freien Arbeitsspeicher hineinpassen. Das Handbuch liefert uns eine ausführliche Anleitung zur Berechnung:

```
Dateigröße + Anzahl der Datensätze * 3
  +32000 < RAM / 2
```

oder mit der etwas handlicheren Formel:

```
((RAM / 2 - 32000) / (DLD + 3)) * DLD
```

(DLD heißt durchschnittliche Länge der einzelnen Datensätze.) Beispiel: In einem 1-MByte-RAM-Gerät hätte der Benutzer annähernd 735 kByte zur Verfügung. Wenn alle Datensätze einer Datei zusammengenommen durchschnittlich 70 Bytes (Erfahrungswert) lang sind, darf die größte Einzeldatei der Kette bis zu 320000 Bytes enthalten. Beweis:

```
((735000 / 2 - 32000) / (70 + 3) * 70 =
                                  321712
```

Die höchstmögliche Anzahl von Kettengliedern ist definiert als *(größte Länge des Suchausdruckes - 44) / 4*. Diese Länge ist derzeit auf 452 festgesetzt, wodurch bis zu 102 Kettenglieder bzw. Einzeldateien möglich sind. Wichtig für alle „Trotzdem-Disketten-Arbeiter": Das Verfahren mit den Kettendateien läßt sich auch auf eine entsprechende Anzahl von Datendisketten anwenden!

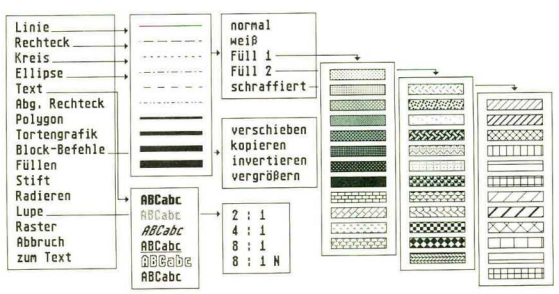
Lassen Sie mich noch einige Feinheiten der STEVE-Datenbank hervorheben, als da wären: Permutieren, also das Vertauschen einzelner Inhalte eines Datensatzes; Zusammenfassen gleicher Datensätze; das Feststellen der Häufigkeit bestimmter Begriffe, um z.B. ein Häufigkeitswörterbuch zu erstellen; oder das Erstellen einer Rangfolge, was besonders für numerische Inhalte interessant ist. Die Abteilung Datenbank in STEVE hält darüber hinaus noch viele andere Sonderfunktionen bereit, die hier zu besprechen den Rahmen sprengen würde. Vielleicht ergibt sich die Gelegenheit, in einem späteren Bericht noch ausführlicher darauf zurückzukommen. Fazit hierzu: Insgesamt ist die Datenbank für den Anfänger gewöhnungsbedürftig, aber die ungewöhnlich reiche Auswahl an Verarbeitungsfunktionen ist erstaunlich. Die Datenbank sucht und sortiert recht schnell. Mangels Vergleich ist diese Aussage natürlich recht subjektiv.

Die Batch-Befehle

Ein Programm zeichnet sich oft durch leichte Bedienbarkeit aus, weswegen die Beliebtheit von grafischen Benutzeroberflächen stetig zugenommen hat. Dennoch ist oft festzustellen, daß ständig die selben oder ähnliche Arbeitsschritte durchzuführen sind. So wird sich ein Überbleibsel aus der guten alten Zeit der zeilenorientierten Befehlseingabe (siehe Kommandozeilen-Interpreter und ähnliche) aller Wahrscheinlichkeit nach in neuere Programme hinüberretten: die Stapelbefehle. Wenn in der Textverarbeitung sogenannte Textbausteine üblich geworden sind, machen Batch-Befehle dort einen Sinn, wo nicht tagtäglich mit der Maus oder irgendwelchen Tastaturkombinationen immer dasselbe von Hand durchgeführt werden soll (wird ja auf die Dauer langweilig). In STEVE wird hierzu (wer hätte es wohl anders gedacht) auch wieder eine Systemtabelle eröffnet, die im Klartext alle Befehle aufnimmt.

Die Grafik in STEVE

Integraler Bestandteil von STEVE sind zwei (!) Grafik-Editoren, die jeweils zwei ganz spezifische Aufgaben wahrnehmen. *Grafik 1* ist für raffinierte Operationen wie Schrifterkennung und Font-Erstellung für DTP ausgelegt (darüber gleich mehr). Bei *Grafik 2* dagegen handelt es sich um ein ganz normales Zeichenprogramm. Unabhängig davon unterschiedet das Programm im Textfenster zwischen zwei verschie-



Das sind die Funktionen des Grafikeditors, die als Pop-Up-Menüs frei positionierbar auf dem Bildschirm erscheinen.



Bitte ausschneiden

12 neue Zeichnungen über Dr. Nibble & Crew.

Format: DIN A3 quer; Umfang: Deckblatt, 12 Kalenderblätter, Kartonrücken
Gebunden: Drahtspiralenbindung
Das Geschenk für alle Computerfans (auch außerhalb der Atari-Welt)
Preis: 14,90 DM plus 5 DM für Porto und Verpackung

Bitte schicken Sie einen Euro-Scheck oder Briefmarken im Wert von 19,90 DM oder einen Zwanzig-Mark-Schein an den ICP-Verlag, Wendelsteinstr. 3, 8011 Vaterstetten.

Absender: _____

denen Zeilenarten: der reinen Text- und der Grafikzeile. Übrigens, Textzeilen lassen sich nachträglich in Grafikzeilen umwandeln, wobei der Text erhalten bleibt, nur umgekehrt geht das leider nicht. Grafikzeilen nehmen immer die gesamte Zeile (oder auch mehr) ein, eine gemischte Text-/Grafik-Zeile ist deshalb nicht möglich. Neben Bildobjekte kann also kein Text (vom Texteditor her) geschrieben werden, außer man fügt ihn im Grafik-Editor zusätzlich an.

Da zunächst (d.h. beim Programmstart) der reine Text-Editor präsent ist, muß STEVE mitgeteilt werden, ob sich zum Geschriebenen auch ein Bild gruppieren soll. Mit der ESC-Taste schaltet man in den Grafikmodus (des Text-Editors) um. (Anmerkung: Ich hätte nicht gerade die ESC-Taste mit dieser Funktion belegt, da sie in anderen Programmen schon fast branchenüblich als „Rettungsanker" oder „Fluchtweg" bei unklaren Programmzuständen Anwendung findet.) Wenn eine Grafikzeile zu wenig ist, können Sie durch fortwährendem Druck auf ESC weitere Zeilen erzeugen. Die eigentliche Grafikarbeit beginnt, wenn der Cursor in einer solchen Zeile steht. Die unabhängigen Textzeilen gibt es ja nach wie vor (auch zwischen den Grafikzeilen), sie bleiben von der ganzen Grafikarbeit unberührt! Ein zweiter Druck auf ESC öffnet eine „andere Welt". Der Grafik-Editor meldet sich mit einer völlig anderen Menüleiste und einer Dialogbox. Diese Box möchte gerne wissen, wie groß die Grafik (und damit auch die einzelne Grafikzeile) werden soll. Eine STEVE-Zeile ist üblicherweise immer 18 Pixel (Bildpunkte) hoch und ein Zeichen 16 Pixel breit. Es kann in der Box nun ein Vielfaches von Höhe und Breite gewählt werden. Die maximal mögliche Ausdehung des Zeichenblattes beträgt in der Breite 16380 Pixel, das sind 1024 Zeichen oder 16 Bildschirme (nebeneinander!).

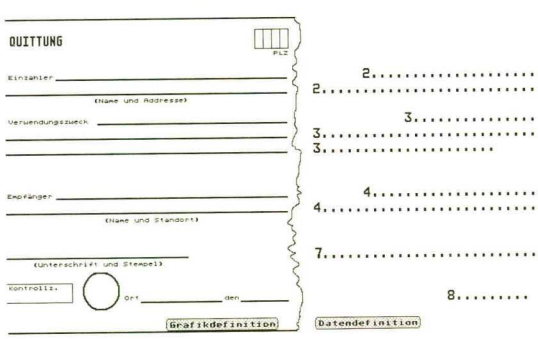
Jeder, der noch kein Grafikprogramm besitzt und gerne in seinen Texten auch Bilder einbinden möchte, trägt sich mit dem Gedanken, ob sich dafür die Anschaffung eines eigenen Programms lohnt. STEVE hält für die Manipulation von Bildern eine große Palette an Funktionen bereit, die an dieser Stelle leider nicht sehr ausführlich beschrieben werden können. (Auch hier droht erneut der Rahmen gesprengt zu werden!) Ein Blick auf Bild Nr. 5 und 6 soll nur einen kleinen Eindruck dieser Vielfältigkeit vermitteln. Fazit hierzu: Die Möglichkeiten von *Grafik 1* sind mehr darauf ausgelegt, vorhandene Bilder (das können auch Zeichensätze sein) zu verändern. So ist alles zu finden von Spiegeln, Invertieren, Kopieren, Drehen und Verschieben, bis hin zu Vergrößern und erkleinern.

Grafik 2 ist mehr ein Programm-Modul für neu zu erstellende Grafik. Es entspricht in seiner Funktionsbreite voll und ganz üblichen Programmen, die separat angeboten werden. Hervorhebenswert erscheint mir hier die Funktion *Tortengrafik*. Es müssen für jedes darzustellende Kreissegment nur die Prozentangabe eingefügt und die gewünschte Rasterdarstellung ausgesucht werden. Wertung: brauchbar für alle, die nicht schon ein eigenes Grafikprogramm besitzen.

Formulare

Eine Mischung von Datenbank und Grafik-Editor und somit eine konsequente Weiterentwicklung der gesamten STEVE-Philosophie stellt der Formulargenerator dar. Ein Formular ist ein grafisch erzeugtes Bild (hauptsächlich mit senkrechten und waagrechten Linien), das später Einträge ähnlich wie bei der Datenbank aufnehmen soll. So gesehen ist ein Formular sogar eine besonders gestaltete Bildschirmmaske, allerdings auf einer Grafikseite erzeugt.

Alle Einträge in ein Formular befinden sich aber noch nicht zwangsläufig in der Form, in der sie später in die Datenbank übernommen werden können. Aber keine Bange, ein spezieller Trenndurchlauf liest die Texteinträge im Grafikformular aus und legt sie „datenbankgerecht" ab. Wichtig zu wissen ist, daß es parallel zur Grafikseite des Formulars eine gleich aufgebaute Systemtabelle gibt, welche die Plazierung der Einträge steuert.



Links im Bild eine grafische Formulardefinition und rechts davon die Systemtabelle für die Platzhalter (beide im Teilausschnitt).

Ein Weg zu DTP?

Wie wir schon gesehen haben, arbeitet STEVE mit Text- und Grafikzeilen weitestgehend unabhängig und dennoch parallel zueinander. Dies ist der erste Weg zum Desktop Publishing. Es paßt eigentlich gut zusammen: hier Textzeilen, dort Grafik, die im Grunde genommen unabhängig voneinander sind. Nun wird im Menüpunkt *Layout* beides zu einem sinnvollen Ganzen „verschweißt", so daß Text und Bild als Einheit dastehen. Weitere Arbeitsschritte sind denkbar, z.B Verschieben, Kopieren usw. Man kennt diese Funkltionen aus dem Grafik-Editor, nur daß jetzt das ganze Arbeitsblatt (also auch der Text) einbezogen wird. In Verbindung mit entsprechenden Druckertabellen und -treibern ist auch der Ausdruck des gestalteten Dokumentes in der gewünschten Form leicht möglich. Wertung: Es muß nicht ein Programm für mehrere Hundert Mark sein, um DTP in ausreichendem Maße verwirklicht zu sehen. STEVE bietet zwar nicht gerade viele DTP-Operationen, sie dürften aber im Umfeld der anderen Möglichkeiten (Text, Grafik, Datenbank) für Durchschnittsanwendungen völlig ausreichend sein.

Schrifterkennung

Eigentlich wollte ich diesen Programmpunkt mit einem üblichen Scanner (EPSON GT-4000) testen. STEVE unterstützt zur Zeit schon eine reiche Auswahl derselben: Handy, SPAT, Marvin oder Panasonic. Nun gut - wenn es nicht direkt funktioniert (meine Schuld, was hab' ich auch einen EPSON), muß es über einen Umweg gehen: Bild einscannen und abspeichern. STEVE untersucht im Grunde als Bild vorliegende Dateien auf Ähnlichkeit mit den Zeichensatzdefinitionen. Übrigens: So funktionieren die derzeit verfügbaren OCR-Programme auch. Muß ich die Abkürzung „OCR" noch einmal erklären? NEIIIIIN!

Dann hab' ich das Programm überlistet: Ich habe es seine eigenen Zeichensatzdateien erkennen lassen! Viel passiert dabei nicht, eine erkannte Zeichenfolge wird lediglich im unteren Bildbereich durch einen anderen Zeichensatz angezeigt. Zwei Funktionen sind hervorhebenswert: *Vergleiche Schriftbilder*, also die normale Schrifterkennung, und *Erstelle Font vom Text*, das Neuanlegen unbekannter Schriften in einer Zeichensatzdatei. Aussagen über die Geschwindigkeit lassen sich

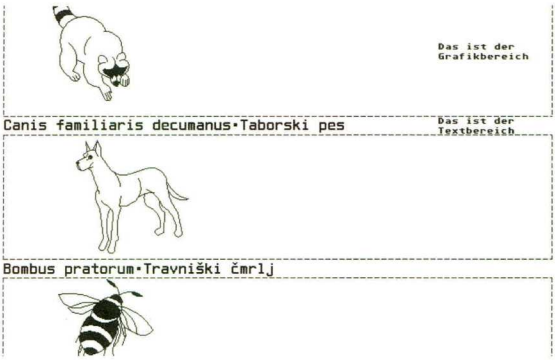


Bild- und Textinformationen in einem STEVE-Dokument bleiben streng voneinander getrennt.

mangels Vergleich auch hierzu nicht machen. Vielleicht kommen wir unter dem Aspekt des zunehmenden Interesses auf das Thema OCR an anderer Stelle noch einmal zurück. Für STEVE-Kunden ist die Tatsache wichtig, daß ihr Programm sich auch in dieser Richtung bewegt und für die Zukunft gerüstet ist.

Wenn sich die Programmierer von STEVE entschließen könnten, die mittlerweile als Standard durchgesetzte IDC-Scanner-Schnittstelle von der Firma Marvin AG einzubinden, wäre ein großer Schritt zu mehr Kompatibilität getan.

Handbuch

Ich habe mir nicht die Mühe gemacht, die Seiten des Handbuchs genau zu zählen, schätzungsweise sind es ca. 280. Es ist eine Wahnsinnsfülle an Informationen über STEVE, ausführlich und mit vielen Beispielen erklärt. An vielen Stellen habe ich gemerkt, daß die Möglichkeiten des Programms sogar den Rahmen des Handbuchs sprengen. Wer etwas experimentierfreudig ist, wird leicht den weniger ausführlich behandelten Funktionen auf die Schliche kommen. Bilder mit Beispielen, Dialogboxen und Menüpunkten machen das Handbuch sehr übersichtlich. Außerdem sind die Kapitel streng strukturiert. Übrigens: Das Handbuch wurde ausschließlich in STEVE geschrieben und macht einen ordentlichen Eindruck.

Abschließende Beurteilung

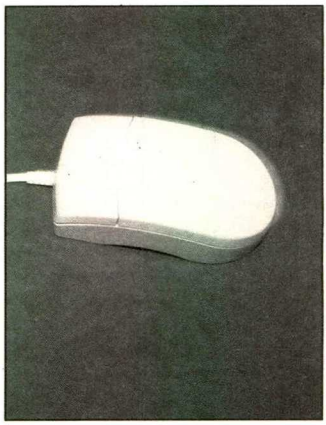
Es tut mir schon fast weh, dem Bericht (und damit auch dem Programm) einen abschließenden Stempel aufdrücken zu müssen. Entgegen der Gepflogenheiten in der Testerbranche habe ich mich eine sehr lange Zeit (14 Tage - ehrlich wahr, großes Indianerehrenwort!) mit STEVE beschäftigt und den Eindruck gewonnen, daß uns hier ein Werkzeug gegeben ist, das viele Anwender nicht für möglich hielten. Schade, daß STEVE aus dem Mittelpunkt des (Kauf-)Interesses etwas verschwunden ist - es ist ein großartiges Programm. Ich muß aber die Schuld (es folgt eine private Meinung), daß es um STEVE zur Zeit still geworden ist, auch der Firma Kieckbusch anlasten, die nicht mit der nötigen Werbe-Power ein ausgesprochen vielfältiges Produkt unterstützt hat. Ich hoffe sehr, daß es bald anders wird!

Das Fehlen einer GEM-Oberfläche ist kein Beinbruch, und wenn man lange genug mit STEVE arbeitet, vermißt man sie nicht mehr. Mir hat die Arbeit mit STEVE großen Spaß gemacht. Und wenn wir uns die Preisfrage stellen, muß ich ehrlich sagen: Mit 498 DM gesellt sich das Programm zu Konkurrenten, denen es an Ausstattungsmerkmalen weit überlegen ist. STEVE mit Schrifterkennung kostet 1398 DM, was mir im Moment nicht ganz einleuchtet, denn so überwältigend waren die OCR-Funktionen (mangels Vergleich mit anderen Programmen) nicht. Dennoch gehört das Grundprogramm (ohne OCR) auf jeden Fall in die Händlerregale; wer noch keine entsprechenden Programme angeschafft hat, sollte durchaus mit STEVE liebäugeln.

DK

Bezugsquelle:
Computer Technik Kieckbusch GmbH
Baumstammhaus
5419 Vielbach
Tel.: 02626/78336

That's a Mouse



Die DTP-Maus mit bis zu 70 % weniger Platzbedarf als andere Mäuse (maximal 5 cm, um bei einem normalen Bildschirm von links oben nach rechts unten zu scrollen)

- für ATARI ST, TT und AMIGA
- Super Auflösung von 290 dpi
- 12 Monate Garantie
- einfach einstecken und fertig, keine Treibersoftware ist nötig
- ergonomisches, elegantes Design
- Mikro-Schalter, dadurch einfacher, exakter Mausklick
- Präzisionskugel, ruhiger und leichter läuft auf fast allen Auflösungen
- haltbar durch ein Gehäuse aus rutschfestem, gehärtetem Kunststoff
- minimale Pflege durch selbstreinigende Mechanik

DM 98,--

unverbindl. empfohlener Verkaufspreis

Hiermit bestelle ich zur sofortigen Lieferung:

____ That's a Mouse a 98,--DM

zuzüglich 6,-- DM Vesandkosten (Ausland DM 10,--)

unabhängig von der bestellten Stückzahl

Heim-Verlag

Heidelberger-Land-Str.194
6100 Darmstadt-Eberstadt
Telefon: 0 61 51 / 5 60 57-58
Telefax: 0 61 51 / 5 60 59

JEDES BIT ZÄHLT

© 1990 by Stefan Hoppe

Unsere Jubiläumsdiskette haben wir zum Anlaß genommen, den Freunden realistischer und historisch relevanter Simulationsspiele einen besonderen Leckerbissen zu servieren. Mit der 2200 im Laufwerk geht die Datenreise zurück ins Jahr 1917: In Ihrer fliegenden Kiste aus Stoff, Sperrholz und Draht werden Sie in den Morgenhimmel aufsteigen und sich dem Roten Baron stellen.

Die Luftkampfsimulation von Richard Spiller ist auf besonderen Wunsch des PD Pool programmiert worden. Aufbauend auf dieser Grundversion wird ein flexibles Rollenspielsystem entstehen, das die Simulation mehrerer – von verschiedenen Spielern geleiteten – Staffeln ermöglicht.

Mit freundlichen Grüßen,

Die "23"

2191

Ramdisk von R. Grothmann läuft unter TOS 1.4. Legt Ordner automatisch an.

Bapf 1.2e Einschaltverzögerung für die Festplatte. Zeichnet Apfelmännchen während des Bootvorgangs.

DCopy 3.2 verwaltet, kopiert und entpackt ARC-Dateien. Vielseitiges Kopierprogramm. Eigene Shell-Oberfläche.

Disk Kit 1.8 Diskette neditor mit attraktiver Oberfläche (s/w).

MP Editor ist klein, hinreichend schnell und vielseitig.

Multi Acc verbindet mehrere Funktion in einem Deskeintrag.

Multi Deluxe 2.0 enthält alle Kontrollfeld-Funktionen und noch mehr. Sehr vielseitig.



Pattern-Editor gibt Sprites und Füllmuster als Quellcode aus. Für C, Assembler und Basic (s/w).

XDIR Multiaccessory mit vielen nützlichen Funktionen. Ordner umbenennen. Dateien und Subdirektories verschieben.

X-Info 1.1 berechnet den Platzbedarf selektierter Dateien. Hilft beim Kopieren auf Diskette.



Edison Utiliy 1.0 enthält mehrere Funktionen in einem Deskeintrag.

Compare 1.1 vergleicht ASCII-dateien und spürt Unterschiede auf. Zahlreiche Hilfs- und Suchfunktionen (s/w).

X-Rechner beherrscht die Grundrechenarten und Prozentrechnen.

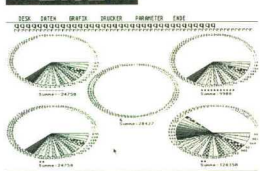
Fuß2End 3.0 konvertiert Fußnoten von WordPlus in Endnoten. Löscht überzählige Lineale. Überträgt WordPlus Texte nach Wordperfekt.

All in One Multiaccessory mit zahlreichen Funktionen.

IBMASCII überträgt Texte nach WordPlus, Skript und Signum. Deutsche Umlaute und ß.

Lupe zeichnet kleine Bilder und Icons mit hoher Präzision Ausgabe als Struktur für C (s/w).

2192

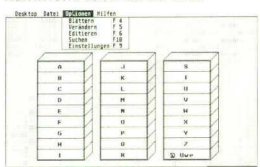


Datari 1.3 wertet Meßreihen aus und stellt sie grafisch dar (s/w).

Discothek 2.0 verwaltet CDs, MCs und LPs auf professionelle Weise (s/w).

Eagledata 1.9 verwaltet Daten aller Art auf einfache Weise. Läuft ab 512Kb.

Kassetten Care 1.0 verwaltet Titel, Interpreten und Kassettennummern. Läuft ab 300Kb.



Uni Dat 1.01 ist eine universell einsetzbare Datenbank.

2193



KC&D 2.0 verwaltet Kurse, Charts & Depot. Zahlreiche Funktionen (s/w).

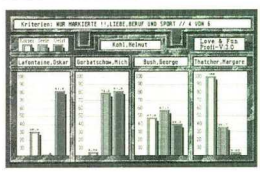
Lohn Control überprüft, ob der Anwender sich in der für ihn günstigsten Steuerklasse befindet.



DyKoVer 1.0 dient der innerbetrieblichen Kostenverrechnung.

Kredit 1.17 führt Berechnungen im Rahmen des Kreditwesens durch.

2194



Love & Fun 3.0 PD zeigt, welcher Partner zu Ihnen paßt: In der Liebe, beim Sport oder am Arbeitsplatz. Die neue Version verfügt über eine eigene Datenbank mit 1000 prominenten Personen, die zum Partnervergleich herangezogen werden können. Die PD-Version erlaubt den Partnervergleich zwischen fünf Personen, die Speicherfunktion ist gesperrt.(s/w).

Sagrotan 4.17 erkennt Computerviren und schützt Datenträger vor neuem Befall. Vergleichsviren, Vergleichsbootsektoren, Minibootprogramme und Programminformationen sind in Bibliotheken enthalten, die vom Anwender gegebenenfalls erweitert werden können (s/w).

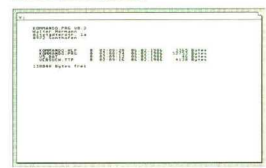
BistDuDa Einschaltverzögerung für SCSI-Platten.

Neustart führt Kalt- und Warmstarts durch. Läuft auch mit der Gemini-Shell.

Rapunzel 0.98 Multiaccessory mit vielen nützlichen Funktionen.

Besonderheiten:	
f	= Läuft nur in Farbe
s/w	= Läuft nur monochrom
E90	= Läuft auch mit Emula 5.1 von Diskette 2090.
e	= in englischer Sprache
J	= Joystick notwendig
MB	= 1MB RAM erforderlich
S	= Shareware
G	= GFA-Quellcode liegt bei
K	= Kontaktkarte eingebaut

2195



Kommando 0.2 erledigt die Abarbeitung von Batch-Dateien.

Ass-Mac 1.1 Macroassembler für Assembler.

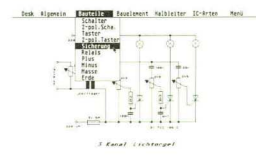
Sprite 1.1 editiert und animiert bis zu 16 Sprites.

Guckrein 0.92 erlaubt das Arbeiten mit Dateien direkt im Desktop. Läuft mit Gemini-Shell.

2196

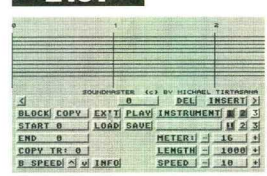


ImageLab 2.0 ist preiswerte Bildverarbeitung für den ST. Lädt zahlreiche Bildformate.



Elektro-Schaltplan Zeichenprogramm mit eigener Bauteil-Bibliothek.

2197



Soundmaster 1.0 erstellt Musikstücke, die in Assembler eingebunden werden können (f, niedrig,

noch nicht ganz Absturzfrei).

STequenz 1.0 Sequenzer mit allen wichtigen Funktionen.

Mididump Soundcreator für Korg DS 8.

U-110 Editor 1.0 für Roland U-110.

Midi Help 1.0 umfaßt drei Midi-Utilities: Sequencer, Arpegiator und Chordmaker (s/w).

E20 Sounds eröffnet 64 zusätzliche Sounds des Roland E20 Keyboards.

Drums ist Midi Drum-Editor und Sequencer (s/w).

2198

WS-DOS 0.4 Lernprogramm und Tool, das den Umgang mit MS-DOS erläutert.

Stundenplan erfaßt Schulstunden und Pausenzeiten. Einfache Bedienung (s/w).

Tausend-1 stellt plus/minus Rechnenaufgaben im ersten Tausender. Verschiedene Übungsmodi sind möglich, sie können durch Vergabe eines Zeitslimits noch erschwert werden (s/w).

RMAT arbeitet wie ein mathematisches Numeriklabor.



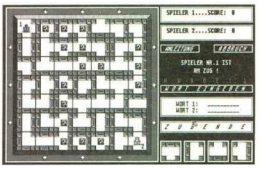
Martins Vokabeltrainer enthält viele deutsche und englische Lektionen. Attraktive Gestaltung. Wiederholte Abfrage unbekannter Vokabeln (s/w).

2199

Mr. Dash ist Paul Dehlis Variante des Boulder-Dash Themas.

Bubble Geschicklichkeitsspiel mit Seifenblasen. (s/w).

Bubbles McGee fordert viel Geschick vom Spieler (f).



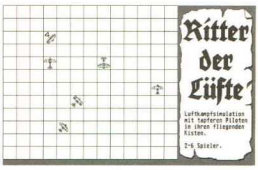
Crazy Ways enthält Labyrinthe, die überwunden werden sollen. Sehr schön gestaltet (s/w).

1-2-3 Spielsteine sollen nebeneinander plaziert werden. 50 Level (s/w).

Gnu Chess Portierung des PD Schachprogramms. Nur Tastatureingabe.

2200

Ritter der Lüfte simuliert Duelle zwischen waghalsigen Piloten in fliegenden Kisten aus Sperrholz, Stoff und Draht. Fokker Dreidecker, Sopwith Camel, SPAD XIII und weitere berühmte Maschinen stehen den 2-6 Piloten zur Verfügung. Unterschiedliche Leistungsmerkmale, aerodynamische Eigenarten und Bewaffnung der Maschinen machen jeden Luftkampf zu einer wahren Zitterpartie. Denn kritische Treffer, Materialermüdung, Ladehemmungen und Pilotenfehler führen nicht selten zum jähen Ende eines morgendlichen Patrouillenfluges. Wer nach erfolgreichem Einsatz den Weg zur heimatlichen Landepiste findet und seine durchlöcherte Maschine sicher aufsetzt, darf sich glücklich schätzen. Notlandungen mit qualmendem und stotterndem Motor, zerfetzten Flächen und kritischen Schäden, gelingen oft nur den erfahrenen Piloten (s/w).



PD-Szene schon gelesen?

Jetzt mit 84 Seiten und 10.000 Auflage. Bei Ihrem Pool-Händler.

PD-Pool sucht noch einige gute Programme zur Veröffentlichung auf den Disketten 2201 - 2210. Die Vorstellung erfolgt gleichzeitig in mehreren großen ST- und PD-Zeitschriften.

Die vorgestellten Disketten erhalten Sie exklusiv bei folgenden PD-Anbietern:

B.I.T.S.
Jagowstraße 17
1000 Berlin 21
030 / 3938203

T.U.M.-Soft&Hardware
Hauptstr. 67
2905 Edewecht
04405 / 6809

EU-SOFT Peter Weber
Josefstraße 11
5350 Euskirchen
02251 / 71897

IDL Software
Lagerstraße 11
6100 Darmstadt 13
06151 / 58912

Simonis Computersysteme
Ubostraße 55
8000 München 60
089 / 8632609 (ab 18 Uhr)

HD-Computertechnik
Pankstr. 61
1000 Berlin 65
030 / 4657028-29

3 1/2 SOFTWARE
Wendenstr. 45
3300 Braunschweig
0531 / 13624

H. Richter Distributor
Hagenerstr. 65
5820 Gevelsberg
02332 / 2706

Weeske Computer
Potsdamer Ring 10
7150 Backnang
07191 / 1528-29 od. 60076

T.S. Service
Szemere Hard&Software
Schleißheimer 127, 8 Mü 40
089 / 3089408

M.Damme -Druck&Computer
Grambeker Weg 40
W-2410 Mölln
04542 / 87258 (ab 16 Uhr)

Intersoft
Nohlstr. 76
4200 Oberhausen 1
0208 / 809014

Eickmann Computer
In der Römerstadt 249
6000 Frankfurt 90
069 / 763409

=PD-Express= J. Rangnow
Ittlinger Straße 45
7519 Eppingen-Richen
07262 / 5131 (ab 17 Uhr)

Schick EDV-Systeme
Hauptstraße 32a
8542 Roth
09171 / 5058-59

Duffner Computer
Habsburgerstr. 43
7800 Freiburg
0761 / 56433

PDST - Michael TWRDY
Kegelgasse 40/1/20 / PF 24
A-1035 Wien
0222 / 75-27-212

Die Zusammenstellung der PD-Pool 2000er Serie unterliegt dem Investitionsschutz: Die Serie darf in der vorgestellten Form nur von den hier angegebenen Pool-Teilnehmern verkauft werden. Im Interesse der PD-Programmierer beachten Sie bitte auch die Nutzungsbestimmungen der verwendeten PD und Shareware.

Händleranfragen erwünscht: Wir suchen noch PD-Anbieter und Fachhändler für gemeinsame Projekte und Veranstaltungen im PD-Bereich.

☐ Bitte senden Sie mir die aktuelle Ausgabe der Zeitschrift PD-Szene, DM 2,50 liegen bei.

☐ Scheck über DM liegt bei, ich erhalte die Ware verpackungs- und versandkostenfrei (Ausland: Bitte Eurocheck in der Landeswährung des Händlers).

☐ Per Nachnahme. Nur Inland! (zuzüglich DM 6,- Nachnahmegebühr).

2021	2031	2041		2061	2071	2081	2091	2101	2111	2121	2131	2141	2151	2161	2171	2181	2191
2022	2032	2042	2052	2062	2072	2082	2092	2102	2112	2122	2132	2142	2152	2162	2172	2182	2192
2023	2033	2043	2053	2063	2073	2083	2093	2103	2113	2123	2133	2143	2153	2163	2173	2183	2193
2024	2034	2044	2054	2064	2074	2084	2094	2104	2114	2124	2134	2144	2154	2164	2174	2184	2194
2025	2035	2045	2055	2065	2075	2085	2095	2105	2115	2125	2135	2145	2155	2165	2175	2185	2195
2026	2036	2046	2056	2066	2076	2086	2096	2106	2116	2126	2136	2146	2156	2166	2176	2186	2196
2027	2037	2047	2057	2067	2077	2087	2097	2107	2117	2127	2137	2147	2157	2167	2177	2187	2197
2028	2038	2048	2058	2068	2078	2088	2098	2108	2118	2128	2138	2148	2158	2168	2178	2188	2198
2029	2039		2059	2069	2079	2089	2099	2109	2119	2129	2139	2149	2159	2169	2179	2189	2199
203C	204C		206C	2070	208C	209C	2100	2110	2120	2130	2140	2150	2160	2170	2180	2190	2200

Diskpreis: DM 8,- *

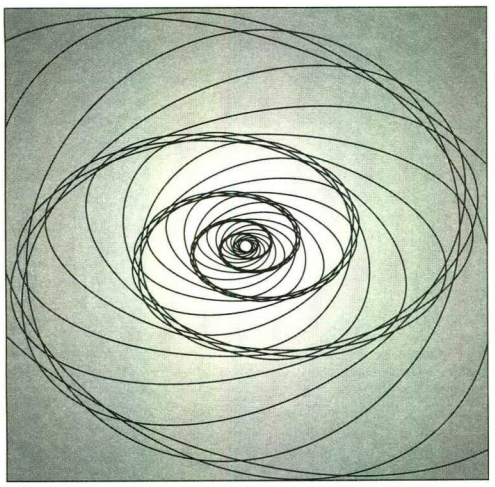
öS 60,-* / sFr 8,-*

* unverbindlich empfohlener Verkaufspreis

Lieferung an meine Adresse:

STC 2/91

Gewünschte Disketten ankreuzen und Bestellschein an einen der oben angegebenen Anbieter einsenden.



PKS-Shell

Ein Schritt in Richtung UNIX

Auch wenn es mancher Atari-Besitzer vielleicht nur ungern zugibt: Es gibt wichtigere Betriebssysteme als das TOS des ST. Oder besser gesagt: Durch ihre Verbreitung haben manche Systeme eine deutlich größere Bedeutung erlangt als TOS. Hierzu zählen in erster Linie MS-DOS und UNIX.

Kompatibilität ist Trumpf

Jedenfalls für IBM-kompatible PCs, die unter MS-DOS arbeiten. Dennoch dürfte MS-DOS trotz seiner weiten Verbreitung wohl kaum das System der Zukunft darstellen, da es aufgrund der Kompatibilität zu den ersten PC-Generationen nicht in der Lage ist, die Leistungsfähigkeit neuerer Rechner sinnvoll zu nutzen. Besonders die Verwaltung heute durchaus üblicher Hauptspeicherkapazitäten von mehr als 1 MB bereitet unter MS-DOS Probleme. Als vor knapp 10 Jahren die ersten IBM-PCs auf den Markt kamen, waren solche Kapazitäten noch undenkbar. Die grafische Oberfläche MS-WINDOWS dürfte für IBM-kompatible Rechner einen Ausweg aus dem Speicherplatz-Dilemma darstellen, aber hier bedarf es zunächst einer Anpassung der Software an die erweiterten Möglichkeiten unter WINDOWS.

System der ersten Stunde

Bleibt also UNIX. Dieses System wurde bereits Anfang der 70er Jahre geboren. Wiege von UNIX waren die Bell Laboratories, die dem amerikanischen Kommunikationskonzern AT&T (American Telephone and Telegraph) angegliedert sind. UNIX sollte den dort beschäftigten Systemprogrammierern eine neue, brauchbare Programmierumgebung zur Verfügung stellen.

Die ersten UNIX-Versionen gehen auf K. Thompson, D. Ritchie und B.W. Kernighan zurück. Die beiden letzteren dürften jedem C-Programmierer ein Begriff sein, wurde diese Programmiersprache doch von Kernighan und Ritchie entwickelt. Dieser Umstand hatte auch einen entscheidenden Einfluß auf die Entwicklung von UNIX: Nachdem die Urversion dieses Systems noch in Assembler geschrieben war, wurden im Laufe der Zeit mehr als 90% des gesamten UNIX-Systems in C codiert.

Ursprünglich war UNIX für den Einsatz auf Großrechnern gedacht. Nachdem die Rechenleistung und Speicherkapazität kleinerer Computer jedoch ständig weiter zunahmen, waren Anfang der 80er Jahre die ersten UNIX-Versionen für Mikrocomputer erhältlich. Auch Atari wird demnächst in den UNIX-Markt einsteigen: Auf der kommenden CeBit (13.3. bis 20.3.91) soll eine UNIX-Version für den Atari TT vorgestellt werden.

UNIX forever

Es mag zunächst seltsam anmuten, daß ein System, das schon ca. 20 Jahre auf dem Buckel hat, in Zukunft weiter an Bedeutung gewinnen wird. Eigentlich würde man auf dem schnellebigen Computermarkt eher das Gegenteil erwarten.

Das Erfolgsgeheimnis von UNIX liegt in erster Linie in der Konzeption des Systems. Der UNIX-Systemkern selber ist recht kompakt. Die meisten Funktionen werden durch externe Dienstprogramme zur Verfügung gestellt. Durch die Verknüpfung der Fähigkeiten dieser Programme können unter UNIX viele Aufgaben sehr flexibel gelöst werden. Man kann durchaus davon sprechen, daß UNIX selbst vom Anwender programmiert werden kann. Dies geschieht ähnlich, wie es beispielsweise unter MS-DOS mit Hilfe von Batch-Dateien (Stapeldateien, Kommandodateien) möglich ist.

Da ein großer Teil des UNIX-Systems in C geschrieben wurde, sind Portierungen auf neue Rechner relativ unproblematisch zu bewerkstelligen. C-Programme sind im Gegensatz zu anderen Programmiersprachen vergleichsweise leicht auf neue Rechnersysteme übertragbar. Lediglich ein kleiner rechnerspezifischer Teil des UNIX-Systems muß an die neuen Gegebenheiten angepaßt werden.

Interessanterweise sind die Quelltexte des UNIX-Systemkerns Public Domain. Wenn man für ein UNIX-System dennoch einiges an Geld hinlegen muß, so zahlt man in erster Linie für die Anpassung an den eigenen Rechner und für die je nach Lieferumfang mehr oder weniger große Zahl an Dienstprogrammen.

Original und Fälschung

Die Lizenz für UNIX besitzt weiterhin AT&T, was diverse andere Firmen nicht daran gehindert hat, eigene UNIX-Ableger mit klangvollen Namen wie XENIX (Microsoft) oder SINIX (Siemens) zu entwickeln. Diese neuen UNIX-Derivate sind zum eigentlichen UNIX nicht immer ganz kompatibel. Inzwischen haben sich jedoch die wichtigsten UNIX-Vertreiber darauf geeinigt, Maßnahmen zur Standardisierung der unterschiedlichen UNIX-Abkömmlinge voranzutreiben. So gibt es seit 1985

im wesentlichen noch drei wichtige UNIX-Richtlinien, die weiter aneinander angeglichen werden sollen:

- X/OPEN Portability Guide, diverse Hersteller
- SVID (System V Interface Definition), AT&T
- POSIX (Portable Operating System for Computer Environments), diverse Hersteller

Die UNIX-Version V.4, die auch für den Atari TT ins Haus steht, stellt einen wichtigen Schritt im Hinblick auf diese Standardisierung dar.

Die PKS-Shell

Nun aber zurück zum ST. „PKS-Shell" - dies klingt im Gegensatz zu den eben angesprochenen Systemen gar nicht mehr nach UNIX, aber der Name täuscht. Soweit es realisierbar und sinnvoll ist, soll es mit diesem Programmpaket möglich sein, auch auf dem Atari ST und TT die Vorteile von UNIX unter TOS zu nutzen. Klar, daß typische UNIX-Eigenschaften wie echtes Multitasking und Multiuser-Betrieb für den Atari nicht in Frage kommen. Hier setzt leider das eigentliche Betriebssystem, das TOS, die Grenzen. Aber um UNIX kennenzulernen und Erfahrungen im Umgang mit UNIX-ähnlichen Betriebssystemen zu sammeln, ist das zunächst einmal nebensächlich.

Der erste Kontakt

Nachdem es bei GEM-unterstützten Programmen nicht unüblich ist, die Programmbeschreibung anfangs links liegen zu lassen (dank Menüleiste und Dialogboxen kommt man ja meist auch so zurecht), ist das Studium des Handbuchs der PKS-Shell unbedingt zu empfehlen. Dies gilt vor allen Dingen dann, wenn man selber noch keine UNIX-Erfahrung besitzt.

Das erste UNIX-Feeling kommt spätestens dann auf, wenn man einen Blick auf den Inhalt der Programmdiskette wirft. Hier finden sich nämlich nahezu beliebig viele Dateien, bei denen es sich in erster Linie um die externen UNIX-Dienstprogramme und Hilfsdateien handelt. Da das UNIX-Konzept darauf beruht, daß die meisten Befehle nicht zum eigentlichen Systemkern gehören, werden solche externen Befehle in Form von Programmdateien bei Bedarf geladen und ausgeführt. Wer nun vermutet, daß aufgrund dieses Prinzips aus Geschwindigkeitsgründen eine Festplatte empfehlenswert ist, hat ins Schwarze getroffen. Außerdem werden

Interne Kommandos

alias
chdir
dirs
echo
errno
eval
exec
exit
export
expr
find
history
keypressed
limits
open
popdir
printf
pushdir
read
readhist
readonly
scrapdir
shift
shinfo
sleep
stty [echoe switch raw onlcr]
tee
test
trap
type
unset

Externe Kommandos

args
banner
cal
chartab
chmod
cmp
comm
cp
cpio
ctags
cut
cw
date
dd
df
diff
du
eval
file
flop
getopt
grep
head
ls
make
memfind
mkdir
more
nm
od
paste
printhex
prof
rm
sed
showcore
sort
strings
strip
tail
touch
tr
version
wait
wc
what

Shellfiles

aliases
backup
basename
clear
dircmp
dirname
install
man
shar

Sprachelemente

for ..; do .. done
until .. ; do .. done
while .. ; do .. done
select ... in ... ; do ... done
case ... in ; ... ; esac
if ..; then ...; [elif ...] [else ...] fi
break
continue
return

Variable der Shell

CDPATH
COLS
DRIVES
ESCAPE
HOME
IFS
LINES
PATH
PS1
PS2
PS3
REPLY
SCAN_CODE
SHELL
SUFFIX
TMPDIR

Ausgabe-Einheiten

aux:
con:
prn:

Bild 1: Befehle und Dienstprogramme der PKS-Shell

von UNIX des öfteren temporäre Dateien erzeugt, die zugunsten einer vernünftigen Arbeitsgeschwindigkeit nicht gerade auf einer Diskette abgelegt werden sollten. Ist keine Festplatte vorhanden, kann man sich zur Not auch mit einer RAM-Disk für diese Daten behelfen. Leider befindet sich keine RAM-Disk im Lieferumfang der PKS-Shell, aber auf dem Public Domain-Sektor gibt es ja genug Programme dieser Art.

Wo die UNIX-Programmdateien sowie eventuelle temporäre Dateien abgelegt werden sollen, kann im Verlauf der Installation der PKS-Shell vom Anwender angegeben werden. Die Installation ist übrigens dank der ausführlichen Beschreibung trotz des aufwendigen Lieferumfangs leicht durchzuführen. In der mir vorliegenden neuen Version V1.02 der Shell wurden jedoch einige Hilfsdateien nicht korrekt installiert. Hier gab es bei der Version 1.01 keine Fehler. (Möglich, daß Ermüdungserscheinungen der Autoren für dieses Problem verantwortlich sind. Wenn man der Erstellungszeit der Dateien Glauben schenkt, so wurden diese nämlich um 3 Uhr nachts zusammengestellt.)

Handbuch überflüssig

Da die PKS-Shell aus einer ganzen Reihe von eingebauten Befehlen und externen Dienstprogrammen besteht (Bild 1), ist es eine wichtige Aufgabe des Handbuchs, diese Befehle vorzustellen und Beispiele zu deren Nutzung zu geben. Dennoch hätte man sich das Handbuch für die PKS-Shell in diesem Punkt fast sparen können - nicht weil es unbrauchbar wäre, sondern weil man sich mit Hilfe des UNIX-kompatiblen *man*-Befehls ausführliche Erklärungen zu allen Funktionen der Shell auf dem Bildschirm anzeigen lassen kann. Ein Blick ins Handbuch erübrigt sich somit in vielen Fällen.

I want more

Wo wir gerade bei der Bildschirmausgabe sind: Hierzu steht unter UNIX der Befehl *more* zur Verfügung. Wie bei einigen anderen Dienstprogrammen wurden die Standardbefehle von UNIX bei der PKS-Shell um neue Eigenschaften erweitert. Das *more*-Kommando ist hierfür ein gutes Beispiel. So geschieht die Bildschirmausgabe von *more* auf einem gesonderten Bildschirm, und es ist mit Hilfe diverser Tasten möglich, den Text beispielsweise vor- und zurückzuscrollen.

Sehr praktisch ist es, daß *more* das WORDPLUS-Dateiformat sowie diverse Bildformate erkennt und solche Dateien im korrekten Anzeigeformat auf dem Bildschirm darstellt. Ruft man more also mit dem Namen einer IMG-Grafikdatei als Parameter auf, wird die Grafik auf dem Bildschirm angezeigt. Bei WORDPLUS-Dateien erfolgt die Ausgabe des Textes mit Schriftattributen. Bilder, die in Textdateien eingebunden wurden, finden jedoch bei der Ausgabe keine Berücksichtigung.

Bildateien können von *more* leider nur auf einem Monochrombildschirm, nicht jedoch in den Farbauflösungen angezeigt werden.

Muttersprache C

UNIX ist zum überwiegenden Teil in C programmiert. Diese Tatsache macht es für den C-Programmierer besonders einfach, Programme zu schreiben, die unter UNIX laufen sollen. Die PKS-Shell stellt einige externe Befehle zur Verfügung, die für den C-Programmierer sehr interessant sein dürften. (Eine Anpassung der Shell an die wichtigsten C-Compiler für den ST sowie an ST PASCAL+ wird übrigens vom Installationsprogramm automatisch durchgeführt.)

So ist es z.B. möglich, mit dem *ctags*-Kommando Verweislisten für C-Quelltexte erstellen zu lassen. Die erzeugten Dateien enthalten alle Funktionsdeklarationen und auf Wunsch auch Typvereinbarungen.

Werden die von *ctags* gelieferten Daten anschließend mit *mkproto* bearbeitet, erhält man eine Datei, in der Funktionsprototypen (gemäß ANSI-Standard) zu allen neu deklarierten C-Funktionen aufgeführt sind. Dies erleichtert die Anpassung älterer C-Quellen an neuere C-Compiler wie TURBO C.

striptease

Neben C-spezifischen Dienstprogrammen bietet die PKS-Shell auch einige Funktionen, die für andere Programmiersprachen geeignet sind.

strip ermöglicht es, die Symboltabelle eines Programms nachträglich aus der Programmdatei zu entfernen. Dies dürfte bei Programmen, die man selber geschrieben hat, zwar kaum notwendig sein, aber ab und zu stößt man auf fremde Programme, die aufgrund einer Symboltabelle unnötig lang sind. Ein prominentes Beispiel stellt der Debugger SID dar.

Ob ein Programm überhaupt eine Symboltabelle besitzt, läßt sich durch das Dienstprogramm *size* feststellen. Man erhält so auch Angaben über die Länge der einzelnen Programmsegmente (TEXT, DATA und BSS).

nm stellt ein weiteres Kommando dar, das mit Symbolen arbeitet. Interessiert man sich dafür, welche Symbole in einer Objekt- oder Programmdatei definiert sind, und in welchem Programmsegment diese Definition erfolgt, so liefert *nm* diese Angaben. Unterstützt wird momentan leider nur das Objektformat von Digital Research, das maximal acht Buchstaben pro Symbol erlaubt und deshalb zunehmend durch neue Formate ersetzt wird. Wünschenswert wäre eine Unterstützung zumindest der wichtigsten neuen Symbolformate wie z.B. des erweiterten Formats von TURBO C.

Kanalisierung

Wie es sich für eine ordentliche Shell gehört, bietet auch die PKS-Shell die Möglichkeit, Ein- und Ausgabekanäle nahezu beliebig umzudefinieren. So können Ausgaben, die eigentlich für den Bildschirm gedacht sind, in eine Datei oder auf den Drucker umgelenkt werden. Benötigt ein Programm umfangreiche Tastatureingaben, lassen sich diese Daten auch aus einer Datei einlesen. Aber Vorsicht: Die Umlenkung von Ein- und Ausgabekanälen ist in der Regel nur bei TOS-Programmen, nicht aber bei GEM-unterstützten Anwendungen möglich. Außerdem arbeitet die Umlenkung von Ein- und Ausgaben erst ab TOS 1.4 halbwegs fehlerfrei. Die alten TOS-Versionen wiesen hier Tücken auf.

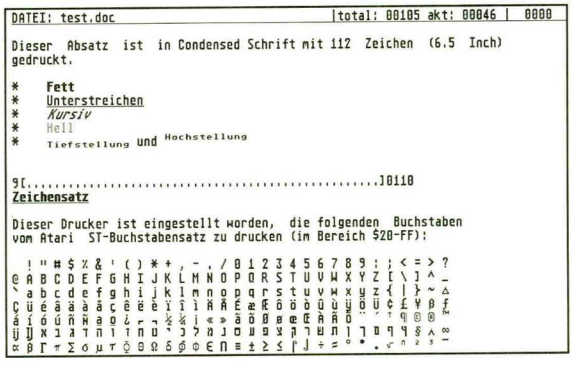


Bild 2: more zeigt WORDPLUS-Datei

VORTEX TECHNOLOGIE UND ZUKUNFT



Mehr Emulation fürs gleiche Geld: DM 498,-*
vortex ATonce mit EGA- und VGA-Monochrom-Graphik[1]

Konfigurieren Sie sich Ihren persönlichen Atari/AT-Computer. Wählen Sie mit welchem Grafik-Mode und welchem Zeichensatz Sie arbeiten möchten (mit der neuen Software Version 3.5 von vortex ATonce). Ganz individuell. Das schnelle Umschalten vom AT-Modus in den Atari-Modus und der volle Protected Mode werden Sie begeistern.

vortex ATonce: ■ Neue Update-Version 3.5 ■ Mit EGA- und VGA-Monochrom-Graphik[1] ■ FontMaster (Möglichkeit den eigenen Zeichensatz zu editieren) ■ vortex HyperSwitch Accessory (Schnelles Umschalten zwischen AT-Mode und Atari-Mode) ■ 80286/16Bit/ 8MHz CPU ■ Norton 6,7 ■ 82% MIPS-Test ■ SMT-Technologie mit vortex CMOS Gate Array, Chip-Level-Emulation und AT-BIOS ■ Windows 3.0 läuft uneingeschränkt im Protected Mode ■ Expanded/Extended Memory-Verwaltung ■ Unterstützung von Fest-/ Wechselplatten-Subsystemen und Floppy-Laufwerken ■ Emulation von Grafik-Modi wie EGA, VGA, CGA, serielle und parallele Schnittstelle und vieles mehr ■ AT-Kompatibilität zum Preis von DM 498,- (*Unverbindlicher empfohlener Endverbraucherpreis) ■ Kostenlose Software-Updates ■ Tägliche Hot-Line

vortex COMPUTERSYSTEME

[1] soweit es die Fähigkeiten des Atari ST/Mega ST zulassen. Alle Firmen- und Produktnamen sind Warenzeichen der jeweiligen Inhaber und urheberrechtlich geschützt.

VORTEX COMPUTERSYSTEME GMBH . FALTERSTRASSE 51-53 . D-7101 FLEIN . TELEFON 07131 / 59 72-0
DTZ DATATRADE AG . LANDSTRASSE 1 . CH-5415 RIEDEN/ BADEN . TELEFON 056/ 82 18 80

Ein weiterer Teil der Dienstprogramme beschäftigt sich mit der Manipulation und Auswertung von Daten. Bei Bedarf sind auch statistische Aussagen über eine Textdatei erhältlich, wenn diese durch Kommandos wie *grep* oder *wc* unter die Lupe genommen wird.

Pfeifen auch für Nichtraucher

Eine nützliche Eigenschaft von an UNIX angelehnten Shells ist die Realisierung sogenannter Pipes. Hierbei handelt es sich um Datenpuffer, die eine Verbindung zwischen mehreren Kommandos herstellen. Sehen wir uns als Beispiel die folgende Kommandozeile an:

```
ls -l | grep „Dez 1990" | wc -l
```

Das Kommando *ls -l* gibt zunächst das Directory des aktuellen Laufwerks inklusive Angaben über die Erstellungszeit der einzelnen Dateien aus. Die Ausgabe von *ls* erfolgt nun aber nicht auf dem Bildschirm, sondern wird über eine Pipe (hierfür steht das Zeichen „|") an den *grep*-Befehl weitergegeben. *grep* extrahiert aus den von *ls* gelieferten Daten alle Zeilen, in denen die Angabe „Dez 1990" enthalten ist. Das Ergebnis wird an *wc* weitergegeben, das die Zahl dieser Zeilen zählt und anschließend auf dem Bildschirm ausgibt. Die obige Kommandozeile zählt somit alle Dateien, die im Dezember 1990 erstellt wurden.

Die Verwendung von Pipes hat in diesem Beispiel dafür gesorgt, daß die Ausgaben der einzelnen Befehle an das jeweils folgende Kommando weitergeleitet wurden und nicht einfach nur auf dem Bildschirm erschienen sind. Es bleibt anzumerken, daß es sich bei den von der PKS-Shell unterstützten Pipes um temporäre Dateien handelt, die alle Ausgaben eines Kommandos aufnehmen und als Eingaben für den nächsten Befehl zur Verfügung stellen. In echten Multitasking-Systemen sind solche Pipes anders realisiert, da bei ihnen mehrere Befehle bzw. Programme gleichzeitig ablaufen können. Dies läßt das TOS des ST jedoch nicht zu.

Eine sehr nützliche Nutzung von Pipes wäre die Kombination der Kommandos *ctags* und *mkproto*. So könnten für einem C-Quelltext direkt Funktionsprototypen erstellt werden. Leider führten Experimente zu Fehlern. *ctags* stürzte bei manchen Parametereingaben ab; beim Versuch, die Ausgaben von *ctags* über eine Pipe an *mkproto* weiterzuleiten, hängte sich das Dienstprogramm beim Schreiben der Prototypdatei auf oder meldete ohne ersichtlichen Grund einen Ausgabefehler.

Ein Blick auf die Uhr

Eine besonders interessante Eigenschaft der PKS-Shell ist die Option, Laufzeitmessungen durchzuführen. Gerade dann, wenn es um die Programmierung zeitkritischer Routinen geht, benötigt man genaue Informationen über die Ausführungszeit eines Programms. Zwar kann man längere Operationen mit einer Stoppuhr messen, aber im Computerzeitalter sollte es bessere Möglichkeiten geben. Leider erlauben es nur wenige Compiler, die Messung von Laufzeiten einzelner Unterprogramme durchzuführen. Mit der PKS-Shell ist es möglich, die Ausführungszeit eines kompletten Programms zu messen. Man erhält so nach dem Programmende automatisch eine Angabe über die seit dem Start verflossene Zeit. Auch die Zeitmessung einzelner Unterprogramme wird mit dem *prof*-Kommando unterstützt. Leider ist dieses nur unzureichend beschrieben und in den mittels *man* einsehbaren Handbuchdateien überhaupt nicht aufgeführt.

Konfiguration

Die Umgebung der PKS-Shell läßt sich leicht an die eigenen Bedürfnisse anpassen. Hierzu existieren unter anderem die sogenannten Shell-Flags. Diese Flags erlauben es, das Verhalten der Shell im Fehlerfall zu bestimmen, und bieten die Möglichkeit, Einfluß auf die Parameterübergabe bei Ein- und Ausgabeoperationen zu nehmen.

Alle Voreinstellungen können in einer sogenannten Profile-Datei abgelegt werden. Diese Datei wird bei jedem Start der Shell ausgewertet, und die in ihr enthaltenen Befehle werden ausgeführt. Auch beim Verlassen der Shell besteht die Möglichkeit, eine spezielle Datei ausführen zu lassen. So können wichtige Einstellungen automatisch beim Programmende gesichert werden.

Unabhängigkeitsbewegung

Nein, es wird nun nicht politisch. Unabhängig sind lediglich die Dienstprogramme der PKS-Shell vom eigentlichen Shell-Programm. Im Klartext: Bei den externen Befehlen der Shell handelt es sich meist um Programme des Typs TTP (TOS Takes Parameters). Es ist also möglich, diese Programme nicht nur aus der PKS-Shell heraus, sondern auch direkt über das Desktop zu starten. Durch geeignete Manipulation der Desktop-Info kann man so das MORE-Dienstprogramm als universelles Ausgabeprogramm für unterschiedliche Dateitypen einsetzen. Wie man dazu vorgehen muß, ist im Handbuch zur PKS-Shell erklärt.

Auch innerhalb von eigenen Programmen kann man die Befehle der PKS-Shell nutzen, sofern diese beim Programmstart aktiv war.

Die Theorie hierzu: Eine Shell richtet an der Adresse der Systemvariablen *_shell_p* einen Zeiger auf ihren Parser ein, der die Shell-Kommandos auswertet. Springt man diese Adresse an und liegt der Pointer auf eine für die Shell gedachte Kommandozeile auf dem Stack, wird diese bewertet, und die Kommandos werden so ausgeführt, als ob man sie per Hand eingetippt hätte.

Dieses Verfahren wird auch von der PKS-Shell unterstützt. Das Handbuch enthält leider nur sehr spärliche Angaben zur Übergabe von Kommandozeilen mittels *_shell_p*, womit wir bei der Programmbeschreibung angelangt wären.

```
 [23] cd c:
 [32] ls -l
AUTO\           drw--u       0 18:44:32  4 Nov 1990
CPX\            drw--u       0 18:44:32  4 Nov 1990
CONTROL.INF     frw--n      24 22:10:30 28 Aug 1990
DCONFIG.PRG     frw--n    6946 14:36:40 25 Nov 1990
DESKICON.RSC    frw--n   23734 21:52:58 23 Sep 1990
DISKDEMO.PRG    frw--n  117350 18:03:50 16 Dez 1990
DISKUS.PRG      frw--n  144564 18:11:08 16 Dez 1990
EMULATOR.ACX    frw--n    7493 00:00:22 15 Nov 1989
HCONFIG.PRG     frw--n    2940 17:31:32 16 Dez 1990
HDDRIVER.PR     frw--n    8630 21:06:08  3 Dez 1990
HDDRIVER.PRG    frw--n    8662 08:35:26 17 Dez 1990
NEWDESK.IN      frw--n    2914 22:16:56 13 Dez 1990
NEWDESK.INF     frw--n    2973 23:11:34 17 Dez 1990
NVDICONF.ACX    frw--n    2925 16:21:16  2 Nov 1990
PROTECT.ACC     frw--n    1468 15:36:40  4 Aug 1990
SNAPSHOT.ACC    frw--n    7239 11:26:44 17 Aug 1989
SPOOLTT.ACX     frw--n    1715 22:22:02  8 Sep 1990
XCONTROL.ACC    frw--n   41033 18:24:40 13 Aug 1990
 [32] ls -l | grep "Dez 1990" | wc -l
7
 [32] size xcontrol.acc
xcontrol.acc        : .text  31546 .data   8602  .bss  10528 .syms      0
 [32]
```

Bild 3: Die PKS-Shell im Einsatz

Die Dokumentation

Das Handbuch der PKS-Shell hat einen Umfang von ca. 180 Seiten und ist in mehrere Abschnitte gegliedert. Neben dem Hauptteil, der sich mit dem Leistungsumfang der einzelnen Befehle und Dienstprogramme beschäftigt, findet man auch einige allgemeine Bemerkungen zum UNIX-Konzept.

Besonders hervorzuheben ist die Beschreibung eines mitgelieferten Shell-Programms, das ein Backup von Dateien erlaubt. Hier wird ausführlich erklärt, wie man aus den Befehlen und Dienstprogrammen der PKS-Shell eine komplexe Anwendung zusammenstellen kann.

Wer sich zu den fortgeschrittenen Computer-Anwendern zählt, dürfte mit dem Handbuch zur Shell gut zurechtkommen, wenn auch an einigen Stellen ausführlichere Erklärungen angebracht wären. Für den Anfänger in Sachen Shell sind Verständnisschwierigkeiten wohl nicht zu vermeiden. Schließlich ist mir hier und da aufgefallen, daß manche Befehlsoptionen fehlerhaft erklärt sind oder vertauscht wurden, was zu Verwirrungen führen kann.

Um einen tieferen Einblick in UNIX zu bekommen, reicht die Programmbeschreibung zur PKS-Shell verständlicherweise nicht aus, hier sollte man zu zusätzlicher Literatur greifen [1],[2]. Auch die Serie „Programmer's Toolbox-Dateien" in der ST-Computer enthält viele Informationen, die in den Umgang mit UNIX-ähnlichen Systemen einführen.

Zum guten Schluß

Die PKS-Shell ist nicht nur eine neue Shell für Atari ST und TT, sondern soll auch auf einen Kontakt mit „echten" UNIX-Systemen vorbereiten. Da ich mich als Nicht-Informatiker ursprünglich kaum mit UNIX beschäftigt hatte, hat mir die Arbeit mit der PKS-Shell einiges gebracht. Inzwischen arbeite ich auch ab und zu an Rechnern, die unter UNIX laufen und profitiere dabei von der UNIX-Erfahrung, die ich durch die PKS-Shell gesammelt habe.

Für UNIX-Interessierte, Programmierer (speziell in C) und diejenigen, die das Arbeiten mit Shells dem Arbeiten mit der Maus vorziehen, ist die PKS-Shell ein Schritt in die richtige Richtung. Einzeln ist die Shell für 168 DM erhältlich, in Kombination mit dem Texteditor PKS-Edit (getestet in der ST-Computer 12/90) ist man mit 248 DM dabei.

US

Bezugsadresse:

Pahlen & Krauß Software
Dieffenbachstraße 32
W-1000 Berlin 61
Tel.: 030/7865945

Literatur:

[1] Detering, „UNIX Handbuch", SYBEX-Verlag
[2] Kernighan, Pike, „Der UNIX-Werkzeugkasten", Hanser-Verlag

Richter

Distributor

wünscht Ihnen ein erfolgreiches gesundes glückliches, kurzum ein einmaliges 1991 !

ACHTUNG NEUE PREISE !

Basisboard MB 12 von 4 auf 6 MB	1548 DM
Potato 4er Romexp.(für Emulatoren)	188 DM
Colibri DIN A4 Scanner incl. Syntex	1790 DM
600 DPI KIT für Laserdrucker	380 DM

Syntex die OCR NEU 199.-DM

AUGUR * REPROK/LAGER * CHILI * 14 MB *

Fordern Sie noch heute weitere Unterlagen an !

Wir werden Sie auch 1991 mit unseren Produkten verzaubern ! Professionelle Produkte, schnelle Lieferung und guter Service werden auch 1991 unser Maßstab sein !

H.Richter Distributor * Hagener Str. 65 * D 5820 Gevelsberg * Fax 02332/2703
Inh. Hans Richter

Gedruckt mit 600 * 300 DPI Laser KIT

Telefon 02332/2706

Preise gültig bis 1.2.1991

Bei uns werben bringt

GEWINN

Sprechen Sie mit uns.
Heim Verlag ☎ 06151/56057

Akzente Softwarevertrieb

Signum !2	335.–	GFA-Basic V 3.5	215.–
Script II	235.–	GFA-Assembler	129.–
Adimens ST plus	314.–	AT-Speed	429.–
Aditalk ST plus	314.–	Speed-Bridge II	48.–

Steckbare Erweiterung auf 2 MB 448.– auf 4 MB 698.–
SCSI Speed Drive 50, 28 ms, SCSI-Port 1348.–
SCSI Ultra Speed Drive 40, 19 ms, 64 KB Cache 1448.–
Preisliste kostenlos !

Akzente Softwarevertrieb
7080 Aalen · Postf. 1672 · Schlehenweg 12
Tel. (0 73 61) 3 66 06 · Fax (0 73 61) 3 66 07

Paint it Black

Mega ST im neuen Gewand



Sagen Sie mal ganz ehrlich: Das beige-graue Büromaschinen-Standard-Gehäuse Ihres ST reißt Sie doch auch nicht vom Hocker? Mich jedenfalls nicht. Und wenn bei rauchenden Zeitgenossen sich der Rechner durch Einfangen des Nikotins ungezählter Zigaretten in die entsprechende Farbe gewandelt hat, dann tut das seinem Anblick ebenfalls keinen Gefallen. Schwarz dachte ich, sei die richtige Farbe für den Silikonkameraden. Ein wenig poppig abgesetzt vielleicht, um nicht die gleiche Langeweile heraufzubeschwören.



Es kam der Tag, an dem ich ihm mit dem Schraubenzieher zu Leibe rückte. Entblättert war er schnell, und so ging's dran, die Gehäuseteile zu reinigen. Eine Behandlung mit feiner Stahlwolle, die nichts zerkratzt, brachte das gewünschte Ergebnis: Zum einen ist der Dreck der Jahre abgetragen, zum andern der Untergrund für den folgenden Lack gut angerauht. Versuche mit Schleifpapier führten zu deutlich sichtbaren Kratzern. Um dem Lack maximale Haftung zu ermöglichen, habe ich die Kunststoffteile anschließend noch mit Reinigungsalkohol entfettet.

Im Farbenhandel gibt es Grundierungen, die speziell für Kunststoffe ausgelegt sind. Denn normale Kunstharz- oder Acryllakke haften nicht ohne weiteres auf Kunststoffuntergründen. Diese müssen entsprechend grundiert werden. Zu beachten ist, daß die Grundierung die Gehäuseteile nicht angreift und auf den endgültigen Lack abgestimmt ist. Ich habe Farbe aus Spraydosen gewählt - sicherlich nicht die umweltfreundlichste Lösung, doch für eine Lackierung mit Kompressor etc. wären die Mengen an Teilen, die ein Atari zu bieten hat, viel zu gering. Andererseits wird man mit dem Pinsel nie die feine Oberfläche erreichen, und Lüftungsgitter etc. werden zur Qual.

Nachdem (natürlich in einem dafür geeigneten Raum, nie im Freien oder gar in der Wohnung!) die Grundierung aufgebracht ist, heißt es ca. einen Tag zu warten, bis der richtige Lack aufgebracht werden kann. Lackieren erfordert einige Übung, und so habe ich mit den Teilen begonnen, auf denen von eventuellen Fehlern nicht viel zu sehen ist: Rechner- und Monitorgehäuse.

Das Auslegen der Sicken auf der Tastatur erfordert eine ruhige Hand und einen dünnen Pinsel. Genauso viel Geduld sollt man beim Abkleben der Embleme haben, die sich nicht entfernen lassen. Bei Drukker und Monitor gibts noch ein Problem: an den Chassis der Geräte hängen Teile, die sich nicht lösen lassen, aber trotzdem lackiert werden müssen. Ich habe die jeweiligen Innereien in Plastikfolie eingeschlagen. Beim Drucker ragten die Hebel und beim Monitor der Rahmen um die Bildröhre heraus. Apropos Monitor: Vorsicht mit der Hochspannung von einigen Tausend Volt an bestimmten Teilen im Innern! Den Monitor vor dem Zerlegen zur Sicherheit einen Tag nicht einschalten!

Tja, und nach der Lackierung dauert's noch zwei bis drei Tage, bis der Lack soweit durchgehärtet ist, daß man die Geräte wieder zusammenbauen kann. Ich habe eine abrieb- und auch ausreichend kratzfeste Oberfläche erhalten. Der Lack blättert also bislang noch nicht. Selbst auf Maus und Tastatur bleiben nur normale Gebrauchsspuren.

Die Krönung sind natürlich die schwarzen Kabel: Für die Tastatur sind sie als Sonderzubehör erhältlich. Eine abgeschnittene und neu eingesetzte Joystick-Verlängerung besorgt das für die Maus auf äußerst preiswerte Art. (Beachten Sie, daß alle Pins durchgeschliffen sind.)

Die Wahl der Farben bleibt letztlich Gesschmackssache - gerade mit Schwarz sollte man natürlich vorsichtig sein, wenn der Rechner am Fenster und evtl. in der Sonne steht. Da kann ihm schon mal zu warm werden. Und wie Sie auf den Photos sehen, habe ich die Tasten selbst vom Färben ausgespart. Selbst wenn ich sie noch lackiert hätte: Die Bezeichnungen darauf zu malen oder gar mit Rubbelbuchstaben aufzubringen, das wollte ich mir nicht antun. Aber vielleicht nimmt sich ja mal ein Tastenkappen-Produzent meiner Design-Wünsche an und bietet welche in etwas ansehnlichen Farben an.

IB

Die neue Generation der TEXTVERARBEITUNG

Write ON-Textverarbeitung

Die herausragende Drucktechnologie:

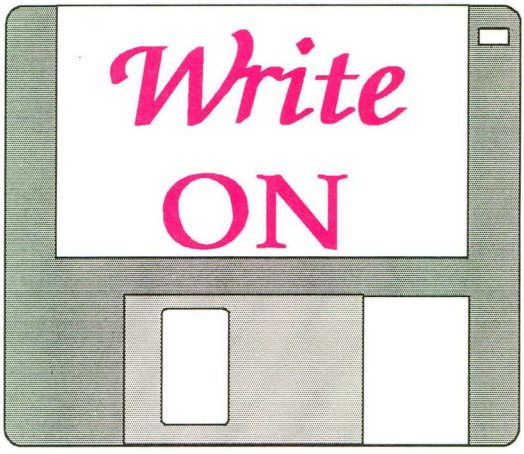
Write ON erlaubt die Nutzung druckerinterner Schriften (auch proportional in Blocksatz) mit grafischen Schriften und Symbolen in einem Dokument. Der Bildschirm zeigt ständig, was auf das Papier kommt. (Vollständiges WYSIWYG)

Das Bedienungskonzept:

Der Einsteiger erlernt *Write* ON einfach und schnell mit der Maus und Pull-Down Menüs. Dabei lernt man bei Bedarf automatisch die schnelle Bedienung aller Funktionen über durchdachte deutsche Tastenkommandos. Als Clou ist es möglich, Befehle, Menüauswahl und Texte nach eigenem Bedarf auf alle Tasten zu legen. Damit können sogar neue Funktionen erzeugt werden.

Das Layoutkonzept:

Write ON hat das praktische Layoutkonzept moderner DTP-Systeme in der Textverarbeitung integriert. Nachträgliche Änderungen sind genauso einfach wie die Voreinstellungen nach Ihrem täglichen Bedarf.



Der Standard:

- Umfangreiches Handbuch zum Einstieg in die Textverarbeitung
- komfortables Installationsprogramm
- automatische deutsche Silbentrennung
- Serienbrieffunktion
- 10 Fonts sind im Lieferumfang enthalten
- eigener Fonteditor ist enthalten
- Fontkonverter für SIGNUM-Fonts
- Textimport Word-Plus und ASCII
- alle gängigen Druckertreiber sind vorhanden
- Import beliebig großer IMG-Grafiken
- Snapschot Accessory zum Import aus anderen Programmen u.v.m.
- flexible Gestaltung durch Absatz und Seitenlayouts
- unterstützt Farb-, Monochrom- und Großbildschirme
- läuft unter allen bekannten TOS-Versionen und unter GEM 2.2
- viele Druckertreiber für 9-, 24-Nadeldrucker SLM 804, HP Desk- und Laserjet
- Serienbrieffunktion in Verbindung mit allen gängigen Datenbanken und Adressverwaltungen u.v.m

Write ON -Die Textverarbeitung mit System:

Wenn Ihr Bedarf an Textverarbeitung und Funktionen im Laufe der Zeit steigt, können Sie jederzeit auf That's Write Profi, That's Write Postcript oder That's Write Postscript Plus umsteigen. Ihre Texte werden ohne Veränderung gelesen, die Bedienung bleibt genauso einfach, lediglich die Ihnen zur Verfügung stehenden Funktionen steigen. Profitieren Sie von diesem Konzept.

Der Preis und die Leistung:

Write **ON**
kostet **148,– DM**

und bietet dem registrierten Benutzer einen Hotline Service. Sollten Sie jetzt noch Fragen zu Write ON haben, dann rufen Sie uns doch einfach einmal an oder lassen sich unser ausführliches Informationsblatt zusenden.

Preise sind unverbindlich empfohlene Verkaufspreise

Heim-Verlag

Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 06151 - 56057
Telefax 06151 - 56059

BESTELL - COUPON

Bitte senden Sie mir

___ *Write* ON a 148,– DM zuzüglich 6,-- DM Versandkosten (Ausland DM 10,–)

Name, Vorname _____

Straße _____

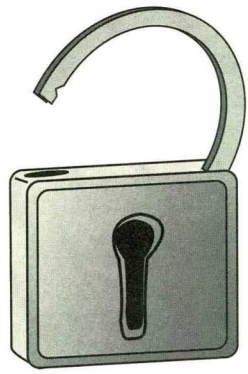
PLZ, Ort _____

unabhängig von der bestellten Stückzahl

oder benutzen Sie die eingeheftete Bestellkarte

in Österreich:
RRR EDV GmbH
Dr. Stumpf Str. 118
A-6020 Insbruck

in der Schweiz:
Data Trade AG
Landstr.1
CH-5415 Rieden-Baden



1st_Lock

Der Schlüsselknecht

Datenverschlüsselung ist ein heikles Thema. Nicht nur in Firmen oder Universitäten müssen wichtige Daten codiert werden, auch im privaten Bereich ist es zunehmend sinnvoll, seine Daten vor fremdem Zugriff zu schützen (Grundgesetz Art. 10, Bundesdatenschutzgesetz etc.). Die Firma LogiLex bietet für den ST ein passendes Programm an, das die Aufgabe der Ver- und Entschlüsselung übernimmt.

Das Arbeitsprinzip von *1st_Lock*, so der Name des Programms, ist recht simpel. Es klinkt sich in die Betriebssystemvektoren für Schreiben und Lesen auf Medien ein und erweitert dadurch das Betriebssystem. Dadurch arbeitet es resident im Hintergrund und bleibt für den bloßen Anwender (fast) immer unsichtbar.

Die Installation des Programms ist denkbar einfach: Ein Programm namens *1st_Lock.Prg* wird in den Auto-Ordner gelegt und ein *1st_Lock.Sys* in das Wurzelverzeichnis der Platte, fertig. Das Hauptprogramm aus dem Auto-Ordner ist lediglich 456 Bytes lang. Daß in diesem Programm nicht alle Routinen enthalten sein können, ist verständlich. Damit wird lediglich *1st_Lock.Sys* nachgeladen, das die eigentlichen Routinen zur Verschlüsselung enthält. Dieses Hauptprogramm ist 25350 Bytes lang und kann damit auch problemlos mit Rechnern mit nur 512 kB betrieben werden.

Los geht's

Nach dem Start aus dem Auto-Ordner muß der Benutzer zunächst sein Paßwort eingeben, um überhaupt auf die Platte zugreifen zu dürfen. Ein falsches Paßwort wird sofort registriert und auf Wunsch in einer Protokolldatei gespeichert, so daß der Superuser des Systems anhand des falsch eingegebenen Paßworts eventuelle Sicherheitslücken des Systems sofort korrigieren kann. Ist die Sicherheitsabfrage überstanden, kann sich der Anwender daran machen, mit seinem ST ganz normal zu arbeiten - bis auf einige eventuelle Kleinigkeiten. So ist es nämlich möglich, dem Benutzer auf einige Partitionen bzw. Laufwerke nur Lesezugriff zu geben oder gar den Zugriff gänzlich zu verweigern. Andererseits ist es natürlich auch problemlos möglich, jedem Anwender den wahlfreien Lese- und Schreibzugriff zu geben.

Vor der ersten Benutzung von 1st_Lock sollte man die Plattenpartitionen „umschlüsseln". Dazu ist es lediglich notwendig, im mitgelieferten Accessory „umschlüsseln" anzuwählen. Danach sind die gewählten logischen Laufwerke nur noch mit 1st_Lock lesbar. Natürlich sollte sich ein verantwortungsbewußter Anwender vor einer solchen Aktion ein Backup machen. Dazu liefert LogiLex gleich ein Backup-Programm mit, das zwar kleine Mängel aufweist, für den normalen Gebrauch jedoch mehr als ausreichend ist. Eine 10 MB-Partition ist in etwa 12 Minuten umgeschlüsselt und kann auch nur von 1st_Lock in den Urzustand zurückgebracht werden.

Schutz perfekt

Nach der Umschlüsselung ist der Schutz perfekt. Wird die Festplatte jetzt ohne 1st_Lock gestartet, kann kein Hacker mehr etwas ausrichten, denn er sieht nur noch „1st_Lock"- und „Logilex"-Dateien, die allesamt keinen Inhalt besitzen. Selbst wenn sich der Unhold auch eine Kopie von 1st_Lock zulegt, kann er nichts ausrichten, denn jede Kopie des Programms codiert ihre Daten anders. Die Chance, hier einen „zufälligen Treffer" zu erlangen, ist ungefähr so groß wie sechs Richtige im Lotto. Voraussetzung hierfür ist natürlich, daß man sein individuelles 1st_Lock-System nicht an Dritte weitergibt. Damit dürfte wohl ein sehr großer Schutz für alle wichtigen Daten gewährleistet sein.

Wie verhält sich 1st_Lock nun beim Betrieb? Recht einfach. Zum einen ist es sehr einfach zu bedienen, da nicht viele Menüpunkte vorhanden sind. Das erleichtert dem Einsteiger die Bedienung des Programms. Eine Tastaturunterstützung wäre zwar wünschenswert, ist jedoch nicht vorhanden, was der Betriebssicherheit aber keinen Abbruch tut.

Weiter oben sprach ich bereits an, daß man für jeden Anwender Zugriffsrechte verteilen kann. Da stellen sich natürlich einige Fragen. Richtig, die Anzahl - 32 Zugriffs-Levels lassen sich frei vergeben. Obwohl diese Zahl auf Anhieb recht klein erscheint, wird sie doch sehr groß, wenn man bedenkt, daß man ein Zugriffs-Level auch mehreren Personen geben kann. Dadurch steigt die Anzahl der möglichen Benutzer gleich um ein Vielfaches.

Protokoll

Treibt tatsächlich ein Hacker sein Unwesen und versucht, in das System einzudringen, werden Sie als Superuser natürlich darüber informiert. Fehlzugriffe werden sofort und unvermeidlich protokolliert. Bei der Anzeige des Fehlzugriffs erhalten Sie die genaue Uhrzeit, den Namen und das falsche Paßwort des Benutzers, denn aus dem Paßwort lassen sich oft Rückschlüsse auf den Hacker ziehen. Auf jeden Fall merken Sie aber, daß der entsprechende Anwender sein Paßwort ändern sollte.

Sicher ist 1st_Lock auch in einer anderen Hinsicht: Das Programm stellt gleich zwei Superuser zur Verfügung. Sollte nämlich der eine Zugang aus Sicherheitsgründen vom 1st_Lock-System gesperrt worden sein, können Sie mit dem zweiten Account den ersten Superuser-Zugang neu installieren. Damit dürfte es dann aber auch wirklich keine Probleme mehr geben.

Sicher können Sie mit 1st_Lock auch in anderer Hinsicht sein, denn Viren haben kaum noch eine Chance. Das Programm verhindert unerlaubte Schreibzugriffe auf Diskette oder Platte (wichtig etwa für Boot-



Bild 1: Im Accessory befinden sich verschiedene Hilfsfunktionen.

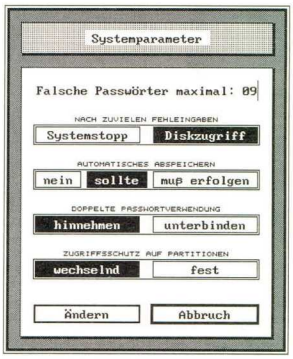


Bild 2: Hier lassen sich unterschiedliche Systemparameter einstellen.



Bild 3: So werden Benutzer legitimiert.

Sektoren) und überprüft ständig alle wichtigen Systemvektoren auf Virenbefall und Veränderung. Unerklärlicherweise ist dieses Feature in der Anleitung selbst nicht beschrieben, sondern nur auf der Verpakkung kurz angedeutet. 1st_Lock ist übrigens resetfest und muß daher nach einem Warmstart nicht neu geladen werden.

Tools

Einige Tools, die man häufig benötigt, werden gleich mitgeliefert. So gibt es das oben bereits erwähnte Backup-Programm *Turtle-Backup*, das für den Normalbedarf vollkommen ausreichend ist. Weiterhin gibt's noch ein „Doppel"-Programm, mit dem sich überprüfen läßt, ob sich auf einer Partition doppelte Dateien finden. Das spart Platz für das Backup. Mit einem weiteren Programm, *Killer*, löschen Sie Dateien nach einem bestimmten Kriterium, etwa alle **.DUP*-Dateien auf Partition C. Zu guter Letzt existiert noch das *Check*-Programm, mit dem sich Check-Summen über Datei(en) errechnen lassen. Dadurch ist leicht feststellbar, ob sich vielleicht irgendwo ein Virus eingeschlichen hat. Leider läßt sich so nicht feststellen, wo sich genau etwas verändert hat, sondern nur, daß innerhalb der gefundenen Datei(en) eine Veränderung aufgetreten ist.

1st_Lock läuft uneingeschränkt auch auf dem TT sowie allen Großbildschirmen. Es bietet einen optimalen Schutz für alle Rechner, auf denen mit wichtigen Daten gearbeitet wird. Der Preis von DM 189,- ist gut kalkuliert und akzeptabel.

MP

Bezugsadresse:

LogiLex
Gerhard Oppenhorst
Eifelstraße 32
5300 Bonn 1
Tel. (0228) 658346

MACH DIR EIN BILD

Mit **TOUCH-UP** erhalten Sie ein vollwertiges Programm für die Bearbeitung von Rasterbildern. Beliebige Bildgrößen mit frei einstellbarer Auflösung und das Erzeugen von **GRAUTON** - Bildern sind einige der Extras, die Touch-Up über das Niveau einer Scanner-Software herausheben. Dazu gibt es natürlich den hochwertigen und handlichen Migraph - **SCANNER**.
Touch-Up: 298,- DM*, Touch-Up und Scanner: 798,- DM*.

* unverbindliche Preisempfehlung



COMPUTERWARE

Im gut sortierten Fachhandel oder bei:
Computerware • Gerd Sender • Weißer Straße 76 • 5000 Köln 50 • Telefon: 02 21 - 39 25 83 • Telefax: 02 21 - 39 61 86
Schweiz: Data Trade AG Zürich, Telefon 0 56 - 82 18 80 • Österreich: Reinhart Temmel GmbH, Telefon 06 62 - 71 81 64

VIDEODAT - und der Computer sieht fern ?

Eigentlich waren die Pioniere der Datenfernübertragung (kurz: DFÜ) seinerzeit angetreten, mit neuartigen Übertragungstechniken den papierlosen Informationsaustausch zu fördern, die Mär vom „papierbefreiten Büro" machte die Runde. Daß der Einzug moderner Datenverarbeitung in unsere Schreibstuben das Papier nicht überflüssig machte (eher das Gegenteil davon), wissen wir bereits.

Dennoch gibt es verschiedene Datenübertragungssysteme, die einerseits einen wahrhaftigen Boom erleben (das gute alte Telefon beispielsweise oder neuerdings Fernkopieren = Telefax), andererseits ein Mauerblümchendasein fristen (Bildschirmtext, kurz: BTX, oder das hoffnungslos veraltete Telex-System), um nur einige zu nennen. Andere Medien sind zu einer wahren Fundgrube der Computerhobbyisten geworden: Mailbox. Welcher DFÜ-Freak hat noch keinen Account (Zugangsberechtigung) einer der bekannten Hobbynetzwerke wie Fido, Magic oder Zerberus?

Selbst ernsthafte Arbeiten werden per Mailbox abgewickelt, es gibt sogar ein weltumspannendes Wissenschaftsnetz (in Deutschland: „WIN", in Europa: „IXI") oder firmeneigene Verbindungen, die oftmals Außenstehenden gerne gegen klingende Münze überlassen werden. Über alle bekannten Informationsaustauschsysteme wacht noch immer ein eifersüchtiger Riese: die Deutsche Bundespost TELEKOM. Gleichgültig, über welche Verfahren Daten transportiert werden sollen, die (liebe) POST ist immer mit dabei. Ob es nun über die normale Telefonleitung oder das exklusive DATEX-Netz geht, selbst bei dem geplanten Mobildatenfunknetz oder per Satellitdirektempfang, immer hält der TELEKOM-Goliath eine Hand auf (für Gebührenzahler) und eine Hand über den Diensten (als Prüfinstitution).

Eine andere Seite modernen Informationstransports läuft aber nicht über die „hoheitlichen" Telefonkabel, sondern kommt quasi kostenlos mit dem gewohnten Fernsehbild bei uns zuhause an. Gemeint ist das System VIDEOTEXT (international TELETEXT, kurz: TXT genannt), das als „Bildschirmzeitung" von speziellen Fernsehgeräten wiedergegeben werden kann. Videotext funktioniert fast so wie das BTX-System der Bundespost (und wird deswegen sehr oft mit ihm verwechselt), hat aber zwei entscheidende Unterschiede: 1. Während beim BTX die Post das Rechnernetz unterhält und verschiedenen Anbietern eine Seitenkapazität vermietet, liegt die Verantwortung für den Inhalt von Videotext bei den einzelnen Rundfunkanstalten. 2. Beim BTX sind der Heimfernseher oder ein Homecomputer mit entsprechendem Decoder per Telefonleitung mit dem Zentralrechner verbunden, und man kann mittels Tastatur in einen eingeschränkten Dialog mit dem System treten. Videotext funktioniert leider nur in eine Richtung, und zwar vom Rundfunksender zum Teilnehmer zuhause.

Nicht nur hören und sehen

Es war im Jahre 1986, als der WDR-Computerclub auf der Kölner Messe C'86 einen Riesengag landete: das Duftfernsehen. (Ja, Sie haben richtig gelesen!) Da flimmerten die herrlichsten Landschaftsbilder über eine Mattscheibe, und wie von Geisterhand wurden einige Duftspraydosen angesteuert, die ein passendes Riecherlebnis verbreiteten. Eine laue Meeresbrise wurde sogar mit einem Ventilator verstärkt.

Obwohl dem duftenden Unterfangen fast nur Schmunzeln entgegengebracht wurde, hatte das Prinzip einen durchaus ernsthaften Hintergrund. Man stelle sich vor: Während einer Lektion des Telekollegs kommen Übungsblätter und wichtige Erläuterungen ganz aktuell aus einem Computerdrucker. Spezielle Versuchsaufbauten zuhause bei den Lernenden sind vom Lehrer im Fernsehstudio ansteuerbar. So sind aber auch naheliegendere Anwendungen parallel zum Fernsehempfang denkbar. Der Westdeutsche Rundfunk (WDR) überträgt seit 1986 während der Sendungen „Computerclub" und „WM - Wissenschaftsmagazin" eine Menge an Zusatzinformationen (ca. 550 kB), von denen der „normale" Zuschauer nichts wahrnimmt. Diese Informationen können wahlweise redaktionell bearbeitete Texte sein, sich als Grafik, Programm-Listing oder fertig lauffähiges Computerprogramm herausstellen.

Ursprünglich hatte Wolfgang Back (Redakteur beim WDR) die Idee, ähnlich wie seine Kollegen bei Radio Hilversum, fertige Computerprogramme seinen Zuhörern möglichst kostengünstig zukommen zu lassen. Man übertrug anstelle von gesprochenen Worten die binären Daten als Audiosignal. Das hatte natürlich den gravierenden Nachteil, daß für diese Zeitspanne jegliche Moderation ausfiel.

Das Verfahren, digitale Impulse als Töne während einer Rundfunk- oder Fernsehsendung zu übertragen, wurde sehr schnell wieder fallengelassen. Eine andere Lösung mußte her. So ersannen Manfred Fillinger und Michael Wiegand in Koordination mit Wolfgang Back (leitender Redakteur) und Christian Bellin (freier Mitarbeiter beim WDR) ein Prinzip, das den freien Raum im Fernsehbildsignal nutzt. Eine neue Idee namens VIDEODAT war geboren. Um zu verstehen, wie das funktioniert, müssen wir einen kleinen Ausflug in die Fachliteratur der Radio- und Fernsehtechniker unternehmen.

Das Fernsehbild nach europäischer Norm ist in 625 Zeilen aufgeteilt (vgl. Bild 1). Würde nun der Kathodenstrahl der Bildröhre jede Zeile nacheinander von oben nach unten auf die Mattscheibe zeichnen, könnten wir dies kaum aushalten, das Bild würde flimmern. So teilt man das darzustellende Fernsehbild in zwei Hälften auf und läßt vom ersten Halbbild nur die Zeilen mit ungerader Nummer, danach vom zweiten Halbbild die Zeilen mit gerader Nummer von oben nach unten beschreiben. Dadurch ist das Bild wesentlich ruhiger. Pro Sekunde werden 25 Vollbilder, d.h. 50 Halbbilder auf diese Weise auf die Mattscheibe gezeichnet.

Was der Bildschirm nicht zeigt

Interessant ist vor allem die Tatsache, daß im Sendesignal noch Zeilen vorkommen, die später überhaupt nicht auf der Mattscheibe zu sehen sein werden (außer man verstellt den sogenannten Bildfangregler). So tragen die Zeilen 1 bis 23, 311 bis 336 und 623 bis 625 keine Bildinhaltsinformationen. Das soll nun aber nicht heißen, daß diese Zeilen keinen Zweck erfüllten. In den sogenannten Austastlücken sind wichtige Impulse versteckt, die u.a. der Bildröhre signalisieren, wo ein neues Halbbild beginnt. Die ersten Zeilen eines jeden Vollbildes dienen meist postinternen Zwecken (Meß- und Prüftechnik) und sind für niemanden zugänglich. In den weiteren Zeilen bis 23 versteckt sich das Videotext-Signal, wenn die entsprechende Rundfunkanstalt diesen Dienst überhaupt ausstrahlt. Das sind zur Zeit lediglich ARD, ZDF, die 3. Programme und SAT1. Sogenannte Bildendeinformationen enthalten die Zeilen 623 bis 625.

Dem Kathodenstrahl muß nun genügend Zeit gelassen werden, um bei jedem Halbbild von der untersten Bildzeile wieder zur obersten zu gelangen. Er wird hierfür einfach dunkel geschaltet, aber Informationen laufen dennoch ein. Diese Zeit könnte man doch andersweitig sinnvoll nutzen, so die Idee der Videodat-Erfinder (siehe oben).

Das Datenfensterchen

Im ersten VIDEODAT-Verfahren aus dem Jahre 1986 konnte man keinen Anspruch auf Informationsraum in den 2 Austastlücken erheben, weil dort beim WDR das Videotext-Signal senderseitig hinzugemischt wird, außerdem wäre der technische Aufwand sehr groß. Es mußte eine Lösung gefunden werden, die es der Redaktion Computerclub erlaubte, die VIDEODAT-Informationen schon bei der Produktion der Sendung dem Aufnahmeband (MAZ) hinzuzugeben.

Um zu verstehen, wie die Realisierung später aussehen konnte, muß man wissen, daß die magnetische Bildaufzeichnung einer Sendung sich bei der Ausstrahlung lediglich in den Zeilen 24 bis 310 (1. Halbbild) und 337 bis 622 (2. Halbbild) wiederfindet. Die anderen Zeilen, also die beiden Austastlücken und das Bildendesignal, kommen erst bei der Erzeugung des Sendersignals während der Ausstrahlung hinzu.

Man hatte einen Weg gefunden: Bis vor kurzem war bei den genannten Sendungen des WDR in der oberen linken Ecke

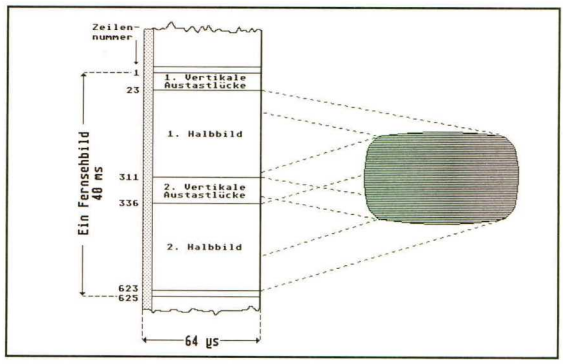


Bild 1: So baut die europäische Fernsehnorm unser TV-Bild zusammen.

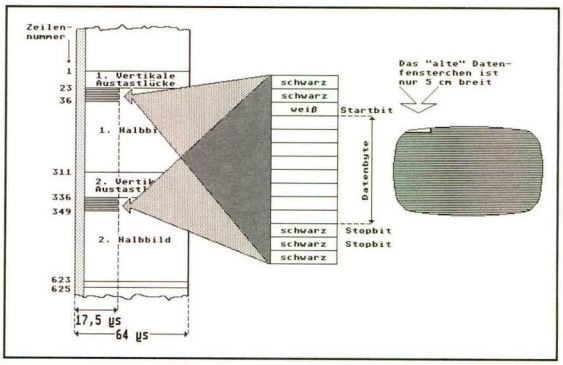


Bild 2: Das veraltete VIDEODAT-Verfahren benutzte noch für jedes Informations-Bit eine eigene Zeile.

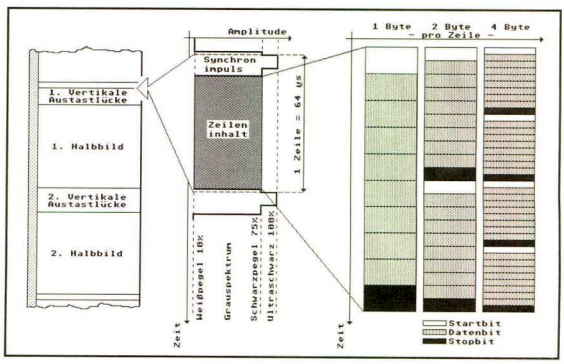


Bild 3: Die Dateninformation liegt im Inhalt unbenutzter Bildzeilen des Fernsehsignals.

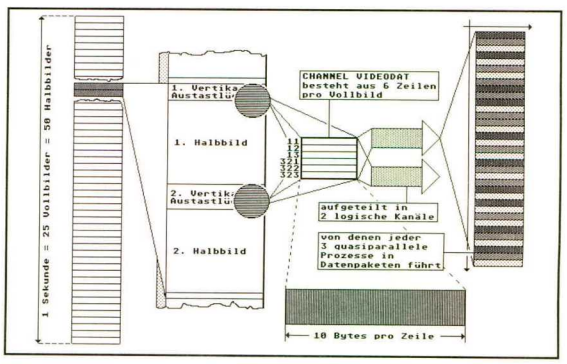


Bild 4: So setzt sich das Signal von CHANNEL VIDEODAT zusammen.



Bild 5: Die genügsamen Pull-Down-Menüs des Programms VIDEODAT 1200.

ein kleiner Balken zu sehen, der abwechselnd schwarz oder weiß leuchtete. Man nannte diesen „digitalen" Balken liebevoll „Datenfensterchen", wobei sich viele Zuschauer über entsprechende Bemerkungen in den jeweiligen Sendungen wunderten. Viele Fernsehgeräte waren so großzügig in der vertikalen Bildaufteilung eingestellt, daß dieser blinkende Balken überhaupt nicht zu sehen war.

Das Rätsel ist schnell gelöst: Dieses erste Verfahren (Name „VD 300") benutzte jeweils 13 Zeilen unterhalb der Austastlücken (also im echten Bildbereich), allerdings nicht in der vollen Zeilenbreite, was auf dem Schirm nur etwa 5 cm (von links) ausmachte (siehe Bild 2). Zeile 25 bzw. 338 war immer ein Weißsignal als Start-Bit, die Zeilen 26 bis 33 und ihr Äquivalent die Zeilen 339 bis 346 bildeten die 8 Informations-Bits (wer weiß es? richtig: 1 Byte!), gefolgt von zwei Stop-Bits (immer schwarz). Für DFÜ-Spezialisten im Kurzcode: „8,N,2".

Jedes Halbbild trug auf diese Weise immer ein Byte, was 50 Bytes pro Sekunde entspricht, oder umgerechnet 400 Bits pro Sekunde, gleich 400 Baud. Das würde bei völliger Fehlerfreiheit in der Übertragung während einer 30-Minuten-Sendung 90.000 gesendeten Zeichen entsprechen, das sind etwa 40 vollgeschriebene DIN A4-Seiten!

Das 2. Verfahren

Im Jahre 1988 wurde von der Firma Wiegand in Brühl ein völlig neues VIDEODAT-Verfahren (Name: VD 1200) fertiggestellt, das mit seinen Leistungsdaten die 400 Baud der ersten Technik in den Schatten stellt. Videodat hat sich nunmehr zum größten Teil in die beiden Austastlücken zurückgezogen (siehe Bild 3). So dienen jetzt (am Beispiel des ersten Halbbildes betrachtet) die Zeilen 7 bis 22 in der Austastlücke und die Zeilen 24 bis 31 im Bildsignal zur Datenübertragung. Was aber das neue Verfahren auszeichnet, ist die andere Benutzung der Zeilen: Jede einzelne Zeile wird in ihrer ganzen Breite genutzt, und die Bytes liegen komplett in einer Zeile. Weitergehende Verbesserungen haben dazu geführt, daß nicht nur 1 Byte, sondern wahlweise 2 oder 4 in eine Zeile gelegt werden können, was sicherlich noch keine Grenze nach oben darstellt (siehe Bild 4).

An der Informationsdarstellung hat sich aber nichts geändert. Die logische 1 (High) bedeutet Schwarzpegel (0,3 Volt), und der logischen 0 (Low) entspricht der Weißpegel (1 Volt). Der Aufbau eines Informations-Bytes besteht nun aus 1 Start- und nur noch 1 Stop-Bit (Kurzcode: 8,N,1).

Der WDR benutzt seit 1988 dieses neue Prinzip und belegt während der Sendungen die Zeilen 24 bis 27. So wird eine Videodat-Information wie beim alten Verfahren schon während der Bildaufnahme im Studio oder beim Bandumkopieren in diese 4 ersten Zeilen des Bildinhaltes hineingegeben. Der Vorteil für den Fernsehteilnehmer liegt darin, die Videodat-Signale mit einem Videorekorder aufzeichnen zu können.

Channel Videodat

Bislang fristete Videodat ein Schlummerdasein und wurde nur zu zwei Terminen im Monat vom Westdeutschen Rundfunk aus dem Dornröschenschlaf erweckt. Inzwischen ist die Zeit des Verschlafens vorbei! Nachdem sich die öffentlich-rechtlichen Rundfunkanstalten noch die Wunden des anderen Schlummerdienstes VideoTEXT lecken und nicht über ihren Schatten springen konnten, zeigt uns eine Privatfirma, wie der Hase läuft: die Firma Wiegand aus Brühl bei Köln. Die Deutsche Bundespost TELEKOM stellt die Sendekapazität in der Austastlücke des privaten Fernsehsenders PRO 7 zur Verfügung, Wiegand produziert den „CHANNEL VIDEODAT" als eigenständige Dienstleistung und bietet entsprechende Hard- und Software dazu an.

Weil PRO 7 keinerlei Vorhaben in Sachen Videotext unternehmen wollte, lag die Übertragungskapazität in den Austastlücken brach. Channel Videodat benutzt (bei PRO 7) die Zeilen 11, 12 und 13. Da mittlerweile ein noch ausgeklügelteres Verfahren Anwendung findet (quasi Verfahren Nr. 3), sind jetzt sogar 10 Bytes pro Zeile machbar. Insgesamt erreicht dieser (physikalische) Datenkanal eine Übertragungsrate von 15.000 Baud, der zur Zeit in zwei logische Kanäle aufgeteilt ist.

Rechnen wir doch einmal nach:

10 Bits/Byte * 10 Bytes/Zeile
* 3 Zeilen/Halbbild * 50 Halbbilder/Sekunde
= 15.000 Bits/Sekunde!

Während der Sendung laufen auf jedem logischen Kanal bis maximal drei Prozesse parallel. So wird üblicherweise Computersoftware mit der schnellsten Rate von 9.600 Baud als sogenannter Vordergrundprozeß gesendet. Texte benötigen nicht so viel Zeit, so daß sie als Hintergrundprozesse etwas langsamer beim Empfänger ankommen können.

Das ganze ähnelt sehr dem Prinzip von DATEX-P: Eine zusammengehörige Datenkette (beispielsweise ein Listing) wird in einzelne Datenpakete mit fest definierter Länge aufgeteilt. Die Datenpakete schickt man mit einer Kennung versehen auf die Leitung. Der Empfangsdekoder setzt nun die einzelnen Datenpakete wieder (wie ein Puzzle) zusammen. So ist es möglich, Datenpakete unterschiedlicher Quellen hintereinander über die Leitung (hier natürlich die Fernsehzeilen) zu schicken und eine quasiparallele Übertragung zu gewährleisten. Logisch dürfte sein, daß ein schnellerer Prozeß in der selben Zeitspanne entsprechend öfter Datenpakete auf die Reise schickt als ein langsamerer.

Die Einbahnstraßen-Mailbox

Wer weiß noch nicht, was eine Mailbox ist? Gut, ganz kurz im Schnellkurs: Eine Mailbox hält für gezielt adressierbare Empfänger (-Personen) Nachrichten bereit. Damit sowohl Empfänger als auch Absender solcher elektronischer Briefe eine Anlaufstelle haben, gibt es Zentralrechner, die entsprechende elektronische Postfächer bereithalten. Dort werden die elektronischen Nachrichten zwischengelagert. Absender und Empfänger einer solchen Mitteilung müssen Zugang zu diesem Speichersystem haben. Abgesehen von den geschlossenen Netzen, in die ein Unbefugter kaum eindringen kann, gibt es auch öffentlich zugängliche Mailbox-Rechner.

Unter den Mailbox-Nutzern besteht eine grobe Unterteilung von zwei Nachrichtentypen: 1. die private Zusendung („eyes

NVDI

Die Lösung

Für alle, die viel und schnell schreiben müssen, oder für die, die viel mit Grafikprogrammen zu schaffen haben, oder die, die einen Großbildschirm benutzen, für alle, die sich immer eine schnellere Bildschirmausgabe gewünscht haben, stellt sich NVDI vor.

NVDI verwandelt Ihren gewöhnlichen ST oder TT in einen Turbo-Rechner, auf den Sie nicht mehr warten müssen.

NVDI enthält ein vollständiges GDOS, wodurch das lästige Vorladen eines solchen Programmes entfällt. Sie bekommen somit summa summarum zwei Programme in einem.

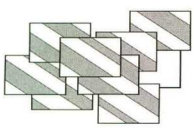
NVDI ist vielfältig und sehr anpassungsfähig. Es arbeitet mit vielen Beschleunigerkarten zusammen (z.B Board 20 von MAXON, HyperCache030 von ProVME). Auch unsauber programmierte Anwendungen behindern die Arbeit von NVDI nicht.

NVDI beschleunigt nicht nur den normalen Schwarzweiß-Modus des ATARI ST, sondern auch andere Auflösungen wie die von OverScan, MegaScreen, MAXON Graphic Adapter oder Matrix-Karte.

NVDI ist die Lösung für viele Aufgaben mit nur einem Rechner. Lassen auch Sie sich verzaubern.

NVDI
Die Lösung
Unverbindliche Preisempfehlung DM 99.-

REVOLVER



Der Profi-Switcher für Ihren ATARI ST. Wo andere Programme den Dienst quittieren, da bietet REVOLVER Sicherheit. Resetfest in jedem Rechnerteil und mit umfangreichen Utility-Funktionen ist REVOLVER ideal für Programmierer, Musiker und Anwender, die mehr aus ihrem ATARI ST machen wollen.

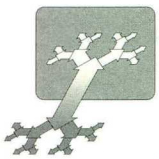
REVOLVER -
Der Profi-Switcher
Unverbindliche Preisempfehlung
DM 79,-

STOP

Einbruch und Datendiebstahl - kein Thema auf dem ST? Mit STop schützen Sie persönliche Daten, Programme oder Artikel- und Kundendateien vor fremden Zugriff. Nur über die Paßwörter ist der Echtzeitzugriff auf die vollständig kodierten Daten möglich. Die Datensicherheit dürfte mit 256 hoch 256 Möglichkeiten gewährleistet sein!

STOP -
Der Datentresor
Unverbindliche Preisempfehlung
DM 129,-

XBoot



XBoot ist ideal für den gestreßten Festplattenbesitzer. Es ermöglicht bei jedem Boot-Vorgang die Auswahl der zu ladenden Accessories und AUTO-Ordner-Programme. So nutzen Sie Ihren ST optimal und verschwenden keinen Speicherplatz durch nicht benötigte residente Programme. Ganz nebenbei lassen sich beliebige GEM-Programme automatisch starten - auch mit den alten TOS-Versionen 1.0 und 1.2!

Doch damit nicht genug: Um nicht bei jedem Booten von der Festplatte sämtliche Einstellungen erneut vornehmen zu müssen, kann für alle Anwendungen ein SET definiert werden. Dadurch beschränkt sich das Ändern der Arbeitsumgebung auf einen einfachen Mausklick oder Tastendruck.

XBoot macht das Unmögliche möglich. Es ist das erste (AUTO-Ordner) Programm für den ST, das den Komfort eines GEM-Programms mit kompletter Mausbedienung bietet.

Und das sagt die Fachpresse über XBOOT

TOS-Magazin 9/90
"Mit XBoot steht dem ST-Anwender ein kleines aber ungemein nützliches Hilsprogramm zur Vefügung."

ST-Magazin 11/90
"Aufgrund seiner vielfältigen Funktionen sollte XBoot jedoch in keinem Autoordner fehlen."

ST-Computer 12/90
"Festplattenbesitzer, die bislang über umständliche Tastaturkommandos die Einstellung ihrer PRGs und ACCs vorgenommen haben und sich mit Bomben und Systemabstürzen herumärgern mußten, können aufatmen. Mit XBoot gehört dies der Vergangenheit an."

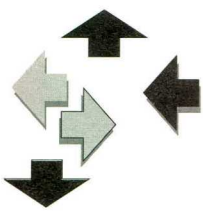
XBoot
Der Boot-Manager
Unverbindliche Preisempfehlung
DM 69.-

SALDO

SALDO ist ein Programm, das Ihnen erlaubt, die Kontrolle Ihrer Finanzen in den Griff zu bekommen. Sie können SALDO für private Zwecke, aber genausogut als Einnahme- und Überschußverwaltung für die gewerbliche Tätigkeit einsetzen. SALDO bietet mit seiner Vielzahl an Funktionen alle nur denkbaren Möglichkeiten, die eingegebenen Daten zu manipulieren. Sie können z.B. sortiert oder aufgesplittert nach verschiedenen Kriterien auf dem Bildschirm dargestellt oder auf dem Drucker ausgegeben werden.
Es würde einfach zuviel, hier jedes einzelne Detail von SALDO aufzuzählen - man muß es gesehen haben.

SALDO
Unverbindliche Preisempfehlung
DM 79.-

INTERLINK ST



INTERLINK ST ist das komfortabelste DFÜ-Programm für den ATARI ST und damit ideal für den Einsteiger und den Profi. So urteilen zumindest die Besitzer, die die Kommunikation und den weltweiten Datenaustausch mit Hilfe von INTERLINK ST nicht mehr missen möchten. Wann gehen Sie auf die Datenreise?

INTERLINK ST -
DFÜ im Griff
Unverbindliche Preisempfehlung DM 79,-

BELA

BELA Computer Layout- und Vertriebs GmbH Unterortstr. 23-25 W-6236 Eschborn Tel: 06196/481944 Fax: 481930

only"), die von einem Absender gezielt nur einem Empfänger zugeleitet werden soll. Dieses Prinzip kennen wir noch von der alten Briefpost, wo jeder Brief einen definierten Absender hat und nur einen vorgegebenen Empfänger kennt. 2. die öffentliche Zusendung („public"), bei der üblicherweise der Absender durchaus bekannt sein kann, aber der (oder die) Empfänger nicht immer festzustellen sind. Am ehesten ließe sich dieses Prinzip mit der Veröffentlichung einer Kleinanzeige in der Zeitung vergleichen.

So haben sich mittlerweile auch in unseren Breiten Mailboxen etabliert, die Nachrichten in „schwarzen Brettern" sammeln (sogenannte Bulletin-Board-Systeme), und jeder, der berechtigten Zugang zu diesem System hat, kann diese „Public-Notes" lesen oder auch nicht. Vorteil: Man kann selbst entscheiden, welche Nachricht lesenswert erscheint. Nachteil: Der Nutzer muß sich die Mühe machen, die Bretter in den Mailboxen zu durchstöbern, um per Befehl die Texte aufzurufen (Optional- oder Holprinzip).

In größeren Netzwerken hat sich ein anderes Verfahren als viel praktikabler erwiesen. Das Mitglied einer Ziel-Mailbox kann sich in eine Nutzergruppe eintragen lassen und erhält ab sofort alle Nachrichten automatisch in sein elektronisches Postfach gesandt, die von irgendwoher für diesen Nutzerkreis adressiert waren. Schwarze Bretter gibt es dort nicht mehr. Vorteil: Man muß die interessierenden Bretter nicht mehr durchsuchen. Nachteil: Das Mailbox-Fach bläht sich sehr schnell mit Mitteilungen auf (Automations- oder Bringprinzip).

Channel Videodat vereinigt die verschiedenen Verfahrensweisen einer Mailbox, einer „Dateneinbahnstraße" gleich, d.h. es bleibt dem Teilnehmer überlassen, welche Inhalte er sich von einer mitgelieferten Software aus dem Kanal „herausfiltern" läßt. Seit Beginn diesen Jahres läuft 24 Stunden täglich, einem Wasserfall gleich, ein nie endenwollender Datenstrom über das Fernsehsignal. Umgerechnet sind das, abzüglich einiger Steuer- und Codierungssignale zwischen 130 und 150 Megabytes Daten pro Tag! Dies ist auch ein Grund, warum der private Fernsehsender PRO 7 als Datenkaravane in Frage kam, denn er sendet ein ununterbrochenes Programm.

Das vielfältige Angebot des Channel Videodat liest sich fast wie das Inhaltsverzeichnis einer Familienillustrierten: Filmlexikon, Börse direkt, Game Box, ddp-Nachrichten, Computermarkt, Shareware-Bibliothek, Kleinanzeigen, Sport, um nur einige zu nennen.

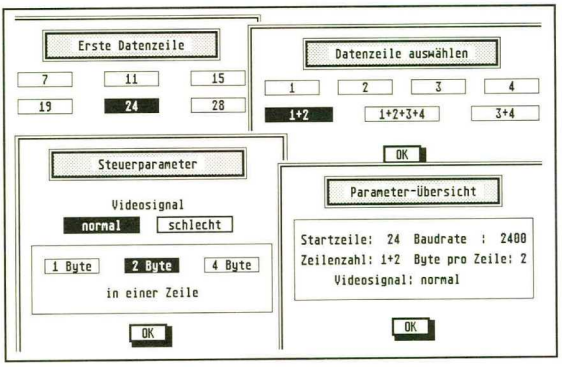


Bild 6: Dies sind die hauptsächlichen Parameter eines Einleseprogramms für das WDR-System „VD 1200".

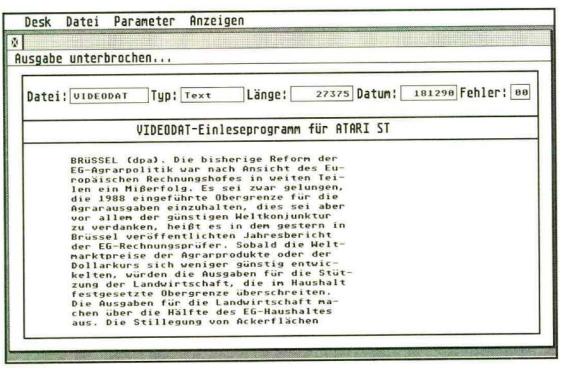


Bild 7: Die Benutzeroberfläche des bisherigen Einleseprogramms

Das Angebot

Ursprünglich waren hauptsächlich sogenannte „Online-Zeitungen" (Textdienste) im Kanal zu finden (Frankfurter Rundschau, deutscher depeschen dienst, Frankfurter Wertpapierbörse). So kam beispielsweise im März 1990 die „elektronische Tageszeitung für Blinde (ETAB)" ins Programm, bei der die Ausgabe des Textes auf ein Sprachmodul oder einen Blindenschriftdrucker (Braille) geleitet werden kann. Mittlerweile geht das Angebot aber weit darüber hinaus, denn auch Grafik und lauffähige Programme (vornehmlich Public Domain bzw. Shareware) sind machbar, ein besonderes Schmankerl für Computerfreaks.

Als weiterer Leckerbissen werden eine Reihe von Computerzeitschriften mitarbeiten, die nicht nur ihre News-Meldungen und Demo-Software, sondern auch Sonderausgaben exklusiv für den Channel Videodat veröffentlichen wollen. Es eröffnet sich damit die Chance für interessierte Computeranwender, ein Forum zu finden, ohne in zahllosen Mailboxen Mitglied zu sein.

Der Fernsehsender PRO 7 wurde nicht zuletzt auch wegen seiner Empfangbarkeit in Europa als „Datenträger" ausgewählt. So hat er einen festen Platz im Breitbandkabelnetz der Deutschen Bundespost TELEKOM. Weitere terrestrische Frequenzen für den Antennenempfang sind bald frei, und über die Satelliten „Astra" (Luxemburg) und „Kopernikus" (Deutschland) ist ein Direktempfang mit Parabolspiegel schon heute europaweit möglich. Noch in diesem Jahr wird ein Versorgungsgrad von 50% der Bevölkerung mit PRO 7 erreicht sein.

Anschluß finden

Als Voraussetzung für den Empfang von Channel Videodat wären zu nennen:

- PC, AMIGA oder ATARI ST mit serieller Schnittstelle
- Empfang des Senders PRO 7
- beliebiger Videoausgang von Fernsehgerät, Videorekorder, TV- oder Satellitenempfänger (Scart, DIN-AV, Chinch, BNC)
- Videokabel zum Dekoder
- serielles Kabel zum Computer

Kernstück der Empfangbarkeit ist ein Videodat-Dekoder, der von der Firma Wiegand, Brühl hergestellt wird. Monatlich verlassen zwischen 10.000 und 15.000 Dekoder das Fließband. Wenn man über-

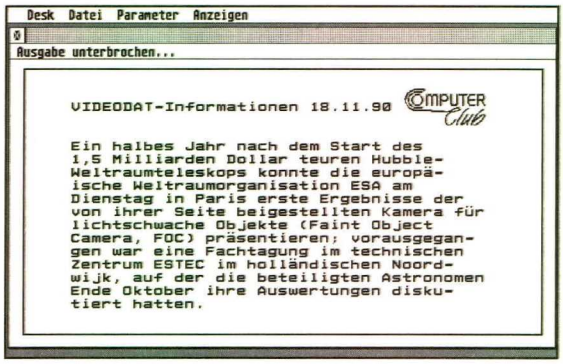


Bild 8: Auch Bildelemente kann eine VIDEODAT-Nachricht enthalten.

legt, daß in den Anfängen von Videodat seit 1986 bereits 30.000 Teilnehmer einen entsprechenden Dekoder lediglich für den Empfang der WDR-Sendungen erwarben, dann erkennt man, welches Interessenpotential sich dahinter verbirgt.

Der Videodat-Dekoder erfüllt gleich mehrere Aufgaben. Zunächst ist er verantwortlich dafür, die richtigen Zeilen aus dem Fernsehsignal herauslesen und kurzzeitig zwischenzuspeichern. Zum gegenwärtigen Stand der Technik kommen die Datenströme mit einer Geschwindigkeit von 15.000 bps (Bits pro Sekunde) als ein physikalischer Kanal herein. Wie wir weiter vorne gesehen haben, teilt sich dieser physikalische in zwei logische Kanäle, von denen der schnellste 9.600 Baud macht. In dieser Geschwindigkeit laufen die Daten an den Computer weiter.

Weiterhin stellt der Dekoder sicher, daß nur solche Inhalte an den Computer fließen, die auch wirklich für diesen Teilnehmer bestimmt sind. Dazu finden 2 parallel arbeitende Verfahren Anwendung: 1. Adressierung. Die Dekoder können individuell zum Empfang verschiedener Dienste freigeschaltet oder gesperrt werden. Dieser „Fernwartungscode" kann bis zu 4 Milliarden Dekoder getrennt ansprechen. 2. Verschlüsselung. Alle Signale im Channel Videodat, auch die sogenannten freien Datendienste sind „gescrambled" (verschlüsselt). Es wurde ein höchst komplizierter Verschlüsselungsalgorithmus (ca. 15 Milliarden Möglichkeiten) angewandt. Damit lassen sich sogar exklusive Benutzergruppen mit Daten bedienen, von denen andere Teilnehmer nichts entziffern können. Ein gutgemeintes Wort an den Hardware-Bastler: Es lohnt sich wirklich nicht zu versuchen, die Videodat-Schaltung der Firma Wiegand entwirren zu wollen und das Gerät nachzubauen.

Wie wir weiter oben festgestellt haben, läuft der Datenstrom des Channel Videodat immer nur in eine Richtung. Das bringt natürlich die Schwierigkeit mit sich, mögliche Fehlübertragungen ausschalten zu müssen, weil es eben keinen Rückkanal gibt, der irgendeine Prüfsumme an den Sender schicken könnte. Außerdem wäre eine individuelle Wiederholung einzelner Datenpakete für einzelne Empfänger total unwirtschaftlich bei tausenden angeschlossener Teilnehmer. Zur Zeit behilft man sich mit dem Trick, die Datenpakete in unregelmäßigen Abständen einfach mehrmals den Empfängern zuzusenden (Zeit-Multiplexing). Bei der extrem hohen Übertragungsrate im Kanal spürt der Anwender gar nichts davon.

Der Videodat-Dekoder, ein Videokabel und die nötige Software sind ab Januar 1991 im ausgesuchten Radio- und Fernsehfachhandel zum Komplettpreis von ca. DM 350 erhältlich. Kostenpflichtige Datenabonnementdienste verschiedener Unteranbieter sind darüber hinaus zu unterschiedlichen Gebühren erhältlich. Verträge hierzu müssen mit der Firma Wiegand gesondert abgeschlossen werden. Das Versprechen von Channel Videodat aber gilt, daß mindestens 50% des Gesamtangebotes dauerhaft von Zusatzgebühren ausgenommen bleiben.

Schlußbemerkung

Wenn man sich in der DFÜ-Landschaft umschaut, ist der Weg zur Information ständig mit Steinen gepflastert. Das Angebot z.B. im Bildschirmtext (BTX) wäre von der Gebührenseite durchaus attraktiv, wenn sich nur genügend Anbieter dort wiederfänden. (Gerüchteweise verlautete, daß BTX ohnehin zum Ende des Jahres 1992 eingestellt werden soll.) An eine weit entfernte Mailbox per Telefon zu gehen, erübrigt sich schon fast wegen der hohen Verbindungsgebühren und der schlechten Übertragungsqualität. Per DATEX-P den Weg zu einer Datenbank zu suchen, scheitert oft an der verwirrenden Gebührenstruktur, oder weil der Anbieter schlichtweg keinen DATEX-Anschluß hat. Videotext hat ein so schmales Angebotsspektrum, daß viele Fernsehbesitzer sich kaum dort umschauen, obwohl es eingebaut ist. Was bleibt denn nun dem Computeranwender für eine Alternative?

VIDEODAT hat durchaus die Chance, die großen Lücken in unserer Telekom-Landschaft sinnvoll zu schließen. Weil der Versorgungsgrad unserer Haushalte mit Fernsehgeräten extrem hoch ist, bietet sich das Fernsehsignal als idealer Träger für Zusatzdienste an. Channel Videodat vereinigt in sich die verschiedenen Arbeitsverfahren einer Mailbox, quasi wie ein gigantisches elektronisches Rundschreiben.

Dennoch darf man die (wenn auch kleinen) Schwächen des Systems nicht verschweigen: Zur Zeit ist zum Empfang des Datenstromes noch ein Fernseher, Videorekorder oder sonstiger TV-Empfänger nötig. Und: Der Kanal von PRO 7 muß beim Empfang der Daten eingestellt sein, so daß andere Fernsehsender während dieser Zeit blockiert sind. Die Firma Wiegand arbeitet aber schon an einem Dekoder mit eingebautem Empfänger, der unseren gewohnten Fernsehabend dann nicht mehr stört. Mein Vorschlag: Baut bitte gleich eine Zeitschaltuhr mit ein.

Womit wir bei einem weiteren Manko des Systems wären: Derzeit muß der Anwender noch bis zur maßgeblichen Uhrzeit ausharren, damit die Daten den Computer erreichen. Wohl dem, der einen Videorekorder sein eigen nennen kann, denn wie wir oben gesehen haben, läßt sich das Signal des Channel Videodat glücklicherweise aufzeichnen (zusätzlich zum eigentlichen Fernsehprogramm).

Mein Fazit: Die Angebotspalette kann sich durchaus sehen lassen und stellt (bei entsprechender Resonanz von Nutzern und Anbietern) so manche kommerzielle Mailbox in den Schatten. Das Prinzip der Übermittlung im Huckepackverfahren mit dem Fernsehbild ist bestechend genial. Wenn die Idee konsequent weiterverfolgt, verbessert und das Angebot noch auf andere Kanäle ausgebaut wird, dann ist Channel Videodat mit Sicherheit *das* interessanteste DFÜ-Medium der Zukunft.

DK

Bezugsquelle:

Wiegand Video-Daten-Systeme
Palmersdorfer Hof 11 - 19
5040 Brühl
Tel.: (02232) 45028

Selbst-Assemblierung

Hast Du keinen, hol' Dir einen...



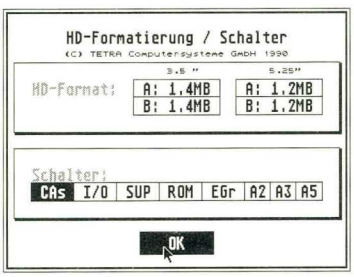
Bereits in Ausgabe 6/90 stellten wir Ihnen den „Manhatten ST" vor - einen ST im Tower mit etlichen Erweiterungen. Der Hersteller dieses Riesengeräts zeigt nun ein Herz für all diejenigen, die nicht das nötige Geld aufbringen können, sich einen kompletten Tower zu kaufen.

Recht so! Warum soll man schließlich Tausende von Märkern für einen Tower berappen, wenn man schon einen eigenen Mega ST hat? Würde es nicht genügen, die vorhandene Platine zu nehmen und in ein leeres Tower-Gehäuse einzubauen? „Das ist der Clou", dachte sich Tetra, und verkauft seitdem leere Towers mit passendem Zubehör als Umbau-Set. Ein solches Set traf auch in der Redaktion ein, woraufhin ein Mega 2 mit 100 MB- und 40 MB-Platte eingebaut werden sollte. Die Größe des Instruments: 456x190x620.

Sesam...

...öffne Dich. Zum Öffnen des Towers müssen sechs Schrauben abgelöst werden. Gesagt, getan, und das Gehäuse, das natürlich komplett aus Metall besteht, ist tatsächlich leer - bis auf ein 220-Watt-Netzteil, mit dem alle einzubauenden Geräte versorgt werden - die Leistung ist mehr als ausreichend. Außerdem sind auf Anhieb sechs $5^1/_4$"-Slots und zwei $3^1/_2$"-Slots für Festplatten und Diskettenlaufwerke zu erkennen. In unserem Fall befinden sich in diesen Slots bereits ein 1,44 MB- ($3^1/_2$") und ein 1,2 MB-Laufwerk ($5^1/_4$"), so daß noch ein $3^1/_2$"- und fünf $5^1/_4$"-Slots freibleiben.

Damit der Einbau problemlos vonstatten gehen kann, liegt eine 60seitige Anleitung bei, die alle Schritte eingehend erläutert.



Mit dem Accessory können HD-Disketten formatiert werden

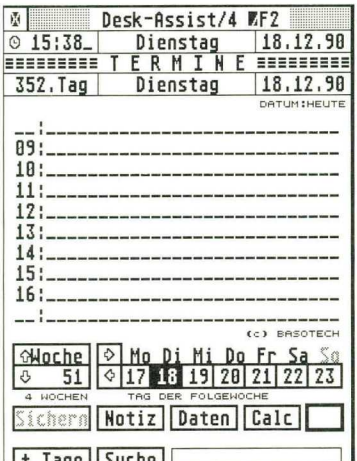


Bild 2: Der Terminplaner von Desk-Assist

Der Aufbau der Anleitung ist übersichtlich und auch für den Laien leicht verständlich. Allerdings sollte man die Anleitung vor dem Einbau seines ST unbedingt einmal ganz gelesen haben, da sonst bei einigen Anweisungen noch Fragen offenbleiben, die erst einige Seiten weiter hinten geklärt werden - ich als ungeduldiger Tester habe das natürlich nicht gemacht und stieß auch prompt auf Probleme - eigene Schuld!

Nachdem die Platine des Mega ST gebührend aus dem nun ausgedienten Originalgehäuse entfernt ist (die Anleitung spricht von „Deassemblierung des Mega ST"; nie zuvor habe ich einen Computer „deassembliert"...), müssen einige Leiterbahnen aufgetrennt und zusätzliche Kabel eingelötet werden - hier ist schon das Ende der Fahnenstange für den reinen Anwender erreicht, der noch nie einen Lötkolben in der Hand hatte: Die Platine des ST ist ziemlich empfindlich und sollte nur von erfahrenen Bastlern bearbeitet werden; besonders die Auftrennung der Leiterbahnen kann haarig werden.

SCSI

Ist, wie in unserem Fall, auch noch ein 1,44- oder 1,2-MB-Laufwerk einzubauen, müssen von der Platine noch 16 MHz abgegriffen werden, was durch Hochbiegen eines Pins und Verlötung desselben mit einem Kabel geschieht. Wer eine SCSI-Platte hat, sollte sich gleich den passenden Host-Adapter bei Tetra mitbestellen. Der Adapter wird auf einer eigenen Metallplatte befestigt und über Flach-

bandkabel mit Quetschverbindern mit dem DMA-Port verbunden, sobald die Mega-Platine eingebaut ist.

Nun legt man an der Rückseite des Gehäuses die Schnittstellen nach außen. Das geht recht einfach, denn auch diese verbindet man durch Flachbandkabel. Zur Befestigung nehmen Sie lediglich die Blinddeckel ab, setzen die Schnittstellen ein und schrauben sie fest. Dabei sollte man die Plazierungshinweise beachten, sonst kommt man bei der späteren Verkabelung in Teufelsküche, weil die mitgelieferten Anschlußkabel fertig konfektioniert sind und somit eine vorgefertigte Länge haben.

Bevor sich die Hauptplatine einbauen läßt, müssen Sie noch alle $3^1/_2$"-Geräte einbauen - später werden die Löcher für die Halteschrauben solcherlei Geräte durch die Hauptplatine verdeckt. Das ist zwar schade, weil es spätere Änderungen und Reparaturen umständlich macht, liegt aber in der Natur eines Towers bzw. der Mega ST-Platine, bei denen die Größen nun mal nicht bis ins Endlose steigerbar sind.

Nachdem die eigentliche Mega-Platine bearbeitet ist, ist die Haupt-Adapterplatine von Tetra anzuschließen. Auf ihr sind alle notwendigen Anschlüsse vorhanden, die an die Rückseite des Towers gelegt werden: SM124, 9poliger SUB-D-Monitorstecker, MIDI-in und -out, RS232, Centronics (I/O-Pin 17 beschaltet), Tastatur, ROM-Port intern und extern (softwaremäßig umschaltbar), SCSI, DMA. Damit es auch ordentlich aussieht und benutzt werden kann, liegen die Anschlüsse, wie gesagt, auf einer gesonderten Platine, die zunächst mit der ST-Hauptplatine verbunden werden muß. Dazu sind einige Stiftverbindungen zwischen MIDI und Monitorstecker einzusetzen, die Platine wird aufgesteckt, fertig.

Simpel

Bevor ich den gesamten Aufbau weiter beschreibe, lassen Sie sich sagen, daß es von hier ab bis zum fertigen Tower nur noch 10 Minuten sind, denn es müssen lediglich einige Kabel eingesteckt werden. Der gesamte Umbau dauert somit für den halbwegs engagierten Bastler vier bis sechs Stunden - eine akzeptable Zeit für einen so entscheidenden Eingriff, wie ich denke.

Jetzt muß man nur noch das Gehäuse zusammenbauen und die Software installieren, damit die HD-Laufwerke auch vernünftig funktionieren. Zum Betrieb ist eine weitere, umfassende Anleitung beigelegt, da Tetra noch ein großes Software-Paket mitliefert. In diesem enthalten sind verschiedene nützliche Programme, so natürlich die Festplatten-Software von ICD, die einen komfortablen Betrieb mit SCSI-Geräten sicherstellt. Weiterhin finden sich ein Accessory zum Formatieren von HD-Disketten aller Art, ein Programm zur Step-Ratenumstellung (wichtig für HD-Laufwerke), eine RAM-Disk, Terminplaner, Notizblock, Druckereinstellung und vieles mehr - kurz: alles, was man bei der täglichen Arbeit braucht. Je nach Ausstattung des Bausatzes wird natürlich auch mehr Software mitgeliefert, etwa für Grafikkarten, MS-DOS-Emulatoren oder Beschleunigerkarten.

Fertig aufgebaut

Das Gerät ist endlich fertig installiert, insgesamt dauert es bei „normal begabten" Bastlern zirka vier bis sechs Stunden, bis die Arbeit erledigt ist. Nun wird noch die Metallummantelung aufgesetzt und festgeschraubt, und der Rechner kann angeschaltet werden. Damit haben wir nun einen ST mit folgenden Merkmalen: Tastaturschloß, Einschaltverzögerung (für Autoboot-Festplatten), alles in einem Gehäuse (nur ein Kabel führt nach außen), alle Schnittstellen an einer Stelle (Rückseite), eingebauter Lautsprecher, starkes Netzteil, Reset- und Turbo-Taster sowie Ein- und Ausschalter an der Vorderseite und nicht zuletzt einen geringeren Stellplatz für das gesamte System.

Alles in allem ist der Tuning-Satz eine nette Sache, die den ST sicherlich aufwertet, besonders dann, wenn dazu noch Erweiterungen wie Beschleuniger oder Emulatoren eingebaut werden. Der Preis von DM 1500,- für das nackte Umbau-Set erscheint mir allerdings etwas hoch gegriffen. So schön der Tower nachher auch aussehen mag, die Einzelteile für ein solches Gerät sind wesentlich preiswerter, wenn auch mit zehnfachem Arbeitsaufwand verbunden. Tatsache ist jedoch, daß ich mich in meinen „neuen" Tower-ST verliebt habe und jedem, der die Investition nicht scheut, das Tuning-Set nur empfehlen kann.

MP

Preise:

Tower Tuning Set: DM 1498,-
Aufpreis schwarzes Gehäuse: DM 298,-
HyperCache ST+: DM 498,-
TEAC-HD-Laufwerk: DM 198,-
84 MB-Seagate-Platte: DM 998,-
SCSI-Hostadapter: DM 198,-
Temperaturlüftersteuerung: DM 148,-
Desk-Assist/4: DM 48,-

Bezugsquelle:

Tetra Computersysteme GmbH
Neuer Markt 27
5309 Meckenheim
Tel. (02225) 17081

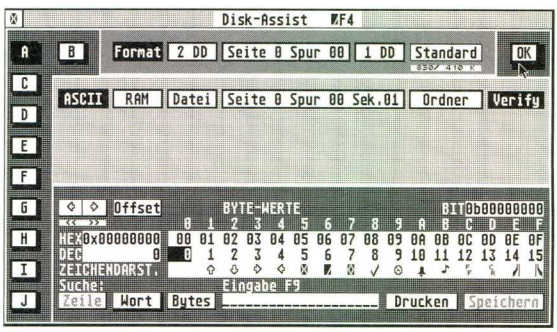


Bild 3: Ein kleiner Monitor

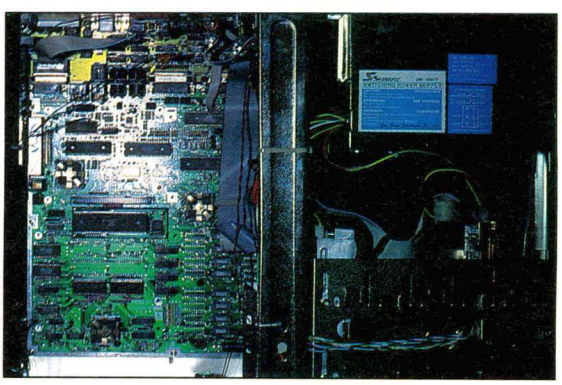


Bild 4: Das Innenleben unserer "Selbstassemblierung"

PUBLIC DOMAIN PAKETE

1 Graphik

Dieses Paket verwandelt Ihren Atari in eine hochwertige Graphikworkstation. Auf 5 Disketten bieten wir die besten PD-Graphik- und Malprogramme. Zum Beispiel:

Spezial Paint, The Manipulator, Butterfly Artist, Mistral, Atari Image Manager, Little Painter, Mandelbrot, Turn PIC, GeoScape 3D, Appehei, Convert, CAD2, ShowTime

Paket PJ1: 34,90 DM

2 Drucker

Werden Sie zum zweiten Gutenberg mit unseren PD- Druckprogrammen. Ob Poster, Briefköpfe, oder stilvolle Disklabels: Dieses Paket hat's in sich! Zum Beispiel:

The Printing Press, Printing Press Clipart, Ideaflist, OutPut, Formular, Brief, Barrell, Quickdruck, KX P1124, DeltaX, Treiber für LQ800, STX80, P6/7, LC-10 Color u.s.w..

Paket PJ2: 34,90 DM

3 Midi

Sequenzer laden, AMP auf 10 stellen, Cubase❖, Cubeat❖, TwentyFour❖ oder Twelve❖ laden und mit unseren 5 Disketten PD-MIDI-Songs ablahren! Zum Beispiel:

Männer - H.Grönemaier, Riders in the Storm - The Doors, Triller - M.Jackson, In the Air Tonight - Phil Collins, One Moment in Time, Goldfinger, Crockett's Theme, America, Ghostbusters u.s.w..

Paket PJ3: 34,90 DM

neu Midi 2

Nach dem großen Interesse an unserem ersten Midi-Paket, haben wir uns entschlossen, ein zweites Midi-Paket zusammenzustellen. Dabei haben wir uns bemüht nur wirklich gute Stücke aufzunehmen. Hier also 5 Disketten gefüllt mit den neuesten und besten PD-MIDI-Songs. Sie werden staunen mit wieviel Perfektionismus einige Stücke eingespielt wurden.

Paket PJ16: 49,90 DM

5 Spiele

Hier bieten wir Ihnen je 5 Disketten mit PD-Spielen quer durch alle Genres.

Farbe: Biker, Greece, Stones, Puzzle, Pente, Tunnel Vision, Adel, Chrome, Droid III u.s.w. / Monochrom: Bouncing Boubles Crystal Cave, Esprit-Demo (4Levels spielbar) Braingame, Rat der Weisen, Stetris, Swapper u.s.w.

Paket PJ5a: 34,90 DM
Paket PJ5b: 34,90 DM

6 PD-Fonts

Wer mit Signum oder Script arbeitet, der sollte sich diese Pakete zulegen. Jedes Paket enthällt 100 P.D.-Zeichensätze. Jeder Zeichensatz liegt als File für 9-Nadeldrucker, 24-Nadeldrucker und Laserdrucker vor.

Paket PJ6a: 49,90 DM
Paket PJ6b: 49,90 DM

7 Disk

Erste Hilfe für den gestressten Disk- /Harddiskbenutzer: 5 Disketten, die vom Harddiskchecker bis zum BackUp alle nötigen PD-Programme enthalten. Zum Beispiel:

FCopy III, Hypercopy, Bitte ein Bit, VDisk, Intram, RAM Disk, Fileselect, MakeDisk, Survey, ARC, Crunch, DARC, SHAR, UUS, ZOO, Packer, Turtle, Corona Shell, Flexdesk, HD Shell, SM-Shell, Best Shell, Thy Shell, Greifer u.s.w.

Paket PJ7: 34,90 DM

8 Clipart

DTP ohne Bilder ist wie GEM ohne Maus. Hier sind 5 Disketten mit **neuen** hochwertigen Graphiken im PAC- Format.



Paket PJ8: 34,90 DM

9 Vector/IMG

Dieses Paket enthält 5 Disketten mit PD-Graphiken im IMG- und Metafile-Vector-Format zum Einsatz unter DTP.



Paket PJ9: 34,90 DM

neu Clipart 2

Paket 14a enthält 5 Disketten gefüllt mit hochwertigen Grafiken im PAC-Format (Bisher in keiner Sammlung). Auf Paket 14b befinden sich auf 10 Disketten Grafiken im IMG-Format (bisher in keiner Sammlung). Beide Pakete enthalten Archivblätter zum schnellen Auffinden der Grafiken.

Paket PJ14a: 34,90 DM
Paket PJ14b: 49,90 DM

neu Signum-Script Tools

Unser neuestes P.D.-Paket haben wir für die Anwender von Signum und Script zusammengestellt. In diesem Paket erhalten Sie jede Menge Grafiken, Zeichensätze und Tools. 6 doppelseitige Disketten, die Ihnen die Arbeit mit Signum und Script erleichern werden.

Houdini, SIC-TO-GEM, SEG SHELL, MASSTAB, LINEAL 24, Funktionstasten, BIG FONT & TURNFONT, SNAPFONT, jede Menge PAC-Grafiken und 25 Font für 9-, 24- und Laserdrucker.

Paket PJ17: 34,90 DM

Bitline

W. Wohlfahrstätter und J. Ohst EDV GbR

Jutta Ohst ■ Nelkenstraße 2 ■ 4053 Jüchen 2 ■ Tel.: 02164/7898

Hardware ✦ Software

CameoST

Daß auch professionelle Software nicht zwangsläufig teuer sein muß, beweisenwir mit unserem CameoST, einem wahren Multitalent.

CameoST, das Musikarchiv

CameoST ist eine Datenbank für CDs, LPs und MCs. CameoST verfügt über flexible Suchfunktionen, die nicht nur alle Beethoveneinspielungen oder alle Versionen von "Just the two of us", sondern auch alle verliehenen LPs in Sekundenschnelle ermittelt. Fest ins CameoST-Konzept eingebunden ist eine Adresskartei, in der Sie allen gespeicherten Personen auch beliebige Eigenschaftskennzeichen zuweisen können. Eine weitere Funktion ist der eingebaute Notizblock. **Einen ausführlichen Testbericht entnehmen Sie bitte der Zeitschrift "PD-Journal" 11/90.**

Superpreis: 59,– DM

Cordless Mouse

Die innovative Infrarottechnik, die das lästige Mauskabel unnötig macht und ein ergonomisches Design stellen das Original weit in den Schatten. Dabei ist die Installation so einfach wie bei der alten Maus und Kompatibilitätsprobleme treten auch nicht auf.

Näheres erfahren Sie in dem Testbericht der "TOS 10/90" bzw. "PD- Journals" 11/90.



Superpreis: 198,– DM

Sonderaktion

Wir bieten Ihnen Vector-Fonts aus eigener Herstellung für Calamus✧ . Damit Sie eine reichhaltige Auswahl an Schriften zu einem wirklich günstigen Preis erhalten, haben wir ein Schriftenpaket für Sie zusammengestellt. Dieses Paket enthält 200 Schriften. Diese Anzeige wurde übrigens mit dem Vektorfont "Serif", der auch in dem Paket enthalten ist, gesetzt und belichtet.

200 Schriften nur: 249,–

✧ Calamus ist eingetragenes Warenzeichen der Firma DMC

Showtime^{Pro}

Dieses graphische Präsentation- und Informatinsprogramm reizt die Möglichkeiten Ihres ST voll aus. Allein 40 absolut flimmerfreie, saubere Umblendalgorithmen, GOTO, REPEAT...UNTIL- Strukturen, Multitasking-Sound, Vernetzung vieler Ataris mittels MIDI und die perfekte Benutzerführung machen dieses Programm zum Novum. Innerhalb kürzester Zeit erstellen Sie perfekte Diashows oder nutzen , durch die beliebigen Sprungmöglichkeiten auf Tastendruck, das Programm zum Aufbau eines Informations- bzw. Lernssystem.

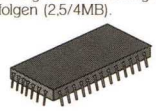
Interesse? Demo 10,– DM

Für weitere Informationen stehen wir Ihnen gerne zur Verfügung.

Einführungspreis: 99,– DM

Speicher

Speichererweiterung für AtariST Computer von 512 KB auf 1 MB oder von 512KB, 1MB auf 2,5/4 MB. Die Speichererweiterung ist voll steckbar und für alle Computertypen (außer MegaST), deren MMU gesockelt ist, geeignet. Die große Aufrüstung kann in zwei Stufen erfolgen (2,5/4MB).



1 MB	198,– DM
2 MB	449,– DM
4 MB	698,– DM

Auf Wunsch übernehmen wir auch den Einbau.

Einbau incl. Porto 50,– DM

Scanner

Bildverarbeitung für jedermann?

Ja. Denn jetzt gibt es Repro Studio junior. Dazu gehört nicht nur ein Logitech Handscanner mit 100-400 dpi, 32 Graustufen, 3 Raster, 105 mm Rasterbreite, sondern auch das professielle Bildverarbeitsungsprogramm Repro Studio ST junior. Das alles erhalten Sie zu einem Preis, den Sie bisher nur für einen Scanner bezahlt hätten.

ReprostudioST junior: 598,– DM incl. Scanner

Hardware

VIDI ST	448,– DM

Vidi-ST digitalisiert Videobilder in Echtzeit durch eigenen Bildspeicher mit 16 Graustufen. Kein Standbild erforderlich. Speichern in allen gängigen Dateiformaten. Es kann jede beliebige Videoquelle angeschlossen werden. (Kamera, Recorder etc.).

Overscan	120,– DM

Testbericht PD-Journal 1/91

LOGIMOUSE	89,– DM
3,5 " Laufwerk	239,– DM
5 1/4 " Laufwerk	289,– DM

Datenbank Textverarbeitung

That`s Write	298,– DM
I.D.A.	358,– DM
Signum!2	a.A.,– DM
Script 2	278,– DM
Daily Mail	169,– DM
STAD	169,– DM
That's Pixel	139,– DM
Tempus Word	569,– DM

Zubehör

10 Disketten 2DD	14,90 DM
50 Disketten	69,00 DM
Monitor Switchbox	49,90 DM
Auto Switchbox	59,90 DM
Druckerkabel 2m	29,90 DM
Scart-Kabel	29,90 DM
Verlängerungskabel Festplatte	49,90 DM
Farbband NEC P6	17,90 DM
Farbband Epson LQ 500,800,850	14,90 DM
Farbband Star LC10	11,90 DM



Software

Syntex	249,– DM

OCR-Schrifterkennung

X-Boot	69,– DM
Neodesk 3.0	89,– DM
Diskus	129,– DM
Tempus	109,– DM
Reprok	549,– DM

Büroorganisation

Steinberg 12	99,– DM
Cubase 2	a.A.

sonstige Steinbergprodukte auf Anfrage.

Verschiedenes

AT-Speed	478,– DM
Speed Bridge	69,– DM
Best 2400L	298,– DM
Best 2448LF	398,– DM

Das DFÜ-Faxmodem incl. Atari-Software.

Fast Filemover	59,– DM

Das schnellste Einzelfilecopy seit Bestehen des ST.

BTX-Software

MultiTerm pro an Modem	158,– DM
MultiTerm pro an D-BT03	236,– DM

Bildschirmtextprogramm in Farbe und s/w.

Scanservice in Profiqualität auf EPSON GT 6000

Hersteller- und Händleranfragen erwünscht.

Wir suchen noch gute Soft- und Hardwareprodukte zum Vertrieb oder zur Vermarktung.

Versandkosten

Der Versand erfolgt per Nachnahme oder Vorauskasse. Natürlich können Sie alle Produkte auch telefonisch per Nachnahme bestellen. Die Versandkosten betragen 3,00 DM bei Scheck- oder Barzahlung und 6,00 DM bei Nachnahme.

02164/7898
0211/429876
BTX: Wohlfahrtstätter

Werner Wohlfahrtstätter ■ Irenenstraße 76c ■ 4000 Düsseldorf 30 ■ Tel.: 0211/429876

Etwas mehr Ruhe, bitte.

Geräuschreduzierung SH205 ohne Geld

Die Atari SH205 macht sich deutlich mit lautem Heulen bemerkbar, sobald sie eingeschaltet wird. Wesentlichen Anteil an der Lärmkulisse hat das Lüftungssystem. Um dem Problem abzuhelfen, wurde bisher vorgeschlagen, eine temperaturabhängige Steuerung der Lüfterdrehzahl einzubauen. Auch der Austausch des Lüfters gegen ein leiseres Exemplar erscheint denkbar.

Eine weitere und zudem kostenlose Möglichkeit, den Lärm drastisch zu verringern, ist die hier beschriebene Optimierung im Lüftungs-kanal durch Entfernen von Strömungshindernissen. Die Optimierung sollte auf die Festplatten der MEGAFILE-Serie übertragbar sein.

Aber Achtung vorm Umbau : Im Gegensatz zu der Situation bei IBM-kompatiblen Computern geht schon beim Öffnen des Atari-Gehäuses eine noch vorhandene Garantie verloren. Vorm Öffnen des Gehäuses ist zunächst die Festplatte zu parken und auszuschalten, danach sind alle äußeren Kabelverbindungen abzuziehen. Die letzte Datensicherung sollte nicht zu lange zurückliegen. Sicher ist sicher. Wer, durch diesen Artikel angeregt, seiner SH 205 mit Messer, Feile oder Säge zuleibe rückt, sollte darauf achten, daß keine Bastelabfälle ins Gehäuse gelangen bzw. dort liegen bleiben.

Lüfter austauschen?

Ein neuer, leiserer Lüfter muß die gleichen offenen Eckdaten (Höhe 6,0 cm x Breite 6,0 cm x Tiefe 2,5 cm, 12 V, 0.12 A) haben und eine annähernd gleiche maximale Luftförderrate aufweisen. Im direkten Vergleich zeigten aber die in Elektronikgeschäften angebotenen Lüfter unterschiedlicher Firmen einen schon mit der Hand spürbaren deutlich niedrigeren Luftdurchsatz, so daß auf den Ersatz des Originallüfters verzichtet wurde. Es mag leisere 'Turbinen' geben, sie wurden jedoch nicht gefunden.

Beseitigung von Strömungshindernissen

Deutlich erkennbar wird der Lüftungsquerschnitt an der Auslaßseite durch Gehäuseteile, insbesondere die Gehäusedecke, verengt. Daher wurden aus dem Gehäuse die Stege des äußeren Lüftergitters herausgeschnitten und das Ober- und Unterteil des Gehäuses nach Anzeichnen der neuen Öffnungsmaße ausgefeilt. Im direkten Vergleich (Bild 1) ist deutlich zu erkennen, wie weit die Ausblasöffnung versperrt wurde. Der erzielte Geräuschminderungseffekt war subjektiv allerdings gering. Zur Erhöhung der Sicherheit sollte/kann wieder ein grobes Gitter in die Lüftungsöffnung des Gehäuses gesetzt werden.

Auch auf der Ansaugseite des Lüfters ist sein Querschnitt durch das Befestigungsblech deutlich verdeckt. Nicht der kräftige Lüfter, sondern die z.T. mitten im Luftstrom stehenden Teile des Befestigungsbleches sind für die Lautstärke und das Heulen verantwortlich - vergleichbar einer Luftschutzsirene, in der ein Rotor sehr dicht an feststehenden Schlitzen vorbeiläuft.

Als Lösung stehen 3 Möglichkeiten zur Auswahl:

1. ersatzloses Entfernen des Befestigungsbleches. Aus Sicherheitsgründen ist dies nicht zu empfehlen.

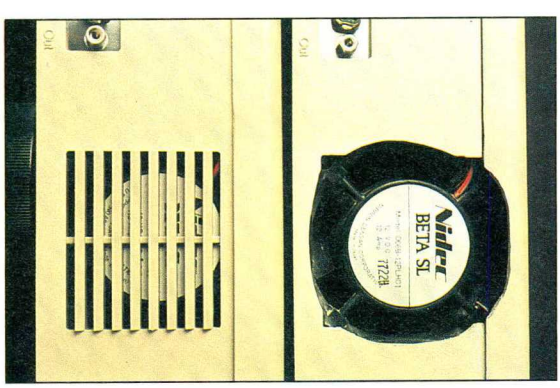


Bild 1 : Die Außenseite der SH205 im direkten Vergleich. Deutlich ist die starke Behinderung des Luftstroms vor dem Ausfeilen des Gehäuses erkennbar.

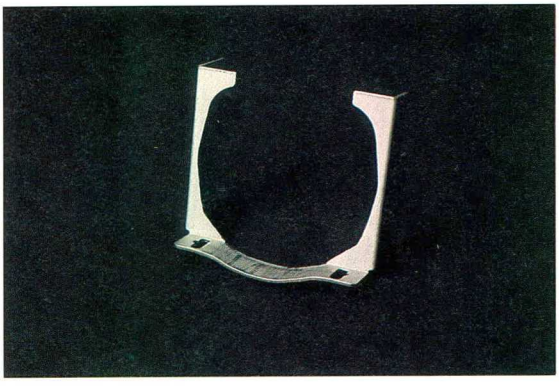


Bild 2 : Das Lüfterbefestigungsblech nach der Bearbeitung

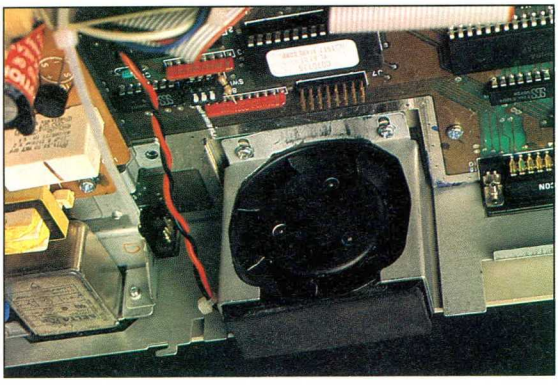


Bild 3 : Das optimierte Befestigungsblech behindert den Luftstrom nicht mehr.

2. Befestigung des Lüfters mit Teppichklebeband an der hinteren Gehäusewand und Entfernen des jetzt überflüssigen Befestigungsbleches. Dies sollte schnell durchführbar sein und ist, wenn nötig, ebenso schnell wieder rückgängig zu machen.
3. Abfeilen/Entfernen der störenden Metallteile des Befestigungsbleches. Für die Optimierung des Befestigungsbleches wurden Teile davon weggefeilt bzw. abgekniffen und die Standfläche des Bleches mit Hilfe einer Flachzange zu einer Senke verformt. Kein Teil des Befestigungsbleches liegt jetzt mehr im Luftstrom. Um das Befestigungsblech wieder festschrauben zu können, wurden die Befestigungsschlitze des Bleches ausgefeilt. Bild 2 zeigt das Blech nach der Optimierung alleine, Bild 3 im eingebauten Zustand. Warum nicht gleich so, Atari?

Da noch ein Schaumstoffrest im Hause war, wurde dieser nach der Wiederbefestigung des großen Abschirmbleches in den freien Raum vorne rechts ausfüllend eingesetzt. Die Maßnahme erscheint sinnvoll, um den „Klangkörper" der SH205 zu dämpfen und durch ein teilweises Verdekken der hier vorhandenen Lüftungsschlitze zu erreichen, daß die Kühlluft vermehrt über Lüftungsschlitze im Gehäuseboden angesaugt wird. Die Effektivität des Schaumstoffeinsatzes wurde nicht einzeln überprüft.

Lohn der Mühen

Das Ergebnis : Endlich ist Musik auch in niedriger Lautstärke hörbar. Die Nachbarn werden es danken. Allerdings ist eine Funktionskontrolle des Festplattenlaufwerks akustisch nur noch auf kurze Entfernung möglich. Anfangs beängstigend (leise).

Noch mehr Ruhe, bitte.

Wem diese Lärmreduktion nicht ausreicht, der kann auf kaufbare Geräuschreduktions-Kits zurückgreifen. Sie werden zu Preisen zwischen DM 30,- und 50,- angeboten, z.B. von den Firmen ddd (Hannover) und Eickmann (Frankfurt am Main). Auch andere Firmen dürften solche Zusätze im Lieferprogramm haben. Diese Kits enthalten in der Regel eine fertige Schaltung zur temperaturabhängigen Steuerung der Drehzahl des Originallüfters und Zusatzteile, z.B. ein neues Schutzgitter und Dämpfungsmaterial. Ein Vergleich der Angebote dürfte lohnend sein.

Für Lärmgeplagte User, die auch vor einer Lötkolbenbenutzung nicht zurückschrecken : Selbstbau der Steuerung nach einem Vorschlag aus der Zeitschrift c't 1989, Heft 10, Seite 136-137. Die erforderliche fertig geätzte Leerplatine kann z.B. bei der Firma eMedia (Hannover, Anzeigen in c't) für DM 9,- + Versandkosten bezogen werden.

Der Zeitaufwand für die Veränderungen wird je nach Lösungsweg mehr oder minder deutlich durch die 'Verpackung' der Atari-Computer und Festplatten erhöht. Kennen Sie einen IBM-kompatiblen Computer, der sich durch mehr Schrauben und Haken vorm Entblättern schützt?

Atari-Laserdrucker

Der Atari-Laserdrucker SLM804 macht im Betrieb noch mehr Lärm als die Festplatten. Versuche, auch diesen leiser zu machen, sollte man aus mehreren Gründen dem Fachhandel überlassen:

1. Der eingesetzte Lüfter hat die seltene Betriebsspannung 24 V. Er dürfte daher schwer auszutauschen sein.
2. Der Lüfter ist so verbaut, daß er ohne Teildemontage des Druckers nur schwer zu entfernen ist.
3. Die Elektronik und die Innereien des Druckers sind sehr empfindlich. Wird z.B. das Controller-Kabel vom Laserdrucker abgezogen, während dieser (bei geöffneter Heckklappe unhörbar) noch angeschaltet ist, kostet dies ca. 940,- DM + Einbau, da die mit SMD-Bausteinen bestückte Hauptplatine im Laserdrucker nicht durch gepufferte Eingänge geschützt ist und bei einem Defekt komplett ausgetauscht wird.

Zur Lärmreduzierung sollte man auf käufliche Stand-By-Schaltungen für den Laserdrucker zurückgreifen oder den neuen, kleineren Atari-Laserdrucker nehmen, der auch Online sehr leise ist und außerdem über einen zusätzlichen geraden Papierweg verfügt. Dies ist ideal für Etikettenbogen und festeres Papier. Als vorläufiger „Testbericht" kann z.B. die Beurteilung des 'Schneider S Printer's in der Zeitschift 'Computer Persönlich', Ausgabe 5/89 (15.02.89), Seite 58-63, dienen. Beide Drucker dürften, auf einer TEC-Maschine aufbauend, zumindest mechanisch sehr ähnlich aufgebaut sein.

Beim einem Blick in die Innereien des alten SLM804 fiel ein Staubfilter an der Ansaugseite des seitlichen Lüfters auf. Er war nach ca. 10000 gedruckten Blättern dicht mit weißen Ablagerungen besetzt. Ein Fall für die Laserdrucker-Anleitung mit Verweis auf die Service-Abteilungen der Atari-Händler zwecks Reinigung.

Wolfgang Waßmus

Die Privaten sind ab sofort kostenlos!



ST ohne Laufwerk, gegen Gebot. (keine Rollschuhe, hab schon 3) Bin nach 8 in der Pinte, dort anrufen.

Festplatte, 4 Jahre alt, kaum gebraucht, mit Rollen (daher nicht mehr ganz fest) im Tausch gegen ATARI ST 1040. Bei Clausen Nachricht hinterlassen.

Rollschuhe, 20MB RENN, mit Verwechselplattenlaufwerk, 900.- DM VHB.

HARDWARE

ATARI ST , 1 Jahr alt, für umsonst oder gegen 1 Paar Rollschuhe. Tagsüber unter dieser Nr. nicht zu erreichen, besser abends.

TT, 50 Mhz , kein 1 Jahr alt, nicht für alles Geld der Welt, oder gegen ein Paar Rollschuhe. Zuhause anrufen.

Kleinanzeigen-Auftrag ST-Computer

Bitte veröffentlichen Sie für mich folgende Kleinanzeige in der angekreuzten Rubrik:

○ Biete an ○ Hardware ○ Software ○ Ich suche ○ Kontakte ○ Tausch ○ Verschiedenes

30 Buchstaben je Standardzeile - incl. Satzzeichen und Wortzwischenräume.
Groß- und Kleinbuchstaben verwenden, fettgedruckte Wörter unterstreichen.

Bearbeitung bei gewerblichen Kleinanzeigen nur gegen Vorausscheck über den entspechenden Betrag (keine Überweisung).
Der Verlag behält sich das Recht vor, die Veröffentlichung zu verweigern, wenn dies erforgerlich erscheint!
Bei Angeboten: Ich bestätige, daß ich alle Rechte an den angebotenen Sachen besitze.

○ Private Kleinanzeige (kostenlos)
○ Gewerbliche Kleinanzeige für 12.- DM je Druckzeile
○ Scheck über DM _____ ist beigefügt

An: Heim Verlag
"ST-Computer-Kleinanzeige"
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Meine Adresse:

Name: _____

Str. _____

Wohnort: _____

Datum,Unterschrift _____

Einkaufsführer

Hier finden Sie Ihren Atari Fachhändler

1000 Berlin

HD COMPUTER-TECHNIK oHG

PD SERVICE — Je Disk 5,- DM

Über 5000 PD-Disketten verfügbar!

z.B. ACS, AMIGA JUICE, AMIGAUBDISK, ANTARES, AUGE 4000, AUSTRIA, BAVARIA, BE, AMI, BRUNOSOFT, CACTUS, CHEMIE, CHIRON, ES-PD, FAUG, FRANZ, GERMAN, GET IT, KIRSCHBAUM, MIDI-PD, OASE, OLLIS GAMES, PANORAMA, PAXUM, PFALZ, PUDOMIX, R-H-S, RPD (=CAM), 6 2 6, SCHATZ, S-DREAMS, TAIFUN, TAURUS, T.B.A.G., TIGER, TIME, TORNADOS, U.G.A., u. a.

und natürlich sämtliche KICKSTART-PD's!

- Speichererweiterungen
- Diskettenlaufwerke
- Festplatten & Turbokarten
- Anwendungssoftware
- Disketten Großhandel
- Fachliteratur & Zubehör
- Desktop Video/Publishing
- Reparatur Service

HD COMPUTERTECHNIK oHG
Pankstraße 61
1000 Berlin 65
Tel.:030/465 70 28

REPARATUR WERKSTATT
1000 Berlin 65 - Pankstr. 42

SERVICE STATIONEN
1/44, Lahnstr. 94, Tel:684 48 31
1/20, Schönwalder Str. 65, Tel:375 60 13

COMPUTERSYSTEME
Schlichting
...die etwas andere Computerei
COMPUTERSYSTEME, PLAYSOFTSTUDIO SCHLICHTING
COMPUTER - SOFTWARE - VERSAND GMBH & CO. KG
ATARI-FACHMARKT
MS-DOS FACHMARKT • NEC FACHHANDEL
Mönchstraße 8 · 1000 Berlin 20
030/786 10 96

1000 Berlin

DATAPLAY
Bundesallee 25 · 1000 Berlin 31
Telefon: 030/861 91 61

COMPUTERSYSTEME
Schlichting
...die etwas andere Computerei
COMPUTERSYSTEME, PLAYSOFTSTUDIO SCHLICHTING
COMPUTER - SOFTWARE - VERSAND GMBH & CO. KG
ATARI-FACHMARKT
MS-DOS FACHMARKT • NEC FACHHANDEL
Wilh.Pieck-Straße 56 · Potsdam
030/786 10 96

Steglitz Schloßstraße
030/79001-418
Ihre Tür zur Zukunft:
karstadt–
computer–center
hardware·software·problemlösungen

PLAYSOFT-STUDIO
Schlichting
...die etwas andere Spielerei
PLAYSOFTSTUDIO SCHLICHTING
COMPUTER - SOFTWARE - VERSAND GMBH
BERLINS NEUE DIMENSION FÜR COMPUTERSPIELE
Katzbachstraße 8 · 1000 Berlin 61
030/786 10 96

1000 Berlin

COMPUTERSYSTEME
Schlichting
...die etwas andere Computerei
COMPUTERSYSTEME, PLAYSOFTSTUDIO SCHLICHTING
COMPUTER - SOFTWARE - VERSAND GMBH & CO. KG
ATARI-FACHMARKT
MS-DOS FACHMARKT • NEC FACHHANDEL
Katzbachstraße 8 · 1000 Berlin 61
030/786 10 96

---

alpha computers g.m.b.h.
u. a. alphatronic, atari, commodore, dai, epson, sord mit pips, nec hard-/software nach maß — servicetechnik
Kurfürstendamm 121a, 1000 Berlin 31 (Halensee)
Telefon 030/8911082

2000 Hamburg

Planen • Beraten • Realisieren
HCAG HABA COMPUTER AG
ATARI Fachhändler
Münsterstraße 9, D–2000 Hamburg 54
Telefon 040/56 19 09–0
Telefax 040/56 19 09–80

G.M.A.
Wandsbeker Chaussee 58
2000 Hamburg 76
GMA
Einziges, autorisiertes Atari-DTP-Center in Hamburg.
Telefon: 040 / 25 12 41 5–7

DTP u. Werbeservice
Soft- und Hardwareversand
Viola Jaap
Berner Heerweg 512
2000 Hamburg 72
Telefon (0 40) 6 44 06 68

2000 Norderstedt

sellhorn
Ulzburger Str. 2
2000 Norderstedt
Tel. 040/527 30 47

2120 Lüneburg

Sienknecht
Bürokommunikation
Beratung - Verkauf - Werkstatt
Heiligengeiststr. 20, 2120 Lüneburg
Tel. 04131 / 46122, Btx 402422
Mo.-Fr. 9⁰⁰-18⁰⁰ und Sa. 9⁰⁰-13⁰⁰

2210 Itzehoe

Der Computerladen
Inhaber Ulrich Bubel · Martin Koppius
Coriansberg 2 · 2210 Itzehoe
Telefon (0 48 21) 33 90 / 91

2300 Kiel

MCC Computer
MCC Computer GmbH Holzkoppelweg 19a
2300 Kiel 1 TEL 0431/54381 FAX 541717

2800 Bremen

MS-DATA
Faulenstraße 48—52
2800 Bremen 1
Telefon (0421) 170577

2940 Wilhelmshaven

Radio Tiemann
ATARI-Systemfachhändler
Markstr. 52
2940 Wilhelmshaven
Telefon 0 44 21 - 2 61 45

3000 Hannover

DATALOGIC COMPUTERSYSTEME
ATARI ST-COMPUTER HARDWARE SOFTWARE
BERATUNG SERVICE VERKAUF
CALENBERGER STR. 26
3000 HANNOVER 1
TEL.: 0511 - 32 64 89

3000 Hannover

COM DATA
Am Schiffgraben 19 · 3000 Hannover 1
Telefon 05 11 - 32 67 36

HD COMPUTER-TECHNIK oHG
HANNOVER
Public Domain Service
Hardware & Zubehör
Fachliteratur
Hildesheimer Str. 118
3000 Hannover 1
Tel.: 0511/809 44 84

3400 Göttingen

Büroeinrichtungs-Zentrum
Wiederholdt
3400 Göttingen-Weende
Wagenstieg 14 – Tel. 05 51 / 38 57-0

4000 Düsseldorf

H O C O
EDV ANLAGEN GMBH
Ellerstraße 155
4000 Düsseldorf 1
Telefon 02 11 / 78 52 13

Hard und Software
Werner Wohlfahrtstätter
Atari
Public Domain
Atari Spiele
Atari Anwender
Ladenlokal
Irenenstraße 76c
4000 Düsseldorf-Unterrath
Telefon (02 11) 42 98 76

BERNSHAUS GmbH
Bürotechnik – Bürobedarf
Cäcilienstraße 2
4000 Düsseldorf 13 (Benrath)
Telefon 02 11 - 71 91 81

4010 Hilden

Beachten Sie unsere Anzeige in diesem Heft!
Weide ELEKTRONIK
Computer · Drucker · Software · Bücher · Service
Gustav-Mahler-Straße 42-44
Tel. (0 21 03) 3 18 80 + 4 12 26

4150 Krefeld

NEERVOORT . EDV
Kommanditgesellschaft
Nordwall 96
D-4150 Krefeld 1
Tel.: 02151-772056
FAX: 02151-770995
BTX: 02151772056
Computer- Hard- & Software
Ein- & Mehrplatzsysteme
Komplettlösungen

4200 Oberhausen

Redakteur
dBMAN
Spectre 128
für TOS, DOS, UNIX, MAC, AMIGA und andere Systeme
by COMPUTER MAI
dBMAN-Vertretung für NRW und BENELUX
ISYS-COMPUTER GbR
Tel.: 02 08 / 65 50 31 · Telefax: 02 08 / 65 09 81
Max-Eyth-Straße 47 · 4200 Oberhausen 11

4320 Hattingen

Ihre Tür zur Zukunft:
KARSTADT **computer-center**
hardware·software·problemlösungen
Hattingen, Große Weilstr. 18-20, Telefon (0 23 24) 2 09 73

Bei uns werben bringt **GEWINN**
Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057
BUF

4430 Steinfurt

CBS GmbH
COMPUTERSYSTEME
Tecklenburger Str. 27
4430 Steinfurt-Burgsteinfurt
☎ 02551/2555

4500 Osnabrück

Heinicke-Electronic
Meller Str. 43 • 4500 Osnabrück
Fax (0541) 58 66 14
Telefon (0541) 58 76 66
Wir liefern Micro-Computer seit 1978

4520 Melle

CBS GmbH
COMPUTERSYSTEME
4430 Steinfurt Tel. 02551/2555
Haferstraße 25 4520 Melle
Tel.: 05422/44788

4600 Dortmund

Elektronik
Computer
Fachliteratur
ATARI-System-Fachhändler
4600 Dortmund 1, Güntherstraße 75, Tel. (02 31) 57 22 84
city-elektronik

ATARI Systemfachhändler
PROFI COMPUTER STUDIO
KARSTADT Aktiengesellschaft
Kampstraße 1 · 4600 Dortmund
Telefon (02 31) 5 43 91

cc Computer Studio GmbH
Atari-Systemfachhändler
PCs von Tandy
Schneider Peacock
Drucker von
Star Brother NEC
Elisabethstr. 5
4600 Dortmund 1
Tel. 0231/528184 Tx 822631 cccsd Fax 0231/528131

4650 Gelsenkirchen

DeskTop Publishing Center
COMPUTERSYSTEME u. ANWENDUNGEN
Hüttenstr. 56
4650 Gelsenkirchen
Telefax: ‹0209› 271584
Telefon: ‹0209› 203420 / 233308

CSA

DTP Lösungen....
Scanner OCR Software Wechselplatten
Großmonitore Spezialanpassungen
TECHNOBOX-CAD Grafikprogramme
030 Zubehör
Festplatten RETOUCHE
Schneidplotter 16 MHz
Modem Digitizer Tower
Händleranfragen erwünscht

4650 Gelsenkirchen-Horst

MENTIS GmbH
Hard- und Software, Literatur
Bauteile, Service, Versand
Groß- und Einzelhandel
Poststraße 15 · 4650 Gelsenkirchen-Horst
Telefon (02 09) 5 25 72

4708 Kamen

D·M COMPUTER
Die jungen Spezialisten für PC
D·M Computer
Weststraße 45
4708 KAMEN
02307 · 17052
BTX 023071518 Telefax 02307 · 18367

4800 Bielefeld

hardware
software
organisation
service
CSF
CSF COMPUTER & SOFTWARE GMBH
Heeper Straße 106 – 108
4800 Bielefeld 1
Tel. (05 21) 6 16 63

5090 Leverkusen

Rolf Rocke
Computer-Fachgeschäft
Auestraße 1
5090 Leverkusen 3
Telefon 0 21 71 / 26 24

5253 Lindlar

DIGITAL SYSTEMS ATARI System-Center
Rheinstr. 15
D-5253 Lindlar
Tel. : (0 22 66) 4114
Fax : 4038
Komplettlösungen für: DTP ■ CAD ■ EBV ■ BÜRO ■ HANDEL ■ INDUSTRIE

5300 Bonn

In Bonn **Ihr** Ansprechpartner für
- Hardware
Monitore·Scanner·Drucker·Massenspeicher
- Software
Arzt-, Branchen-, Individual- und PD-Prg
- Zubehör
für EDV, Kommunikation und Büro
- Beratung
individuell und preiswert

ATARI ST Messe-Neuheiten
MS-DOS
Computer & Beratung Behnck
(0228) 67 70 21
Wir beraten und verkaufen
Mo- Fr 15-20Uhr, Sa 9-12Uhr oder nach Vereinbarung!

5500 Trier

bürocenter LEHR
Güterstraße 82 · 5500 Trier
☎ 06 51 / 20 97 10
Fordern Sie unsere Zubehör-Liste an!

5600 Wuppertal

MEGABYTE
COMPUTERVERTRIEBSGESELLSCHAFT MBH
Kleiner Werth 18
5600 WUPPERTAL 2
Tel. (0202) 592064
Nordstr. 57
5630 REMSCHEID 1
Tel. (02191) 21033
Fax (02191) 21034

5650 Solingen

MegaTeam
Computer-Vertriebs-OHG
Kölbach – Finke
Hardware – Software – Zubehör – Service
Rathausstraße 1–3 · 5650 Solingen 1
Telefon (02 12) 4 58 88 · Fax (02 12) 4 73 99

5800 Hagen

ATARI
wir machen Spitzentechnologie preiswert.
Vertragshändler Axel Böckem
Computer + Textsysteme
Eilper Str. 60 (Eilpezentrum) · 5800 Hagen
Telefon (0 23 31) 7 34 90

6000 Frankfurt

WAIZENEGGER
Büroeinrichtungen
Kaiserstraße 41
6000 Frankfurt/Main
Tel. (0 69) 27 30 6 - 0

Eickmann Computer
***Der* Atari-System-Fachhändler !**
z.B.: Festplatten von 30 MB bis 110 MB für Atari ST und Mega ST, Zusatzaufrüstungen für Ihre Festplatten bis 110 MB, Umrüstung Ihres SM 124 in einen EM 124 Multisync für alle Auflösungen, Slotkit für PC 1...
besuchen Sie unser Fachgeschäft:
In der Römerstadt 249
6000 Frankfurt 90-Praunheim
Telefon (069) 763409

6100 Darmstadt

Heim
Büro- und Computermarkt
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon (0 61 51) 5 60 57

6120 MICHELSTADT

COMPUTER SERVICE – SCHWARZER – 6120 Michelstadt – Bahnhofstraße 40

ATARI ST / TT / PC PORTFOLIO
Desktop Publishing
Drucker Festplatten 19" Monitore Scanner
Software Speichererweiterungen Zubehör
Bürodrehstühle – Büroeinrichtungen
Tel. 06061/73601 FAX 06061/73602

6204 Taunusstein

COMPUTER-CENTER der SYNTHESIZER STUDIO Jacob GmbH
COMPUTER • DTP
KEYBOARDS • MIDI
SOFTWARE
ZUBEHÖR
EIGENER SERVICE
ATARI SYSTEM-CENTER
6204 Taunusstein-Neuhof • Industriegebiet Triebgewann
Georg-Ohm-Straße 10 Tel. 06128/7 30 52 • Fax 7 30 53

6240 Königstein

KFC
COMPUTERSYSTEME
Wiesenstraße 18
6240 Königstein
Tel. 0 61 74 - 30 33
Mail-Box 0 61 74 - 53 55

6250 Limburg

PAULY
... das Zentrum der Bürowelt ...
DTP-Center • Salzgasse 6 • 6250 Limburg
☎ 06431/5004-0 • FAX 06431/5004-10

Bei uns werben bringt
GEWINN
Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057 BUF

6400 Fulda

Schneider ATARI Commodore
WEINRICH
BÜRO · ORGANISATION
Ronsbachstraße 32 · 6400 Fulda
Telefon (06 61) 4 92 - 0

6457 Maintal

LANDOLT - COMPUTER
Beratung - Service
Verkauf - Leasing
Finanzierung
ATARI star
6457 Maintal-Dörnigheim Robert-Bosch-Straße 14
Tel. (06181) 4 52 93 Fax (06181) 43 10 43
Mailbox (06181) 4 88 84 Btx *2 98 99#

6520 Worms

orion
Computersysteme GmbH
6520 Worms · Friedrichstraße 22
Telefon 0 62 41 / 67 57 - 58

6700 Ludwigshafen

MKV Computermarkt
Bismarck-Zentrum
6700 Ludwigshafen
Telefon 06 21 - 52 55 96

6720 Speyer

Elzkorn Computer
ATARI *Desktop Publishing Center*
Vom Portfolio über ST bis zum TT alles lagermäßig und vorführbereit auf 10 Anlagen!
3K • DMC • Eizo • NEC • Protar • Spectre • Vortex
6720 Speyer • Auestr. 20
Tel. 06232 / 32435 • Fax 41398
Mo - Fr. 9 - 12 u. 15 - 18.30, Sa. 9 - 14 Uhr

7000 Stuttgart



7100 Heilbronn

Seel's **Computer-Welt**
Am Wollhaus 6
7100 Heilbronn
Tel. 0 71 31 - 6 84 01 - 02

7150 Backnang

Computer-Fans finden bei uns alles von:
Servicestation Vertragshändler Computer-Systeme Software-Hardware
commodore Schneider COMPUTER DIVISION ATARI
WERSKE Das Elektrohaus am Marktplatz
Potsdamer Ring 10 7150 Backnang Tel. 0 71 91 / 15 28

Bei uns werben bringt
GEWINN
Sprechen Sie mit uns.
Heim Verlag ☎ 0 61 51 / 5 60 57 BUF

7312 Kirchheim/Teck

Comp & Phone
Computerfachhandel
Alleenstraße 66
7312 Kirchheim/Teck
Tel.: 07021/3949 - Fax: 07021/53933

7475 Meßstetten

Ihr ATARI-Systemhändler im Zollern-Alb-Kreis
HEIM + PC-COMPUTERMARKT
HARDWARE · SOFTWARE · LITERATUR
SCHEURER
ATARI COMMODORE CUMANA DATA BECKER
MULTITECH RITEMAN SCHNEIDER THOMSON
7475 Meßstetten 1 · Hauptstraße 10 · 0 74 31 / 6 12 80

7500 Karlsruhe

MKV GMBH
Kriegsstraße 77
7500 Karlsruhe
Telefon (0721) 8 46 13

ERHARDT Am Ludwigsplatz
Am Ludwigsplatz · 7500 Karlsruhe 1 · Tel. (0721) 1608-0

7700 Singen

Udo Meier Computersysteme
Ringstraße 4
Telefon (07731) 68222

7750 Konstanz

neu: ATARI Desk Top Publishing-Center
computer - fachgeschäft
rösler
Rheingutstr. 1 • ☎ 0 75 31-2 18 32

7890 Waldshut-Tiengen

hettler-data
service gmbh
Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon 07751/3094

7918 Illertissen

bidtech gmbh
technische Informationssysteme
Computerladen

Marktplatz 13
7918 Illertissen
07303/5045

8000 München

ludwig
COMPUTER + BÜROTECHNIK
COMPUTER · SOFTWARE · PERIPHERIE
BERATUNG · TECHN. KUNDENDIENST
INGOLSTÄDTER STRASSE 62L
EURO INDUSTRIE-PARK • 8000 MÜNCHEN 45
TELEFON 089 / 31 81 95-0 • TELETEX 89 83 41

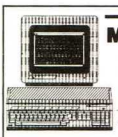
City Studio
Rindermarkt 6, 8000 München 2
Tel. 089 / 2 60 98 01

schulz computer
Schillerstraße 22
8000 München 2
Telefon (089) 59 73 39
Beratung · Verkauf · Kundendienst

8032 Gräfelfing

ProMarkt
Pasinger Straße 94 8032 Gräfelfing
Telefon 089 - 8548823
Fax 089 - 8541764

8150 Holzkirchen



MÜNZENLOHER GmbH
Tölzer Straße 5 * 8150 Holzkirchen
Tel.(08024)1814 * Fax.(08024)4879

ATARI-SCHNEIDER-NEC
PANASONIC-PHILIPS
TOSHIBA PORTABLE-LUCKY GOLDSTAR
Hard- und Software in großer Auswahl

Service und Beratung sind bei uns Inklusive

8200 Rosenheim

ludwig
COMPUTER + BÜROTECHNIK
Kufsteiner Str. 11, 8200 Rosenheim
Tel. (08031) 38 00 30
Fax (08031) 1 53 34

8400 Regensburg

WITTICH
COMPUTER GMBH
Luitpoldstr. 2
8400 Regensburg



Tel. (0941) 562530
Fax (0941) 562510

8423 Abensberg

WITTICH
COMPUTER GMBH
Tulpenstr. 16
8423 Abensberg
☎ 0 94 43 / 4 53



8500 Nürnberg

hib
hib Computer GmbH
Äußß. Bayreuther Str. 57–59
D–8500 Nürnberg 10
Telefon 0911/99514–0
ATARI Desk Top Publishing Center

8520 Erlangen

Computerservice Decker
Meisenweg 29 - 8520 Erlangen
Telefon 09131 / 42076

8600 Bamberg



Das Haus der großen Auswahl
BÜRO-ZENTRUM A+R KUTZ
8600 Bamberg · Am Kranen 12a
Tel. 09 51 / 2 78 08 · Fax 09 51 / 2 78 00

8700 Würzburg

SCHMOLL BÜROTEAM
Hardware · Software
Service · Schulung
computer center
am Dominikanerplatz
Ruf (09 31) 30808-0

8720 Schweinfurt

Uhlenhuth GmbH
Computer + Unterhaltungselektronik
Albrecht-Dürer-Platz 2
8720 Schweinfurt
Telefon 09721 / 65154

8900 Augsburg

Adolf & Schmoll Computer
Unser Plus: Beratung u. Service
Schwalbenstr. 1 · 8900 Augsburg-Pfersee
Telefon (08 21) 52 85 33 oder 52 80 87
Computer Vertriebs- und Software GmbH

Bei uns werden bringt
GEWINN



Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057
BUF

O-5800 Gotha

ASCELL
DTP & MIDI Studio GmbH
ATARI - System - Center
5800 Gotha, Waltershäuser Str. 69
Tel.: 00622 / 55666
Mo - Mi: 11 - 18 Uhr, Do: 11 - 20 Uhr

O-6300 Ilmenau/Thür.

Concept
PROFESSIONAL COMPUTERS
ATARI - Fachhändler
MS - DOS
Kopierer, Drucker, Telefaxgeräte
usw., usw., usw.....
Porzellanstraße 30 · 6300 Ilmenau/Thür. · Tel. 2965

O-7500 Cottbus

COMPUTER & ART
Hard- und Softwarevertriebs GmbH
O-7500 Cottbus
Thälmannplatz 48
Telefon: Cottbus 23696
DTP-Center
ATARI Systemhändler

O-8027 Dresden

Dresdner Computer Center
Ihr erster ATARI System-Fachhändler in Dresden
Kaitzerstraße 82
8027 Dresden Tel.: 47 88 65

Bei uns werben bringt
GEWINN
Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057 BUF

ÖSTERREICH

A-1030 Wien

Ihr ST-Fachhändler in Wien
Computer-Studio
Wehsner Gesellschaft m.b.H
A-1030 Wien
Landstraßer Hauptstraße 2
Hilton-Einkaufspassage

A-1040 Wien

Ihr ST-Fachhändler in Wien
Computer-Studio
Wehsner Gesellschaft m.b.H
A-1040 Wien · Paniglgasse 18-20
Tel. (0222) 505 78 08, 505 88 93

A-1050 WIEN

KNEISZ GmbH
Schönbrunnerstr. 110
1050 Wien
Tel: 55-75-13
Tel: 55-13-80
Fax: 55-29-59/24
ATARI Friends
Autorisierter Händler

A-1180 Wien

Ihr kompetentes Atari-Fachgeschäft in Wien
COMPUTING
Tel. (0222) 408 52 56
Fax: (0222) 408 99 78
A-1180 Wien - Schulgasse 63

A-2340 MÖDLING

BESTENS BETREUT
bei
BÖHM Ges.m.b.H.
Ihr
ATARI Systemfachhändler
! 2 Jahre Garantie !
A-2340 Mödling, Hauptstr. 10
☎ 02236-86230 - Fax: 25035
DER WEG ZU UNS LOHNT SICH

Erfolgreich werben
Sprechen Sie mit uns.
Heim-Verlag ☎ (0 61 51) 5 60 57 BUF

A-2700 Wr. Neustadt

BESTENS BETREUT
bei
BÖHM Ges.m.b.H.
Ihr
ATARI Systemfachhändler
! 2 Jahre Garantie !
A-2700 Wr.Neustadt, Bahng.42
☎ 02622-20151
DER WEG ZU UNS LOHNT SICH

SCHWEIZ

CH-1700 Fribourg

FRIDAT SA INFORMATIQUE
ehem. Softy Hard's Computershop
VOTRE SPECIALISTE
Rte des Grives 4
1700 Granges-Paccot/Fribourg
Tel. 0041 (0)37 26 66 28
Fax. 0041 (0)37 26 61 06

CH-2503 Biel

UE **URWA ELECTRONIC**
Computer Hard- und Software
Ihr ATARI ST Spezialist
in der Schweiz.
☎ 032/41 35 35
Bözingenstraße 133, 2504 Biel

CH-3072 Ostermundingen

Neu
im Zentrum – Ostermundigen
..der Computerladen
in Bern–Ostermundigen
Als offiz. Atari–Computer–Fachhändler helfen wir Ihnen durch freundliche und kompetente Beratung und Support einzusteigen.
computer atelier dellsperger
Beratung – Verkauf – Support – Schulung – Datensysteme
ATARI
NEC EPSON CITIZEN HEWLETT PACKARD Computer Drucker
cad computer atelier dellsperger
Zentrum – Alpenstrasse 1
CH–3072 Ostermundigen
Telefon: 031 31 00 32

CH-4313 Möhlin

BCR Computerdienst
Bahnhofstrasse 63
CH-4313 Möhlin

Computersysteme
EDV-Beratung
Installationen
CAD Anlagen
Datenpflege + Service

ATARI
NEC
star

Tel. 061 88 30 32 FAX 061 88 30 03

CH-4500 Solothurn

Fluri Informatik

Hard- & Software, Zubehör
ATARI Schulungszentrum
Desktop Publishing
Systemlösungen

Schänzlistr. 4 4500 SOLOTHURN 1
Tel. 065 / 23 68 58 Fax. 065 / 23 16 57

CH- 4625 Oberbuchsiten

STECTRONIC M. Steck
Electronic-Computer-Shop

Hauptstr. 104/137
CH-4625 OBERBUCHSITEN
Tel. 0 62 / 63 17 27 + 63 10 27

LUXEMBURG

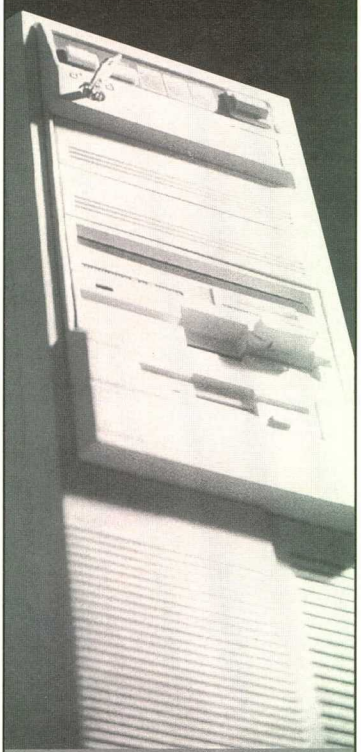
Ihr Spezialist + Service für
Computer
Commodore
Schneider
Atari
7 av. Victor Hugo · Luxembourg · Tel 20148
bürodatik

Inserieren in der ST-COMPUTER bringt Erfolg.
Über 100.000 Druckauflage

Heim Verlag

MANche Leute
HATTEN einmal
einen ganz normalen **ST...**

...jetzt erleben sie die Faszination des
Tower Tuning Set und
bringen ihren Mega ST so richtig in Form.



Komplett vorbereitetes Tower mit 220 W Netzteil, Bedienungselementen, Kabel, Adapterplatinen, Softwarepaket und Handbuch zum einfachen Selbsteinbau vorhandener Mega ST Boards, Festplatten usw; Adapterplatinen führen alle Standardschnittstellen heraus, zusätzlich SCSI-Out-Adapter, SubD-Buchse für Multisync-Monitore inkl. Farb-/Mono-Umschaltung, Parallelport mit Treiber, Scanneranschluß beschaltet, Tastaturschloß, ROM-Port intern und extern, per Software schaltbar, 8 Stromversorgungsanschlüsse, 8 softwaremäßig steuerbare Schaltausgänge: Logik für HD-Floppies, hard- und softwaremäßige Turboschaltung und erweiterte Grafikauflösung; Hardware-Bildschirmschonschaltung, Bootlogik, NF-Verstärker. DM **1498,-**

TETRA®
Computersysteme GmbH

Neuer Markt 27 Tel.: (0 22 25) 1 70 81
D-5309 Meckenheim Fax: (0 22 25) 1 70 83

Zum Glück noch
rezeptfrei!



Wirkt nachhaltig gegen chronischen Ärger mit der Buchhaltung.

Wirkstoffe: 100.000e wohldosierter Bytes

Anwendungsgebiete:
Problemlose Einnahme-Überschuß-Rechnung (fibuMAN e + m) und Finanzbuchhaltung nach dem neuesten Bilanzrichtliniengesetz (fibuMAN f + m)

Nebenwirkungen:
exzellente Verträglichkeit mit:
fibuSTAT - graphische Betriebsanalyse
faktuMAN - modulares Business-System

Gegenanzeigen:
Verschwendungssucht, akute Aversionen gegen einfache und übersichtliche Buchhaltung

fibuMAN Programme gibt es schon ab DM 398,-
* unverbindliche Preisempfehlung Atari ST, Preise für fibuMAN MS-DOS® und Apple Macintosh® auf Anfrage

Testsieger in DATA WELT, 6/89
4 MS-DOS® Buchführungsprogramme im Prüfstand; davon 3 mit 8.23, 8.25, 8.65 Punkten (max. 10)
fibuMAN mit der höchsten Punktzahl des Tests 9.35

fibuMAN begeistert Anwender wie Fachpresse!
Nachzulesen in: c't 4/88, DATA WELT 3/88, 6/88, 5/89, 6/89, ST-COMPUTER 12/87, 12/88, ST-MAGAZIN 4/88, 10/88, ATARI SPECIAL 1/89, ATARI MAGAZIN 8/88, ST-PRAXIS S/89, ST-VISION 3/89, PC-PLUS 5/89

NEU
1ST fibuMAN
Die Einsteiger-Buchführung
DM 148,-*

NOVOPLAN
Hardtstraße 21 · 4784 Rüthen 3
Tel. (02952) 80 80 * (0161) 2215791
Telefax (02952) 32 36

Senden Sie mir für fibuMAN ○ INFO ○ Demo mit Handbuch ich arbeite mit dem System ○ MS-DOS 5, ATARI ○ Macintosh
Mein Name:
Firma:
Straße/Nr.:
PLZ/Ort:
Demo mit Handbuch DM 65,-

Kleinanzeigen

Biete Hardware

HD-System f. ST, 60MB M-ALIA-IF nach CT, ohne Netzteil, 600,–, SM 124, 150,–, GLP 100, Sony CPDIOOOE, 400,–. Abends 0. WE (06131) 368774

MEGA ST4, SM 124, PC-Speed, Vortex HDplus 30, NEC P6+, 5,25" Lfw, TOS 1.0-1.4, Zubehör, VS, Tel. (02 09) 78 51 00

Netzt. P531 45,–; Ersatz-IC Shift DMA, MMU, GLUE a 85,–; FDC1772 30,–; Tast. 40,–; Sound 15,–; AtariMaus 55,–; DRAM 41256 2,80; Tastatur 520ST 30,–; TOS 1.0 35,–; 1.4 65,– (6 IC); 1.2 45,– (2 IC). Tel. (08238) 4817

PC-Speed, neuw. 250,– DM Tel. (0931) 57030, ab 16 Uhr

Coprozessor **MC 68881,** 12 MHz. 170,– DM. (09 13) 5 52 04

Laserdrucker SLM 804 für Atari günstigst abzugeben, kaum benutzt, mit Originalverpackung und Garantie. 40% unter Neupreis für nur 1900,– DM. Tel: (089) 34 91 56 oder Tel: (07 11) 710363

TOS 1.4 mit Patches aus ST-Computer 1+10/90 in 6 ROMs für 60 DM bei Vorkasse, sonst + NN: Postfach 3345, 2900 Oldenburg

SM 124 mit Garantie bis 4/91 also nagelneu für 300,– DM Andreas Wieneke (02 02) 43 10 00

Speichererweiterung für ATARI ST 2.5 MB: DM 450,–; 4 MB: DM 880,–, Einbau + 50,– DM, mit Garantie heiler engineering (07 21) 37 66 22

SCSI-Platte Lacom 84 MB/19ms, Quantum P80S, Stahlgeh., ger. Lüfter, Softw.; 7 Mon. benutzt, noch **15 Mon. Drive-Garantie;** Neu 2200,–, jetzt VB 1500,– M. Leidig PF 1820, 8650 Kulmbach

HEGA-ST-Tastaturen, Floppies, Netzteile, Mäuse neu u. original. Rauch (07031) 52607

Einige **MEGA-ST-Leergehäuse** zum Einbau von Festplatten usw. Rauch (07031) 52607

2 Hypercache (350,— u. 500,–). Einige **SM 124,** neu (280,–), gebr. (150,–). Rauch (07031) 52607

MEGA ST2, SM 124, TOS 1.4, Vortex HD20plus, original 1st Word 3.15, Tempus 2.0, ST-Pascal 2.0, PD, Bücher, VB 1650,– (09 11) 33 88 24, nach 19 Uhr

VERKAUFE: Profibank mit 1,5 MB, Akku Profiram mit 380 KB, Treiber alles komplett bestückt für 500, Tel: (06 11) 806913

Videodigitizer „Easytizer" für 200,– Tel. (0931) 409679

Speichererweiterung für 260/520 ST 2.5 MB: DM 399,–; 4 MB: DM 749,– Einbau, 12 Monate Garantie: DM 50, heiler engineering (07 21) 37 66 22

PC-Speed, Version 1.4 mit Buch: PC-Speed Know-how 295,– DM Tel.: (02 11) 354876, ab 18 Uhr

Blitter-Chip 50,- Tel. (069) 5 07 16 94

MEGA ST4 / HD 30 / LC24 / Wordperfect, Calamus, Zubehör; alles neu Preis: 3500,– DM Tel. (05224) 3715

Rainbow TOS 1.4 mit ST-Computer Patches 70,– **Schaltpläne** ST/ Mega/SM124 je 25,– **3 MByte** Erweiterung 400,– (04351) 44252

MEGA ST4, TOS 1.4, HyperCache, FPU, SM 124: VB 2300,–; **SPECTRE GCR:** VB 700,–; NEC P6: VB 500,– **1040 ST,** TOS 1.4, SM 124: VB 800,– W. Ditz, Roterweg 4, 658 Idar-Oberstein, Tel. (06781) 46639

Turbo C 68881, GFA und OMICRON BASIC Comp. 68881, PC-Tast. Modul+ Softw., Tempus, Aladin 3.0, Starwriter ST. Zus.: VB 350,– W. Ditz, Roterweg 4, 6580 Idar-Oberstein, Tel. (06781) 46639

MEGA ST4, TOS 1.4, eingebaute CSH 105MB Festplatte, gegen Höchstgebot Tel. (07348) 53 12, ab 20 Uhr

Atari MEGA ST4 mit Coprocessor 68881, SM 124, Megafile 60MB, QWERTY Tastatur, STAR LC10-24, Software, wie neu, 3600 DM. Tel. (089) 3 15 91 39, ab 18 Uhr

I/O-INTERFACE ITF-32, 32 TTL-komp. Ein- u. Ausgänge f. Ind.-Steuerungen, Alarmanlage etc. J. Sieber, Raffelterssteige 15, 7100 Heilbronn, Tel. (07131) 161151

80MB Quantum Festplatte 950,–, 44MB SyQuest Wechselplatte für 950,– DM. 1MBit RAMs für 8,50 DM das Stück. Alle Teile sind neu. Tel. (030) 6184457

520 STF, 4MB, SF 314, SF 354, TOS 1.4, RTS, Omicron BASIC, VB 1500 DM. Tel. (07306) 2964 oder (07 11) 649351

STE-Speichererweiterung auf 2MB DM 240 / 4MB DM 540 Manthe Michael, Salinenstraße 46, 7730 VS-Schwenningen

Die neue Megafile 30: nur 598,– SM 124: nur 148,–!! (09372) 72588

Biete Software

Alle PD's aus ST-Computer für DM 2,– je Disk, 10 Disks DM 19,– Versandk. DM 4,– (bei Nachnahme 7,–) PD-Point Oelde A. Wessolek Schorlemer Straße 55, 4740 Oelde

100 Disketten 3,5" 2DD 93,– DM Festplatten a. Anfr. (02922) 82195

Adimens 2.3 99,– Profimat 65,– Tel. (07 11) 88 35 87

PD AB 1,-DM, (06421) 62763, Hans-M:06

ST-MATH VB; Megamax C E-System 100 DM; RTOS-Pearl+GKS 200 DM ALLES ORG. SOFTWARE; (0841) 69836

Originale für Atari ST (inkl. Handbuch, etc.), 1st ADDRESS 64,50; OPTIMIZER 59,90; ALADIN 3.0 TIMEWORKS 1.12 (PostScript); HERMES 35,–; CALAMUS 1.09 (auch 1MB), ADI-PROG 2.3 (GFA); R. Riesenhuber, Auf dem Kamp 2, 2358 Kaltenkirchen, Tel. (04191) 4320

ORIGINALE: LDW-Power-Calc, MCC Pascal je 120 DM, ST-Math: 50 DM, ST-Analog: 45 DM. Tel. (06 11) 400672 (abends)

Lottofee * komf. Lottoprogramm * inkl. aller Voll + VEW-Systeme * Sp77 Ausfüllen der Scheine * Alle Zahlen ab 1955, Statistik, uvm. * DM 30, Scheck, HvB, Pilgerborn 53, 5106 Roetgen

Preisbrecher! ST-PD-Software! ST-Comp., 2000er, J- u. V-Serie jede Disk nur 2,– DM, Abo 1,80, Anrufen u. Tel.: (02 01) 733497

######## **PD-SOFTWARE** ######### auch SIGNUM- und GRAFIK-Pakete KATALOG-DISK 3DM/Kurz-INFO ODM bei: B. Jürgensen, Holmberg 4, 2398 Harrislee

8051-MAKRO-ASSEMBLER, schnell, komfortabel, vielseitig, **SPS-Befehlseinbindung,** Info kostenl. ST oder MS-DOS 190,– DM; E. Hakemann, Barnstorfer Str. 91, 2849 Goldenstedt

ST-Pascal+ (150,–), Quick Dialog (15,–), Laser C (150,–), Tempus 1.1 (25,–), Wordplus 3.15 (100,–), Flugsim II (45,–), Impossible Mission II (25,–), Kennedy Appr. (25,–). Tel: (089) 2 72 41 87, ab 19 Uhr

Supergünstig DESK-ASSIST/4 Terminplaner, Adressendatei, Calculator, Drukkerspooler/-voreinstellung, Diskettenformatierung bis 830 K uvm. 130seitiges Handbuch, **DM 38,–** Tel. (02225) 17081

Technobox CAD2 V.1.4 VB 1350 DM Wordperfect 4.1 VB 300 DM Scheibenkleister 2 VB 50 DM Hardwarebuch (M&T) VB 50 DM ST-Pascal+und Buch (M&T) VB 180 DM Alexander, (02 02) 62 80 36, ab 19 Uhr

PD-Einzelkopierservice, 0,8 Pf/KB, über 1000 PRGs, Katalogdisk gegen Rückporto (DM 1,70) anfordern! H. Schilling, Willbecker Str. 67, 4006 Erkrath

ICH PROGRAMMIERE IHR (!) PROGRAMM! Anfragen an Marco Nissen, Hainweg 20, 2264 Süderlügum

Anwender- u. Spielesoftware
Laufend n. Programme a. Lager
Riesenauswahl an Original
Soft- u. Hardware
Gratiskatalog anf. o. anrufen
W. Wünsch
Soft- u. Hardware,
(07231) 766595
Friedenstraße 212, 7530 Pforzheim

- PD-Disks aus ST-Compu.(S) und PD-
- Journ. (J) je 2,50 DM / Abo 2 DM,
- Porto: Vk 4 DM od. NN 8 DM/
- Katalogdisk 4 DM bei: S. Heigert,
- Berndesallee 6, 6501 Heidesheim,
- Tel. (06132) 5461 (18-20 Uhr)

Public-Domain-Software
alle bekannten Serien
Preis 3,40 DM inkl. Diskette
Lieferung innerh. 24 Stunden!
Liste auf Diskette lieferbar
Tel. (04134) 8689

That's Pixel, das Grafikprogramm für den Atari ST wegen Systemwechsels für DM 60,– zu verkaufen. Tel. (04921) 51188 nach 18 Uhr und WE

EIGENES GESCHÄFT? Wir geben unseren Soft- u. Hardwarehandel aus persönl. Gründen zum 31.12.90 auf. PD-Sammlung (1500 Disk. für ATARI ST) aus ST-Computer, ATARImagazin, XEST, PD-Journal (Reihen J, D, V), GFA-Club, PD-Pool für 3700,– DM. Zusatzangebot: PD's + 2200 Kd-Adressen mit Faktura + PD-Katalog auf Diskette + Anschreiben aller Kd. durch uns, 4900,– DM. CHIFFRE

Editorprogramm für Technics KN 800/ Atari ST für 170,– DM VB, 2 Stck. Seagate ST 412 Festpl. 11 NB à 50,– DM, Rack für 19 Zoll Einbaugeräte 250,– DM. Tel. (02 31) 10 16 96

1st Word+/1st Mail 2.02 Original DM 70,–, Tel. ab 18 h: (09874) 1325

1st Word+/1st Mail 2.02 mit Registrierkarte, neu, DM 90,–**Adimens 2.3** mit Registrierkarte, neu, DM 120,– Tel. ab 18 h: (09874) 1325

Suche Software

Lernprogramm für Jurastudium P. Egbers, Stahlstraße 9, 3000 H. 71

Sportspiele für ST zu kaufen gesucht. Nur Originale! Liste an G. Harleß, Sonnenstraße 14, 8806 Neuendettelsau, oder Tel. (0 98 74) 13 25

Bridgeprogramme für 1040 ST. Angebote: Boers, Raadhuispl. 6, NL-6367 ED Voerendaal, 00 31 45 75 20 16

Verschiedenes

Löse ST-PD-Sammlung (150 Disk) auf! Liste von: Ulf Baumart, Bonhoeffer Str. 4, 4172 Straelen

ST-Computer Hefte 1/88-12/90 für 120 DM zu verkaufen Tel. (02222) 63244

An Programmierer u. Softwarehäuser! Sachbuchautor (Profi) schreibt oder korrigiert Ihre Handbücher. R. D. Zens (02266) 44228

Vermittle gebrauchte Hard- und Software aller Computersysteme zum Verkaufen und Kaufen! INFO kostenlos: Tel. (07371) 7167

„Atari ST Profibuch", Sybex. Neuwertig! DM 35,– Tel. (06152) 83112, n. 18 Uhr

Calamus-Fonts zu tauschen ges. Biete: Script, Excelsia, Baskerville, Fraktur, Royal, Avanti. Angebote an G. Harleß, Sonnenstraße 14, 8806 Neuendettelsau, Tel. ab 18 h: (09874) 1325

Kontakte

Wer hat NEC P2 am Atari ST und kann mir helfen? Tel. (02223) 1567

Suche Kontakte zu ST-Usern im Großraum Leipzig/Halle zwecks Erfahrungsaustausch (PC-Speed), Claus Möckel, Ernst-Thälmann-Straße 164, O-Leipzig



INHALT

Schnelle Strings
GFA-BASIC ..... Seite 74

Korrekturhilfe selbst gemacht
Modula-2 ..... Seite 78

Ganz kurzes Accessory - die Zweite
Assembler ..... Seite 90

Speicherverwaltung noch effizienter
C ..... Seite 93

SCHNELLE STRINGS

Andreas Hollmann

Hier stelle ich Ihnen zwei Routinen zum Laden und Sichern von String-Feldern im ASCII-Format vor. Der aufmerksame GFA-BASIC-Programmierer wird da sofort sagen: „Gips doch schon!". Nämlich in Form der Befehle STORE und RECALL. Doch leider haben diese Befehle die Einschränkung, dass nur die Übertragung von kompletten String-Feldern möglich ist. Will man aber nur einen Block laden/sichern, muss man auf diese sonst sehr komfortablen Befehle leider verzichten.

Zusätzlich sollten die Routinen (so ganz nebenbei) natürlich auch schneller sein, so daß der Anwender erst gar keine Zeit mehr hat, zur Kaffeetasse zu greifen.

Die schnellste Möglichkeit, Daten zu sichern, ist das Schreiben in einem einzigen Block; der dazugehörige Befehl heißt *BSAVE*. Bei numerischen Feldern läßt sich dies einfach programmieren: man berechnet die Adresse, an der die Inhalte der Feldelemente stehen, und aus deren Anzahl und dem Platzbedarf pro Element die Blocklänge. Dieser Speicherbereich wird nun mit *BSAVE* gesichert, was erheblich schneller geht als mit einer herkömmlichen Routine.

Das Sichern von numerischen Feldern ist deshalb so einfach, weil die einzelnen Feldelemente hintereinander im Speicher stehen. Aber genau dies ist bei einem String-Feld nicht der Fall. Warum das so ist, zeigt die Variablenstruktur eines String-Felds in Bild 1. Da, wo bei einem numerischen Feld die Inhalte der einzelnen Feldelemente stehen, befinden sich beim String-Feld lediglich die Deskriptoren der einzelnen Strings. Jeder dieser Deskriptoren ist 6 Bytes lang. Die letzten 2 Bytes geben die Länge des Strings an, die ersten 4 Bytes enthalten die Adresse, an der die Zeichenkette sich im Speicher befindet.

Der eigentliche Inhalt der Strings befindet sich also in einem anderen Speicherbereich als das String-Feld. Diese Trennung ist notwendig, da der Speicherverbrauch eines Strings im Gegensatz zu dem einer numerischen Variablen nicht konstant ist. Wird ein String verlängert, kann man in den meisten Fällen nicht einfach den zugefügten Teil dahinterschreiben. An dieser Stelle könnte sich bereits ein anderer String befinden, welcher dann überschrieben würde. Der verlängerte String wird an einer anderen Stelle im Speicher abgelegt, und der Bereich des 'alten' Strings muß freigegeben werden.

Mit der Zeit enstehen Lücken im Speicher, wodurch die Speicherverwaltung immer aufwendiger und uneffektiver wird. Von Zeit zu Zeit führt GFA-BASIC deshalb eine sogenannte Garbage-Collection durch. Bei dieser Müllsammlung werden die Strings wieder zusammengerückt und die Lücken geschlossen. Am Ende des Speichers entsteht dann wieder ein größerer zusammenhängender freier Bereich. In diesen können dann geänderte Strings geschrieben werden, und das Spiel geht von vorne los.

Nun könnte man auf den Gedanken kommen, daß die einzelnen Strings nach einer Garbage-Collection hintereinander im Speicher stehen. Im Prinzip ist das der Fall, doch leider nicht in der richtigen Reihenfolge. Die Garbage-Collection eliminiert nur die Lücken, ordnet die Strings jedoch nicht in aufsteigender Folge, wozu auch? Selbst wenn die Reihenfolge eingehalten würde, könnte man den Block nicht als ASCII-Datei sichern, da sich am Ende jeden Strings ein sogenannter 'Backtrailer' befindet. Dieser enthält die Anfangsadresse des Strings und dient zur schnelleren Durchführung der Garbage-Collection.

Noch ein Wort

Eine weitere Hürde ist die wortweise Speicherorganisation der Strings. Im Unterschied zur Programmiersprache C wird das Ende eines Strings nicht durch den ASCII-Code 0 gekennzeichnet. In GFA-BASIC beginnen und enden Strings immer auf Wort-Adressen. Die tatsächliche Länge des Strings ist im Deskriptor vermerkt. So schleppen Strings mit einer ungeraden Länge immer ein überflüssiges Byte am Ende mit. Der Vorteil dieser Methode ist jedoch eine hohe Geschwindigkeit der gesamten String-Verwaltung, ein Aspekt, der das eine Byte-Anhängsel rechtfertigt.

Der einzige Ausweg ist die Verwendung eines zusätzlichen Speicherbereichs, in den alle Strings in aufsteigender Reihenfolge durch CR und LF voneinander getrennt kopiert

werden. Die Länge dieses Speicherblocks berechnet sich ganz einfach aus der Summe der Längen aller Strings plus je 2 Bytes für die Zeilenende-Kennung. Nun liegt das Feld in einem Block vor und kann mit *BSAVE* gesichert werden. Das Kopieren des Felds benötigt zwar auch eine gewisse Zeit, diese wird aber durch das schnellere Sichern mehr als aufgeholt. Steht nicht genügend Speicher zur Verfügung, wird das Feld in herkömmlicher Weise geladen/gesichert.

Im Programm wird zuerst der freie GEMDOS-Speicher angezapft; ist hier nicht genug frei, erfolgt das Laden/Sichern mit der normalen, langsamen Methode. Nach Beendigung des Datentransfers wird der allozierte Speicherblock ggf. wieder freigegeben.

Die Benchmarks in Tabelle 1 zeigen das Geschwindigkeitsverhältnis der einzelnen Sicherungsmethoden. Der Vergleich mit Tempus kann sich sehen lassen, hier werden zum Sichern auf Diskette ca. 14.5 s und auf Platte ca. 1.5 s benötigt.

Als nächstes soll mit der gleichen Methode ein Stringfeld von Diskette/Festplatte eingelesen werden. Die Länge des Speicherblocks entspricht hier genau der Dateilänge. Nach dem Laden der Datei müssen Anfang und Ende der einzelnen Strings ermittelt werden, in einer Schleife sucht man nach dem ASCII-Code 13, der das Zeilenende markiert. Jetzt sind die Anfangsadresse und die Länge des zu kopierenden Strings bekannt. Doch leider kann nicht mit dem schnellen *BMOVE*-Befehl kopiert werden, da die Zieladresse im GFA-BASIC-Variablenbereich nicht bekannt ist.

Bisher ist mir keine Möglichkeit eingefallen, wie man den freien Speicher im String-Bereich erfragen könnte (entsprechende Tips erbeten). Ohne diese Abfrage ist für GFA-BASIC aber jede Kopieraktion in den Variablenbereich eine Einmischung in die inneren Angelegenheiten und daher nicht erfolgversprechend. Es bleibt also nur der Transfer mit *CHAR{ }*, welches nur byteweise und dadurch langsamer kopiert, und einer String-Zuweisung übrig.

Da *CHAR{ }* zum Erkennen des String-Endes den ASCII-Code 0 benötigt, wird vorher der ASCII-Code 13 durch eben diesen ersetzt.

Jetzt kann der String seinen neuen Platz im Feld einnehmen. Durch das byteweise Kopieren ist zu befürchten, daß die Beschleunigung der Laderoutine nicht so groß ist wie beim Sichern. Eine weitere Bremse ist das byteweise Suchen nach dem Ende eines Strings.

Man sollte bedenken, daß eine ASCII-Datei wesentlich unstrukturierter aufgebaut ist als ein Feld. Bei einem Feld existieren in den Deskriptoren Einträge für Länge und Position der einzelnen Elemente. In einer ASCII-Datei dagegen sind alle Daten einfach seriell angeordnet. Will man sich Auskunft über ein bestimmtes Element verschaffen, muß zu zeitaufwendigen Suchroutinen gegriffen werden. Selbst eine so einfache Abfrage wie die Feststellung der Zeilenanzahl ist nicht möglich, ohne vorher die ganze Datei geladen und durchsucht zu haben. Bei einem Feld hingegen ist nur ein einziger Lesezugriff am Anfang der Deskriptorenliste erforderlich. Das byteweise Suchen nach dem Zeilenende (ASCII 13) und das byteweise Lesen des Strings mit *CHAR{ }* werden sich also negativ auf das Laufzeitverhalten der Laderoutine auswirken.

Die Benchmarks für das Laden der Testdatei (Tabelle 2) zeigen den erhöhten Zeitverbrauch. Beim Laden von Festplatte überwiegt die höhere Datenübertragungsrate, das Kopieren dauert hier länger. Bei der Arbeit mit Disketten läßt sich hingegen noch eine geringe Steigerung erzielen. Es bleibt jedem Programmierer selbst überlassen, ob ihm die 28% Steigerung den längeren Programmcode wert sind. Da aber das Laden von Daten ohnehin schneller als das Schreiben ist, kann man auch gut mit der normalen Sicherungsroutine leben. Die Zeiten wurden übrigens mit den compilierten Routinen gemessen. Das Sichern dauert im Interpreter nur wenig länger, beim Laden dagegen verbraucht das byteweise Suchen nach dem String-Ende etwa die 22fache Zeit. Haben Sie beim Test im Interpreter also etwas Geduld.

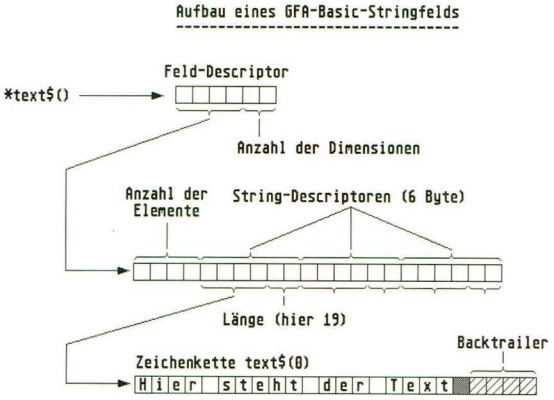


Bild 1: So ist ein String-Feld in GFA-BASIC intern aufgebaut. Der abgebildete String ist zwar nur 19 Zeichen lang, belegt aber wegen der Festlegung auf Wortgrenzen 20 Bytes (das zusätzliche Byte ist grau markiert). Am Ende des Strings befindet sich der sogenannte Backtrailer, der auf den String-Anfang zurückzeigt und eine schnellere String-Verwaltung ermöglicht.

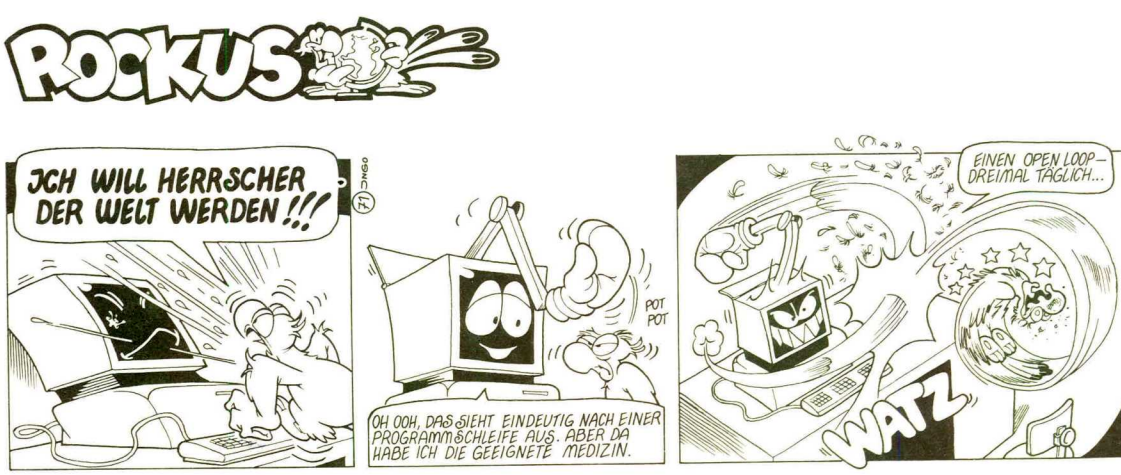
Methode	Diskette		Festplatte	
	[s]	[%]	[s]	[%]
Befehl STORE	36,65	100	3,45	100
normal	36,95	99	4,10	84
schnell	13,61	269	1,44	240

Tabelle 1: Die Zeiten zum Sichern der ASCII-Datei. Zum Test wurde eine Datei mit 1000 Zeilen zu je 80 Zeichen unter TOS 1.4 benutzt; die Festplatte war eine Protar ProFile 30 SCSI-Platte.

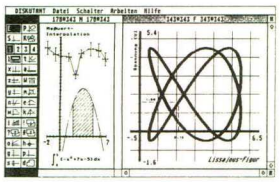
Methode	Diskette		Festplatte	
	[s]	[%]	[s]	[%]
Befehl RECALL	9,42	100	1,51	100
normal	13,41	70	4,82	31
schnell	7,38	128	2,40	63

Tabelle 2: Die Zeiten zum Laden einer ASCII-Datei. Beim Laden von der Festplatte holt GEMDOS wieder auf, hier ist der RECALL-Befehl schneller (er bietet aber leider trotzdem kein blockweises Laden).

```
' ************************************************
' * zwei Routinen zum Laden und Sichern          *
' * von Stringfeld-Blöcken als ASCII-Datei       *
' * Programmautor: Andreas Hollmann              *
' * Sprache:       GFA-Basic                     *
' * Copyright:     MAXON Computer GmbH 1991      *
' ************************************************
RESERVE 128000   !sonst kein GEMDOS-Speicher frei
DIM txt$(1000)
'
FOR i&=0 TO 999                          !Feld füllen
  txt$(i&)=STRING$(80,"A")
NEXT i&
'
' Aufrufe zum Laden/Sichern:
' block_load("a:\test.txt",0,1000,txt$())
block_save("a:\test.txt",0,1000,txt$())
'
ERASE txt$()
RESERVE
END
'
PROCEDURE block_save(path$,ln1%,lns%,VAR txt$())
  ' Stringfeld-Block als ASCII-Datei sichern
  ' ----------------------------------
  ' Parameter:
  '  path$   = Pfad und Dateiname der ASCII-Datei
  '  ln1%    = erste Zeile des Blocks
  '  lns%    = Länge des Blocks
  '  txt$() = Stringfeld
  ' ----------------------------------
  LOCAL p_dsc%,ln%,buf_size%,p_buf%,len%
  ' Zeilenanzahl > max. Feldindex?
  IF ln1%+lns%>DIM?(txt$())
    lns%=DIM?(txt$())-ln1%
  ENDIF
  ' Adresse der String-Descriptoren berechnen:
  p_dsc%=LONG{*txt$()}+4
  ' erforderliche Länge des Puffers berechnen:
  FOR ln%=ln1% TO ln1%+lns%-1
    ADD buf_size%,LEN(txt$(ln%))
  NEXT ln%
  ADD buf_size%,(lns%)*2   !+2 Byte für CR LF
  '
  ' Feld in GEMDOS-Block kopieren und sichern:
  IF MALLOC(-1)>=buf_size%   !Speicher frei
    p_buf%=MALLOC(buf_size%) !her damit
    p_bufstart%=p_buf%       !Adresse sichern
    FOR ln%=ln1% TO ln1%+lns%-1
      len%=WORD{ADD(p_dsc%,4)} !Stringlänge lesen
      BMOVE {p_dsc%},p_buf%,len% !String kopieren
      ADD p_dsc%,6               !6 Byte weiter
      ADD p_buf%,len%            !+ Stringlänge
      BYTE{p_buf%}=13            !ASCII 10 = LF
      INC p_buf%
      BYTE{p_buf%}=10            !ASCII 13 = CR
      INC p_buf%
    NEXT ln%              !Alle Zeilen sind kopiert
    ' Block sichern und Speicher freigeben:
    BSAVE path$,p_bufstart%,buf_size%
    ~MFREE(p_bufstart%)
  ELSE                            !nicht genug Speicher
    OPEN "o",#1,path$             !normal sichern
    FOR ln%=ln1% TO ln1%+lns%-1
      PRINT #1,txt$(ln%)
    NEXT ln%
    CLOSE #1
  ENDIF
RETURN
'
PROCEDURE block_load(path$,ln1%,lns%,VAR txt$())
  ' ASCII-Datei als Stringfeld-Block laden
  ' ----------------------------------
  ' Parameter:
  '  path$   = Pfad und Dateiname der ASCII-Datei
  '  ln1%    = erste Zeile des Blocks
  '  lns%    = Länge des Blocks
  '  txt$() = Stringfeld
  ' ----------------------------------
  '
  LOCAL buf_size%,p_buf%,p_bufstart%,ln%,p_str%
  '
  OPEN "i",#1,path$
  buf_size%=LOF(#1)              !Dateilänge abfragen
  CLOSE #1
  '
  IF MALLOC(-1)>=buf_size%    !genug Speicher frei
    p_buf%=MALLOC(buf_size%) !her damit!
    p_bufstart%=p_buf%
    '
    BLOAD path$,p_buf%         !Datei laden
    '
    FOR ln%=ln1% TO ln1%+lns%-1
      p_str%=p_buf%            !Stringanfang merken
      ' Stringende suchen (compiliert ca. 22mal
      ' so schnell wie im Interpreter):
      DO UNTIL BYTE{p_buf%}=13         !Stringende
        INC p_buf%
      LOOP
      BYTE{p_buf%}=0         !Ende für CHAR{} setzen
      txt$(ln%)=CHAR{p_str%}  !String kopieren
      ADD p_buf%,2        !ASCII 13,10 überspringen
      ' falls Ende des GEMDOS-Blocks erreicht:
      EXIT IF p_buf%>=p_bufstart%+buf_size%
    NEXT ln%
    ~MFREE(p_bufstart%)   !GEMDOS-Block freigeben
  ELSE          !nicht genug Speicher, normal laden
    OPEN "i",#1,path$
    DO UNTIL EOF(#1)
      LINE INPUT #1,txt$(ln%)
      INC ln%
    LOOP UNTIL ln%>ln1%+lns%
    CLOSE #1
  ENDIF
  '
RETURN
```



Der Diskutant



- *Abbildungen aus ℝ in ℝ u. ℝ in ℝ×ℝ*
- *Funktionsgraphen und Wertetabellen*
- *analytische (!) Differentiation*
- *numerische Integration*
- *numerische Kurvendiskussion*
- *Interpolation und Approximation*
- *Animation (mathemat. Trickfilme!)*
- *integrierter Taschenrechner*
- *komfortable GEM-Benutzeroberfläche*
- *ausführliches deutsches Handbuch*
- *läuft mit SW- und Farb-Monitor*
- *Ermäßigung für Schüler: 25,- DM*

Perfekte Kurvenanalyse mit dem ATARI ST!

c't 9/90: "Empfehlenswert!"

Fordern Sie kostenlose Informationen an!

Der Diskutant »standard« nur *88,- DM**
Der Diskutant »de luxe« nur *148,- DM**
Demo-Version (incl. Versand) nur *20,- DM*
**+Versandkosten: 3,90 DM (Ausland 15,90 DM)*

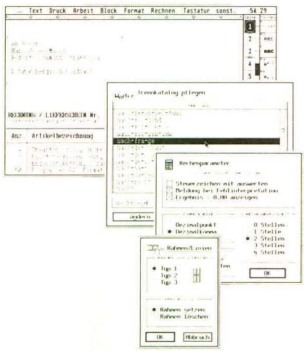
Friedemann Serbass Software
Kennwort STC
Hüninger Straße 28
1000 Berlin 33

WRITER ST

Neu Version 2.0

WRITER ST wurde speziell für Personen entwickelt, die täglich eine große Anzahl an Briefen, Texten, Rechnungen oder kleineren Dokumentationen schreiben müssen, wie klein- und mittelständische Betriebe, Handwerker, Ärzte und Anwälte. Durch die konsequente Einbindung in die graphische Benutzeroberfläche GEM ist sie für den Einsteiger leicht und schnell zu erlernen.

- Die kommerzielle Textverarbeitung auf dem ATARI ST
- Rechnen und Fakturieren im Text
- integrierte Formularverwaltung
- Makroverwaltung mit bis zu 32.000 Makros (Artikel, Adressen...)
- Serienbriefschreibung (Mail-Merge) mit Schnittstelle zu Datenbanken
- vielfältige zeilen- und spaltenweise Blockoperationen
- bis zu 4 frei belegbare Tastaturen
- eigene Zeichensätze verwendbar
- lernfähiger Trennkatalog
- eigene Briefkopferstellung
- komfortable Druckeranpassung
- lauffähig auch auf Großbildschirmen
- und vieles, vieles mehr



komplett 189,-DM incl. Mwst.

SSD-SOFTWARE
M. Schmitt-Degenhardt
Gregorstr. 1 - D-5100 Aachen
Tel. 0241/602898

Schweiz: DTZ DataTrade AG - Landstr. 1 - CH-5415 Rieden/Baden - Tel. 056/821880
Österreich: Haider Computer & Peripherie - Grazer Str. 63 - A-2700 Wiener Neustadt - Tel. 02622/24280-0
Frankreich: LOG-ACCESS - 44 rue du Temple - F-75004 Paris - Tel. 42777456

Wonderland
Computerservice Höger * Postfach 1051 * 6912 Dielheim
Tel. 06227-63587

vortex DataJet-AT	**Nr. 0704**	**DM**	**1.398,-**
40 MB Festplatten-Subsystem DataJetX40 plus ATonce			
vortex ATonce			
AT-Emulator für Atari ST und Mega ST	**Nr. 0900**	**DM**	**428,-**
Steckadapter für Mega ST	**Nr. 0901**	**DM**	**69,-**
Steckadapter für 1040 STE	**Nr. 0902**	**DM**	**89,-**

Wonderland bietet zehn Tage uneingeschränktes Rückgaberecht. 24 Stunden/7 Tage Bestellservice: Anrufbeantworter oder Postkarte/Brief. Mail-Order-Versand: Keine Beratung, kein Ladenverkauf. Lieferung nach Vorauskasse (Euroscheck max. DM 400,-) oder Post-Nachnahme. Volle Gewährleistung. Versandkosten pro Lieferung innerhalb BRD DM 10,-- . Auslandsbestellungen nur gegen Vorauskasse: Kosten nach Aufwand.

ATARI mit 16 MHz !

Speedup-16

>>> Das Projekt aus der c't 10/90 Seite 330 direkt vom Autor ! <<<
>> Ohne Cache trotzdem superschnell und dadurch fast 100% -ig kompatibel ! <<
> Funktioniert mit diversen PC/AT - Emulatoren zusammen auch im 16 MHz Modus ! <
Umschaltbar 8 /16 MHz entweder softwaremäßig oder durch einen einfachen Schalter !

Komplettbausatz incl. Treiber **DM 99.-** Fertigbaugruppe **DM 149.-**
Dallas Chipsatz **DM 22.-** Leiterplatte **DM 19.-** Sockelsatz **DM 6.50**
EPROM-Satz (Programmierservice) 6 Stück 27C256-120 nur **DM 40.-**

CEW Computer Electronic Helga Winter **Tel: 0906/ 21855**
Perchtoldsdorferstraße 12 8850 Donauwörth **Fax: 0906/ 1560**

ATARI-ST * ATARI-ST *

Dies ist Ihr **GUTSCHEIN** : **Eprommer als Bausatz nur 98.--**
fordern Sie kostenlos und unverbindlich Ihre " KUNDENKARTE"
+ unseren Gesamtkatalog für ATARI, AMIGA, C64/128 + IBM !
Hier ein **SCHNUPPERANGEBOT** :
10 Disketten 3.5 (System angeben)
voll mit Software
für nur **10.--** (Scheck oder Schein)
Katalog + Kundenkarte
liegen natürlich kostenlos bei !

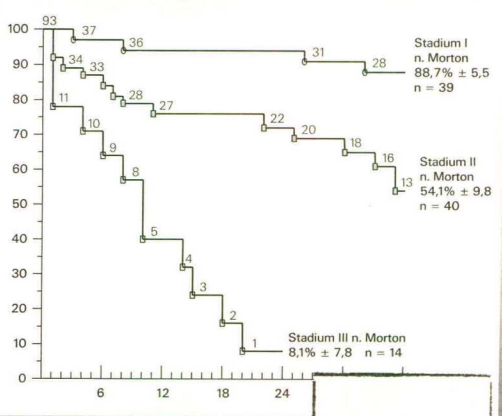
149.–
Atari ST Eprommer
Händleranfragen erwünscht !



DATA2000 Datentechnik GmbH + Co. KG
W-5800 HAGEN 1, Stresemannstraße 14-16, (Nähe Bahnhof)
Hotline 02331 / 370947/48/49/50 Fax. 330568 Mo.-Fr. 9-16.45
Ladenzeiten : Mo.-Fr. 10-13°°/14-18.30 Sa. 9-14 (Langer S.16°°)

ST Statistik

Das ideale Werkzeug für Wissenschaft und Statistik



- **Uni- und multivariate Verfahren von Crosstabs bis zur Clusteranalyse**
- **Grafiken in 2D und 3D mit Grafikeditor, Schnittstellen zu Text- und Grafikprogrammen sowie Datenbanken**

ST-Statistik kostet: DM 348,–
Demo: DM 30,–

SciLab GmbH · Isestraße 57 · 2000 Hamburg 13
Telefon: 040 - 4 60 37 02 · Fax: 040 - 47 93 44

KORREKTURHILFE SELBST GEMACHT

Wolfgang Keller

Immer taucht das Problem auf, Texte schnell auf Rechtschreibfehler hin überprüfen zu müssen. Auf Basis des hier vorgestellten Moduls lassen sich schnell Korrekturhilfeprogramme schreiben. Damit die Erstellung eines Grundwortschatzes nicht so aufwendig wird, wurde ein Format gewählt, das zu dem von First Word 1.8 ('SPELLING.DIC') vollständig kompatibel ist.

Machen wir uns zunächst einmal klar, was für Operationen auf einem Wörterbuch überhaupt gebraucht werden. Weiter interessiert noch, mit welchen Daten in welchem Umfang wir es hier zu tun haben. Für den voraussichtlichen Umfang eines Wörterbuches müssen wir für ein sinnvolles Arbeiten wohl mindestens 20.000 Wörter veranschlagen, sollten aber auch in der Lage sein, weit mehr aufzunehmen. Aus Geschwindigkeitsüberlegungen heraus ist noch zu fordern, daß die ganze Datenstruktur komplett im Speicher gehalten werden soll.

Speicherhunger

Wollte man ein Wörterbuch mittlerer Größe, also etwa mit 30000 Wörtern, einfach mit dem von Modula definierten Datentyp String = *ARRAY [0..79] OF CHAR* verwalten, so ergäbe das einen Speicherverbrauch von über 2MB. Dies ist natürlich völlig untragbar. Da in einem normalen deutschen Text Wörter im Mittel selten länger als 15 Buchstaben sind würden sich in den 2MB also etwa 85% nutzlose Zeichen tummeln. Viel ließe sich schon dadurch sparen, daß man die reinen Zeichenketten direkt hintereinander in den Speicher schreibt. Damit Wortenden erkannt werden können, müssen die Zeichenketten nullterminiert sein. Aufgrund der Tatsache, daß die String-Verwaltung die Grundlage für ein Wörterbuch bildet, kann man noch fordern, daß die Zeichenketten alphabetisch sortiert gespeichert werden. Unter diesen Randbedingungen läßt sich der Speicherhunger des Wörterbuches noch sehr viel weiter einschränken.

Diät

Bereits mit einem einfachen Kompressionsverfahren, das die Redundanz zwischen aufeinanderfolgenden Zeichenketten ausnutzt, läßt sich der Speicherbedarf für ein einzelnes Wort im Mittel auf weniger als 4 Zeichen reduzieren.

In Bild 1 ist die Idee, die dem Kompressionsverfahren zugrunde liegt, veranschaulicht. Wie das Kompressionsverfahren arbeitet, möchte ich an drei willkürlich gewählten, aber sortierten und 'ähnlichen' Wörtern demonstrieren. Die Wörter: 'Fenchel', 'Fenster' und 'Fensterglas' sind die Ausgangsdaten. Die drei Wörter würden als nullterminierte Zeichenketten genau 28s Byte Speicherplatz verbrauchen. Sehen wir uns nun die Wörter genauer an, d.h. suchen wir jeweils die Position desjenigen Buchstabens der als erster nicht mehr mit dem Zeichen gleicher Position im vorhergehenden Wort übereinstimmt. Das ergibt für den Vergleich von 'Fenchel' mit 'Fenster' den Wert 3, für 'Fenster' und 'Fensterglas' den Wert 7. (Gezählt wird hier ab Null - nicht ab Eins!). Komprimiert gespeichert wird also das ganze Wort 'Fenchel', aber dann anstatt der Null als Kennzeichnung für das Wortende schreiben wir direkt die 3 in den Speicher. Von dem Wort 'Fenster' muß dann nur noch der Rest, der nicht bereits in dem Wort 'Fenchel' vorgekommen ist, abgespeichert werden - also nur die Zeichen 'ster'. Jetzt wiederholen wir das ganze noch für 'Fenster' und 'Fensterglas' und erhalten als Position der kleinsten Nichtübereinstimmung die 7 und den Rest 'glas'. Betrachten wir das Ergebnis, zeigt sich deutlich die Einsparung. Die unkomprimierten Zeichenketten haben genau 28 Bytes beansprucht, die komprimierten Daten bringen es dagegen nur noch auf 17 Zeichen.

Sparbuch

Das heißt immerhin, 50% werden eingespart. Da in einem großen Wörterbuch die aufeinanderfolgenden Wörter jedoch meist sehr viel ähnlicher sind, lassen sich mit dieser Kompression normalerweise deutlich mehr als 60% einsparen. Da mit dem Wörterbuch ausschließlich die korrekte Schreibweise eines Wortes, nicht aber richtige Groß-/Kleinschreibung bzw. der korrekte Satzbau überprüft werden soll, werden der Einfachheit halber alle Wörter nur in Kleinschrift abgespeichert.

Aus dieser Art der Komprimierung ergibt sich aber eine Einschränkung: Da Wortendemarkierungen auch nur als ein Byte abgelegt werden, können sie nur Werte zwischen einschließlich Null und dem Ordinalwert des alphabetisch kleinsten Buchstabens annehmen. Das ist eigentlich der Buchstabe 'A' im ASCII-Zeichensatz. Er hat den Ordinalwert 65. Doch da Wörter nur in Kleinschrift gespeichert werden, ist das für unseren Fall der Ordinalwert des Buchstabens 'a', also 97. D.h. daß das Wörterbuch eigentlich Wörter mit beliebiger Länge verwalten kann. Die Nichtübereinstimmungsposition darf aber höchstens Position 96 sein. Das heißt für das praktische Arbeiten mit handelsüblichen deutschen Wörtern ergibt sich dadurch keinerlei spürbare Einschränkung. Oder wie oft kommen in

Ihren Briefen so Bandwurmwörter wie Donaudampfschifffahrtsgesellschaftskapitänswitwenrentenversicherungsgesellschaft vor? (aber selbst dieser Wurm hat nur lächerliche 78 Zeichen - also kein Problem für das vorgestellte Modul.)

Einfügen

Da nun soweit alles klar sein sollte, möchte ich kurz auf die Einzelteile des Moduls *Dictionary* eingehen. Zuerst ist da der Teil, der das Einfügen von Wörtern erledigen muß. *InsertDic* sucht als erstes die Stelle, an der das neue Wort einzufügen ist. Kennt man die Stelle, müssen die relativen Änderungen neu berechnet und im Wörterbuch Platz geschaffen werden für die Daten. Einen konkreten Fall des Worteinfügens sehen Sie in Bild 2. In die komprimierte Zeichenfolge aus Bild 1 soll nun das Wort 'Fennek' (kleiner Wüstenfuchs) eingefügt werden.

Eigentlich muß man in einem Wörterbuch ja nichts mehr löschen, denn dort sollten eigentlich nur korrekte Wörter stehen. Für den Fall, daß doch irgendein Lurmi Blödsinn mit aufgenommen hat, gibt es die Prozedur *DeleteDic*, die einen Eintrag entfernt. Das Löschen eines Wortes erfolgt (zu großen Teilen) analog zum Einfügen, nur daß bis auf die Wortsuche alles in umgekehrter Reihenfolge abläuft.

Graue Haare

Soweit sind wird ja noch auf keinerlei ernste Probleme gestoßen, die einem Programmierer graue Haare wachsen lassen könnten. Aber es gibt sie dennoch. Schließlich will man ja nicht nur Wörter in das Wörterbuch aufnehmen oder löschen. Man kann ja auch auf die Idee kommen, einen ganzen Text Wort für Wort mit den im Wörterbuch vorhandenen Daten zu vergleichen. Schon ein mittelgroßer Text würde mit der Suchmethode wie sie von *InsertDic* verwendet wird, Minuten - wenn nicht Stunden - brauchen! Denn hier haben wir es mit einem Algorithmus zu tun, der einen quadratischen Aufwand hat. D.h. bei doppelt so großem Wörterbuch dürfen wir beim Vergleichen mit einem Text schon viermal so lange warten. Das liegt daran, daß die komprimierten Daten bei der Suche sequentiell dekomprimiert und verglichen werden müssen.

Es muß also was Schnelleres her - und wenn man 1st_Word kurz mal bei der Arbeit zusieht, wird auch klar, daß das wirklich geht. Was also ist zu tun? Eine Lösung des Dilemmas ist z.B., daß man sich jedes m.te Wort des Wörterbuches in einem kleinen nichtgepackten Wörterbuch sortiert speichert und dann mit binärer Suche zuerst in diesem schnellen Wörterbuch sucht. In diesem muß zusätzlich die Position verzeichnet werden, ab der im komprimierten Wörterbuch gesucht werden soll. Im schnellsten Fall, wenn $m=1$ ist, würde das zwar heißen, wir hätten uns die ganze Komprimierung sparen können. Aber bei geschickter Wahl vom 'm' irgendwo zwischen 15 und 100 läßt sich der Speicherverbrauch klein halten und ein enormer Geschwindigkeitsgewinn verzeichnen. Wählt man $m=15$, so läßt sich 1st_Word, das seine Korrekturarbeit sehr zügig erledigt, immer noch deutlich schlagen. Das schnelle Wörterbuch wird komplett von dem Modul *FastDic* verwaltet, kann aber durchaus auch alleine und in anderem Kontext Verwendung finden.

Benehmen

Jetzt möchte ich noch die Feinheiten bzw. die Unarten des vorgestellten Moduls besprechen. Die Unart gleich vorweg - das Modul bemüht sich zwar, Fehler zu erkennen, reagiert aber grausam und unerbittlich mit einem HALT, es verzichtet also auf eine Fehlerbehandlung. Diese Stellen können noch nach Belieben verfeinert werden - hätten hier aber einfach den Rahmen gesprengt. Da sich hier auch nichts algorithmisch Interessantes versteckt, vertraue ich dabei einfach auf Ihre Programmierkenntnisse. Jetzt zu den Feinheiten. Wie schon gesagt hat 1st_Word exakt das gleiche Datenformat bei der Version 1.8 verwendet - kann aber nur Wörter bis zu 32 Zeichen aufnehmen bzw. korrigieren. Aus Geschwindigkeitsgründen ging ich immer davon aus, daß die übergebenen Daten korrekt sind. Zeichenketten werden als nullterminierte Zeichenfolgen von Kleinbuchstaben angenommen und behandelt. Da der Suchbaum von den Funktionen Einfügen und Löschen nicht aktualisiert wird, ist eine gemischte Anwendung mit Suchen nicht erlaubt. Vor der Suche muß der Suchbaum immer neu angelegt werden. Alle Routinen, die nicht direkt mit dem Wörterbuch bzw. dem Suchbaum zu tun haben, sind der Übersicht halber im Modul *DicBase* untergebracht. Die Übertragung auf andere Programmiersprachen bzw. Modulasysteme sollte ebenfalls keine allzuschwere Aufgabe darstellen.

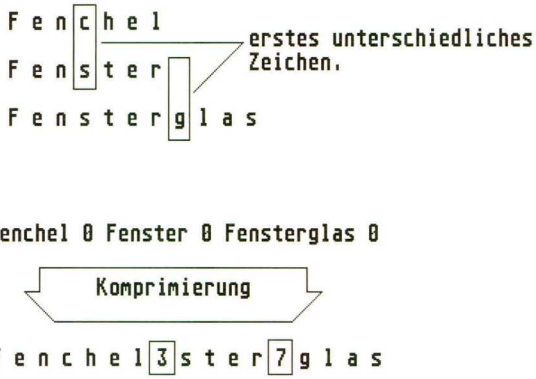


Bild 1: So funktioniert das Kompressionsverfahren.

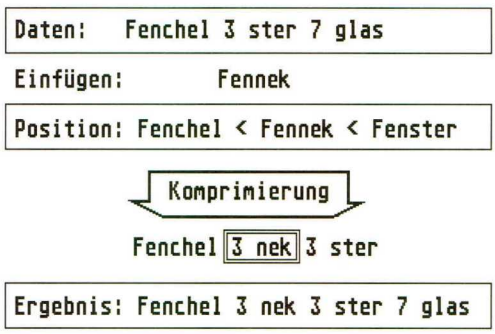


Bild 2: So wird ein Wort eingefügt.

```
 1:  DEFINITION MODULE Dics;
 2:
 3:  FROM Strings    IMPORT  String;
 4:  FROM MOSGlobals IMPORT  FileStr;
 5:
 6:  TYPE    Dic;                (* opaker Typ !!! *)
 7:          ErrorState = (ok, outOfMem, fileErr);
 8:          queryProc = PROCEDURE (Dic, VAR ARRAY OF
                        CHAR): BOOLEAN;
 9:
10:  PROCEDURE CreateDic (VAR d: Dic; over, step:
            LONGCARD);
11:
12:  (*  Liefert in 'd' eine Kennung für das
         Dictionary.
13:   *  'over' gibt an wieviel Speicherplatz beim
         Anlegen alloziert                    →
```

```
14:   *  werden soll,  'step' um wieviel der
         Speicherplatz bei Überlauf
15:   *  vergrößert werden sollte (vorgesehen aber
         noch nicht implementiert.)
16:   *  Als Wert für over gilt etwa - pro 10.000
         Worte 50 KB.
17:   *)
18:
19:   PROCEDURE LoadDic (fName: FileStr; VAR d: Dic;
                         over, step: LONGCARD);
20:
21:   (*  Lesen eines Dictionarys, Parameter - siehe
         Create. *)
22:
23:   PROCEDURE SaveDic (fName: FileStr; d: Dic);
24:
25:   (*  Schreiben eines Dictionarys. *)
26:
27:   PROCEDURE ReleaseDic (VAR d: Dic);
28:
29:   (*  Gibt den von 'd' belegten Speicher frei. *)
30:
31:   PROCEDURE InsertDic (d: Dic; REF x: ARRAY OF
                           CHAR; VAR scs: BOOLEAN);
32:
33:   (*  Wenn noch Platz war wird 'x' eingetragen. *)
34:
35:   PROCEDURE DeleteDic (d: Dic; REF x: ARRAY OF
                           CHAR);
36:
37:   (*  Löscht den Eintrag 'x'. *)
38:
39:   PROCEDURE FindDic (d: Dic; REF x: ARRAY OF
                         CHAR): BOOLEAN;
40:
41:   (*  Sucht nach 'x' - TRUE falls gefunden, FALSE
         sonst.
42:    *  SpellMode muß vorher eingeschaltet sein.*)
43:
44:   PROCEDURE QueryDic (d: Dic; x: ARRAY OF CHAR;
              elemProc: queryProc);
45:
46:   (*  Liefert 'elemProc' die Zeichenketten aus 'd'
         ab Stelle 'x[0]'.
47:    *  Abgebrochen wird falls 'elemProc' FALSE
         liefert oder das Ende
48:    *  erreicht wurde. Falls 'x' ein Leerstring ist
         wird das
49:    *  gesamte Dic durchlaufen.
50:    *)
51:
52:   PROCEDURE SetSpellMode (d: Dic; every: CARDINAL);
53:
54:   (*  every > 0 schaltet den SpellMode ein. (=0
         aus). *)
55:
56:   PROCEDURE CountWords (d: Dic): LONGCARD;
57:   END Dics.
```

Listing 1: DICS.D: Das Hauptmodul

```
 1:   IMPLEMENTATION MODULE Dics; (* Megamax Modula-2,
                                    2.2 *)
 2:
 3:   (* Format eines Dictionarys: (Regulärer Ausdruck)
 4:    *
 5:    *   Dictionary    ::= [offset, coded Strings]*
 6:    *   coded strings ::= 0C, x, [delta CHAR*]*,  x
         aus {y|IsLower (x)}
 7:    *   delta (a[0..n], b[0..m]) ist das minimale i,
         mit a[i] # b[i].
 8:    *)
 9:
10:   FROM SysUtil0    IMPORT  ClearVar;
11:   FROM Block       IMPORT  Clear, Copy;
12:   FROM Characters  IMPORT  IsAlpha, IsLower;
13:   FROM SYSTEM      IMPORT  ADDRESS, ADR;
14:   FROM MOSGlobals  IMPORT  OutOfMemory, FileStr;
15:   FROM Storage     IMPORT  Enlarge, ALLOCATE,
                                DEALLOCATE;
16:   FROM Strings     IMPORT  Assign, Relation, Lower,
                                Length, String;
17:   FROM FastDics    IMPORT  FastDic, CreateFastDic,
                                CloseFastDic, Insert,
18:                             GetPos;
19:   FROM DicBase     IMPORT  FileToMem, MemToFile,
                               ByteCopy, Check, cmp, inc,
20:                             CardSize, void0, CPtr,
                                FILESIZE, allocate;
21:
22:   CONST startChar = 'a';    fail = ' failed';
23:         MinSize   = 236L;
24:         enoughMemStr = 'enoughMem failed';
25:
26:   TYPE  Dic_Rec = RECORD
27:                     every : CARDINAL;
28:                     tree  : FastDic;
29:                     dic   : POINTER TO ARRAY
                                [0L..100L*100L*100L]
                                OF CHAR;
30:                     size, max, step : LONGCARD;
31:                     OffsetTable   : ARRAY
                                [0C..377C] OF LONGCARD;
32:                   END;
33:
34:         Dic = POINTER TO Dic_Rec;
35:
36:         strRec = RECORD
37:                    Str       : String;
38:                    Len, RLen : CARDINAL;
39:                    StartPos  : LONGCARD;
40:                  END;
41:
42:         strTab = ARRAY [0..2] OF strRec;
43:
44:         moveRec = RECORD
45:                     ReadPos, MaxPos      :
                                LONGCARD;
46:                     vorIx, aktIx, nachIx :
                                CARDINAL;
47:                     StrTab               :
                                strTab;
48:                   END;
49:
50:         TYPE Zustand = (zahl,char);
51:
52:   PROCEDURE DicAdr (d: Dic; i: LONGCARD): CPtr;
53:     BEGIN
54:       RETURN CPtr (ADR (d^.dic^ [i]));
55:     END DicAdr;
56:
57:   PROCEDURE CardToMem (card: CARDINAL; adr:
                           ADDRESS);
58:     BEGIN
59:       ByteCopy (ADR (card), CardSize, adr);
60:     END CardToMem;
61:
62:   PROCEDURE MemToCard (adr: ADDRESS): CARDINAL;
63:     VAR card: CARDINAL;
64:     BEGIN
65:       ByteCopy (adr, CardSize, ADR (card));
66:       RETURN card;
67:     END MemToCard;
68:
69:   PROCEDURE Get_Index_Adr (d: Dic; i: LONGCARD):
                              CPtr;
70:     BEGIN
71:       RETURN ADR (d^.dic^ [i]);
72:     END Get_Index_Adr;
73:
74:   PROCEDURE Get_Offset_Adr (d: Dic; c: CHAR):
                ADDRESS;
75:     BEGIN
76:       RETURN ADR (d^.dic^ [d^.OffsetTable [c]]);
77:     END Get_Offset_Adr;
78:
79:   PROCEDURE Get_Offset_Pos (d: Dic; c: CHAR):
                LONGCARD;
80:     BEGIN
81:       RETURN d^.OffsetTable [c];
82:     END Get_Offset_Pos;
83:
84:   PROCEDURE Get_0CHAR_Pos (d: Dic; c: CHAR):
                LONGCARD;
85:     BEGIN
86:       RETURN d^.OffsetTable [c] + 2L;
87:     END Get_0CHAR_Pos;
88:
89:   PROCEDURE Get_Word_Pos (d: Dic; c: CHAR):
                LONGCARD;
90:     BEGIN                                    →
```

DER ETWAS ANDERE VERSAND!

24-Stunden Service!

Wir garantieren, daß jede Bestellung spätestens 24 Stunden nach Eingang unser Haus verläßt, sofern verfügbar. Auf alle gekauften Artikel erhalten Sie natürlich volle Garantie. Wir führen jede verfügbare Hard- und Software für den Atari ST, sowie alle Bücher. Hier ein kleiner Auszug aus unserem reichhaltigen Programm:

SPIELESOFTWARE:
Anarchy 60,-
Back to the Future II 85,-
Badlands 80,-
Block Out 65,-
Botics 65,-
Cadaver 85,-
Chaos Strikes Back 80,-
Dragonflight 85,-
Elite 65,-
F-16 Falcon 80,-
F-16 Falcon Mission Disk II .. 65,-
F-19 Stealth Fighter 95,-
F-29 Retaliator 75,-
Finale 85,-
Flight Simulator II deutsch 85,-
jede Scenery Disc dazu 45,-
Flood 85,-
Geisha 90,-
Gold of the Aztecs 60,-
Immortal 65,-
Imperium 85,-
Kaiser 120,-
Kick off II 65,-
Leisure Suit Larry III 115,-
Loopz 70,-
Lords of Doom 90,-
On the Road 85,-
Operation Stealth 85,-
Pang 90,-
Party Time Compilation 80,-
Pirates 80,-
Player Manager 65,-
Populous 85,-
Powermonger 95,-
Psion Chess 75,-
Roque Trooper 85,-
Sim City 85,-
Space Quest III 95,-
Sporting Gold Compilation ... 80,-
Their Finest Hour 95,-
Turrican 65,-
Wings of Death 85,-
Zak McKracken 75,-

ATARI POWER PACK
Compilation mit 20 Super-spielen, original von Atari ... 135,-
z.B. Gauntlet II, Outrun, Space Harrier, Starglider, Afterburner

OXYD-der Nachf.von Esprit. Diskette als PD frei kopierbar nur 5,-
Buch, ab 10. Level ein Muß . 50,-

Der Hit – Disketten zum Schleuderpreis: 10 Stck. Pack.
3,5" NO NAME MF2DD 8,50
5,25" NO NAME MD2D 5,00
3,5" Fuji MF2DD 20,-

UNSER TIP DES MONATS:
NINTENDO Gameboy
incl. „Tetris" 169,-
NINTENDO Telespiel
incl. "Super Mario Bros" .. 199,-
ATARI LYNX* incl.
Com-Lynx-Kabel 299,-
für alle Geräte:
Spiele + Preise auf Anfrage

PUBLIC DOMAIN:
Aus unserer umfangreichen PD-Sammlung für den ATARI ST kostet **jede Diskette ..... nur 5,-**

*Gerät ohne ZZF-Nummer

CAD/Graphik:
Arabesque 275,-
Convector 245,-
Omikron DRAW 125,-
STAD 175,-
That's Pixel 145,-

DTP/Textverarbeitung/Editoren
CAD 3D Cyber Studio 175,-
Calamus Outline Art 395,-
Edison 165,-
Lektorat 148,-
PKS-Write 195,-
Script 195,-
Script II 295,-
Signum! 2.0 440,-
uns.Zusatzprogr.f.Signum!/STAD
Headline, Großschriften 95,-
Fontmaker 95,-
Convert 95,-
SDO merge 50,-
SDO index 50,-
SDO preview 50,-
SDO graph 50,-
Script-Trenn 50,-
MetaxMap 50,-
Tempus 2.10 125,-
That's Write 345,-
1st Word plus 3.15 245,-
1st Proportional 115,-

Datenbanken/Tabellenkalkul.:
Easy Base 245,-
LDW-Power Calc 2.0 345,-
Steuer Tax '90 95,-
That's Address 185,-
Themadat 245,-

Programmiersprachen:
Omikron Compiler 175,-
ST Pascal plus 245,-
Turbo C 2.0 ..... ab 245,-

Utilities:
Anti Virus Kit 95,-
BTX-Manager 3.02 385,-
Copy Star 3.0 165,-
Interlink 75,-
Mortimer 75,-
Neodesk 3.0 95,-
Revolver 75,-
Wordflair 245,-

MS-DOS auf dem Atari ST:
PC-Speed (Einb. Aufpreis) . 395,-
AT-Speed (Einb. Aufpreis) . 498,-

Zubehör:
Staubschutzhaub.Kunstleder für:
ATARI SM 124 30,-
ATARI 1040 o. Mega Tast.je 20,-
ATARI 260/520 ST 15,-
Mega ST Set Monitor + Tast. 50,-
andere Monitore + Drucker .. a.A.
Media Box 3,5" f. 150 Disks . 40,-

Alternative Mäuse:
Logitech-Maus Pilot 95,-
Atari-Maus 78,-
Marconi Trackball 195,-

Hardware:
Monitorumschalter 60,-
NEC P 60 Drucker 1395,-
NEC P 20 Drucker 850,-
Profile 20 Festplatte 599,-
Profile 30 Festplatte 898,-
Profile 40 Festplatte 998,-
Profile R44 Wechselplatte 1798,-
Atari STE, 1 MB 795,-
Atari STE, 2 MB 1195,-
Atari STE, 4 MB 1495,-
Atari MEGA ST 1 995,-
Atari SC 1224 Colormonitor 585,-
Atari SM 124 S/W-Monitor .. 350,-

Kostenlose Kataloge für PD, Bücher, Hardware und Software bitte getrennt unter Angabe Ihres Computertyps anfordern. Lieferung per NN zzgl. 7,- DM Versandkosten. Bei Vorauskasse zzgl. 3,- DM, ab 100,- DM Bestellwert versandkostenfrei. Auslandsversand grundsätzlich zzgl. 15,- DM Versandkosten. Auf Wunsch auch UPS-Versand.

COMPUTER-VERSAND
Schlichting
...der etwas andere Versand

Rund um die Uhr: ☎ 030 / 786 10 96
Playsoft-Studio-Schlichting
Computer-Software-Versand GmbH
Postanschrift / Ladengeschäft: Katzbachstraße 8
D-1000 Berlin 61
Fax: 030/786 19 04 · Händleranfragen erwünscht

ATARI
1040 STFM + SM 124 998,-
1040 STE + SM 124 ..... 1.398,-
Mega ST 1 + SM 124 ..... 1.298,-
Mega ST 2 + SM 124 ..... 1.998,-
Mega ST 4 + SM 124 ..... 2.398,-
Megafile 30 798,-
Megafile 60 ..... 1.249,-
PC-Speed 349,-
AT-Speed 498,-
Vortex AT-Emulator 449,-
Super Charger 498,-

VORTEX-FESTPLATTEN
Neu / Datajet 30 ..... 1.098,-
Neu / Datajet 60 ..... 1.598,-

EPSON
LX-400 429,-
EZB LX-800 LQ 400/500 198,-
LQ 400, 24 Nadel A4 629,-
LQ-550, dito 749,-
LQ-850 + dito 1.298,-

NEC
NEC P 7+ 24-Nadel A3 ..... 1.379,-
NEC P60 24-Nadel A4 ..... 1.398,-
NEC P70 24-Nadel A3 ..... 1.698,-
Farb Option P6+/P7+/P60/P70 279,-

STAR
LC 24-10 649,-
EZB LC 10 / 24 229,-

SOFTWARE ATARI ST
GFA-BASIC 3.0 EWS ST 179,-
GFA-BASIC 2.0 EWS ST 44,-
GFA-Assembler ST 135,-
GFA-Draft plus ST 309,-
Turbo C 2.0 ST 198,-
Turbo C 2.0 Pro ST (Paket) 389,-
Debugger / Assembler 229,-
Signum!2wei 369,-
Stad 159,-
Megamax C 349,-
Modula 2 349,-
Superbase Professional 359,-
Superbase 2 198,-
LDW Power-Calc 219,-
Publishing Partner d 219,-
Cyber Paint 2.0 109,-
Devpac Assembler 2.0 128,-
CADproject Professional d 539,-

SCHUTZHAUBEN
aus hochwertigem Kunstleder ANTHRAZIT
ATARI 1040 / 260 / 520 24,95
Floppy SF 314 / 354 22,95
Monitor 124 / 125 27,95
Mega ST Tastatur 24,95
Mega ST-Set 49,95
EPSON FX86/800/LX86/LQ500/850 24,95
EPSON FX1000/1050/LQ1050/2550 27,95
NEC P6/7 P6+/P7+ P60/P70 27,95
STAR NL10/LC10/10c/24-10 24,95
PANASONIC 1080/91/1092/1592 24,95

Achtung - Preisänderungen vorbehalten!!
Versand nur per Nachnahme, zzgl. Versandkosten
Abholung nur nach tel. Voranmeldung möglich

TORNADO Computer Vertrieb
Wangenerstraße 99, 7980 Ravensburg
Tel. 0751 / 3951 · Fax 0751 / 3953

Bossart – Soft presents

SIDUS – ASTRUM

Das professionelle Sternwarten-Programm auf der ATARI ST & TT Linie. Die Sternkarten können auf Drucker oder Plotter bis DIN A0 (8-farbig) auf Papier gebracht werden.
SIDUS ASTRUM das Hilfswerkzeug für jeden Astronomen.

Sidus D Demo mit 3 Handbücher (wird beim Kauf der Vollversion angerechnet) sFr. 150.--
Sidus ST Vollversion inkl. Ephemeriden, Planetarium, unzählige Karten usw. sFr. 1200.--

VTX – Kombi ST & VTX – Adress

VTX-Combi ST ist das Programm für VTX/BTX Kommunikation, mit VTX-Adress können Sie die Adressdaten für ein beliebiges Programm konvertieren. (Frei Programmierbar) Der Preis für das Ideale VTX/BTX Programm-Paar sFr. 208.--

Wir führen auch weiter alle grossen PD Serien und ST-Computer Sonderdisketten. Verlangen Sie unsere Liste. Unser Angebot wird Sie überzeugen.

BOSSART – SOFT
P.B. 5146
CH-6020-Emmenbrücke-3
Tel. 041 / 45'82'84

ARTI COLOR ©
Das Multiplus - Creativ-System für **DTP**

* **Farbtransfer**
Laserdrucke + Fotokopien in über 100 Farben einfärben!

* **Laminieren**
Schützen u. Veredeln Ihrer Dokumente bis DIN A 3.

* **Kaschieren**
Aufziehen auf Karton Lackieren mit Lackfolie - Toner einglänzen -Displayherstellung

* **Binden**
Heißkleben von Loseblattwerken zu Broschüren in Mappen.

* **Abziehbilder**
Fürs Auto, Betrieb, Büro selbstklebend, magnetisch

* **ArtiCards**
Für Urkunden, Visitenkarten- selbstklebende Folien u. Papiere.

Handy-Scooter CT 40



Multiplus-Creativ-System Gerät



ARTICOLOR
Manfred Elzner
Großmattenstraße 12
D – 7813 Staufen
Tel. 07633 / 82413
Fax 07633 / 50701

DESKTOP Color
FARBTRANSFERFOLIEN
und Ihr Schwarz-Weiß
LASERDRUCKER oder
Fotokopierer bringt ***Farbe*** auf.
In Gold / Silber / Kupfer oder Rot, Blau, Grün. ..... (z.Zt. über **50** Farben)

Einsteiger Set:
10 x 10 versch. Farbfolienstreifen =
100 Streifen (je 21cm L. / 4cm B.)
Bedienungsanleitung mit Tips u. Tricks.
1Bg./120 St.Fixierklebepunkte.
DM 59,– *inklusive Porto/ Nachnahme und Verpackungskosten.*

Manfred Elzner · Druck+Sachen
Großmattenstraße 12
D– 7813 Staufen
Tel. 07633 / 82413 · Fax 50701

```
91:       RETURN d^.OffsetTable [c] + 4L;
92:     END Get_Word_Pos;
93:
94:   PROCEDURE Get_Max (d: Dic; c: CHAR): LONGCARD;
95:     VAR max: LONGCARD;
96:     BEGIN
97:       max := Get_Offset_Pos (d, inc (c));  DEC
              (max);  RETURN max;
98:     END Get_Max;
99:
100:  PROCEDURE SaveDic (fn: FileStr; d: Dic);
101:    VAR offset : LONGCARD;
102:        c      : CHAR;
103:    BEGIN
104:      FOR c := startChar TO MAX (CHAR) DO
105:        IF IsLower (c) THEN
106:          offset := Get_Offset_Pos (d, inc (c));
107:          DEC (offset, Get_Offset_Pos (d, c));
108:          CardToMem (SHORT (offset),
                         Get_Offset_Adr (d, c));
109:        END;
110:      END;
111:      MemToFile (fn, d^.dic, d^.size);
112:    END SaveDic;
113:
114:  PROCEDURE ReleaseDic (VAR d: Dic);
115:    BEGIN
116:      DEALLOCATE (d^.dic, 0); DEALLOCATE (d, 0); d
                                  := NIL;
117:    END ReleaseDic;
118:
119:  VAR G_pos, G_cnt: LONGCARD;
120:      G_lcnt      : CARDINAL;
121:
122:  PROCEDURE QueryDic (d: Dic; str: ARRAY OF CHAR;
123:                      elemProc: queryProc);
124:
125:    PROCEDURE queryAux (c: CHAR): BOOLEAN;
126:
127:      VAR i, pos  : LONGCARD;
128:          max     : LONGCARD;
129:          j       : CARDINAL;
130:          cPtr    : CPtr;
131:          zstr    : String;
132:          zustand : Zustand;
133:      BEGIN
134:        G_lcnt := 0;
135:        pos := Get_Word_Pos (d, c);  cPtr :=
                 Get_Index_Adr (d, pos);
136:        zustand := char;  zstr[0] := c; j := ORD
                   (cPtr^); INC (cPtr);
137:        max := Get_Max (d, c);
138:        IF pos > max  THEN RETURN TRUE; END;
139:        FOR i := pos + 1L TO max DO
140:          IF zustand = zahl THEN
141:            zstr[j] := 0C; j := ORD (cPtr^); INC
                       (cPtr);
142:            zustand := char;
143:            G_pos := i; INC (G_cnt);
144:            IF ~elemProc(d,zstr) THEN RETURN
                                        FALSE; END;
145:            INC (G_lcnt);
146:          ELSE
147:            zstr[j] := cPtr^; INC (cPtr); INC (j);
148:          END;
149:          IF cPtr^ < startChar THEN zustand :=
                                        zahl; END;
150:        END;
151:        G_pos := Get_Max (d, c); zstr [j] := 0C;
152:        RETURN elemProc (d, zstr);
153:      END queryAux;
154:
155:    VAR c: CHAR;
156:    BEGIN
157:      IF str[0] = 0C THEN
158:        FOR c := startChar TO MAX (CHAR) DO
159:          IF IsLower (c) AND ~queryAux (c) THEN
                                        RETURN; END;
160:        END;
161:      ELSE
162:        voidO := queryAux (str[0]);
163:      END;
164:    END QueryDic;
165:
166:  PROCEDURE count (d: Dic; VAR voidS: ARRAY OF
                     CHAR): BOOLEAN;
167:    BEGIN
168:      RETURN TRUE;
169:    END count;
170:
171:  PROCEDURE CountWords (d: Dic): LONGCARD;
172:    BEGIN
173:      G_cnt := 0;  QueryDic (d, '', count);
                      RETURN G_cnt;
174:    END CountWords;
175:
176:  PROCEDURE insert (d: Dic; VAR zstr: ARRAY OF
                      CHAR): BOOLEAN;
177:    BEGIN
178:      IF (G_lcnt MOD d^.every = 0) THEN  Insert
            (d^.tree, zstr, G_pos);  END;
179:      RETURN TRUE;
180:    END insert;
181:
182:  PROCEDURE SetSpellMode (d: Dic; every: CARDINAL);
183:    BEGIN
184:      d^.every := every;
185:      voidO := CreateFastDic (d^.tree, every,
                   CountWords (d));
186:      QueryDic (d, '', insert);
187:      CloseFastDic (d^.tree);
188:    END SetSpellMode;
189:
190:  PROCEDURE CreateDic (VAR d: Dic; over, step:
                LONGCARD);
191:    VAR j : CARDINAL;
192:        c : CHAR;
193:    BEGIN
194:      allocate (d, SIZE (d^));
195:      allocate (d^.dic, MinSize + over);
196:      j := 0;
197:      FOR c := 1C TO MAX (CHAR) DO
198:        d^.OffsetTable [c] := j;
199:        IF IsLower (c) THEN
200:          d^.dic^[j]   := 0C;
201:          d^.dic^[j+1] := 4C;
202:          d^.dic^[j+2] := 0C;
203:          d^.dic^[j+3] := c;
204:          INC (j, 4);
205:        END;
206:      END;
207:      d^.step  := step;
208:      d^.max   := MinSize + over;
209:      d^.size  := MinSize;
210:      d^.every := 0;
211:    END CreateDic;
212:
213:  PROCEDURE LoadDic (fn: FileStr; VAR d: Dic;
214:                     over, step: LONGCARD);
215:
216:    VAR c      : CHAR;
217:        offset : LONGCARD;
218:        scs    : BOOLEAN;
219:    BEGIN
220:      allocate (d, SIZE (d^));    (* falsches Format
                                         bzw. fehler-*)
221:      d^.size := FILESIZE (fn);   (* hafte Daten wer
                                         den nicht    *)
222:      allocate (d^.dic, d^.size+over); (*erkannt!*)
223:      FileToMem (fn, d^.dic, d^.size);
224:      d^.max := d^.size + over;
225:      d^.step := step;
226:      ClearVar (d^.OffsetTable);
227:      offset := 0;
228:      FOR c := 1C TO MAX (CHAR) DO
229:        IF IsLower (c) THEN
230:          d^.OffsetTable [c] := offset;
231:          INC (offset, MemToCard (DicAdr (d,
                 offset)));
232:        ELSE
233:          d^.OffsetTable [c] := offset;
234:        END;
235:      END;
236:    END LoadDic;
237:
238:  PROCEDURE UpdateOffsetTable (d: Dic; c: CHAR;
                                   shift: INTEGER);
239:    BEGIN
240:      FOR c := inc (c) TO MAX (CHAR) DO
241:        INC (d^.OffsetTable [c], shift);
242:      END;
243:      INC (d^.size, shift);
244:    END UpdateOffsetTable;
245:
```

→

```
246:  PROCEDURE enoughMem (d: Dic; c: CHAR; shift:
              CARDINAL;
247:                       pos: LONGCARD; VAR scs:
                                BOOLEAN);
248:
249:    (* Falls ein fehlerfreies STORAGE.Enlarge
          vorliegt läPt sich
250:     * diese Prozedur leicht flexibler gestalten!
          Das aktuelle Enlarge
251:     * hat aber leider so seine Probleme.
252:     *)
253:    BEGIN
254:      Check (d^.max - d^.size <= LONG (shift), 0,
                enoughMemStr);
255:      UpdateOffsetTable (d, c, shift);
256:      Copy (DicAdr (d, pos), d^.size-pos, DicAdr
              (d, pos + LONG (shift)));
257:    END enoughMem;
258:
259:  PROCEDURE GetDeltaPos (REF a, b: ARRAY OF CHAR):
              CARDINAL;
260:    VAR i: CARDINAL;
261:    BEGIN
262:      FOR i := 0 TO MAX (CARDINAL) DO
263:        IF (a[i] # b[i]) THEN  RETURN i;  END;
264:      END;
265:    END GetDeltaPos;
266:
267:  PROCEDURE GetRestLen (REF s: ARRAY OF CHAR; pos:
              CARDINAL): CARDINAL;
268:    VAR i: CARDINAL;
269:    BEGIN
270:      i := 0;
271:      IF s[0] # 0C THEN
272:        REPEAT
273:          INC (i);  INC (pos);
274:        UNTIL (pos > HIGH (s)) OR (s[pos] = 0C);
275:      END;
276:      RETURN i;
277:    END GetRestLen;
278:
279:  PROCEDURE Compact (REF a: ARRAY OF CHAR; al,ar :
                        CARDINAL;
280:                     REF b: ARRAY OF CHAR; bl    :
                              CARDINAL;
281:                     REF c: ARRAY OF CHAR; cl,cr :
                              CARDINAL;
282:                     VAR x: ARRAY OF CHAR; VAR xl,
                              shift: CARDINAL);
283:
284:    (* kompaktiert 'b' und 'c' zu 'x' *)
285:
286:    VAR delta, j, i: CARDINAL;
287:    BEGIN
288:      delta := GetDeltaPos (a,b); (* Es gilt:
                 delta >= 1, bl >= 1 *)
289:      x[0] := CHR (delta); j := 1;
290:      FOR i := delta TO bl-1 DO
291:        x[j] := b[i]; INC (j);
292:      END;
293:      delta := GetDeltaPos (b,c);
294:      IF delta # 0 THEN      (* am ende einfügen *)
295:        x[j] := CHR (delta);
296:        xl  := j + 1;
297:        shift := j + cl - delta - cr;
298:      ELSE
299:        xl := j;  shift := j;
300:      END;
301:    END Compact;
302:
303:  PROCEDURE InitMoveRec (d: Dic; REF str: ARRAY OF
                          CHAR;
304:                       VAR MoveRec: moveRec);
305:    BEGIN
306:      WITH MoveRec DO
307:        ReadPos := Get_Word_Pos (d, str [0]);
308:        MaxPos  := Get_Offset_Pos (d, inc (str
                     [0]));
309:        vorIx   := 0; aktIx  := 1; nachIx := 2;
310:        StrTab [vorIx].Str := ' ';
311:        StrTab [vorIx].Len := 1;
312:        StrTab [vorIx].RLen := 1;
313:        StrTab [vorIx].StartPos := Get_0CHAR_Pos
                                       (d, str [0]);
314:        StrTab [aktIx]  := StrTab [vorIx];
315:        StrTab [nachIx] := StrTab [vorIx];
316:      END;
```

```
317:    END InitMoveRec;
318:
319:  PROCEDURE Move (d: Dic; VAR MoveRec: moveRec);
320:    VAR i: CARDINAL;
321:    BEGIN
322:      WITH MoveRec DO
323:        vorIx  := (vorIx  + 1) MOD 3;
324:        aktIx  := (aktIx  + 1) MOD 3;
325:        nachIx := (nachIx + 1) MOD 3;
326:        IF ReadPos >= MaxPos THEN
327:          StrTab [nachIx].Str := 377C;
328:          StrTab [nachIx].StartPos := MaxPos;
329:          StrTab [nachIx].Len  := 0;
330:          StrTab [nachIx].RLen := 0;
331:        ELSE
332:          StrTab [nachIx].Str      := StrTab
                                          [aktIx].Str;
333:          StrTab [nachIx].StartPos := ReadPos;
334:          StrTab [nachIx].RLen     := 0;
335:          i := ORD (d^.dic^ [ReadPos]);
336:          StrTab [nachIx].Len := i;
337:          INC (ReadPos);
338:          REPEAT
339:            StrTab [nachIx].Str [i] := d^.dic^
                                          [ReadPos];
340:            INC (StrTab [nachIx].RLen);
341:            INC (ReadPos);
342:            INC (i);
343:          UNTIL (d^.dic^ [ReadPos] < startChar) OR
                                  (ReadPos > MaxPos);
344:          StrTab [nachIx].Len     := i;
345:          StrTab [nachIx].Str [i] := 0C;
346:        END;
347:      END;
348:    END Move;
349:
350:  PROCEDURE InsertDic (d: Dic; REF str: ARRAY OF
                         CHAR; VAR scs: BOOLEAN);
351:
352:    VAR MoveRec     : moveRec;
353:        cStr, zstr  : String;
354:        cLen, shift : CARDINAL;
355:        rel         : Relation;
356:    BEGIN
357:      scs := TRUE;
358:      IF (str[0] = 0C) THEN RETURN; END;
359:      Assign (str, zstr, scs);
360:      InitMoveRec (d, zstr, MoveRec);
361:      WITH MoveRec DO
362:        Move (d, MoveRec);
363:        LOOP
364:          Move (d, MoveRec);
365:          CASE cmp (ADR (zstr), ADR (StrTab
                                  [aktIx].Str)) OF
366:          | equal : RETURN;
367:          | less  : EXIT;
368:          ELSE END;
369:        END;
370:        Compact (StrTab [vorIx].Str, StrTab
                    [vorIx].Len, StrTab [vorIx].RLen,
371:                 str, Length (str),
372:                 StrTab [aktIx].Str, StrTab
                            [aktIx].Len, StrTab
                            [aktIx].RLen,
373:                 cStr, cLen, shift);
374:        enoughMem (d, str[0], shift, StrTab
                      [aktIx].StartPos, scs);
375:        IF ~scs THEN HALT; END;
376:        ByteCopy (ADR (cStr), cLen, Get_Index_Adr
                     (d, StrTab [aktIx].StartPos));
377:      END;
378:    END InsertDic;
379:
380:  PROCEDURE FindDic (d: Dic; REF str: ARRAY OF
                       CHAR): BOOLEAN;
381:
382:    VAR i, pos, max: LONGCARD;
383:        j          : CARDINAL;
384:        zstr       : String;
385:        cPtr       : CPtr;
386:        rel        : Relation;
387:    BEGIN
388:      IF d^.every = 0 THEN  RETURN FALSE;  END;
389:      Assign (str, zstr, void0);  Lower (zstr);
390:      GetPos (d^.tree, zstr, pos); IF pos = MAX
                (LONGCARD) THEN  RETURN TRUE;  END;
```

→

```
391:      max := Get_Max (d, zstr[0]);
392:      j := Length (zstr);
393:      cPtr := Get_Index_Adr (d, pos);
394:      IF pos > max  THEN RETURN TRUE; END;
395:      FOR i := pos TO max DO
396:        INC (j);
397:        IF cPtr^ < startChar THEN
398:          zstr [j-1] := 0C;
399:          rel := cmp (ADR (zstr), ADR (str));
400:          IF    rel = equal   THEN  RETURN  TRUE;
401:          ELSIF rel = greater THEN  RETURN  FALSE;
                                       END;
402:          j := ORD (cPtr^);
403:        END;
404:        INC (cPtr);
405:        zstr [j] := cPtr^;
406:      END;
407:      zstr [j+1] := 0C;
408:      RETURN cmp (ADR (zstr), ADR (str)) = equal;
409:    END FindDic;
410:
411:  PROCEDURE DeleteDic (d: Dic; REF str: ARRAY OF
                          CHAR);
412:
413:    VAR MoveRec         : moveRec;
414:        writePos        : LONGCARD;
415:        shift           : INTEGER;
416:        from, to        : ADDRESS;
417:        delta, i, cLen  : CARDINAL;
418:        zstr            : String;
419:        c               : CHAR;
420:
421:    BEGIN
422:      Assign (str, zstr, voidO); c := zstr[0];
423:      InitMoveRec (d, c, MoveRec);
424:
425:      (* Vorbed: akt <= nach < str *)
426:      (* Return: nach = str        *)
427:      (* Insert: akt < str < nach  *)
428:
429:      WITH MoveRec DO
430:        Move (d, MoveRec);
431:        (*$W-*)
432:        REPEAT
433:          Move (d, MoveRec);
```

```
434:          IF cmp (ADR (zstr), ADR (StrTab
                    [aktIx].Str)) = equal THEN
435:            IF StrTab [nachIx].Len = 0 THEN
436:              shift := - (StrTab [aktIx].RLen)+1;
437:            ELSE
438:              writePos := StrTab [aktIx].StartPos;
439:              delta    := GetDeltaPos (StrTab
                                [vorIx].Str, StrTab
                                [nachIx].Str);
440:              d^.dic^ [writePos] := CHR (delta);
441:              cLen := 0;
442:              FOR i := delta TO StrTab
                            [nachIx].Len-1 DO
443:                INC (writePos);
444:                INC (cLen);
445:                d^.dic^ [writePos] := StrTab
                              [nachIx].Str[i];
446:              END;
447:
448:  (* shift = neue Länge - alte Länge *)
449:  (*        = (1+aRLen + 1+cRlen/neu) - (1+aRLen+ 1+
                bRlen +1+cRlen);*)
450:  (*        = (cRlen/neu) - (bRlen +1+cRlen);*)
451:
452:              shift := cLen - (StrTab [aktIx].RLen
                             + 1 +
453:                             StrTab [nachIx].RLen);
454:            END;
455:            from := Get_Index_Adr (d, ReadPos); to
                       := from;
456:            INC (to, shift);
457:            Copy (from, d^.size - ReadPos, to);
458:            UpdateOffsetTable (d, c, shift);
459:            RETURN;
460:          END;
461:        UNTIL cmp (ADR (zstr), ADR (StrTab
                          [aktIx].Str)) = less;
462:        RETURN;
463:      END;
464:    (*$W=*)
465:    END DeleteDic;
466:  END Dics.
```

Listing 2: DICS.I: Der zweite Teil des Hauptmoduls

```
  1:  DEFINITION MODULE DicBase;
  2:
  3:  FROM SYSTEM     IMPORT  ADDRESS;
  4:  FROM Strings    IMPORT  String, Relation;
  5:
  6:  TYPE CPtr = POINTER TO CHAR;
  7:       SPtr = POINTER TO String;
  8:
  9:  CONST CardSize = 2;
 10:        AdrSize  = 4;
 11:
 12:  VAR voidS: String;
 13:      voidC: CARDINAL;
 14:      voidO: BOOLEAN;
 15:
 16:  PROCEDURE FileToMem (VAR fn: ARRAY OF CHAR;
 17:                       VAR adr: ADDRESS;VAR n:
                                  LONGCARD);
 18:
 19:  (*  Liest 'fn' nach 'adr'. Bei adr=NIL wird
          alloziert *)
 20:
 21:  PROCEDURE MemToFile (VAR fn: ARRAY OF CHAR; adr:
                          ADDRESS; n: LONGCARD);
 22:
 23:    (*  Schreibt Daten ab 'adr' ('n' = Größe) *)
 24:
 25:  PROCEDURE allocate (VAR adr: ADDRESS; size:
                          LONGCARD);
 26:
 27:    (*  ALLOCATE mit eingebauter Fehlerabfrage *)
 28:
 29:  PROCEDURE Check (err: BOOLEAN; res: INTEGER;
```

```
                       txt: ARRAY OF CHAR);
 30:
 31:    (*  Entspricht IF err THEN WriteString
          (GetErrMsg + txt),TermProcess *)
 32:
 33:  PROCEDURE inc (c: CHAR): CHAR;
 34:
 35:    (* Liefert 'c'+1 zurück *)
 36:
 37:  PROCEDURE CardToMem (card: CARDINAL; adr:
                           ADDRESS);
 38:  PROCEDURE MemToCard (adr: ADDRESS): CARDINAL;
 39:
 40:   (* kopiert 'card' nach 'adr', bzw. umgekehrt. *)
 41:
 42:  PROCEDURE cmp (a, b: CPtr): Relation;
 43:
 44:    (* schneller Vergleich von Nullterminierten
          Strings *)
 45:
 46:  PROCEDURE ByteCopy (from: ADDRESS; n: CARDINAL;
                         to: ADDRESS);
 47:
 48:    (* schnelle Copyroutiene - nur für kleine 'n'
          verwenden.*)
 49:
 50:  PROCEDURE FILESIZE (VAR fn: ARRAY OF CHAR):
                         LONGCARD;
 51:
 52:  END DicBase.
```

Listing 3: DICBASE.D: Die Grundfunktionen...

NEUE VERSION

Steuern sparen mit

STeuer Tax 90

Das unentbehrliche Programm zur richtigen Berechnung der Lohn- und Einkommmenssteuer (mit den neuen 90er Vorschriften) für alle Steuerzahler mit einem ATARI ST-Computer

Version 2.90

Läuft voll unter GEM, mausgesteuert, einfache Benutzung, Eingabe an Steuerformulare angepaßt, Auswertung auf Monitor oder Drucker, mit vielen Hilfen, so daß auch der Laie mit seinem ATARI schnell und mühelos seine Steuer berechnen kann. Alle Eingaben und Auswertungen können abgespeichert und später wieder aufgerufen werden. Die Version 2.90 ist geeignet für den "normalen Anwender", der für sich seine Steuer berechnen will.
S/W- oder Farbmonitor. Update-Service für die Folgejahre.

DM 98,–

alle Preise sind unverbindlich empfohlene Verkaufspreise

Version 3.90

Mandantenfähig. Alle Merkmale wie Version 2.90 jedoch zusätzlich mit einer Datenbank, dadurch mandantenfähig.Pro doppelseitiger Diskette können ca. 250 Mandanten abgespeichert werden, auf 20 MB Harddisk ca. 6.600 ! Die Version 3.90 eignet sich besonders -aber nicht nur- für Steuerberater, Lohnsteuervereine, Buchführungshelfer, Versicherungsvertreter, usw. Darüber hinaus auch für Selbstständige sehr interessant, die mehrmals im Jahr bzw. ständig einen Überblick über ihre Steuerbelastung haben wollen.

DM 159,–

Bitte benutzen Sie die eingeheftete Bestellkarte

Update Service

Steuer Tax-Besitzer erhalten die neue Version 2.90 oder 3.90 gegen Rücksendung ihrer registrierten Originaldiskette zum Preis von 35,-- DM zuzüglich 6,-- DM Versandkosten. (Die Lieferung erfolgt nur gegen Übersendung eines Schecks in Höhe von 41,– DM).

DM 35,–

Für ihre Bestellungen benutzen Sie bitte die im Heft eingefügte Bestellkarte oder den Bestellcoupon auf dieser Seite.



PC/AT Speed ... Gewußt wie

Aus dem Inhalt:

- Das Betriebssystem MS-DOS
- Einbau von PC/AT-Speed
- Installation der System-Software
- Apassung des DOS-Betriebssystems (DOS 3.3 und 4.0)
- Arbeiten mit MS-DOS
- Tips und Tricks beim Umgang mit Emulatoren
- Ausblick auf künftige Weiterentwicklungen

PC/AT-Speed
Gewußt wie
- Perfekte Installation
- MS-DOS auf dem ATARI ST
- Tips und Tricks

Hardcover
ISBN 3-923250-95-9
Bestell-Nr. B-450

DM 49,–

Heim-Verlag

Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Tel.: 0 61 51 / 5 60 57-58
Fax: 0 61 51 / 5 60 59

BESTELLCOUPON

Bitte senden Sie mir:
____ STeuer Tax 2.90 a 98.– DM
____ STeuer Tax 3.90 a 159.– DM
____ STeuer Tax Update a 35.– DM
____ PC/AT-Speed Buch a 49.– DM

Name, Vorname _____

Straße, Hausnr. _____

PLZ, Ort _____

zuzüglich Versandkosten DM 6,-- Ausland DM 10,–, unabhängig von der bestellten Stückzahl

in der Schweiz:

Data Trade AG
Landstr.1

CH-5415 Rieden-Baden

in Österreich:

RRR EDV GmbH
Dr. Stumpf Str.118

A-6020 Innsbruck

```
 1:  IMPLEMENTATION MODULE DicBase; (* SYSTEM:
                                Megamax Modula-2, 2.2 *)
 2:
 3:  FROM EasyGEM0    IMPORT  FormAlert;
 4:  FROM PrgCtrl     IMPORT  TermProcess;
 5:  FROM SYSTEM      IMPORT  TSIZE, ADDRESS, ADR;
 6:  FROM Storage     IMPORT  ALLOCATE, DEALLOCATE;
 7:  FROM Binary      IMPORT  WriteBytes, ReadBytes,
                              FileSize;
 8:  FROM Directory   IMPORT  SplitPath, PathExists,
                              CreateDir;
 9:  FROM Strings     IMPORT  Length, Compare,
                              Relation, String, Concat, Assign;
10:  FROM Files       IMPORT  File, Access, Open,
                              Close, Create, State,
11:                           GetStateMsg, ReplaceMode;
12:  FROM MOSGlobals IMPORT   PathStr, FileStr,
                              fReadFault, GeneralErr,
13:                           OutOfMemory;
14:
15:  CONST lmax = 200;
16:        endBut  = '][Abbruch]';
17:        sign    = '[1][';
18:        fail          = ' failed';
19:        concStr       = 'Conc failed';
20:        fileToMemStr  = 'FileToMem failed';
21:        openStr       = 'Open failed';
22:        closeStr      = 'Close failed';
23:        createStr     = 'Create ';
24:        makePathStr   = 'MakePath ';
25:
26:  TYPE  LString = ARRAY [0..lmax] OF CHAR;
27:
28:
29:  PROCEDURE allocate (VAR adr: ADDRESS; n:
                         LONGCARD);
30:    BEGIN
31:      ALLOCATE (adr, n); Check (adr = NIL,
                                   OutOfMemory, '');
32:    END allocate;
33:
34:  PROCEDURE NthPos (c: CHAR; VAR s: ARRAY OF CHAR;
                       n: CARDINAL): CARDINAL;
35:
36:  (*  Sucht nach n-maligem auftreten von 'c' in
        dem String 's'.
37:   *  Bei Erfolg wird die n.te Position
        zurückgegeben, ansonsten 0.
38:   *)
39:
40:    VAR nth, i : CARDINAL;
41:    BEGIN
42:      nth := 0;
43:      FOR i := 0 TO Length (s)-1 DO
44:        IF s[i] = c THEN INC (nth); END;
45:        IF n = nth THEN RETURN i; END;
46:      END;
47:      RETURN 0;
48:    END NthPos;
49:
50:  PROCEDURE Conc3 (s1, s2, s3: ARRAY OF CHAR):
                     LString;
51:    VAR s   : LString;
52:        scs : BOOLEAN;
53:    BEGIN
54:      Concat (s1, s2, s, scs); Check (~scs, 0,
                 concStr);
55:      Concat (s, s3, s, scs);  Check (~scs, 0,
                 concStr);
56:      RETURN s;
57:    END Conc3;
58:
59:  PROCEDURE Conc5 (s1, s2, s3, s4, s5: ARRAY OF
                     CHAR): LString;
60:    BEGIN
61:      RETURN Conc3 (Conc3 (s1,s2,s3),s4,s5);
62:    END Conc5;
63:
64:  PROCEDURE Check (err: BOOLEAN; errNo: INTEGER;
                     txt: ARRAY OF CHAR);
65:    VAR errMsg: String;
66:    BEGIN
67:      IF err THEN
68:        errMsg := '';
69:        IF errNo < 0 THEN GetStateMsg (errNo,
                           errMsg); END;
70:        FormAlert (1, Conc5 (sign, txt, '|',
                      errMsg, endBut), voidC);
71:        TermProcess (errNo);
72:      END;
73:    END Check;
74:
75:  PROCEDURE inc (c: CHAR): CHAR;
76:    BEGIN
77:      INC (c); RETURN c;
78:    END inc;
79:
80:  PROCEDURE CardToMem (x: CARDINAL; adr: ADDRESS);
81:    BEGIN
82:      ByteCopy (ADR (x), CardSize, adr);
83:    END CardToMem;
84:
85:  PROCEDURE MemToCard (adr: ADDRESS): CARDINAL;
86:    VAR x: CARDINAL;
87:    BEGIN
88:      ByteCopy (adr, AdrSize, ADR (x)); RETURN x;
89:    END MemToCard;
90:
91:  PROCEDURE cmp (a, b: CPtr): Relation;
92:    BEGIN
93:      LOOP
94:        IF    a^ < b^ THEN  RETURN  less;
                             (* nur C-Strings *)
95:        ELSIF a^ > b^ THEN  RETURN  greater;
96:        ELSIF a^ = 0C THEN  RETURN  equal;  END;
97:        INC (a); INC (b);
98:      END;
99:    END cmp;
100:
101: PROCEDURE MakePath (VAR path: ARRAY OF CHAR);
102:
103: (*  Diese Prozedur erzeugt, falls noch nicht
       vorhanden 'path' *)
104:
105:   VAR part   : String;
106:       lpath  : String;
107:       res    : INTEGER;
108:       i, pos : CARDINAL;
109:
110:   BEGIN
111:     SplitPath (path, lpath, voidS);
112:     IF NOT PathExists (lpath) THEN
113:       i := 1;
114:       REPEAT
115:         INC (i);
116:         Assign (lpath, part, void0);
117:         pos := NthPos ('\', lpath, i);
118:         IF pos > 0 THEN
119:           part[pos+1] := 0C;
120:           IF ~PathExists (part) THEN
121:             part[pos] := 0C;
122:             CreateDir (part, res);
123:             Check (res # 0, res, Conc3
                        (makePathStr, lpath, fail));
124:           END;
125:         END;
126:       UNTIL Compare (part, lpath) = equal;
127:     END;
128:   END MakePath;
129:
130: PROCEDURE open (VAR f: File; VAR fn: ARRAY OF
                    CHAR; mode: Access);
131:   BEGIN
132:     Open (f, fn, mode);
133:     Check (State (f) # 0, State (f), Conc3
               ('Open ', fn, fail));
134:   END open;
135:
136: PROCEDURE create (VAR  f   : File;
137:                    VAR fn   : ARRAY OF CHAR;
138:                        m    : Access;
139:                        rpm  : ReplaceMode);
140:
141:   BEGIN
142:     MakePath (fn);
143:     Create (f, fn, m, rpm);
144:     Check (State (f) # 0, State (f), Conc3
               (createStr, fn, fail));
145:   END create;
146:
147: PROCEDURE close (VAR f: File);
148:   BEGIN
149:     Close (f); Check (State (f) # 0, State (f),
                          closeStr);
```

→

```
150:    END close;
151:
152:  PROCEDURE FileToMem (VAR fn: ARRAY OF CHAR;
153:                        VAR adr: ADDRESS; VAR n:
                                   LONGCARD);
154:
155:    VAR bytesRead : LONGCARD;
156:        f         : File;
157:
158:    BEGIN
159:      open (f, fn, readOnly);
160:      IF adr = NIL THEN
161:        n := FileSize (f);
162:        allocate (adr, n);
163:      ELSE
164:        Check (n < FileSize (f), OutOfMemory,
                   fileToMemStr);
165:      END;
166:      ReadBytes (f, adr, n, bytesRead);
167:      Check (n # bytesRead, fReadFault,
                 fileToMemStr);
168:      close (f);
169:    END FileToMem;
170:
171:  PROCEDURE MemToFile (VAR fn: ARRAY OF CHAR; adr:
              ADDRESS; n: LONGCARD);
172:
173:    VAR f   : File;
174:        res : INTEGER;
175:    BEGIN
176:      create (f, fn, readWrite, replaceOld);
177:      WriteBytes (f, adr, n);
178:      Check (res # 0, res, Conc3 ('MemToFile ', fn,
                 fail));
179:      close (f);
180:    END MemToFile;
181:
182:  PROCEDURE FILESIZE (VAR fn: ARRAY OF CHAR):
              LONGCARD;
183:
184:    VAR size : LONGCARD;
185:        f    : File;
186:    BEGIN
187:      open (f, fn, readOnly); size := FileSize
                (f); close (f);
188:      RETURN size;
189:    END FILESIZE;
190:
191:  (*$L-*)
192:  PROCEDURE ByteCopy (from: ADDRESS; n: CARDINAL;
              to: ADDRESS);
193:    BEGIN
194:      ASSEMBLER
195:            MOVE.L  -(A3),A1        ; A1 := to
196:            MOVE.W  -(A3),D0        ; D0 := n
197:            MOVE.L  -(A3),A0        ; A0 := from
198:            BRA     start
199:  copy      MOVE.B  (A0)+,(A1)+
200:  start     DBF     D0, copy
201:      END;
202:    END ByteCopy;
203:  (*$L=*)
204:
205:  END DicBase.
```

Listing 4: DICBASE.I: ...und I/O-Routinen

```
  1:  DEFINITION MODULE FastDics;
  2:
  3:  FROM DicBase      IMPORT  CPtr, SPtr;
  4:
  5:  TYPE FastDic = RECORD
  6:                   STab      : CPtr;
  7:                   akt_STab  : CPtr;
  8:                   max_STab  : CPtr;
  9:                   PTab      : POINTER TO ARRAY
                                   [0..MaxCard] OF SPtr;
 10:                   akt_PTab  : CARDINAL;
 11:                   max_PTab  : CARDINAL;
 12:                 END;
 13:
 14:  PROCEDURE CreateFastDic (VAR fd: FastDic; every:
               CARDINAL;
 15:                          words: LONGCARD):
                                                BOOLEAN;
 16:
 17:  (*  Legt eine FastDic an. Mit every und words
          wird der der
 18:   *  Speicherverbrauch berechnet.
 19:   *)
 20:
 21:  PROCEDURE CloseFastDic (VAR fd: FastDic);
 22:
 23:  (*  Der noch übriggebliebene Speicher wird
          zurückgegeben.
 24:   *  Der Aufruf ist optional und muß nach dem
          letzten Insert erfolgen.
 25:   *)
 26:
 27:  PROCEDURE ReleaseFastDic (VAR fd: FastDic);
 28:
 29:  PROCEDURE Insert (VAR fd: FastDic; VAR zStr:
                       ARRAY OF CHAR; pos: LONGCARD);
 30:
 31:  (*  Fügt [zstr,pos] in den Suchbaum ein. *)
 32:
 33:  PROCEDURE GetPos (fd: FastDic; VAR zStr: ARRAY
              OF CHAR; VAR pos: LONGCARD);
 34:
 35:    (*  Sucht nach zStr (bzw. nach nächstkleinerem
            Element) und liefert
 36:     *  das gefundene Element und dessen
            Nachfolgerposition zurück.
 37:     *)
 38:  END FastDics.
```

Listing 5: FASTDIC.D: Hier gibt's die schnelle...

```
  1:  IMPLEMENTATION MODULE FastDics; (* Megamax
                                        Modula-2, 2.2 *)
  2:
  3:    (* StrTab: ARRAY in dem Nullterminierte Strings
           und deren Positionen im Dic
  4:     *          gespeichert werden.
  5:     *
  6:     *              +-----------------
  7:     *              | [CHAR,...,CHAR] [0C] [LONGCARD]
  8:     *              +-----------------
  9:     *              | [CHAR,...,CHAR] [0C] [LONGCARD]
 10:     *              +-----------------
 11:     *  Bottom *--> | ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
 12:     *
 13:     *              | ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
 14:     *              +-----------------
 15:     *)
 16:
 17:  FROM SysUtil0    IMPORT  ClearVar;
 18:  FROM Storage     IMPORT  DEALLOCATE;
 19:  FROM SYSTEM      IMPORT  ADR, ADDRESS;
 20:  FROM Strings     IMPORT  Length, String, Relation;
 21:  FROM XInOut      IMPORT  WriteS, WriteSSL,
                               WriteLn, WriteSSSL;
 22:  FROM DicBase     IMPORT  AdrSize, voidO, Check,
                               allocate, SPtr,
 23:                           CPtr, ByteCopy;
 24:
 25:  CONST minLen = 354L;           insFailST =
                          'Insert failed STab overflow';
 26:        wLen   = 25L;           insFailPT =
                          'Insert failed PTab overflow';
 27:        lowers = 59;
 28:
 29:  PROCEDURE Insert (VAR fd: FastDic; VAR zstr:
                        ARRAY OF CHAR; pos: LONGCARD);
 30:    VAR l: CARDINAL;
 31:    BEGIN
 32:      WITH fd DO
 33:        l := Length (zstr) + 1;
 34:        Check (akt_STab + ADDRESS (LONG (l)) >
                   max_STab, 0, insFailST);
 35:        Check (akt_PTab > max_PTab, 0, insFailPT);
 36:        PTab^ [akt_PTab] := SPtr (akt_STab);
                                INC (akt_PTab);
 37:        PTab^ [akt_PTab] := SPtr (akt_STab);
 38:        ByteCopy (ADR (zstr), l,    akt_STab);
                      INC (akt_STab, l);
 39:        ByteCopy (ADR (pos), AdrSize, akt_STab);
                      INC (akt_STab, AdrSize);
 40:      END;
                                                  →
```

```
41:     END Insert;
42:
43:   PROCEDURE cmp (a: CPtr; b: CPtr; VAR found:
                     SPtr): Relation;
44:     VAR old: SPtr;  (* kleineres Element merken *)
45:     BEGIN
46:       old := SPtr (b);
47:       LOOP
48:         IF    (a^ < b^) THEN RETURN less;
49:         ELSIF (a^ > b^) THEN  found := old;
                                  RETURN greater;
50:         ELSIF (a^ = 0C) THEN RETURN equal;  END;
51:         INC (a); INC (b);
52:       END;
53:     END cmp;
54:
55:   PROCEDURE GetPos (fd: FastDic; VAR zstr: ARRAY
                        OF CHAR; VAR pos: LONGCARD);
56:
57:   (*  Vorbedingung: mind. ein Element existiert
          das <= 'zstr' ist.
58:    *  Der Rest ist binäre Suche auf einem
          sortierten Array
59:    *)
60:
61:     VAR j,k,l,i: INTEGER;
62:         found  : SPtr;
63:     BEGIN
64:       i := 0; j := fd.akt_PTab - 1;
65:       REPEAT
66:         k := (i + j) DIV 2;
67:         CASE cmp (ADR (zstr), CPtr (fd.PTab^ [k]),
                                  found) OF
68:           less    : j := k - 1;
69:         | equal   : pos := MAX (LONGCARD); RETURN;
70:         | greater : i := k + 1;
71:         END;
72:       UNTIL i > j;
73:       l := Length (found^) + 1;
74:       ByteCopy (found,l,ADR(zstr)); INC (found,l);
```

```
75:       ByteCopy (found, AdrSize, ADR (pos));
76:     END GetPos;
77:
78:   PROCEDURE CreateFastDic (VAR fd: FastDic; every:
                CARDINAL;
79:                            words: LONGCARD):
                                    BOOLEAN;
80:     VAR n: CARDINAL;
81:     BEGIN
82:       n := SHORT (words DIV LONG (every));
83:       WITH fd DO
84:         akt_PTab := 0;  max_PTab := (n + lowers);
85:         allocate (PTab, max_PTab * AdrSize);
86:         allocate (STab, minLen + wLen * LONG (n));
87:         akt_STab := STab;  max_STab := STab;
88:         INC (max_STab, minLen + wLen * LONG (n));
89:       END;
90:       RETURN TRUE;
91:     END CreateFastDic;
92:
93:   PROCEDURE CloseFastDic (VAR fd: FastDic);
94:     BEGIN
95:       WITH fd DO
96:         DEALLOCATE (PTab, (max_PTab - akt_PTab) *
                        AdrSize);
97:         max_PTab := akt_PTab;
98:         DEALLOCATE (STab, (LONGCARD (max_STab) -
                        LONGCARD(akt_STab)));
99:         max_STab := akt_STab;
100:      END;
101:    END CloseFastDic;
102:
103:  PROCEDURE ReleaseFastDic (VAR fd: FastDic);
104:    BEGIN
105:      DEALLOCATE (fd.PTab, 0);  DEALLOCATE
                      (fd.STab, 0);
106:      ClearVar (fd);
107:    END ReleaseFastDic;
108:  END FastDics.
```

Listing 6: FASTDIC.I: ...Verwaltung des Suchbaums

Fleiger Schneider Hofmann Forster

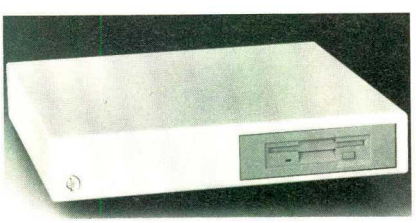
CATCH COMPUTER GbR

Ludwigsallee 1b, 5100 Aachen

Telefon 0241-157393

Professionelle SCSI-Systeme:

Technisch wie optisch für höchste Ansprüche:



CCHD 84 S	84 MB	24ms	Echtzeituhr	1398.-
CCWD 44	44 MB	Wechselplatte, Uhr, Medium		1798.-

Andere Komponenten, Kombinationen auf Anfrage.

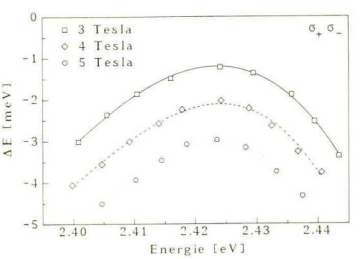
Selbstverständlich besitzen unsere SCSI-Systeme die allgemeine Genehmigung nach Verf. 1046 des ZZF.

Professionelle Hardware:

CCMB 4	RAM-Erweiterung auf 4 MB (bei 1MB Hauptspeicher).	444.-
AdSPEED	16 MHz Beschleuniger von ICD, Cache, softwaremäßig umschaltbar auf 8 MHz.	598.-

Auf Wunsch auch mit Einbau. Fordern Sie unsere kostenlosen Produktinformationen an!

Diagramm 3.0



Sehr gute Druckqualität - Druckraster 1650*1120 Pixel - ASCII-Datenformat mit max. 20 Spalten - Lineare und logarithmische Achseneinteilung - Verwendung von SIGNUM-Zeichensätzen - 12 verscheiden Symbole in beliebiger Größe - viele Linienformen - Eingabe von Funktionen - Darstellung von Meßfehlern - Gitter - Drucker-Warteschlange - Spline-Interpolation und Spline Approximation - Ausgabe in Hardcopy und IMG-Format zur Einbindung in SIGNUM oder CALAMUS - Balkendiagramme mit Schraffur und Stapelung - Treppendiagramme - Lineare Regression - Polynomfit bis Grad 6 - Eingabe von Datenpunkten mit der Maus - Zeichnen von Linien mit der Maus - Radiergummi mit einstellbarer Größe - Setzen und Löschen einzelner Pixel - Bedienung auch über die Tastatur - Druckertreiber für 9 und 24 Nadeldrucker, HP-Laserjet und Atari-Laserdrucker.

Dr. Axel Nöthe
Ringstraße 49
4620 Castrop-Rauxel
Tel.: 02305/12044

Preis 149,- DM
Update 50,- DM
+ Versandkosten

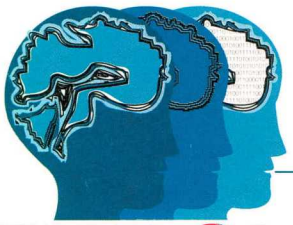
Sprechen Sie PROLOG?

Nein? Könnten Sie aber!

MAXON PROLOG **bietet Ihnen den idealen Einstieg; denn es besitzt eine kompakte Entwicklungsumgebung mit eingebautem, schnellem Editor, Debugger und einem On-Line-Hilfesystem.**

Mit MAXON PROLOG öffnen Sie sich die Welt der symbolischen regelorientierten Datenverarbeitung. Einsatzgebiete sind u.a. algebraisches Rechnen (32 Bit-Integer-Artihmetik), Expertensysteme und Rapid-Prototyping. Doch auch „typische" ST-Anwendungen lassen sich mit MAXON PROLOG verwirklichen, da die Routinen des Betriebssystems (AES, VDI, GEMDOS, BIOS, XBIOS) problemlos angesprochen werden können.

MAXON PROLOG ab Versand inkl. Porto u. Verpackung DM 305,50. Auslandsbestellungen nur gegen Vorauskasse



MAXON PROLOG

Voll integrierter Prolog-Interpreter

MAXON PROLOG eignet sich aufgrund der vielen mitgelieferten Quelltexte, darunter die Entwicklungsumgebung selbst, auch hervorragend als Lehr- und Lernsystem. Der 2500 LIPS schnelle Interpreter kennt alle von Clocksin und Mellish definierten Prädikate. MAXON PROLOG läuft auf allen ATARI STs mit mindestens 1MB Speicher.

DM 298,-

unverbindlich empfohlener Verkaufspreis

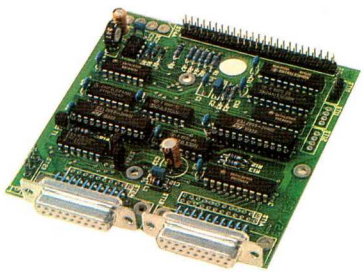
MAXON computer gmbh

Schwalbacher Str. 52 • 6236 Eschborn
Tel.: 06196/48 2811 • Fax: 06196/41885

DER MAXON SCSI-ADAPTER

SCHLICHT UND EINFACH GUT

Der MAXON SCSI-Adapter ist das brandneue Festplatten-Interface für die ATARI ST-Serie, das den Einsatz zukunftssicherer SCSI-Festplatten am ST nun auch für den kleinen Geldbeutel erschwinglich macht. Trotz - oder gerade wegen - seines einfachen und geradlinigen Aufbaus bietet es volle Kompatibilität zum SCSI-Befehlssatz und glänzt gleichzeitig durch höchste Übertragungsraten. Zusätzlich gewährleistet dieses technische Design in Verbindung mit der ausgereiften Software eine extreme Zuverlässigkeit, die höchste Datensicherheit auch im ständigen harten Einsatz garantiert. Damit ist der MSA die optimale und zukunftssichere Interface-Erweiterung auch für Ihren ST.



Der MAXON SCSI-Adapter

- erreicht traumhaft hohe Übertragungsraten von 1000 KByte/sec. ohne und 860 KByte/sec. mit Zylinderwechsel.
- macht das angeschlossene SCSI-Gerät uneingeschränkt bootfähig.
- wird mit leistungsfähiger und komfortabler Software ausgeliefert.
- unterstützt alle SCSI-Kommandogruppen.
- bietet einen gepufferten DMA-Bus.
- ermöglicht den Anschluß von bis zu vier SCSI-Geräten.
- ist kompatibel zu allen erhältlichen SCSI-Festplatten.
- besitzt einen per Schalter aktivierbaren Hardware-Schreibschutz.
- ist als Fertiggerät wie auch als Bausatz zu einem sensationellen Preis erhältlich.

Warum sollten Sie mehr für weniger ausgeben?
Mit dem MAXON SCSI-Adapter erhalten Sie modernste SCSI-Technologie für wenig Geld!

Erhältlich ist der MSA in folgenden Versionen:

Als Komplettgerät mit Treiber- und Initialisierungs-Software sowie Anleitung, zum Einbau in den ST oder ein externes Gehäuse
Bestell-Nr.: 90 0810 unverbindliche Preisempfehlung **DM 259.-**

Als Bausatz, bestehend aus Platine, 2 programmierten GALs, Software und Anleitung
Bestell-Nr.: 90 0811 unverbindliche Preisempfehlung **DM 149.-**

MAXON Computer
Schwalbacher Str. 52
6236 Eschborn
Tel: 06196/481811

GANZ KURZES ACCESSORY - DIE ZWEITE

Harald Günterberg

Als Assembler-Programmierer fühlte ich mich natürlich herausgefordert, ein 'bis aufs letzte Byte optimierte Accessory' [Jürgen Stessun: Ein ganz kurzes Accessory, ST Computer 10/89] noch kürzer zu schreiben. Dabei sollten keine undurchsichtigen Tricks zur Anwendung kommen, sondern effektiver Assembler-Code, der sich durch Reduzierung auf das allernötigste auszeichnet.

Mit den resultierenden 264 Bytes Länge ist noch längst nicht das Minimum erreicht; in Zeile 33 des Listings ist erläutert, wie man weitere 2 Bytes einsparen kann; in Zeile 35 ist ein langer Sprung (bsr), der wahrscheinlich durch Umstrukturierung optimiert werden kann. Ich hab's stehen gelassen (was einmal steht und gut läuft...).

Da unser *MINI_ACC* verschiedene AES-Funktionen aufruft und wir keine Bibliothek hinzulinken wollen, müssen die AES-Parameter von Hand gesetzt werden. Das Control-Array muß stets mit 5 Werten belegt werden: die Funktionsnummer nach *control [0]* sowie die Anzahl der Parameter in den übrigen Feldern nach *control [1]* bis *[4]*. Hinzu kommen, je nach Funktion, noch weitere Parameter, die in *intin* und *addrin* übergeben werden. In meiner MINI_ACC-Version verwende ich den *movep*-Befehl zur Übergabe der Parameter in *control [1]* bis *[4]*. Dadurch verkürzt sich die Handarbeit auf *move.l#$01010100,d1*.

Noch ein Wort zum *movep*-Befehl: Dieser Befehl transportiert Bytes mit einer Schrittweite von 2 Bytes. Anders gesagt: Je ein Byte wird wortweise (an einer geraden oder an einer ungeraden Adresse beginnend) in den Speicher geschrieben oder gelesen. Diese Befehlsverwendung ist für die GEM-Parameterübergabe eindeutig die kürzeste und außerdem sehr übersichtlich. Leider kamen frühere Versionen des GFA-Assemblers mit dem *movep*-Befehl nicht zurecht, doch seit der Version 1.3 ist dieser Fehler behoben.

Zur Speicheradressierung wird Register A5 mit dem Label *contrl* geladen und im weiteren Verlauf nicht geändert, sonst müßte es gerettet oder neu geladen werden. Alle benötigten Adressen werden dann durch symbolische Offsets angesprochen. In den Zeilen 18-25 werden die Konstanten dazu vereinbart, die Werte entstehen durch Addition der benötigten Distanz zum vorhergehenden. Verständlicher wird die Berechnung der Konstanten, wenn man sie mit den Labels im BSS-Segment vergleicht. Dem aufmerksamen Leser wird nun auffallen, daß das Label *evntmsg* im Text-Segment gar nicht angesprochen wird. Doch dies geschieht in Zeile 44 mit *lea.l evnt(a5),a0*. In der Tat kann bei der Programmierung mit symbolischen Displacements auf das Setzen von Marken im BSS-Segment verzichtet werden. Man muß nur die Konstanten korrekt berechnen und genügend Speicherplatz anlegen. Es muß auch nicht jedes Displacement als Konstante definiert sein, z.B. wird *intin [1]* mit *move.w #1,int+2(a5)* erreicht. Im vorliegenden Quelltext wird in Zeile 38 die ID-Nummer der Applikation mittels *move.w apid(a5),int(a5)* von *global [2]* nach *intin [0]* übertragen.

Für die Überlegungen der Optimierung ist interessant, daß eine Routine im Programm häufig aufgerufen wird (*aes*), in der die sich wiederholende Arbeit erledigt wird. Daher sind die spezifischen Parameter der Funktionsaufrufe sehr schnell in Datenregister geladen (Zeile 31 und 32). Die Programm-ID (Identifikationsnummer) und die Nummer des Accessory-Eintrag müssen nicht zwischengespeichert werden. Die ID ist ohnehin im Global-Feld eingetragen (macht GEM bei der Anmeldung). Korrekterweise müßte man auf Fehlermeldungen, besonders bei *menu_register* entsprechend reagieren, indem man das Accessory 'auf Eis' legt, also statt *evnt_mesag* dann *event_timer* mit einer langen Zeitdauer aufrufen. Es kann ja durchaus vorkommen, daß sich vorher schon sechs andere Accessories angemeldet haben.

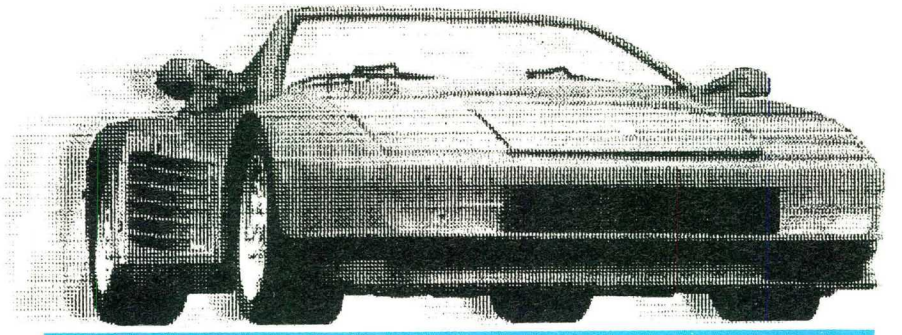
Die Warteschleife des Accessories beginnt automatisch mit dem AES-Aufruf *evnt_mesag* und wird nur dann verlassen, wenn das Ereignis *AC_OPEN* (Zeile 46) gemeldet wird. GEM sorgt schon dafür, daß nur die Meldungen, die unser Programm betreffen, im Message-Puffer landen. Dann geht's sofort weiter mit der Ermittlung des freien Speicherplatzes; dieser wird auf eher konventionelle Weise nach dezimal und dann nach ASCII gewandelt. Leider ist mir auch keine universelle Routine eingefallen, die von der Länge her mithalten könnte. Diese Wandlungsroutine ist auf 7 dezimale Ausgabestellen festgelegt, wobei allerdings führende Nullen unterdrückt werden. Sie ist aber

ganz ordentlich optimiert: D4 wird mit *moveq.l #0,d4* initialisiert und dient dann als Flag für führende Spaces. Der Zähler D1, der ja die Dezimalstelle repräsentiert, wird einfach (und schnell) zu D4 dazugeodert. Ab dem ersten Ergebniswert werden dann dezimale Ziffern ausgegeben.

Etwas unästhetisch, aber unvermeidbar (hat jemand eine elegante Lösung?), ist die Langwortdivision in Zeile 69-71. Beim ersten Durchgang entsteht ein Überlauf, da der Quotient größer als die zugelassene Wortlänge ist (der Startwert für 7 Stellen ist 1.000.000). Bvc heißt: ist das Überlaufflag nicht gesetzt, (war also die Division fehlerfrei), dann mache weiter mit der Schleife *dbra...* Sonst ersetze das Fehlergebnis durch den zweiten Wert 100.000. Im vorliegenden Beispiel entsteht kein Divisionsrest, sonst müßte man das Ergebnis auf den Quotienten mit *andi.l #$FFFF,d2* oder auch mit *ext.l d2* bereinigen.

Die restlichen Zeilen sind im Quelltext erläutert, das minimal angelegte BSS-Segment reicht natürlich nur für die vorliegenden Funktionsaufrufe; wenn es nicht auf die absolute Minimierung ankommt wie hier, sollten die Felder immer reichlich bemessen sein.

```
 1:  ; MINI_ACC.Q        (GFA-Assembler 1.3)
 2:  ; ----------------------
 3:  ; Ein kürzeres Accessory (Länge: 264 Bytes)
 4:  ; zur Anzeige des freien Speichers.
 5:
 6:  ; 30.09.89 von:    Harald Günterberg
 7:  ;
 8:  ;
 9:  ;         (c) MAXON Computer GmbH 1990
10:
11:
12:  ; Für andere Assembler lediglich die Direktiven
13:  ; (.ds.w, .TEXT, etc.) anpassen.
14:
15:  ; vorwiegende Adressierungsart: indirekt mit
       symbol. Displacement
16:  ; Konstantendeklaration: symbolische Offsets ab
       Speicher-Basis 'contrl'
17:
18:  con        = 0
19:  glob       = con+2*12
20:  apid       = glob+4
21:  int        = glob+2*15
22:  iout       = int+2*2
23:  addr       = iout+2*2
24:  aout       = addr+1*4
25:  evnt       = aout+1*4
26:
27:  .TEXT
28:
29:  lea.l      stack(pc),sp    ; Für Accessory nötig
30:  lea.l      contrl(pc),a5   ; BASIS-Register laden
31:  moveq.l    #10,d0          ; appl_init
32:  move.l     #$00010000,d1   ; wird per 'movep.l'
                                   transportiert
                          ;kürzer: moveq.l #1,d1
33:  swap     d1
34:  bsr      aes             ;GEM setzt ap_id autom
35:  moveq.l    #35,d0          ; menu_register
36:  move.l     #$01010100,d1
37:  move.w     apid(a5),int(a5) ;global[2]=unsere ID
38:  lea.l      name(pc),a0     ; A0 nach Addrin
39:  bsr.s      aes         ; ACC-Nummer uninteressant
40:
41: doacc:
41:  moveq.l    #23,d0          ; evnt_mesag
43:  move.l     #$00010100,d1
44:  lea.l      evnt(a5),a0     ; Nachrichtenpuffer
45:  bsr.s      aes
46:  cmpi.w     #40,evnt(a5)    ; AC_OPEN?
47:  bne.s      doacc           ; mehr braucht's nicht!
48:
49:  moveq.l    #-1,d0          ; jetzt starten wir's
50:  move.l     d0,-(sp)        ; wieviel Speicher da?
51:  move.w     #$48,-(sp)      ; MALLOC
52:  trap       #1              ; Rückgabewert in D0
53:  addq.l     #6,sp
54:
55:  lea.l      mem(pc),a0    ; D0 in String eintragen
56:  moveq.l    #6,d3         ; 7 Stellen brauchen wir
57:  moveq.l    #0,d4         ; Flag für Spaces
58:  move.l     #1000000,d2   ; Startwert (7. Stelle)
59: loop:
       moveq.l   #-1,d1       ; Zähler
60: dec:
     addq.w     #1,d1         ; (D1-#1) mal ...
61:  sub.l      d2,d0         ; ...subtrahiert
62:  bpl.s      dec           ; noch >0 weitermachen
63:  add.l      d2,d0       ; D0 wieder positiv machen
64:  or.w       d1,d4         ; Zähler und Flag...
65:  bne.s      digit         ; Ziffer ausgeben
66:  moveq.l    #(' '-'0'),d1 ; sonst Leerzeichen
67:  digit:
      addi.b    #'0',d1
68:  move.b     d1,(a0)+      ; Space oder Ascii-
                                 Ziffer schreiben
69:  divu.w     #10,d2        ; nächste Stelle
70:  bvc.s      div_ok        ; bei der ersten
                                 Division ensteht ein
71:  move.l     #100000,d2    ; Überlauf, bei folgen-
                                 den kein Rest!!
72:  div_ok:
      dbra      d3,loop       ; 7mal
73:
74:   moveq.l   #52,d0        ; Ergebnis per
                                 Alert ausgeben
75:   move.l    #$01010100,d1
76:   move.w    #1,int(a5)    ; Defaultbutton
77:   lea.l     alert(pc),a0
78:   bsr.s     aes
79:   bra.s     doacc         ; zur ACC-Warte
                                 schleife zurück
80:
81:  aes:
      move.l    a0,addr(a5)   ; Addrin besetzen...
82:   move.w    d0,(a5)       ; Funktionsnummer
83:   movep.l   d1,3(a5)      ; ...und
                                 die Parameterliste
84:   lea.l     aespb(pc),a0  ; spart Relozierbyte
85:   move.l    a0,d1         ; Parameterblock in D1
86:   move.w    #$c8,d0       ; und AES-Call in D0
87:   trap      #2
88:   rts
89:
90:   .DATA
91:
92:  aespb:     .DC.l contrl,global,intin,intout,
                      addrin,addrout
93:
94:
95:  name:      .DC.b '  Mem',0
96:
97:  alert:     .DC.b „[0][ „
98:  mem:       .DC.b „          ][Byte]",0
99:
100:           .BSS
101:
102: contrl:    .DS.w 12
103: global:    .DS.w 15
104: intin:     .DS.w 2  ; Diese SparmaPnahmen sind
                           nur für Demozwecke -
105: intout:    .DS.w 2  ; sonst immer richtig dimen-
                           sionieren (128 Words)!
106: addrin:    .DS.l 1
107: addrout:   .DS.l 1
108: evntmsg:   .DS.w 8   ; Messagepuffer
109:            .DS.l 10  ; auch lieber größer
                            dimensionieren
110: stack:
111:            .END
```



Performance, die begeistert

HYPERCACHE TURBO+

Der neue Hypercache Turbo+ läßt keine Wünsche mehr offen !!

- Neu ist z.B. die Fast Rom Option oder die 8MHz/16MHz Umschaltung bei laufenden Programmen durch Accessory, Hotkey oder Schalter. Bei der Umschaltung wird gleichzeitig das Cache Ram aktiviert/deaktiviert. Das bedeutet 100% Kompatibilität zu allen Programmen

- Standardmäßig ist jeder Hypercache Turbo+ mit einem CMOS-Prozessor ausgestattet.

- Durch die neue Bauform paßt der Hypercache Turbo+ in jeden Rechner - auch in den STE.

- Hypercache Turbo+ - die Summe jahrelanger Erfahrung

Das Original. Nur 498,- DM

Vertrieb Ausland:

Compo Software GmbH
Ritzstraße 13
5540 Prüm
Telefon : 0 65 51 - 62 66
Fax : 0 65 51 - 63 39

Distribution und Verlag:

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt - Eberstadt
Telefon : 0 61 51 - 5 60 57
Fax : 0 61 51 - 5 60 59

Eine Produktion der Pro VME

SPEICHERVERWALTUNGEN NOCH EFFIZIENTER

Klaus Elsbernd

Haben Sie schon einmal versucht, ungefähr 191158 Objekte gleicher Grösse vom Betriebssystem anzufordern? Nein? Macht nichts. Irgendwann werden Sie auch einmal dazu kommen. Aber wenn das doch der Fall war, sollten Sie diesen Artikel nicht nur überfliegen. Dieses Problem tritt meist nur bei Programmen auf, die mit dynamischen Datenstrukturen „rechnen" müssen.

Oft ist man versucht, diese über ein festes Array den eigenen Routinen zur Verfügung zu stellen. Jedoch ist ein solches Verfahren zu statisch, als daß es in den meisten Fällen befriedigen würde. Vielleicht braucht man ja auch nur die Hälfte? Wer weiß. Hängt ganz davon ab, was der Anwender so alles mit dem Programm anstellt.

Speicherverwaltung ist ein Problem, daß man meist dem Betriebssystem oder der Programmiersprachen-Library überläßt. Diese gehen damit recht stiefmütterlich um. Aus diversen Artikeln u.a. in dieser Zeitschrift [1, 2] und Büchern wissen wir alle von der mangelhaften Speicherverwaltung mittels der sogenannten Memory-Deskriptoren (MDs) des GEMDOS. Also griffen die Hochsprachen-Compiler zu unterstützenden Maßnahmen in Form eigenener, vorgeschalteter Routinen, die die Anzahl der *malloc*-Aufrufe an das Betriebssystem reduzieren. Aber auch das ist nicht immer der Weisheit letzter Schluß.

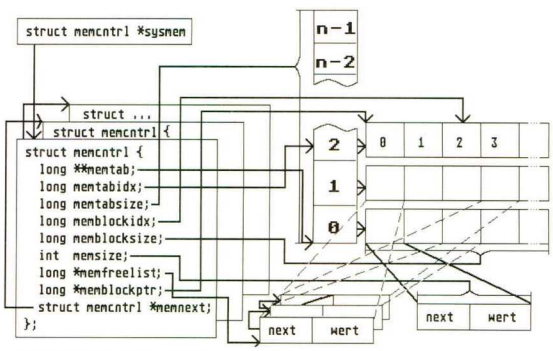
Daß diese nicht immer befriedigen, erkannte auch Hans-Jürgen und stellte eine eigene Implementierung für TURBO-C vor. Sein Ansatz besteht im Prinzip aus der Verringerung von Verwaltungsinformationen und einer verbesserten Rückführung von nicht mehr benötigtem Speicher. Die Nachteile dieses Algorithmus' sind seine Festlegung auf eine bestimmte Speichergröße (die sich bei meinem Problem nicht im voraus berechnen läßt) und der nicht unbeträchtliche Verwaltungsaufwand, um die Freispeicher effektiver zurückführen zu können. Die hier vorgestellte Speicherverwaltung setzt an einer anderen Stelle der Problematik an und ist relativ zum Implementierungsaufwand die flexibelste und schnellste ihrer Art.

Historie

Für ein Projekt mußte ich in meinem LISP-System „My-Lisp" eine neue Datenstruktur (defstruct) einführen. Diese verlangte die Reservierung einer bestimmten Anzahl Bytes für ein Objekt dieses Typs. Ich verließ mich ganz auf die Speicherverwaltungen des Compiler-Systems (nicht TURBO C), die ja auch korrekt funktionierten. Bei meinen Zeitmessungen, die ich für dieses Projekt unternahm, machte ich eine mir anfangs unerklärliche Feststellung: Der Aufruf einer Lisp-Funktion dauerte beim ersten Mal 3.125 Sekunden. Beim 2. Mal nur noch 1.670 Sekunden. Dies ließ sich aufgrund der Eigenschaften der Funktion genauestens erklären (Makroexpansion). Jedoch bei weiteren Aufrufen, die sich programmtechnisch nicht von dem 2. Aufruf unterschieden, stieg die benötigte Zeit zur Lösung immer weiter an. In einem anderen Beispiel erhöhte sich die Verarbeitungszeit von 4 Min auf 1.5 Stunden (90 Minuten). Ich zweifelte an meiner Zeitmessung.

Die Erklärung...

... kam nach einer kurzen Analyse des Lisp-Systems. Die Funktion verursachte bei ihren Berechnungen Tausende von Aufrufen der Speicherverwaltung. Jetzt stelle man sich einmal vor, es seien PI mal Daumen 317 Objekte à 12 Bytes (im folgenden zumeist Records genannt), die für einen Aufruf dieser Funktion benötigt würden. Ein zweiter Aufruf benötigte wiederum 317 * 12 Bytes.



Komplette Speicherverwaltung

Die Verwaltung des Speichers hat also mindestens 632 Records unter ihrer Aufsicht. Ein weiterer Aufruf der Funktion erhöht die Anzahl auf 3*317= 951. Und so weiter. Beim 13. Funktionsaufruf sind es mithin 3170+951= 4121 Records à 12 Bytes. Jeder Kartenspieler weiß, daß er beim Einsortieren einer neuen Karte umso länger braucht, je mehr er davon schon auf der Hand hat. Genauso sieht es bei den Speicherverwaltungen aus. Bei der Bearbeitung einer Speicheranforderung wird nach einem mehr oder weniger effektiven Algorithmus überprüft, welcher Speicherblock der Freispeicherliste für die Zuteilung der Anforderung am zweckmäßigsten ist (First-Fit, Best-Fit usw.). Die Überprüfung dieser Seicherbereiche dauert umso länger, je länger die zu verwaltende Liste ist. Welch Weisheit.

Nochmals das Problem...

... in komprimierter Form. Es wird eine große Zahl von gleich großen Speicherbereichen benötigt. Die genaue Zahl läßt sich nicht im voraus bestimmen. Sie ist von Programmaufruf zu Programmaufruf verschieden. Ebenso ist die Größe dieser Bereiche erst zur Laufzeit feststellbar. Geschwindigkeit hat das Primat über Speicherverbrauch (auch ein 1040er hat eine Menge Speicher). Dieses Problem tritt dann auf, wenn man dynamische Datenstrukturen aufbaut, deren Speicherbereiche über eine Funktion vom System (hier und im folgenden ist damit sowohl die GEMDOS-Speicherverwaltung als auch die in der Programmiersprache integrierte gemeint) angefordert werden sollen. In Pascal wird dafür die Funktion *new(*record)* verwendet.

Die Lösung...

... ist eigentlich recht einfach und wurde z.B. in einfacherer Form in [3] benutzt. (Gute Lösungen sind immer einfach.) Wenn es auch in der Problembeschreibung so klingt, als ob hier der Speicher verschwendet würde, so ist dies, wie wir weiter unten sehen werden, nicht der Fall. In der Tat arbeitet der Algorithmus speichersparender als andere, wenn nur eine genügend große Zahl von Records benötigt wird.

Im Prinzip besorgen wir uns von der Speicherverwaltung (also über *malloc*) einen zusammenhängenden Speicherbereich (Block), der eine gewisse größere Menge unserer benötigten Strukturen umfaßt. Diese können wir als ein einfaches Array verwalten, indem wir bei Bedarf das nächste freie Element zur Verfügung stellen.

Ist der Block voll, wird ein neuer vom System angefordert, und die Zuteilung erfolgt nun aus diesem. Die Verwaltung des gerade aktuellen Blocks erledigen wir über eine Tabelle. (Eine Verkettung als sequentielle Liste durch Zeiger am Anfang des Blockes ist leicht zu implementieren.) Wir erhalten also ein zweistufiges System, über das uns Speicher zugeteilt wird. Das Ziel ist eine möglichst flexible Verwaltung, die nicht durch irgendwelche Konstanten beschränkt wird. Damit ist sowohl eine Anpassung an die eigenen Programme als auch an die dynamischen Anforderungen innerhalb eines Programmlaufes möglich. Insbesondere ist die Größe des benötigten Records flexibel. Gleiches gilt für Art und Umfang verschieden großer Datenstrukturen.

Als Informationen werden folglich benötigt:

- die Adresse der Tabelle der Blöcke *memtab*. In ihr werden die Blöcke in die Verwaltung eingeklinkt.
- der Index innerhalb der Tabelle der Blöcke *memtabidx*. Dieser zeigt auf den Block, aus dem die Zuteilung erfolgt.
- die maximale Anzahl der Blöcke *memtabsize*, die über *malloc* vom System geordert werden sollen. Diese Begrenzung ist eigentlich nicht nötig, aber sie ermöglicht eine rechtzeitige garbage-collection (Müllbeseitigung), wenn Objekte auch wieder freigegeben werden müssen. Sonst wird solange Speicher vom System angefordert, bis die Library-Routine abwinkt. Das ist insofern fatal, als es auch andere Routinen gibt, die Speicher benötigen. Auch diese könnten dann nicht mehr bedient werden. Wird vorher abgebrochen, haben sie jedoch noch eine Chance.
- die Adresse des augenblicklichen Blockes *memblockptr*. Dadurch wird die Zugriffsgeschwindigkeit auf einen neuen Record beschleunigt.
- der dazugehörige aktuelle Index *memblockidx* innerhalb des aktuellen Blockes.
- diverse statistische Informationen, wie die Anzahl der belegten Elemente aller Blökke *memstatuentries*, die Zahl der noch freien Elemente *memstatfentries* nach einer garbage-collection, und die Zahl der zur Verfügung stehenden Elemente *memstatentries*. Letztere unterscheiden sich erst dann von *memstatuentries*, wenn das System keinen weiteren Speicher zur Verfügung stellt und man den Müll (garbage) einsammeln und verwenden muß.
- da wir verschiedene solcher Speicherorganisationen für unterschiedlich große Datenstrukturen verwalten wollen, verketten wir die Kontrollinformationen miteinander und benötigen deshalb einen Zeiger auf das nächste Speicherverwaltungselement *memnext*.

Größe der Datenbereiche	> n
4-22	263
24-54	264
56-84	265
86-...	256

Tabelle 1 für BLOCKSIZE = 8kB ohne Berücksichtigung des zusätzlichen Programmcodes

Größe der Datenbereiche	> n
4-10	263
12-26	264
28-42	265
44-56	256
58-...	267

Tabelle 2 für BLOCKSIZE = 4kB ohne Berücksichtigung des zusätzlichen Programmcodes

Größe der Datenbereiche	> n
4-18	327
20-42	328
44-68	329
70-...	330

Tabelle 3 für BLOCKSIZE = 8kB mit Berücksichtigung des zusätzlichen Programmcodes

Der Algorithmus

Wenn nun ein Programm eine bestimmte Datenstruktur benötigt, die sehr häufig vorkommt, ruft es das Unterprogramm *freshmem* auf. Es erwartet die Adresse einer oben beschriebenen Speicherverwaltungsstruktur. Um diese zu erhalten, wird in einer verketteten Liste mit Hilfe der Funktion *searchmemcntrl* nach einer passenden Struktur gesucht.

Btx/Vtx-Manager

Btx/Vtx: Nase vorn

in der Welt der Telekommunikation mit dem *Btx/Vtx-Manager V3.0.*
Sie wollen Ihr Konto verwalten, Bestellungen aufgeben, eine Urlaubsreise buchen ...
Entdecken Sie jetzt die neuen komfortablen Wege, die Ihnen der *Btx/Vtx-Manager* (als intelligente Komplettlösung) mit dem Abruf aktuellster Informationen und Daten rund um die Uhr liefert.
Ausführliche Informationen erhalten Sie bei Ihrem Atari-Fachhändler oder direkt von uns.

Atari ST Btx/Vtx-Manager V3.0 für 389,– DM *an Postmodem bzw.* 289,– DM *an Akustikkoppler/Hayes-Modem.* (FTZ-Zulassung beantragt). Unverbindliche Preisempfehlungen.

Drews EDV + Btx GmbH
Bergheimerstraße 134 b
D-6900 Heidelberg
Telefon (0 62 21) 2 99 00
Fax (0 62 21) 16 33 23
Btx-Nummer 0622129900
Btx-Leitseite * 2 99 00 #

Bildschirmtext

d Drews

MEGA 2→ MEGA 4 DM 398.––
1040 STE auf 2/2.5 MB DM 298,––
1040 STE auf 4 MB DM 498,––
Wir nehmen Ihre alten Simm-Module in Zahlung!

Aufrüstungen 260/520/1040/MEGA 1 auf 2 – 5 MB	ab 348,––
MEGA-CLOCK kompatibel zur MEGA-ST-Uhr	99,––
ICD AdSpeed 16 Mhz Accelerator – Superleistung auf engstem Raum	
CMOS-CPU, 32 KB Data/Tag Cache, Fast-ROM-Option	578,––
1040STE & SM124	1098,––
1040STE mit 2 MB & SM124	1298,––
1040STE mit 4 MB & SM124	1498,––
AT-Speed	478,––
Vortex ATonce	478,––
PC-Tastatur anschlußfertig	378,––

GENG TEC
GengTec Gerald Geng
Teichstraße 20 4020 Mettmann
Tel. 02104/22712 FAX 02104/22936

Vortex Datajet Festplatten	ab DM	1099,—
GFA-Basic EWS V3.5 dt. (Interpr.+ Com.)	DM	229,—
That's Write Profi dt. - Textverarb. V1.5	DM	289,—
SPC-Modula II V1.42 (2.0)	DM	329,—
Turbo-C mit Ass. + Sourcedebugger V2.o dt.	DM	349,—
Signum II deutsch	DM	a.A.
Interlink ST-DFÜ-Programm	DM	69,—
Turbo St-Software Blitter dt. V1.8	DM	79,—
AT-Speed MS-DOS-Emulator V2.21	DM	409,—
BTX-Manager V3.02 dt./an DBT03	DM	299,—
N-N-Disk 3.5-Z DD.....DM –,99 Psion Chess	DM	59,95
Spiele (Restposten)	ab DM	10,—
LDW-Power Calc dt. DM 209,— Cyber Paint 2	DM	109,—
Amstrad 24-Nadeldrucker LQ 3500 di dt.	DM	499,—
Megamex Modula II dt.	DM	309,—

Kostenlose Prospekte, auch für Amiga und IBM von

CWTG **Joachim Tiede**
Bergstraße 13 • 7109 Roigheim
Tel./BTX 06298/3098 von 17–19 Uhr

Leserservice



DM 12,-
für zwei Monate

Enthält alle Listings und Programme – keine Tipparbeit mehr!

Die Diskette zur ST-Computer

Alle zwei Monate erscheint die Monatsdiskette der ST-Computer. Auf ihr sind alle Listings und Programme enthalten, die in zwei aufeinanderfolgenden Ausgaben abgedruckt sind, z.B. Januar/Februar oder März/April. Ausnahme bildet die Diskette zur sommerlichen Doppelnummer der ST-Computer, die nur einen Monat abdeckt.

Ab dieser Ausgabe kostet eine Monatsdiskette nur noch DM 12,-. Wir haben für Sie nachgerechnet:

2 * ST-Computer = DM 16,-
1 * Monatsdiskette = DM 12,-
---
2 Monate voll informiert = DM 28,-

Sie sehen, für nur DM 14,- pro Monat sind Sie immer auf dem Laufenden und sparen sich lästige Tipparbeit. Und der Clou: Die Lieferung erfolgt versandkostenfrei. Bestellen Sie schon jetzt die Monatsdiskette der Januar/Februar-Ausgabe 1991 der ST-Computer für DM 12,- (nur gegen Vorauskasse).

Bestellung unter: **Heim Verlag**
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Wer will, kann bei bekannten Record-Größen diese auch fest „verdrahten" und schon passende Variablen anlegen. Für beide Vorgehensweisen finden Sie ein Beispiel in Listing 2.

searchmemcntrl durchsucht die Liste der Speicherverwaltungsblöcke sequentiell nach dem zugehörigen Verwaltungsblock. (Das Suchverfahren bricht ab, wenn *size >= mem->memsize* ist.) Wird keine solche gefunden, d.h. keine Verwaltungsstruktur verwaltet Speicherbereiche der Größe *sizeof(struct myrec)*, wird eine solche eingerichtet.

Diese Aufgabe erledigt *newmemcntrl*. Die Routine ermittelt aufgrund der Größe der gewünschten Datenstruktur die Blockgröße als ein ganzzahliges Vielfaches in der Nähe von „willkürlich" 8 kB. Weiterhin werden die Verwaltungsinformationen in der Speicherstruktur auf Anfangswerte gesetzt. Nachdem *searchmemcntrl* die neue Speicherstruktur eingekettet hat, wird diese zurückgegeben.

Zurück zu *freshmem*. Diese Funktion soll ein Objekt einer Datenstruktur liefern. Dazu haben wir ihr ja die zugehörige Speicherverwaltung übergeben. Das Unterprogramm unterscheidet zwei Phasen:

- **Phase 1:** Es gibt noch Speicher vom System, oder die maximale Anzahl von Blökken ist noch nicht erreicht.
- **Phase 2:** Es gibt keinen Speicher vom System mehr, oder die maximale Anzahl der vom Programm zu benutzenden Blöcke (falls es eine solche Grenze geben soll) ist erreicht. Dann werden nur noch Records aus einer verketteten Liste zur Verfügung gestellt. Diese muß dann aus den nicht mehr verwertbaren Elementen aufgebaut sein.

freshmem überprüft zuerst, ob wir uns noch in Phase 1 befinden. Das ist der Fall, wenn das System noch Speicherplatz für die Blöcke zur Verfügung stellen konnte. Anfangs ist dies (fast) immer so. Dann wird überprüft, ob es noch freie Records innerhalb eines Blokkes gibt (mem->memblockidx++ < mem->memblocksize). Ist dies nicht der Fall, wird ein neuer Block vom System angefordert und an der Stelle *mem->memtabidx* in der Tabelle *mem->memtab* eingetragen. Der Aufruf von *memset* löscht den Block. Zurückgeliefert wird nun der erste Record innerhalb dieses Blockes.

Es kann aber passieren, daß entweder keine neuen Blöcke mehr angefordert werden dürfen (dies ist bei entsprechender Dimensionierung der *memtab* mittels des Makros *CELLTABSIZE* ziemlich unwahrscheinlich), oder daß es keinen freien Speicherplatz mehr dafür gibt. In beiden Fällen wechseln wir zur Phase 2. Von nun an können wir nur noch Datenstrukturen zur Verfügung stellen, die von unserem Anwendungsprogramm freiwillig zurückgegeben worden sind. Dies tat es (hoffentlich) mittels des Unterprogramms *freemem*. Es kann auch eine sogenannte garbage-collection *collectgarb* aufgerufen werden, die das Aufsammeln des entsprechenden Speichermülls besorgt. Nun wird immer nur von der Freispeicherliste das 1. Element zurückgegeben. Hier nun der Vollständigkeit halber eine Einschränkung des Verfahrens (die eigentlich keine ist). Die minimale Speichergröße für Records ist die Größe eines Zeigers, also im allgemeinen 4 Bytes.

Eine einfache Modifikation des Unterprogramms *freshmem* erlaubt die sofortige Nutzung zurückgegebener Records; hierzu werden am Anfang die Zeilen aus Listing 3 eingefügt. Aus der Phase 2 kehren wir niemals in die Phase 1 zurück.

Was bringt's

Diese Frage hat zwei Seiten. Zum einen:

- Wie steht es mit dem Speicherverbrauch? Die Verwaltung des Systems muß dem Verbrauch der Blockverwaltung gegenübergestellt werden. Aus [1] wissen wir, daß die benötigten Records in TURBO-C mittels 8 Bytes verkettet werden. Für „n" solcher Datenstrukturen benötigen wir also „8n" Bytes. Hinzu kommen, und das werden wir vernachlässigen, 16 Bytes für initiale Verwaltungseinträge je vom System benötigten Block.

Das hier vorgestellte System benötigt eine Tabelle der zu verwaltendenen Blöcke, also CELLTABSIZE*sizeof(long **) == 4*CELLTABSIZE Bytes, sowie eine Verwaltungsstruktur [sizeof(struct memcntrl) == 46 Bytes] und zusätzlich sizeof(sysmem) == 4 Bytes. Weiterhin sind die vom System benötigten Verwaltungs-Bytes zu berechnen. Diese schätzen wir mit jeweils 8 Bytes je Block nach unten ab. Wann lohnt sich also diese Verwaltung? Genau dann, wenn

```
8*n > 4*CELLTABISZE + 46 +
 4 + 8*(trunc(n/anzblk)+1)
```

mit

```
anzblk = max (BLOCKSIZE /
sizeof(struct myrecord),1)
```

:=> mit

```
sizeof(struct myrecord)
                    < BLOCKSIZE

8*(n - trunc((n * sizeof
   (struct myrecord))/
   BLOCKSIZE)) >4*
   CELLTABISZE + 50 + 8
```

CELLTABSIZE kann aufgrund der normalerweise maximalen Ausbaustufe von 4MB für den Atari maximal 4MB/8kB = 512 Bytes sein.

```
:=>
n - trunc((n * sizeof
    (struct myrecord))/
    BLOCKSIZE) > 256 +
    50/8 + 1
```

macht ungefähr

```
n - n * sizeof(struct
 myrecord)/BLOCKSIZE > 256
 + 50/8 + 1 = 263.25

n*(1 - sizeof(struct
 myrecord)/BLOCKSIZE)
 > 263.25

n > 263.25 / (1 - sizeof
 (struct myrecord)/
 BLOCKSIZE)
```

In den Tabellen 1 und 2 kann man für verschiedene Größen eigener Datenstrukturen sehen, ab wann sich diese Speicherverwaltung lohnt, differenziert nach Größe der jeweiligen Blöcke. Wie man sieht, lohnt es sich bereits schon nach wenigen Datensätzen gleicher Größe.

Eigentlich müssen wir bei unseren Überlegungen auch noch den Speicherverbrauch der programmierten Routinen berücksichtigen. Diese schlagen unter TURBO-C mit ca. 512 Bytes zu Buche. Daraus ergibt sich ein Korrekturfaktor von 512/8 = 64 für die rechte Seite der Ungleichung. (Daten hierzu finden sich in Tabelle 3.)

- Ein anderer Aspekt ist die Geschwindigkeit des ablaufenden Programms. Diese hat sich in meinen Fall mehr als verdoppelt. Im Durchschnitt besteht die Allokation eines neuen Records aus dem zumeist nicht benötigten Zugriff auf die entsprechende Verwaltungsstruktur und einem Array-Zugriff.

Die Adresse wird einfach durch Shift-Operationen und einfache Addition des Indexes ermittelt. Die Zeit für einen Unterprogrammaufruf und die Abfrage der Phase kommen hinzu. Insgesamt ist dieses Verfahren wesentlich schneller als das Durchsuchen einer Freispeicherliste mittels eines womöglich dem Problem nicht adäquaten Verfahrens (First/Best-Fit ...). Hinzu kommt, daß bestimmte Verwaltungsalgorithmen die freigegebenen Speicher recht stiefmütterlich behandeln.

Abschlußbemerkung

Wie man wieder mal sieht, kommt es bei der effizienten Programmierung von Problemen weniger auf eine hochoptimierte Befehlsfolge und „geniale" Registerallokation oder eingestreuten ASM-Anweisungen als auf einen intelligenteren Algorithmus an. Die hier vorgestellten Routinen

sollten leicht in Programme eingebunden werden können. Anwenderschnittstellen sind nur die Unterprogramme *freshmem*, *searchmemcntrl* und *freemem*. Der Programmcode hält sich in Grenzen (ca. 512kB) und ist zudem noch portabel. Für alle Programme, die mit einer größeren Anzahl von gleichen Datenstrukturen kämpfen, ist dies im Vergleich zu den Systemroutinen eine bessere Lösung. Wer allerdings extrem optimieren möchte, kann dieses Verfahren mit dem in [1] vorgestellten kombinieren (oder sogar mittels Assembler-Anweisungen implementieren).

Literatur:

[1] Hans-Jürgen Richstein, Memory Manager - Leistungsfähige Speicherverwaltung in Turbo-C, ST-Computer 1990, Nr. 11

[2] A. Esser: TOS Intern, ST-Computer Sonderheft Nr. 2

[3] Sharam Hekmatpour, Lisp - A portable implementation. Prentice Hall 1989

```
/*control-Block for allocation of memory of same size*/

struct memcntrl { /*!*/
   long **memtab;   /* current memtab */
   long memtabidx;  /* points to a memtab entry */
   long memtabsize; /*maximum of entries of memblocks*/
long memblockidx;  /* points to a memblock entry */
long memblocksize; /* number of entries in one block */
unsigned int  memsize;       /* size (> 4) of a memblock
                                entry */
long *memfreelist;/*points to a list of free entries */
long *memblockptr;   /* points to the current block */
long memstatentries; /* Number of used block entries */
long memstatuentries;/* Number of used blocks */
long memstatfentries;/* Number of free blocks */
struct memcntrl *memnext;/* next memory-definition */ }
```

Listing 1

```
struct myrecord { /* das ist meine Datenstruktur */
   struct myrecord *meinZeiger;
   long             meineZahl;
   int              meinIndex; };

typedef struct myrecord *MYrec;/* memory description */

struct memcntrl sysmem;
#define CONVmyrec(x) (Myrec)(x)

MYrecord mkrec ()  /*! make a record object */ {
   register MYrec rec;
      rec = CONVmyrec(freshmem(searchmemcntrl((long)
            (sizeof(MYrec)))));
      /* oder alternativ, wenn es nur einen Record-typ
         gibt:
         rec = CONVmyrec(freshmem(&sysmem));
         wobei in einer Initialisierung sysmem
vorbelegt
         werden muß.
      */
   rec->meinZeiger = NULL; /* irgendwelche
                              Initialisierungen */
   rec->meineZahl = 0;
   rec->meinIndex = 0;
   return(rec);
} /* mkrec */
```

Listing 2

```
long *
freshmem (mem)  /* returns a fresh memory-block */
kernmem mem;
{
   long *memblockptr;

      if (mem->memfreelist != NULL) {
          mem->memstatentries--;
          mem->memstatuentries--;
          mem->memstatfentries--;
          memblockptr = mem->memfreelist;
          mem->memfreelist = (long *)
                             (*mem->memfreelist);
          return(memblockptr);
      }
```

Listing 3

```
 1:  /* kompaktes Beispielprogramm, daß die Anwendung
 2:   * der Speicherverwaltung an einem Beispiel
 3:   * testet.
 4:   * Die Verwaltungsroutinen können ohne weiteres
 5:   * eigene Programme eingebunden werden.
 6:   * Kompiliert mit TURBO-C
 7:   *
 8:   * Klaus Elsbernd
 9:   *
10:   * (c) MAXON Computer 1991
11:   */
12:
13:  #include <stdio.h>
14:  #include <aes.h>
15:  #include <stdlib.h>
16:  #include <string.h>
17:
18:     /* size of cell table */
19:  #define CELLTABSIZE     512
20:     /* Number of Bytes in one block */
21:  #define BLOCKBYTES      8192
22:
23:  /* control-blk for alloc of memory of same size*/
24:  struct memcntrl { /*!*/
25:    long **memtab;          /* current memtab */
26:    long memtabidx;         /* idx in memtabentry */
27:    long memtabsize;        /* max of memblocks */
28:    long memblockidx;       /* idx memblockentry */
29:    long memblocksize;      /* no entries in block */
30:    unsigned int  memsize;/* size memblock entry */
31:    long *memfreelist;      /* points to ... */
32:    long *memblockptr;      /* points to curr blk */
33:    long memstatentries;  /* no allocated entries*/
34:    long memstatuentries; /* no of used entries */
35:    long memstatfentries; /* no of free entries */
36:    struct memcntrl *memnext;/* next memory-def */
37:  };
38:
39:  /* memory description */
40:  typedef struct memcntrl *kernmem;
41:
42:  /* start of memory-control */
43:  struct memcntrl *sysmem;
44:
45:  /* das ist meine Datenstruktur */
46:  struct myrecord {
47:          struct myrecord *meinZeiger;
48:          long             meineZahl;
49:          int              meinIndex;
50:  };
51:
52:  typedef struct myrecord    *MYrec;
53:
54:  #define CONVmyrec(x) (MYrec)(x)
55:
56:  void faterr(char *str),  /* ANSI-C läPt grüPen */
57:       freemem(long size, long *ptr);
58:  char *new(unsigned int size);
59:  kernmem newmemcntrl(long size),
60:          searchmemcntrl(long);
61:  long *collectgarb(void), *freshmem(kernmem);
62:  MYrec mkrec(void);
63:
64:  char err_memory[] =
65:         „[1][No more | memory available!][Abort]";
66:  int phasel = 1;       /* block allocation phase */
67:
68:  void
69:  faterr (str)  /*  prints a fatal error message */
70:  char *str;
71:  {
72:          form_alert(1,str);
73:          exit(1);
74:  } /* faterr */
75:
76:  char *
77:  new (size)  /* ——— allocates 'size' bytes */
78:  unsigned int size;
79:  {
80:     char *res;
81:
82:          /*! alloc minimal 1 byte */
83:          if (!size) size = 1;
84:          if ((res = malloc(size)) == NULL) {
85:             /* terminate block allocation phase */
86:             phasel = 0;
87:             { register kernmem mem = sysmem;
88:               while (mem != NULL) {          →
```

```
89:                 mem->memtabsize =
90:                                 mem->memtabidx + 1;
91:                 mem = mem->memnext;
92:               }
93:             }
94:             collectgarb();
95:             /* try again */
96:             if ((res = malloc(size)) == NULL)
97:               faterr(err_memory);
98:         }
99:         return(res);
100: } /* new */
101:
102: kernmem
103: newmemcntrl (size)  /* new memory control blk */
104: long size;
105: {
106:    kernmem mem = (kernmem)
107:       new((unsigned int)sizeof(struct memcntrl));
108:    long idx = (int)(BLOCKBYTES/size);
109:
110:   mem->memtab = (long **)
111:                 new((unsigned int)
112:                    (CELLTABSIZE*sizeof(long *)));
113:   mem->memtabidx = -1;
114:   mem->memtabsize = CELLTABSIZE;
115:   if (idx == 0) idx = 1;
116:   mem->memblockidx = mem->memblocksize =
117:                                 BLOCKBYTES/size;
118:   mem->memsize = (unsigned int)size;
119:   mem->memfreelist = mem->memblockptr = NULL;
120:   mem->memnext = NULL;
121:   mem->memstatentries = mem->memstatuentries =
122:                         mem->memstatfentries = 0;
123:   return(mem);
124: } /* newmemcntrl */
125:
126: long *
127: freshmem (mem)  /*  returns a fresh memory-blk */
128: kernmem mem;
129: {
130:    long *memblockptr;
131:
132:   if (phase1) { /* in this phase storage is still
133:                  * available */
134:       if (mem->memblockidx++ >=mem->memblocksize)
135:           {       /* get it from current block */
136:
137:           if (++mem->memtabidx < mem->memtabsize)
138:               {/* create a new block */
139:               if ((mem->memtab[mem->memtabidx]
140:                   = mem->memblockptr
141:                   = (long *)malloc((size_t)
142:                                   (mem->memsize *
143:                                mem->memblocksize))
144:                   ) == NULL) {
145:                   mem->memtabsize=mem->memtabidx;
146:                   /* run out of storage ->
147:                    * phase2 */
148:                   goto phase2;
149:               }
150:               memset(mem->memblockptr,'\0',
151:                      (size_t)(mem->memsize *
152:                               mem->memblocksize));
153:               mem->memblockidx = 1;
154:           }
155:           else goto phase2;
156:       }
157:       /* count number of demanded memory-entries*/
158:       ++mem->memstatentries;
159:       ++mem->memstatuentries;
160:       memblockptr = mem->memblockptr;
161:       (char *)mem->memblockptr += mem->memsize;
162:       return(memblockptr);
163: phase2:  /* in this phase storage is exhausted */
164:       phase1 = 0;
165:   }
166:   if (mem->memfreelist == NULL) {
167:       /* try garbage collecting */
168:       if (collectgarb() == NULL)
169:           faterr(err_memory);
170:   }
171:
172:   /*! count number of demanded block entries */
173:   ++mem->memstatuentries;
174:   mem->memstatfentries--;
175:   memblockptr = mem->memfreelist;
176:   mem->memfreelist = (long *)(*mem->memfreelist);
177:   return(memblockptr);
178: } /* freshmem */
179:
180: kernmem
181: searchmemcntrl (size)  /* search for cntrl blk */
182: register long size;
183: {
184:    register kernmem mem = sysmem,
185:             *pmem = &sysmem;
186:
187:   while (mem != NULL && mem->memsize < size) {
188:     /* save a pointer to the last controlblock */
189:     pmem = &mem->memnext;
190:     mem = mem->memnext;
191:   }
192:   if (mem == NULL || mem->memsize != size ){
193:       /* insert new memory */
194:       *pmem = newmemcntrl(size);
195:       (*pmem)->memnext = mem;
196:       mem = *pmem;
197:   }
198:   return(mem);
199: } /* searchmemcntrl */
200:
201: void
202: freemem (size,ptr)  /* free a record im mem-blk*/
203: long size;
204: long *ptr;
205: {
206:    kernmem mem = searchmemcntrl(size);
207:
208:         /* concatenate memory-blocks */
209:         *ptr = (long)mem->memfreelist;
210:         mem->memfreelist = ptr;
211:         mem->memstatfentries++;
212: } /* freemem */
213:
214: long *
215: collectgarb ()  /* ——— collect garbage */
216: {
217:         /* was auch immer hier gemacht wird,
218:          * (wie Müll eingesammelt werden kann),
219:          * es sollte die Freispeicherliste
220:          * zurückgegeben werden */
221:         return(sysmem->memfreelist);
222: } /* collectgarb */
223:
224: MYrec
225: mkrec ()  /*! ——— make a record object */
226: {
227:    register MYrec rec;
228:
229:   rec = CONVmyrec(freshmem(searchmemcntrl((long)
230:                    (sizeof(struct myrecord)))));
231:   /* oder alternativ, wenn es nur ein Record-typ:
232:      rec = CONVmyrec(freshmem(&sysmem)); */
233:   rec->meinZeiger = NULL;
234:   rec->meineZahl = 0;
235:   rec->meinIndex = 0;
236:   return(rec);
237: } /* mkrec */
238:
239: void
240: main ()  /* - Programm test memory-allocation */
241: {
242:    int i, j;
243:    register MYrec startrecord, record;
244:
245:    /* Schleife, die in jedem Durchlauf 387
246:     * Datenstrukturen anfordert und diese
247:     * anschliePend wieder freigibt.
248:     */
249:    for (i = 0; i < 3000; i++) {
250:      for (startrecord = mkrec(),
251:           record = startrecord, j = 0;
252:           j < 387; j++) {
253:        record->meinZeiger = mkrec();
254:        record = record->meinZeiger;
255:      }
256:      for (record = startrecord, j = 0;
257:           j < 387; j++) {
258:        startrecord = record->meinZeiger;
259:        freemem((long)(sizeof(struct myrecord)),
260:                (long *)record);
261:        record = startrecord;
262:      }
263:    }
264: } /* main */
```

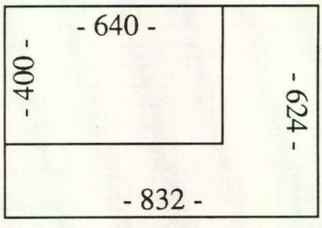
Mega Screen

Ein Monitor für alle Auflösungen

für den Mega ST erweitert niedrige, mittlere und hohe Auflösung, z.B.: 640x350 in mittlerer Auflösung, 832x624 in hoher Auflösung. Die Auflösung kann vom Desktop aus geändert werden. Volle Kompatibilität auch bei Programmen, die nur mit der Original-Auflösung arbeiten, da diese eingestellt werden kann.

Leistungen:

- Erhöhung der darstellbaren Pixelauflösung des ATARI Mega ST in Verbindung mit „Multisync" Monitoren
- Sämtliche Original-ST-Auflösungen mit 75 - 90 Hz (je nach verwendetem „Multisync" Monitor). Somit flimmerfrei auch bei Farbprogrammen
- Bis zu 24 verschiedene Auflösungen zur Auswahl (Auflösung vom Desktop aus änderbar)
- Konfigurationsprogramm zur individuellen Einstellung der verschiedenen Auflösungen
- Umschalten zwischen Farb- und Monochrombetrieb mit Auto-Monitor-Switchbox vom Desktop aus möglich
- Mehr als doppelte Anzahl an Bildpunkten auf dem Bildschirm
- Volle Nutzung des Blitters bei jeder Auflösung
- Integrierter, abschaltbarer Bildschirmschoner
- Auch Spiele laufen, da die Grafikkarte abschaltbar ist
- Läuft mit TOS 1.2 und TOS 1.4
- Getestet in C'T 3/90



Die preiswerte Grafikkarte
DM 298,-

That's PiXel

Nur ein Malprogramm?

Es gibt für den ST ca. 200 „Malprogramme". Davon sind etwa 25 in der Lage, auch Grafiken, die größer als der Bildschirm sind, zu bearbeiten. Die wenigsten erlauben einen bequemen Umgang mit dem Bild. Kein weiterer Kommentar!

- Bildgröße nur durch Speicher begrenzt
- **Bildmanipulation nicht durch Monitor begrenzt**
- Montage mehrerer Bilder, auch aus Fremdprogrammen zu einem großen Bild
- Maus- und Tastaturbedienbar
- Viele praktische Hilfs- und Zeichenfunktionen
- UNDO Puffer (abschaltbar für größte Bilder)
- Automatische Abschaltung der Ganzseitendarstellung, falls das Bild auf den Monitor paßt
- Läuft unter SM124, SM194, OverScan! und mit MegaScreen
- Import verschiedener Bildformate (IMG, PI3, PAC,
- DOO, TPI, PIC,...) die Liste der Treiber wird laufend ergänzt
- Bilder einfach und schnell bearbeiten, die ideale Ergänzung zur Scannersoftware
- Die ideale Ergänzung zu vielen Programmen am Markt

Das Grafiktool mit Konzept!

- **Gleichzeitig** und daher **übersichtlich** bearbeitet man Ganzseitendarstellung, Lupe und 1 : 1-Darstellung
- **Automatisches Verschieben** erspart das sonst mühevolle Scrollen, während man eigentlich zeichnen möchte. Die Geschwindigkeit ist per Tastendruck auf Alternate, Shift und Control wählbar
- **Kürzeste Ladezeit des Programms!** Erspart lästiges Warten beim Programmwechsel
- **Schnelles Scrollen,** Bild laden, sichern, Block kopieren, verschieben,... ermöglicht fließendes Arbeiten. Unkomplizierte, **einfache Bedienung** mit Übersicht
- **Kurze Einarbeitung** wird durch das verständliche Konzept ermöglicht. Ausdruck auf allen gängigen Druckern in verschiedenen Auflösungen möglich. Auf Wunsch Page-Preview vor Druck

– Alle genannten Preise unverbindliche Preisempfehlungen –

DM 148,-

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Hotline
Info + Betreuung
COMPO SOFTWARE GMBH
Ritzstr. 13
5540 Prüm
Tel.: 0 65 51 / 62 66

BESTELL-COUPON

Bitte senden Sie mir: _____ MegaScreen à DM 298,-
_____ That's Pixel à DM 148,-
zzgl. Versandkosten DM 6,- (Ausland DM 10,-) unabhängig von der bestellten Stückzahl

Name, Vorname _____

Straße, Hausnr. _____

PLZ, Ort _____

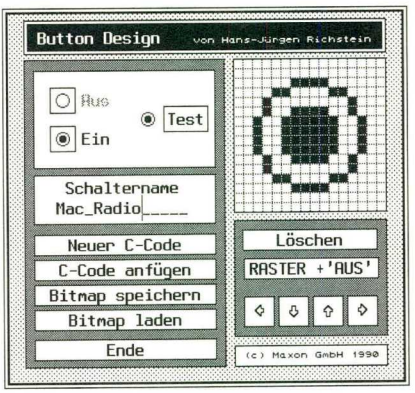
Oder benutzen Sie die in ST-Computer eingeheftete Bestellkarte

In Österreich:
RRR EDV GmbH
Dr. Stumpfstraße 118
A-6020 Innsbruck

In der Schweiz:
Data Trade AG
Landstraße 1
CH - 5415 Rieden-Baden

Den eigenen Knopf durchsetzen

Button-Design-Editor



Mittlerweile gehört es schon zum guten Ton, daß man seiner Benutzeroberfläche auch auf dem ST ein gewisses Macintosh-Flair verpaßt. Kleine Kästchen, die beim Selektieren nur vorsichtig mit einer filigranen Linie durchkreuzt werden, Radio-Buttons, die nicht - wie beim ST sonst üblich - einem Volksempfänger entlehnt scheinen, sondern eher den Miniaturbedienungselementen moderner HiFi-Anlagen entsprechen - alle Bemühungen Ataris bzw. DRIs, dem Anwaltsheer von Apple durch klobige Entstellungen etwaiger Parallelen im Userinterface zu entrinnen, scheinen nun von findigen Programmierern unterlaufen zu werden.

Dabei sind die GEM-Entwickler selbst schuld. Schließlich haben sie diesen geheimnisvollen Objekttyp *USERDEF* (oder manchmal auch *PROGDEF*) geschaffen. Gut, mit dürftiger Dokumentation und völligem Fehlen einer Unterstützung durch irgendein mir bekanntes Resource Construction Set konnte diese Objektart lange Jahre ihr Dasein im Schatten der kantigen Knöpfe fristen. Aber zum Glück kann man mit wenig Programmieraufwand all diese Hindernisse überwinden und seiner Kreativität bei dem Entwurf eigener Knöpfe freien Lauf lassen. Dazu dient zum einen der hier vorgestellte *Button Design*-Editor, mit dem Sie selbst eigene Knöpfe im ein- und ausgeschalteten Zustand entwerfen können, und zum anderen ein kurzes Stückchen C-Code, das Sie in Ihre Programme per *#include* einbinden und das sich dann um diese Knöpfe kümmert. Dabei ändert sich überhaupt nichts an der bisherigen Programmierung des Resource-Verhaltens. Sogar bestehende Programme lassen sich kinderleicht 'umrüsten'.

Arbeitsteilung

Wenn das AES beim Zeichnen der einzelnen Objekte einer Resource auf ein Objekt vom Typ *USERDEF* (user defined) stößt, macht es erst einmal Kaffepause und läßt die Applikation irgendetwas tun. Was das im einzelnen ist, ist dem AES völlig egal. Der Zeiger *ob_spec* eines USERDEF-Objektes zeigt auf eine *USERBLK*-Struktur (Bild 1), in der wiederum ein Zeiger auf eine Routine steht (- stehen muß, sonst gibt es Bomben...). Dorthin wird dann verzweigt, und genau da unterlaufen wir sämtliche Konventionen und malen auf den Schirm, was uns gefällt.

Die zweite Strukturkomponente *ub_parm* ist ein frei verfügbarer Parameter, der zumindest für das AES unwichtig ist, uns jedoch erlaubt, dort beliebige Details über das Objekt abzulegen. Prinzipiell könnte man diese Struktur beliebig mit weiteren Komponenten vergrößern, da man sie sowieso explizit anlegen muß. Wie gesagt, Resource Construction-Programme unterstützen diese Datenstruktur nicht.

Wie man an der Definition in Bild 1 sieht, steht vor dem Funktionszeiger noch die Anweisung *cdecl*. Dies bedeutet, daß unsere Funktion Parameter in GEM-Manier auf dem Stack erhält und nicht, wie sonst in Turbo-C üblich, möglichst in Registern. Der also auf dem Stack mitgegebene Parameter ist ein Zeiger auf eine PARMBLK-(Parameterblock-)Struktur (Bild 2), in der unsere Funktion Informationen über jenes Objekt erhält, mit dem AES selbst nichts zu tun

```
typedef struct
{
  int cdecl (*ub_code)(PARMBLK *parmblock);
                /* Zeiger auf Zeichenfunktion */
  long ub_parm; /* frei verfügbarer Parameter */
} USERBLK;
```

Bild 1: Die 'User-Block'-Struktur

```
typedef struct
{
  OBJECT  *pb_tree;      /* Zeiger auf den
Objektbaum */
  int     pb_obj;        /* Nummer des Objekts  */
  int     pb_prevstate;  /* vorh. Objektstatus  */
  int     pb_currstate;  /* neuer Objektstatus  */
  int     pb_x, pb_y, pb_w, pb_h;   /* Objektkoordinaten   */
  int     pb_xc, pb_yc, pb_wc, pb_hc; /* Clipping-Koordinate */
  long    pb_parm;       /* der Parameter aus der
                            USERBLK-Struktur    */
} PARMBLK;
```

Bild 2: Die Parameterblock-Struktur

haben will. Neben Größe und Lage des Objektes sind dort die Baumadresse und Objektnummer enthalten, weiterhin der Clipping-Bereich für Zeichenroutinen, aktueller und vorheriger Objektstatus und eine Kopie des *ub_parm*-Parameters aus dem Userblock. Letzteres ist zwar überflüssig, da wir über die Objektadresse auch so wieder an diesen Parameter gelangen könnten, aber es vereinfacht zumindest den Zugriff darauf.

Der vorherige Status wird deshalb mitgeliefert, damit man nicht immer das gesamte Objekt neu zeichnen muß. Sind vorheriger und neuer Status identisch, wünscht das AES, daß das gesamte Objekt neu gezeichnet wird, beispielsweise für das Wiederherstellen des gesamten Formularinhaltes nach einer Verschiebeoperation. Unterscheiden sich die beiden Werte jedoch, so braucht auch nur der für selektiertes und deselektiertes Aussehen unterschiedliche Teil neu gezeichnet zu werden. Bei komplexen Objekten kann dies Zeit sparen.

Alles, was unsere Routine nun zu tun hat, ist das Zeichnen des Objektes, das den angegebenen Status (selektiert, umrandet o.ä.) repräsentiert. Das Verwalten des eigentlichen Objektstatus' *ob_state* bleibt weiterhin Aufgabe des AES. Zwei Sachen muß unsere Routine allerdings beachten: Es dürfen keine AES-Aufrufe gemacht werden, da die globalen Arrays mit dem aktuellen Zustand des Baumzeichenvorganges zerstört würden; und wir müssen eine Null zurückliefern und so zumindest vorgeben, alles sei paletti. Etwaige Fehlerrückmeldungen sind zwar im Prinzip durchaus vorgesehen, würden das AES aber so überraschen, daß es die Arbeit sofort einstellen würde.

Auf AES-Routinen können wir leicht verzichten, da in dem Parameterblock bereits die absoluten Bildschirmkoordinaten für die Lage des Objektes angegeben werden. Diese hätten wir sonst ja erst mittels der AES-Funktion *objc_offset* aus den auf das Elternobjekt bezogenen Relativkoordinaten des Objektes ermitteln müssen. Das VDI mit seinen Raster- und Zeichenoperationen können wir jedoch voll nutzen.

Knopfarbeit

Nun könnte man seinen - sagen wir - Macintosh-Knopf mit Kreis- und Linienfunktionen hinmalen, zwei verschiedene Routinen für die zwei gängigen Knöpfe. Möchte man noch andere Knöpfe verwirklichen, kämen noch mehr Routinen hinzu.

Deswegen habe ich mir ein etwas universelleres Verfahren ausgedacht. Mittels des hier vorgestellten *Button Design*-Editors entwirft man zunächst das Aussehen eines Knopfes im eingeschalteten und ausgeschalteten Zustand. Der Editor erzeugt daraus ein C-Listing wie in Bild 3 gezeigt, an das man beliebig viele weitere Knopfdaten anfügen kann. Jeder Knopf wird in einer sog. *SWITCH*-Struktur abgelegt, die aus einer *on*- und einer *off*-Komponente besteht. Als Parameter *ub_parm* trägt man nun einen Zeiger auf die gewünschte *SWITCH*-Struktur ein, so daß das *USERDEF*-Objekt fortan ein entsprechender Schalter ist.

Nur nicht knopflos werden

Alles schön und gut, aber bevor irgendeine eigene Routine zeichnen kann, müssen ja erst einmal die USERDEF-Objekte in der Resource auftauchen. Leider können wir nicht schon beim Entwurf der Resource solch ein Objekt erzeugen. Vielmehr müssen wir einen Platzhalter einsetzen, der charakteristisch für den gewünschten Schalter ist und später dann durch ein USERDEF-Objekt ersetzt wird.

```
/*************************/
/* Schalter-Datenstruktur */
/*************************/
typedef struct
{
  int on[16];
  int off[16];
} SWITCH;
/*************************/
/*    Schalter-Bitmaps    */
/*************************/
SWITCH Runder_Knopf = {
  0x0000,0x03c0,0x0c30,0x1008,0x23c4,0x27e4,0x4ff2,0x4ff2,
  0x4ff2,0x4ff2,0x27e4,0x23c4,0x1008,0x0c30,0x03c0,0x0000,
  0x0000,0x03c0,0x0c30,0x1008,0x2004,0x2004,0x4002,0x4002,
  0x4002,0x4002,0x2004,0x2004,0x1008,0x0c30,0x03c0,0x0000
};
```

Bild 3: Beispielausgabe des 'Button-Design'-Editors

Alles, was unsere Zeichenroutine zu tun hat, ist die Übertragung der dem Objektstatus *selektiert* oder *nicht selektiert* entsprechenden Bitmap an die Objektposition auf dem Bildschirm. Dieselbe kurze Zeichenroutine kann dann beliebig viele Knopftypen bearbeiten, ohne daß eine Zeile Code darin geändert werden muß. Zusätzlich wird auch noch ein Rand um diese Bitmap herum gezeichnet, falls im Objektstatus-Parameter das *OUTLINED*-Bit gesetzt ist.

Dieses Verfahren hat obendrein den Vorteil, daß ohne Änderungen am Programm die bislang benutzten Knöpfe jederzeit durch andere Formen ersetzt werden können - Tapetenwechsel braucht der Mensch. Sie müssen lediglich die Bitmaps austauschen. Dies könnte durch ein kleines Konfigurationsprogramm sogar vom Benutzer eines Programmes selbst vorgenommen werden, so daß er sich mittels eines Button-Editors seine Oberfläche selbst gestalten könnte.

Bild 4 verdeutlicht dieses Verfahren. Da die Bitmaps der Schalter eine Größe von 16 * 16 Pixeln haben, erzeugen wir an der gewünschten Position ein BOXCHAR-Objekt, das sich mittels Raster-Arretierung (Character-Snap) leicht auf eine solche Größe bringen läßt (zwei Rasterpositionen breit und eine hoch). Dieser Objekttyp hat den Vorteil, daß nur die reine OBJECT-Struktur angelegt wird und keine weiteren Daten wie beispielsweise die relativ große TEDINFO-Struktur. Statt eines Zeigers auf eine weitere Struktur finden wir in der Objektspezifikation *ob_spec* direkt den Buchstaben, den man für dieses Objekt eingeben kann. Anhand dieses Buchstabens erkennen wir später den gewünschten Schaltertyp.

In Bild 4 ist dies beispielsweise *o* für den Mac-Radiobutton, *x* für die durchkreuzten Boxen, *e* für einen Ein/Aus-Schalter und *b* für einen kleinen Binärschalter. Diese Buchstaben-Knöpfe können bzw. müssen nun zunächst genauso präpariert werden

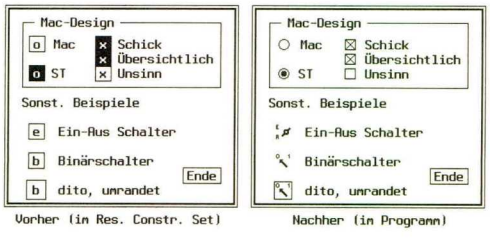


Bild 4: Platzhalter beim Formularentwurf werden später ersetzt

wie sonst auch, d.h. damit sie auch auswählbar sind, muß man *Selectable* einschalten. Man kann sie zusätzlich als Radio-Button deklarieren oder ihnen eine Umrahmung verordnen. Weitere Eigenschaften wie *Crossed*, *Checked* oder *Disabled* werden nicht unterstützt, sind ja aber auch für solche Objekte generell nicht sinnvoll, sondern eher für Knöpfe mit Texten darin gedacht. Für eigene Entwicklungen können diese Flaggen jedoch nützlich sein. So könnte man aufgrund eines gesetzten *Checked*-Bits beispielsweise eine bestimmte Tonfolge beim Umschalten ertönen lassen usw.

Knopf hoch

Jetzt ist es schon fast geschafft. Das Resource-File ist fertiggestellt, so daß bei Programmstart nun nur noch die BOXCHAR-Paten gefunden werden müssen, um sie in USERDEF-Objekte zu verwandeln und mit einer USERBLK-Struktur zu versehen. Schauen Sie sich dazu den Button Manager *BUT_MNGE.C* in Listing 1 an. In Zeile 41 sind die verwendeten Buchstaben der Platzhalter in einer Zeichenkette definiert. In der selben Reihenfolge finden sich die Adressen der zuzuordnenden SWITCH-Strukturen in der Belegung des USERBLK-Arrays in den Zeilen 43-49. Die dazugehörigen Bitmap-Daten wurden in Zeile 18 mit *#include* eingefügt.

Sie müssen nun lediglich die Datei *BUT_MNGE.C* in Ihr Programm einbinden und nach dem Laden der Resource die Funktion *init_buttons* aufrufen, die dann alle Arbeit erledigt. Danach brauchen Sie sich um nichts mehr zu kümmern. Als Parameter erwartet diese Funktion von Ihnen die Kennung der geöffneten VDI-Arbeitsstation und ein Array mit den Zeigern auf die Bäume der Resource, das mit einem Nullzeiger abgeschlossen ist.

Bei der Initialisierung werden die MFDBs (Memory Form Description Block) für die Rasteroperationen initialisiert und für jeden Baum die Funktion *convert_tree* aufgerufen. Diese hangelt sich dann rekursiv durch den Baum und macht BOXCHAR-Objekte ausfindig, die dann mittels der Funktion *find_switch* daraufhin untersucht werden, ob der dazugehörige Buchstabe Repräsentant für einen unserer Schalter ist. Wenn ja, wird dieses Objekt in den Zeilen 85 bis 87 in ein USERDEF-Objekt verwandelt und ein Verweis auf die dem gefundenen Schalter zugeordnete USERBLK-Struktur eingetragen, die den Zeiger auf die Zeichenroutine *draw_procedure* und als Parameter den Zeiger auf die entsprechende Schalterstruktur enthält.

Die eigentliche Zeichenroutine (und mehr wäre nicht nötig, wenn es uns die verfügbaren Resource Construction Programme nicht so schwer machen würden...) finden Sie in den Zeilen 113 bis 162. Als erstes muß das Clipping gesetzt werden, also das Zeichenbegrenzungsrechteck, das im Parameterblock ausdrücklich angegeben wird. Übergibt man einer Funktion wie z.B. *objc_draw* ein Clipping-Rechteck der Ausdehnung 0, wird das aktuelle Clipping verwendet und keines explizit neu eingestellt - und so machen wir es denn auch (Zeile 126).

Danach wird ein kleines Rechteck als Rahmen gezeichnet, falls das *OUTLINED*-Bit im Objektstatus gesetzt ist. Nun wird als Quellraster je nach Objektstatus entweder die Ein- oder Ausschaltkomponente der zugeordneten SWITCH-Struktur ausgewählt. Den Zeiger auf diese Struktur finden wir in der Kopie *pb_parm* des USERBLK-Parameters *ub_parm*. Dann wird mittels *vrt_cpyfm* die Schalter-Bitmap auf den Schirm kopiert - und schon sind wir fertig.

Vorerst verwenden wir als Farben für die gesetzten Pixel der Bitmap den Farbindex 1 und für die gelöschten Pixel den Index 0. Beides ist in den Zeilen 118 und 119 festgelegt, der geneigte Farbenfreund mag sich hier ein individuelleres Verfahren ausdenken, um anders eingestellte Farben der BOXCHARs zu übernehmen. Denkbar wäre ein weiterer Parameter in der USERBLK-Struktur, in dem die in *obspec* angegebenen Farbinformationen gerettet werden, bevor hier der USERBLK-Zeiger eingetragen wird. Allerdings muß das Array dann für jede auftretende Farbkombination eines Schalters einen eigenen Eintrag erhalten, und die *find_switch*-Routine würde etwas aufwendiger.

Montagsdemo

Listing 2 demonstriert, wie einfach der Button Manager in ein eigenes Programm eingebunden werden kann. Dabei sind nur die Zeilen 12,15,21 und 41 bis 47 neu, der Rest ist der Weg, den alle GEM-Applikationen einmal gehen müssen. Es handelt sich hierbei um das Programm zur Darstellung des Formulars aus Bild 4, das vier Schaltertypen beinhaltet.

In Zeile 15 wird der Button Manager eingebunden und in Zeile 21 ein Array zur Aufnahme der Baumadressen und einer Abschluß-Null angelegt. Dieses wird nun in den Zeilen 41 bis 45 initialisiert und dann zusammen mit der Kennung der geöffneten VDI-Arbeitsstation der Funktion *init_buttons* übergeben. Dann ist auch schon alles erledigt, mehr muß man sich und dem Programm nicht antun.

Man verwendet die Knöpfe dann genauso wie andere auch, weswegen man auch eine bestehende Resource leicht umstricken kann und so in den Genuß eines neuen Outfits der Benutzeroberfläche kommt, ohne irgendeine Zeile Code ändern zu müssen. Man kann auch wie gewohnt durch Zurücksetzen des *SELECTABLE*-Bits den Schalter in der momentanen Position sperren - alles wie gehabt.

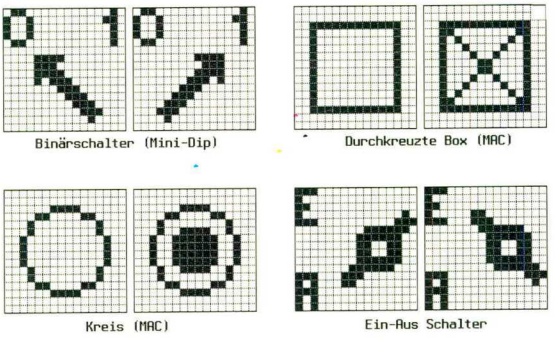


Bild 5: Einige Beispiele für selbstdefinierte Schalter

Knopfüber hinein...

...ins Vergnügen kann man sich nun stürzen, denn mit dem Button Design-Editor kann man seiner Kreativität beim Entwurf eigener Schalter freien Lauf lassen. Einige Beispiele zeigt die Bild 5. Weitere Anregungen wären kleine DIP-Schalter, die aneinandergereiht wie ein echter DIP aussähen und so die praxisnahe (komfortable?) Einstellung irgendwelcher Adressen oder Geräteparameter erlauben würden. Für Freunde römischer Spiele gibt es dann noch die auf- bzw. abwärts zeigenden Daumen usw.

Der Editor ist leicht zu bedienen. Man muß zwei Bitmaps edieren: das Aussehen im eingeschalteten und das im ausgeschalteten Zustand. Die gerade aktive, also im Raster manipulierbare Stellung ist jeweils fett gedruckt, die andere hell. Durch Anklicken einer der Umrahmungen aktivieren Sie die jeweilige Position und ko-

Alles Lesens-Werte

Experimente am Schreibtisch



Durch schnelle Computer kann seit geraumer Zeit die Simulation von Naturvorgängen realistisch vorgeführt werden.
Mit diesem Buch werden Sie in die Lage versetzt, in Ihrem ATARI ST Galaxien kollidieren zu lassen, ohne daß der Himmel einstürzt, oder gar die gewagtesten chemischen Experimente zu riskieren, ohne daß gleich das Haus in die Luft gesprengt wird.
Natürlich darf bei allem Spieltrieb die Theorie der Gesetzmäßigkeiten nicht zu kurz kommen. Ausführlich dokumentierte Listings in GFA-BASIC geben einen Einblick in die programmiertechnische Seite.

Robert Becker
Computersimulationen
337 Seiten und Diskette
DM 59,-
ISBN 3-927065-03-X

Mehr als ein Buch! Mehr als nur Software!

2. überarbeitete Auflage ab Ende Dezember erhältlich

Mittlerweile zählt Scheibenkleister II bereits zu den Standardwerken für den ST. Die Erfolgsautoren Claus Brod und Anton Stepper beschreiben auf fast 900 Seiten alles, was man über Floppies, Festplatten, CD-ROMs und andere Massenspeicher wissen muß. Das Buch beinhaltet einen großen Kursteil, in dem Themen wie Floppy- und Festplattenprogrammierung (über Betriebssystem und direkte Controllerprogrammierung), Aufzeichnungsverfahren, Anschluß von Fremdlaufwerken und sogar Justierung und Reparaturhinweise von Laufwerken u.v.m. Zusätzlich wird eine Diskette mit 1.2 MB Software mitgeliefert, die keine Wünsche offen läßt:
Track- und Sektormonitor, bootfähige RAM-Disk, Hyperformat bringt bis zu 950 kByte, Festplattentreiber CBHD.SYS (Laden von Accessories von beliebigem Laufwerk) u.v.m.



Brod/Stepper
Scheibenkleister II
872 Seiten
und Diskette
DM 89,-
ISBN 3-927065-00-5



Fürs Programmieren unentbehrlich

Die Art und Weise wie benutzererstellte Daten behandelt werden, ob beim Speichern, beim Sortieren oder beim Suchen, ist ein zentrales Thema in der Informatik. Grund genug für Dirk Brockhaus, den verschiedenen Strukturen und Algorithmen auf mehr als 400 Seiten auf den Leib zu rücken.
Angefangen mit den Grundlagen, wie einfachste Strukturelemente, über komplexe Zahlen, sowie verschiedene Methoden für Suchen und Sortieren, bis hin zur Verarbeitung großer Datenmengen umspannt dieses Buch den gesamten Themenbereich.

Dirk Brockhaus
Datenstrukturen
403 Seiten und Diskette
DM 59,-
ISBN 3-927065-02-1

Das Buch zum Handbuch



OMIKRON.BASIC, mittlerweile als ATARI-BASIC bei jedem ST beigefügt, wird in diesem Buch ausführlich beschrieben.
Eine kurze Anleitung für den Neuling führt in die grundlegenden Kennisse der BASIC-Programmierung ein.
Es folgt die Beschreibung der Befehle, Programmierkniffe mit vielen Beispielen, sowie Aufgaben und Lösungen.
Sprites werden erklärt, die Overlaytechnik zeigt das Auslagern langer Programmteile nebst Laden bei Gebrauch.
Was sind Bibliotheken (Libraries)? Das Mitbenutzen fremder Bibliotheksfunktionen, Aufrufe von TOS und GEM, sowie GDOS wird kein Geheimnis bleiben.
Ein weiterer Schwerpunkt ist die Sound- und Grafikprogrammierung unter OMIKRON-Basic, mit Klarheit über Metafiles, IMG-Format usw. Als Spezialität folgt die Beschreibung von internen Multitasking-Befehlen.

Clemens Hoffmann
OMIKRON.BASIC 3.0
355 Seiten
und Diskette
DM 59,-
ISBN 3-927065-01-3

MAXON

Bestellcoupon MAXON Computer GmbH Schwalbacherstr. 52 6236 Eschborn Tel.: 06196/481811

Name/Vorname

Straße

Ort

Unterschrift

Hiermit bestelle ich:

- ☐ SCHEIBENKLEISTER II DM 89,00
- ☐ COMPUTER-SIMULATIONEN DM 59,00
- ☐ OMIKRON.BASIC 3.0 DM 59,00
- ☐ DATENSTRUKTUREN DM 59,00

Versandkosten: Inland DM 7,50
Ausland DM 10,00

Auslandbestellungen **nur** gegen Vorauskasse
Nachnahme zuzgl. DM 4,00 Nachnahmegebühr.

- ☐ Vorauskasse
- ☐ Nachnahme

Vertrieb in der Schweiz: DTZ DataTrade AG Langstrasse 94 Postfach 413 CH-8021 Zürich Tel.: 01/242 80 88 Fax.: 01/291 05 07
Vertrieb in Österreich: Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling Tel.: 06244/7081-17 Fax.: 06244/7188-3
Vertrieb in Frankreich: AROBACE 2 Rue Piemontesis F-75018 Paris Tel: 1/42235044 Fax 1/42545631

pieren sie ins Raster. Pixel werden wechselweise gelöscht oder gesetzt, wobei ein Modus bei gedrückter Taste erhalten bleibt, so daß man dann kontinuierlich malen oder löschen kann.

Mit den Pfeil-Knöpfen läßt sich das gesamte Raster - wer hätte es erahnt - in der korrespondierenden Richtung verschieben. *Löschen* erzeugt eine Bezierkurve nach einem speziellen Algorithmus...- äh - nein, ich höre gerade, *Löschen* löscht lediglich das Raster!

Raster + Aus bzw. *Raster + Ein* "oder"t die Bitmap der gerade inaktiven Schalterstellung in das Raster. Dies ist recht praktisch, da meistens nur einige Teile gegenüber der anderen Position zur Veränderung anstehen.

Durch Anklicken der Taste *Test* kann man sich den Schalter in Aktion anschauen, indem beide Bitmaps wechselweise in den Bereich neben der Taste kopiert werden. Nun muß man sich noch für einen Schalternamen entscheiden, unter dem eine SWITCH-Struktur angelegt wird. Die ganze Herrlichkeit kann man zunächst als Bitmap speichern, um sie so später noch einmal modifizieren zu können. Dazu erzeugt man eine Datei mit dem Extender *BBM* (Button BitMap), in der zunächst 16 Bytes *On*-Bitmap abgelegt werden, gefolgt von 16 Bytes *Off*-Bitmap und 14 Bytes Schalternamen.

Als nächstes erzeugt man einen C-Code mit Extender *BTN* (BuTtoN). Ist es der erste Schalter, so benutzt man *Neuer C-Code*, wobei vor den eigentlichen Daten noch die Datenstruktur *SWITCH* in eine neue Datei geschrieben wird, so wie in Bild 4. Weitere Schalter fügt man mit *C-Code anfügen* an, wobei nur die Schalterdefinition selbst an eine bestehende Datei angehängt wird. Man kann diesen Editor übrigens prima auch als Mausform-Editor verwenden!

Knopfschmerzen

Vor den Erfolg haben die Götter das Listing 3, 4 und 5 gestellt. Die Resource brauchen Sie nicht selbst zu erstellen, da diese für das Pixel-Raster sehr genau dimensioniert sein muß, was eine ziemliche Fummelei ist. Listing 4 enthält eine decompilierte Fassung und ist auch nicht besonders lang. Dazu gehören noch einige Definitionen in Listing 5.

Kühlen Knopf bewahren

Obwohl es nun so herrlich einfach ist, eine Vielzahl von Knöpfen in die Benutzeroberfläche seiner Programme zu integrieren, gilt wie so oft: weniger ist mehr. Eine Überhäufung mit unterschiedlichen Knöpfen verwirrt Benutzer nur. Hinter der Verwendung bestimmter Formen muß eine klare Konzeption stecken. So werden beispielsweise beim Macintosh die bekannten runden Knöpfchen immer nur als Radio-Buttons verwendet, die kleinen durchkreuzbaren Quadrate (*Check-Buttons*) immer für unabhängig anwählbare Parameter. Solche Prinzipien sollte sich jeder auch auf dem ST zu eigen machen.

Hans-Jürgen Richstein

Literatur:

Aumiller/Luda/Möllmann: GEM-Programmierung in C, Markt&Technik Verlag 1987

Jankowski/Reschke/Rabich: Atari ST Profibuch, Sybex-Verlag 1989

Szczepanowski/Günther: Das große GEM-Buch zum Atari ST, Data Becker 1985

```
 1:  /*********************************************/
 2:  /* BUT_MNGE.C                                */
 3:  /* 'Button Manager'  zum Verwalten selbstent-*/
 4:  /* worfener Knöpfe in Resourcen              */
 5:  /*                                           */
 6:  /* Übersetzt mit Turbo C 2.03                */
 7:  /*                                           */
 8:  /* von Hans-Jürgen Richstein                 */
 9:  /*      (c) 1990 Maxon Computer GmbH         */
10:  /*********************************************/
11:
12:  #include <vdi.h>
13:  #include <aes.h>
14:
15:  /* Die vom 'Button-Design' Editor erzeugten Daten
16:   und die Struktur SWITCH werden hier eingefügt */
17:
18:  #include "switches.btn"
19:
20:  #define RASTERBREITE 16
21:  #define RASTERHOEHE 16
22:  #define RASTER_WORTE 16
23:
24:  #define COL_SET 1
25:  #define COL_RESET 0
26:
27:  /*------- Prototypen ----------*/
28:  void init_buttons(int v_handle,OBJECT *trees[]);
29:  void convert_tree(OBJECT *tree,unsigned int ob);
30:  int find_switch(char pate);
31:  int cdecl draw_procedure(PARMBLK *pb);
32:  /*-----------------------*/
33:
34:  int vdi_handle; /* Für die Zeichen- und Raster-*/
35:  MFDB screen_mfdb,bitmap_mfdb; /* operationen   */
36:
37:  /* Hier werden nun die Platzhalter in einem
38:     String eingetragen und in gleicher Reihenfolge
39:     folgen die dazugehörigen User-Blocks */
40:
41:  char paten[] = "oxbe";  /* 4 Schalterarten */
```

```
42:
43:  USERBLK user_defined[] =
44:  {
45:    draw_procedure,(long) &Mac_Radio,
46:    draw_procedure,(long) &Mac_Button,
47:    draw_procedure,(long) &Binaerschalter,
48:    draw_procedure,(long) &Ein_Aus
49:  };
50:  /*-------------------------*/
51:  void init_buttons(int v_handle,OBJECT *trees[])
52:  {
53:    int i = 0;
54:    screen_mfdb.fd_addr = NULL; /* = Bildschirm */
55:    bitmap_mfdb.fd_w = RASTERBREITE;
56:    bitmap_mfdb.fd_h = RASTERHOEHE;
57:    bitmap_mfdb.fd_wdwidth = 1;
58:    bitmap_mfdb.fd_stand = 1;
59:    bitmap_mfdb.fd_nplanes = 1;
60:    vdi_handle = v_handle;
61:    while (trees[i]) convert_tree(trees[i++],0);
62:  }
63:  /*-------------------------*/
64:  void convert_tree(OBJECT *tree,unsigned int ob)
65:  {
66:   /* Durchsucht alle Kinder von 'ob' nach BOXCHARs
67:   die als Pate für die neuen Buttons stehen und
68:   substituiert diese mit einem Zeiger auf die da-
69:   zugehörige USERBLK-Struktur */
70:
71:    static int schalter;
72:    int akt_child = tree[ob].ob_head;
73:
74:    while (akt_child != ob)
75:    {
76:      if (tree[akt_child].ob_type == G_BOXCHAR)
77:      {     /* Boxchar ? */
78:
79:        schalter = find_switch(tree[akt_child].
80:                               ob_spec.obspec.character);
81:
82:        if (schalter >= 0)
```

→

```
83:       {            /* So einen haben wir ! */
84:
85:         tree[akt_child].ob_type = G_USERDEF;
86:         tree[akt_child].ob_spec.userblk =
87:                         &user_defined[schalter];
88:       }
89:     }
90:     if (tree[akt_child].ob_head >= 0)
91:       convert_tree(tree,akt_child); /* Rekursiv
92:                                        durchhangeln */
93:     akt_child = tree[akt_child].ob_next;
94:   }
95: }
96: /*-------------------------*/
97: int find_switch(char pate)
98: {
99:  /* Sucht die zu einem Buchstaben gehörende
100:     Schalterstruktur. Liefert -1, wenn für diesen
101:     Buchstaben kein Button vorhanden ist. */
102:
103:   char *cur = paten;
104:   int ctr = 0;
105:   while (*cur)
106:   {
107:     if (*cur == pate) return ctr;
108:     cur++; ctr++;
109:   }
110:   return -1;
111: }
112: /*-------------------------*/
113: int cdecl draw_procedure(PARMBLK *pb)
114: {
115:   /* Dies ist die eigentliche Zeichenroutine */
116:   int pxy[8], col_index[2];
117:
118:   col_index[0] = COL_SET;
119:   col_index[1] = COL_RESET;
120:
121:   /* Evt. Clipping setzen */
122:   pxy[0] = pb->pb_xc; pxy[1] = pb->pb_yc;
123:   pxy[2] = pxy[0] + pb->pb_wc - 1;
124:   pxy[3] = pxy[1] + pb->pb_hc - 1;
125:
126:   if (pxy[2] || pxy[3])
127:     vs_clip(vdi_handle,1,pxy);
128:
129:   if (pb->pb_currstate & OUTLINED) /* Rahmen ? */
130:   {
131:     pxy[0] = pb->pb_x - 2; pxy[1] = pb->pb_y - 2;
132:     pxy[2] = pxy[0] + RASTERBREITE + 3;
133:     pxy[3] = pxy[1] + RASTERHOEHE + 3;
134:
135:     vsl_color(vdi_handle,COL_SET);
136:     vsf_interior(vdi_handle,FIS_HOLLOW);
137:     vsf_perimeter(vdi_handle,1);/* Rand sichtb.*/
138:     v_bar(vdi_handle,pxy);
139:   }
140:                   /* Ein- oder Ausschalter */
141:   bitmap_mfdb.fd_addr =
142:       (char *) ((pb->pb_currstate & SELECTED) ?
143:                 ((SWITCH *) (pb->pb_parm))->on :
144:                 ((SWITCH *) (pb->pb_parm))->off);
145:
146:   pxy[0] = pxy[1] = 0; /* Quellrasterkoord. */
147:   pxy[2] = RASTERBREITE - 1;
148:   pxy[3] = RASTERHOEHE - 1;
149:
150:   pxy[4] = pb->pb_x; pxy[5] = pb->pb_y;
151:   pxy[6] = pxy[4] + RASTERBREITE - 1; /* Ziel- */
152:   pxy[7] = pxy[5] + RASTERHOEHE - 1;  /* koord.*/
153:
154:   v_hide_c(vdi_handle);    /* Maus aus */
155:
156:   vrt_cpyfm(vdi_handle,MD_REPLACE,pxy,
157:             &bitmap_mfdb,&screen_mfdb,col_index);
158:
159:   v_show_c(vdi_handle,1); /* und wieder an */
160:
161:   return 0;    /* Sehr wichtig !!! */
162: }
```

Listing 1: Der Laufzeitcode

```
1:  /*********************************************/
2:  /* Demo zur Verwendung des 'Button Managers' */
3:  /*                                           */
4:  /* von Hans-Jürgen Richstein                 */
5:  /*      (c) 1990 Maxon Computer GmbH         */
6:  /*********************************************/
7:
8:  #include <vdi.h>
9:  #include <aes.h>
10: #include <stddef.h>
11:
12: #define NUM_TREES 1 /* Anzahl der Bäume im RCS */
13:
14: #include "demo.h" /* Resource-Definitionen */
15: #include "but_mnge.c" /* Button-Manager */
16:
17: int  appl_id,msgbuf[8],v_handle, /* Gem-Krams */
18:    work_in[] = {1,1,1,1,1,1,1,1,1,1,2},
19:    work_out[57],dummy,x,y,w,h;
20:
21: OBJECT *trees[NUM_TREES + 1]; /* Anzahl der Bäume
22:                                  + Abschluss-Null */
23: /*-------------------------*/
24: int main(void)
25: {
26:   unsigned int i;
27:
28:   appl_id =  appl_init();
29:   if (appl_id != -1)
30:   {
31:     v_handle = graf_handle(&dummy,&dummy,
32:                            &dummy,&dummy);
33:     v_opnvwk(work_in,&v_handle,work_out);
34:     if (v_handle != 0)
35:     {
36:       graf_mouse(ARROW,0);
37:       if (rsrc_load("demo.rsc"))
38:       {
39:         /* Array mit den Baumadressen belegen */
40:
41:         for (i = 0;i < NUM_TREES; i++)
42:           rsrc_gaddr(0,i,&trees[i]);
43:
44:         /* Array-Abschluss mit Nullzeiger */
45:         trees[NUM_TREES] = NULL;
46:
47:         init_buttons(v_handle,trees); /* Jetzt
48:                                          passiert's */
49:
50:         /* Hier geht es dann ganz normal weiter,
51:           als wenn es keine neuen Buttons gäbe */
52:
53:         form_center(trees[0],&x,&y,&w,&h);
54:         form_dial(FMD_START,0,0,0,0,x,y,w,h);
55:         objc_draw (trees[0],0,MAX_DEPTH,x,y,w,h);
56:         form_do(trees[0],-1);
57:         form_dial(FMD_FINISH,0,0,0,0,x,y,w,h);
58:       }
59:       v_clsvwk(v_handle);
60:     }
61:     appl_exit();
62:   }
63:   return (0);
64: }
```

Listing 2: Anwendungsbeispiel

```
1:  /*********************************************/
2:  /* BUT_DSGN.C                                */
3:  /* 'Button Designer' zum Entwurf eigener     */
4:  /* Knöpfe für Resourcen                      */
5:  /*                                           */
6:  /* Übersetzt mit Turbo C 2.03                */
7:  /*                                           */
8:  /* von Hans-Jürgen Richstein                 */
9:  /*      (c) 1990 Maxon Computer GmbH         */
10: /*********************************************/
11:
12: #include <vdi.h>
13: #include <aes.h>
14: #include <stdio.h>
15: #include <tos.h>
16: #include <string.h>
17:                                          →
```

```
#include "but_dsgn.h"
#include "but_dsgn.rsh" /* Resource File
                           Dekompilat, erzeugt
                           mit "RSC2CSRC" von
                           Erik Dick,Kaisers-
                           lautern              */
/*------- Prototypen ----------*/

void hauptprogramm(void);
void handle_form(int sel_obj);
void redraw(int tree_nr,int obnr);
void edit_on_switch(void);
void edit_off_switch(void);
void save_bitmap(void);
void load_bitmap(void);
void save_c_code(char *mode);
void make_header(FILE *dest);
void make_raster(void);
void clear_raster(void);
void copy_to_raster(void);
void move_up(void);
void move_down(void);
void move_left(void);
void move_right(void);
void switch_test(void);
void flip_pixel(int mx, int my);
void copy_bitmap(char *src,int obj);
void main_init(void);

/*------- Einige Makros ----------*/

#define Pfeil graf_mouse(ARROW,0)
#define Biene graf_mouse(BUSYBEE,0)
#define TRUE 1
#define FALSE 0

#define PIXELBREITE 7  /* Im Mal-Raster */
#define RASTERBREITE 16
#define RASTERHOEHE 16
#define RASTER_WORTE 16

/*-------------------------*/
int  appl_id,msgbuf[8],vdi_handle,/* GEM Krams */
     work_in[] = {1,1,1,1,1,1,1,1,1,1,2},
     work_out[57],dummy,mouse_x,mouse_y;

char path[128],   /* Puffer für Pfadnamen */
     f_name[14] = "",
     *schaltername;

unsigned int *on_bitmap,*off_bitmap,
    *switch_bitmap; /* Die akt. Switch-Bitmap */

int cur_switch,     /* Das akt. Switch-Obj.   */
    test_status;
    color_index[] = {1,0};

enum {TOGGLE,ON,OFF} drawing_mode;

MFDB screen_mfdb,bitmap_mfdb;

/*-------- Alert - Texte --------*/
char noname_alert[] = "[1][ |"
                      " Es wurde noch kein Name für|"
                      " diesen Schalter eingegeben!]"
                      "[ Ok ]";
/*-------------------------*/
int main(void)
{
  int pxy[4];

  appl_id = appl_init();  /* Standard Initiali-
                             sierung im GEM */
  if (appl_id != -1)
  {
    vdi_handle = graf_handle(&dummy,&dummy,
                             &dummy,&dummy);
    v_opnvwk(work_in,&vdi_handle,work_out);
    if (vdi_handle != 0)
    {
      /* Default - Clipping setzen */
      pxy[0] = pxy[1] = 0;
      pxy[2] = work_out[0];
      pxy[3] = work_out[1];
      vs_clip(vdi_handle,1,pxy);

      main_init(); /* Erst mal 'klar Schiff' */
      Pfeil;
      hauptprogramm();

      v_clsvwk(vdi_handle);
    }
    appl_exit();
  }
  return (0);
}
/*-------------------------*/
void hauptprogramm(void)
{
  int x,y,w,h,sel_obj,m_button;

  form_center(TREE[MAINFORM],&x,&y,&w,&h);
  form_dial(FMD_START,0,0,0,0,x,y,w,h);
  form_dial(FMD_GROW,x+w/2,y+h/2,10,10,x,y,w,h);
  redraw(MAINFORM,0);
  make_raster();

  do
  {
    graf_mkstate(&dummy,&dummy,&m_button,&dummy);
    if (!m_button) drawing_mode = TOGGLE;

    sel_obj = form_do(TREE[MAINFORM],
                (drawing_mode == TOGGLE) ?
                                  SWTCHNME : -1);
    graf_mkstate(&mouse_x,&mouse_y,
                                &dummy,&dummy);
    sel_obj &= 0x7fff; /* Doppelklicks
                                  ignorieren */
    handle_form(sel_obj);

  } while(sel_obj != ENDE);

  form_dial(FMD_SHRINK,x+w/2,y+h/2,10,10,
                                  x,y,w,h);
  form_dial(FMD_FINISH,0,0,0,0,x,y,w,h);
}
/*-------------------------*/
void handle_form(int sel_obj)
{
/* Verzweigt aufgrund des selektierten Objektes
   in die entsprechenden Routinen.            */

  switch (sel_obj)
  {
    case RASTER   : flip_pixel(mouse_x,mouse_y);
                    break;
    case BUTNEIN  :
    case BUTNAUS  : if (sel_obj == BUTNAUS)
                      edit_off_switch();
                    else
                      edit_on_switch();
                    make_raster();
                    redraw(MAINFORM,EINTEXT);
                    redraw(MAINFORM,AUSTEXT);
                    redraw(MAINFORM,ORSWITCH);
                    break;
    case CLEARALL : clear_raster(); break;
    case ORSWITCH : copy_to_raster(); break;
    case MOVEUP   : move_up(); break;
    case MOVEDOWN : move_down(); break;
    case MOVELEFT : move_left(); break;
    case MOVERGHT : move_right(); break;
    case TEST     : switch_test(); break;
    case NEWCODE  : save_c_code("w"); break;
    case APPEND   : save_c_code("a"); break;
    case SAVEIMG  : save_bitmap(); break;
    case LOADIMG  : load_bitmap(); break;
  }

  switch (sel_obj)
  {
   /* Raster-Update nach diesen Aufrufen */

    case CLEARALL: case ORSWITCH: case MOVEUP:
    case MOVEDOWN: case MOVELEFT: case MOVERGHT:

        make_raster();
        copy_bitmap((char *) switch_bitmap,
                                  cur_switch);
        break;

   /* Komplett-Update nach diesen Aufrufen */
```

→

RIEMANN

Symbolisches Algebra- und Programmiersystem

RIEMANN ist ein außergewöhnlich flexibles Algebraprogramm mit Numerik- und Grafikroutinen und einer eigenen Programmiersprache für ATARI ST-Computer.

Symbolische Mathematik:
Algebra, beliebig genaue rationale Arithmetik, Lösen von Gleichungen und lin. Gleichungssystemen; trigonometrische und hyperbolische Funktionen; Differentiation, Integration, Grenzwerte; Summen- und Produktbildung, Vektor- und Matrixoperationen.

'Formula Modelling': neue einzigartige Methode zur Manipulation mathematischer Ausdrücke mit der Maus; komplette hochgenaue Fließkommaarithmetik; Grafikfunktionen; Pattern Matching (Mustererkennung); eingebauter Volleditor.

RIEMANN ist gleichzeitig eine leistungsfähige interaktive symbolverarbeitende LISP-ähnliche Programmiersprache mit leicht erlernbarer Syntax.

RIEMANN ist kompatibel zu dem berühmten Algebraprogramm muMATH-83/ muSIMP-83.

mitgelieferte Zusatzpakete für Debugging, Differentialgleichungen, Vektoranalysis und allgemeine Relativitätstheorie (Tensorrechnung), sowie eigener PD-Pool für RIEMANN-Routinen unserer Kunden.

Sehr günstiges Wartungs- und Updateabonnement, bester Service bei Problemen oder Fragen.

Testberichte c't 1/90, ST-Computer 4/90, TOS-Magazin 5/90, PD-Journal 7/8-90.

RIEMANN kostet 238.- DM, gegen Nachweis für Schüler und Studenten sogar nur 168.- DM. Der Versandkostenanteil beträgt 5.50 DM.

Bestellungen mit Verrechnungsscheck oder gegen Nachnahme (zzgl. Nachnahmegebühr) an

SOFTWARE B&N ENTWICKLUNG

Begemann & Niemeyer
Softwareentwicklung GbR
Göllnitzer Str. 12
7500 Karlsruhe 41
Tel. 0721 / 40 47 03

Oder bestellen Sie sich einfach eine automatische Demonstrationsdiskette für DM 10,-. Fordern Sie auch unsere ausführliche, kostenlose Informationsschrift an.

GALACTIC

CHAOS!

Das kennen Sie doch auch?

Irgendwo in den 2773 Dateien auf der Dokumenten-Partition haben Sie die Unterlagen, in denen etwas zum Geschäftsabschluß steht.

Oder: in welchem verd... Modul ihres Riesenprogramms haben Sie die Funktion xy_ungeloest(????) definiert?





Diese und viele andere Suchprobleme löst jetzt RETRIEVE.

Dateien, Ordner oder ganze Partitionen (natürlich auch Disketten) nach Stichwörtern durchsuchen, Querverweistabellen aufbauen und die gefundenen Dateien in einen integrierten komfortablen Editor laden, das alles kann RETRIEVE.

Die geladenen Dateien können editiert, neu gespeichert und gedruckt werden, ganz und blockweise.

RETRIEVE ist für uns zum unentbehrlichen Hilfsmittel geworden!

und das alles für 69 DM!

RETRIEVE ist ein Accessorie für hohe und mittlere Auflösung.

Versandbedingungen: Inland 8.00 DM Nachnahme, 5.00 DM Vorkasse
Ausland: nur Vorkasse +10 DM Porto/Verpackung

GALACTIC - Stachowiak, Dörnenburg und Raeker GbR
Burggrafenstr. 88 - 4300 Essen 1
Tel.:0201/27 32 90 oder 710 18 30- Fax:0201/710 19 50

Preise gültig ab 15.01.91

NEU
Einbaufestplatten für MEGA ST

AF-3 188.-
3,5'Floppy,720KB,TEAC

AF-5 218.-
5.25'Floppy,720/360KB,TEAC

AHS-50 998.-
49MB, 600KB/s, 40ms, Seagate

AHS-85 1298.-
84MB, 600KB/s, 24ms, Seagate

AHS-52L 1148.-
52MB, 950 KB/s, 12ms,Quantum

AHS-105L 1648.-
105MB,1050 KB/s 12ms Quantum

Einbaufestplatten für MEGA ST
DMA weiter nutzbar

AHM-52L 998.-
52MB, 950 KB/s, 12ms,Quantum

AHM-105L 1498.-
105MB,1050 KB/s 12ms Quantum

AHS-2000 2548.-
210MB,1100KB/s,10ms, Quantum

AHS-3000 3998.-
330MB, 1150KB/s,17.5ms,Imprimis

AHS-7000 5498.-
660MB, 1250KB/s,15.5ms,Imprimis

AHW-44 1498.-
44MB, 550 KB/s, 20ms, Wechselplatte incl. Medium, SyQuest

HW-SQ400 198.-
Wechselmedium 44 MB

AP-Speed 598.-
ICD 16 MHz - Erweiterung

AP-ASPME 198.-
Speichererweiterung STE auf 2 MB

2 Jahre Garantie

unschlagbarer Service

schnelle Lieferzeit

schnelle Massenspeicher zu günstigen Preisen

VOM SPEZIALISTEN
Festplatten und Laufwerke

TRINOLOGY EMS

Alle hier angebotenen Produkte sind komplett anschlußfertig. Auf Systeme mit Quantum- bzw. SyQuest-Laufwerken geben wir 2 Jahre Garantie, andere 12 Monate.

FSE
Schmiedstraße 11
D-6750 Kaiserslautern
Tel. 0631 / 67096-98
Fax: 0631 / 60697
Händleranfragen erwünscht
FRANK STRAUSS ELEKTRONIK

```
192:      case NEWCODE: case APPEND:
193:      case SAVEIMG: case LOADIMG:
194:          redraw(MAINFORM,0); make_raster();
195:          break;
196:    }
197:  }
198:  /*-------------------------*/
199:  void main_init(void)
200:  {
201:    objc_init(); /* Nötig, da wir die Resource
202:                    nicht mit 'rsc_load' laden */
203:
204:    /* Formular und Variablen vorbelegen    */
205:
206:    path[0] = Dgetdrv() + 'A'; /* Aktuellen Pfad */
207:    path[1] = ':';             /* ermitteln      */
208:    Dgetpath(&path[2],0);
209:    strcat(path,"\\");
210:
211:    /* Diese Zeiger sparen Schreibarbeit */
212:
213:    on_bitmap = (unsigned int *)
214:                    TREE[MAINFORM][BUTNEIN]
215:                        .ob_spec.bitblk->bi_pdata;
216:    off_bitmap = (unsigned int *)
217:                    TREE[MAINFORM][BUTNAUS]
218:                        .ob_spec.bitblk->bi_pdata;
219:    schaltername = TREE[MAINFORM][SWTCHNME]
220:                        .ob_spec.tedinfo->te_ptext;
221:
222:    schaltername[0] = 0; /* Eingabefeld löschen */
223:    edit_off_switch(); /* Startedierung ist der
224:                          deselektierte Schalter */
225:    drawing_mode = TOGGLE;
226:
227:    screen_mfdb.fd_addr = NULL; /* = Bildschirm */
228:    /* Quelle für Rasterkopien */
229:    bitmap_mfdb.fd_w = RASTERBREITE;
230:    bitmap_mfdb.fd_h = RASTERHOEHE;
231:    bitmap_mfdb.fd_wdwidth = 1;
232:    bitmap_mfdb.fd_stand = 1;
233:    bitmap_mfdb.fd_nplanes = 1;
234:  }
235:  /*-------------------------*/
236:  void redraw(int tree_nr,int obnr)
237:  {
238:  /* Zeichnet Obj.'obnr' in Baum 'tree_nr' neu */
239:
240:    objc_draw(TREE[tree_nr],obnr,MAX_DEPTH,
241:              0,0,0,0); /* = Default-Clipping */
242:  }
243:  /*-------------------------*/
244:  void clear_raster(void) /* Raster löschen */
245:  {
246:    int i;
247:    for (i = 0; i < RASTER_WORTE;
248:                          switch_bitmap[i++] = 0);
249:  }
250:  /*-------------------------*/
251:  void copy_to_raster(void) /* Jeweils anderen   */
252:  {                         /* Knopf hereinodern */
253:    unsigned int i,*source;
254:    if (switch_bitmap == on_bitmap)
255:      source = off_bitmap;
256:    else
257:      source = on_bitmap;
258:    for (i = 0; i < RASTER_WORTE;
259:                    switch_bitmap[i] |= source[i++]);
260:  }
261:  /*-------------------------*/
262:  void move_up(void) /* Ein Pixel nach oben */
263:  {
264:    int i;
265:    for (i = 0;i < RASTER_WORTE - 1;i++)
266:          switch_bitmap[i] = switch_bitmap[i + 1];
267:    switch_bitmap[RASTER_WORTE - 1] = 0;
268:  }
269:  /*-------------------------*/
270:  void move_down(void) /* Ein Pixel nach unten */
271:  {
272:    int i;
273:    for (i = RASTER_WORTE - 1;i > 0;i--)
274:          switch_bitmap[i] = switch_bitmap[i - 1];
275:    switch_bitmap[0] = 0;
276:  }
277:  /*-------------------------*/
278:  void move_left(void) /* Ein Pixel nach links */
279:  {
280:    int i;
281:    for (i = 0;i < RASTER_WORTE;
282:                          switch_bitmap[i++] <<= 1);
283:  }
284:  /*-------------------------*/
285:  void move_right(void) /* Ein Pixel nach rechts */
286:  {
287:    int i;
288:    for (i = 0;i < RASTER_WORTE;
289:                          switch_bitmap[i++] >>= 1);
290:  }
291:  /*-------------------------*/
292:  void switch_test(void)
293:  {
294:    /* Kopiert wechselweise den selektierten oder
295:       deselektierten Schalter ins Testfeld. */
296:    int dum,mb;
297:    if (test_status)
298:    {
299:      test_status = 0;
300:      copy_bitmap((char *) on_bitmap,TESTAREA);
301:    }
302:    else
303:    {
304:      test_status = 1;
305:      copy_bitmap((char *) off_bitmap,TESTAREA);
306:    }
307:    do
308:    { /* Und auf Loslassen der Maustaste warten */
309:      graf_mkstate(&dum,&dum,&mb,&dum);
310:    } while(mb);
311:  }
312:  /*-------------------------*/
313:  void edit_on_switch(void)
314:  {
315:    /* Schaltet auf Edierung des selektierten
316:       Schalters um (ohne 'redraw') */
317:    cur_switch = BUTNEIN;
318:    switch_bitmap = on_bitmap;
319:    TREE[MAINFORM][AUSTEXT].ob_state = DISABLED;
320:    TREE[MAINFORM][EINTEXT].ob_state = NONE;
321:    TREE[MAINFORM][ORSWITCH].ob_spec.free_string
322:                                  = "RASTER +'AUS'";
323:  }
324:  /*-------------------------*/
325:  void edit_off_switch(void)
326:  {
327:    /* Schaltet auf Edierung des deselektierten
328:       Schalters um (ohne 'redraw') */
329:    cur_switch = BUTNAUS;
330:    switch_bitmap = off_bitmap;
331:    TREE[MAINFORM][EINTEXT].ob_state = DISABLED;
332:    TREE[MAINFORM][AUSTEXT].ob_state = NONE;
333:    TREE[MAINFORM][ORSWITCH].ob_spec.free_string
334:                                  = "RASTER +'EIN'";
335:  }
336:  /*-------------------------*/
337:  void make_raster(void)
338:  {
339:    /* Zeichnet in den Button 'RASTER' ein Gitter
340:       und füllt alle Punkte, die in der aktuellen
341:       'switch_bitmap' gesetzt sind */
342:    int pxy[4],x_min,y_min,x_max,y_max,mask,x,y;
343:
344:    objc_offset(TREE[MAINFORM],RASTER,
345:                                &x_min,&y_min);
346:
347:    x_max = x_min+TREE[MAINFORM][RASTER].ob_width;
348:    y_max = y_min+TREE[MAINFORM][RASTER].ob_height;
349:
350:    vsl_udsty(vdi_handle,0xaaaa); /* Punktiert */
351:    vsl_type(vdi_handle,7);
352:    v_hide_c(vdi_handle);
353:
354:    pxy[0] = x_min + PIXELBREITE; pxy[1] = y_min;
355:    pxy[2] = pxy[0]; pxy[3] = y_max - 1;
356:
357:    while (pxy[0] < x_max)
358:    {
359:      v_pline(vdi_handle,2,pxy);
360:      pxy[0] += PIXELBREITE + 1;
361:      pxy[2] += PIXELBREITE + 1;
362:    }
363:
364:    pxy[0] = x_min; pxy[1] = y_min + PIXELBREITE;
365:    pxy[2] = x_max - 1; pxy[3] = pxy[1];
```

→

```
366:
367:    while (pxy[1] < y_max)
368:    {
369:      v_pline(vdi_handle,2,pxy);
370:      pxy[1] += PIXELBREITE + 1;
371:      pxy[3] += PIXELBREITE + 1;
372:    }
373:
374:    for (x = 0;x < RASTERBREITE;x++)
375:      for (y = 0;y < RASTERHOEHE;y++)
376:      {
377:        pxy[0] = x_min + x * (PIXELBREITE + 1);
378:        pxy[1] = y_min + y * (PIXELBREITE + 1);
379:        pxy[2] = pxy[0] + PIXELBREITE - 1;
380:        pxy[3] = pxy[1] + PIXELBREITE - 1;
381:
382:        mask = 0x8000 >> x;
383:        vsf_interior(vdi_handle,
384:                     (switch_bitmap[y] & mask
385:                                  ? FIS_SOLID
386:                                  : FIS_HOLLOW));
387:        vr_recfl(vdi_handle,pxy);
388:      }
389:
390:    v_show_c(vdi_handle,0);
391:  }
392:  /*-------------------------*/
393:  void flip_pixel(int mx, int my)
394:  {
395:    /* Vertauscht die Farbe des Pixels unter der
396:       Maus. Der Drawing-Mode wird auf den akt.
397:       Pixelwert gesetzt und bis zum Loslassen der
398:       Maustaste beibehalten */
399:    int pxy[4],x,y,raster_x,raster_y,mask,col;
400:
401:    objc_offset(TREE[MAINFORM],RASTER,&x,&y);
402:    x = (mx - x)/(PIXELBREITE + 1);
403:    y = (my - y)/(PIXELBREITE + 1);
404:    objc_offset(TREE[MAINFORM],RASTER,
405:                              &raster_x,&raster_y);
406:
407:    mask = 0x8000 >> x;
408:    switch (drawing_mode)
409:    {
410:      case TOGGLE : switch_bitmap[y] ^= mask;
411:                    break;
412:      case ON     : switch_bitmap[y] |= mask;
413:                    break;
414:      case OFF    : switch_bitmap[y] &= ~mask;
415:                    break;
416:    }
417:    col = switch_bitmap[y] & mask;
418:    drawing_mode = (col ? ON : OFF);
419:
420:    pxy[0] = raster_x + x * (PIXELBREITE + 1);
421:    pxy[1] = raster_y + y * (PIXELBREITE + 1);
422:    pxy[2] = pxy[0] + PIXELBREITE - 1;
423:    pxy[3] = pxy[1] + PIXELBREITE - 1;
424:
425:    vsf_interior(vdi_handle,(col ? FIS_SOLID
426:                                 :FIS_HOLLOW));
427:    v_hide_c(vdi_handle);
428:    vr_recfl(vdi_handle,pxy);
429:    copy_bitmap((char *) switch_bitmap,cur_switch);
430:    v_show_c(vdi_handle,0);
431:  }
432:  /*-------------------------*/
433:  void copy_bitmap(char *src,int obj)
434:  {
435:    /* Kopiert die Bitmap eines Schalters auf den
436:       Schirm. Ein normales 'redraw' funktioniert
437:       bei 'BITBLK'-Objekten leider nicht im
438:       'replace'-Modus...    */
439:
440:    int pxy[8];
441:
442:    bitmap_mfdb.fd_addr = src;
443:    pxy[0] = pxy[1] = 0;
444:    pxy[2] = RASTERBREITE - 1;
445:    pxy[3] = RASTERHOEHE - 1;
446:
447:    objc_offset(TREE[MAINFORM],obj,&pxy[4]
448:                                  ,&pxy[5]);
449:    pxy[6] = pxy[4] + RASTERBREITE - 1;
450:    pxy[7] = pxy[5] + RASTERHOEHE - 1;
451:
452:    v_hide_c(vdi_handle);
```

```
453:    vrt_cpyfm(vdi_handle,MD_REPLACE,pxy,
454:                         &bitmap_mfdb,&screen_mfdb,
455:                          color_index);
456:    v_show_c(vdi_handle,1);
457:  }
458:  /*-------------------------*/
459:  void save_bitmap(void)
460:  {
461:    /* Schreibt die Bitmap für einen Schalter und
462:       seinen Namen */
463:    int exit;
464:    FILE *dest;
465:
466:    strcat(path,"*.BBM");
467:    if (fsel_input(path,f_name,&exit)
468:        && (exit == 1) && f_name[0])
469:    {
470:      Biene;
471:      *(strrchr(path,'\\') + 1) = 0;
472:      strcat(path,f_name);
473:      dest = fopen(path,"wb");
474:      if (dest)
475:      {
476:        fwrite(on_bitmap,sizeof(int),RASTER_WORTE,
477:                                              dest);
478:        fwrite(off_bitmap,sizeof(int),RASTER_WORTE,
479:                                              dest);
480:        fwrite(schaltername,1,14,dest);
481:        fclose(dest);
482:      }
483:      Pfeil;
484:    }
485:    *(strrchr(path,'\\') + 1) = 0; /* Pfadnamen
486:                                      korrigieren */
487:  }
488:  /*-------------------------*/
489:  void load_bitmap(void)
490:  {
491:    /* Liest die Bitmap für einen Schalter und
492:       seinen Namen */
493:    int exit;
494:    FILE *dest;
495:
496:    strcat(path,"*.BBM");
497:    if (fsel_input(path,f_name,&exit)
498:        && (exit == 1) && f_name[0])
499:    {
500:      Biene;
501:      *(strrchr(path,'\\') + 1) = 0;
502:      strcat(path,f_name);
503:      dest = fopen(path,"rb");
504:      if (dest)
505:      {
506:        fread(on_bitmap,sizeof(int),RASTER_WORTE,
507:                                              dest);
508:        fread(off_bitmap,sizeof(int),RASTER_WORTE,
509:                                              dest);
510:        fread(schaltername,1,14,dest);
511:        fclose(dest);
512:        copy_bitmap((char *) off_bitmap,BUTNAUS);
513:        copy_bitmap((char *) on_bitmap,BUTNEIN);
514:        redraw(MAINFORM,SWTCHNME);
515:      }
516:      Pfeil;
517:    }
518:    *(strrchr(path,'\\') + 1) = 0; /* Pfadnamen */
519:  }                                /* korrigieren*/
520:  /*-------------------------*/
521:  void save_c_code(char *mode)
522:  {
523:    /* Schreibt den C-Code für den aktuellen
524:       Schalter. Wenn nicht angefügt wird, wird
525:       zusätzlich der Datenstrukturheader erzeugt */
526:
527:    int exit,i;
528:    FILE *dest;
529:
530:    if (schaltername[0] == 0)
531:    {
532:      form_alert(1,noname_alert);
533:      return;
534:    }
535:
536:    strcat(path,"*.BTN");
537:    if (strrchr(f_name,'.'))
538:      strcpy(strrchr(f_name,'.') + 1,"BTN");
539:
```

→

Hard&Soft

leise+schnell (ab15ms)
SCSI Tools
vorbildlicher Service
2 Jahre Garantie

DAS VORBILD

Unser System setzt Maßstäbe

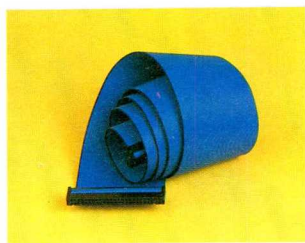
in Leistung, Zuverlässigkeit und Geschwindigkeit, ist kaum zu hören und natürlich vollkommen im Atari ST Design gehalten. Die ausgefeilte, grafische Benutzerführung macht die Bedienung spielend leicht.
... mit weniger Qualität sollten Sie sich nicht zufrieden geben.

Lieferbare Ausführungen

SCSI Ultra Speed Drive	
52 MB · 17 ms · 64 KB Cache	1498,-
85 MB · 19 ms · 64 KB Cache	1998,-
105 MB · 17 ms · 64 KB Cache	2198,-
120 MB · 15 ms · 64 KB Cache	2498,-
170 MB · 15 ms · 64 KB Cache	2998,-
210 MB · 15 ms · 64 KB Cache	3498,-
SCSI Speed Drive	
49 MB · 28 ms	1398,-
85 MB · 24 ms	1648,-
Wechselplatte SCSI Speed Drive 44	1998,-
Streamer SCSI Speed Drive 155	2298,-

Festplatten am Atari TT

Durch ACSI/SCSI Umschaltung und SCSI Tools 2.0 Betrieb am SCSI Port des TT möglich.

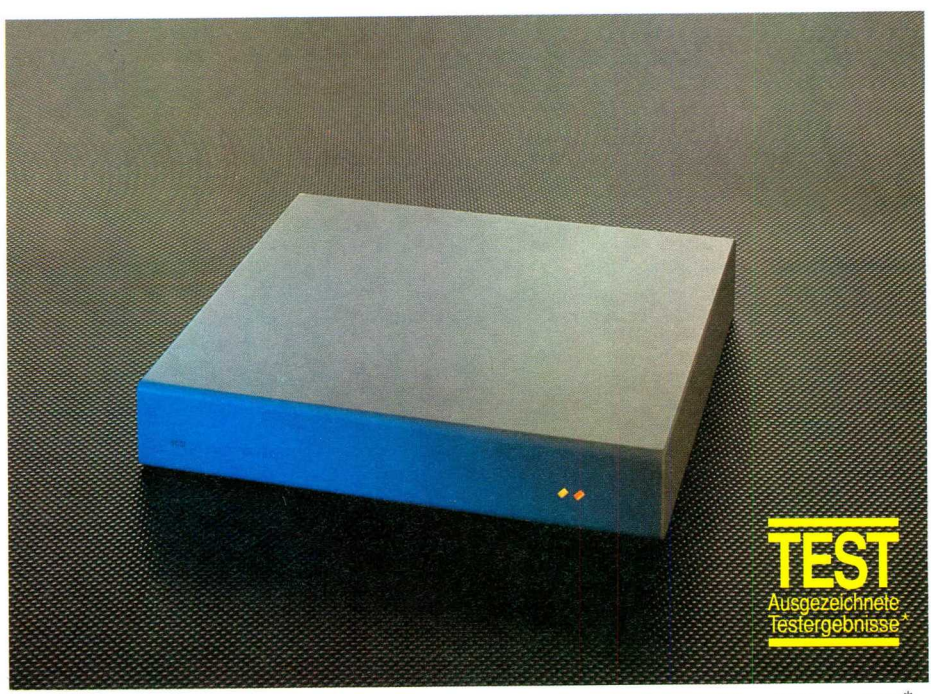


intern voller SCSI Standard



SCSI Schnittstelle an Geräterückseite herausgeführt

SCSI Ultra Speed Drive



»... ein sehr gutes Preis/Leistungsverhältnis« (ST Computer 4/90)*
»Fazit – ein Referenzmodell auf dem Festplattenmarkt« (ST Magazin 8/90)

Obere Münsterstr. 33 — 35
4620 Castrop-Rauxel
Telefon (0 23 05) 1 80 14
Fax 3 24 63

HARD & SOFT A. HERBERG

ATARI-SYSTEM-CENTER

Unsere Vertretungen im Ausland:

Schweiz:
DTZ DATATRADE AG
Landstraße 1
CH 5415 Rieden

Dänemark:
Indigo 2
Montanade 29 E
DK 8000 Arhus C

und unsere Fachhändler in Deutschland:

Sellhorn Computerhaus
Ulzburger Str. 2
2000 Norderstedt

G.M.A mbH
Wandsbecker
Chaussee 58
2000 Hamburg 76

M & R Computer
Vertriebs GmbH
Eppendorfer Weg 56
2000 Hamburg 20

Der Computerladen
Coriansberg 2
2210 Itzehoe

MCC Computer
Lindenhofsgarten 1
2900 Oldenburb

Data Division
Calenbergerstr. 26
3000 Hannover

Com Tech
Gebr.-Woge-Str. 16/17
3220 Alfeld/Leine

H & S Wohlfahrtstätter
Irenenstr. 76 c
4000 Düsseldorf 30

Hendrik Haase
Wiedfeldstr. 77
4300 Essen

Wolf Kirsten
Hammerstr. 11
4400 Münster

CC Computerstudio
Schneider
Elisabethstr. 5
4600 Dortmund

AB computer
GmbH & Co KG
Mommsenstr. 72
5000 Köln 41

Roda Soft
Bahnhofstr. 6
5120 Herzogenrath

Scan Studio Kluge
Hasenbuschstr. 2
5132 Übach-Palenberg

Rückemann
Soft + Tronic
Grundstr. 63
5600 Wuppertal

Application Arts
Eisenbahnstr. 133
6072 Dreieich

Heim Bürofachmarkt
Heidelberger
Landstr. 194
6100 Darmstadt

KFC Computer
Wiesenstr. 18
6240 Königsstein

DEOS Software GmbH
Kappelweg 18
7453 Burladingen 5

Computerfachgeschäft
Rössler
Rheingutstr. 1
7750 Konstanz

Softhansa
Unterbergstr. 22
8000 München 90

Lauterbach
Josephsplatz 3
8000 München

ATAN GbR
Postfach 10 65
8031 Gilching

EDV Partner Strehlow
Brahmstr. 4
8034 Germering

Fischer & Bach GmbH
Münchener Str. 41
8200 Rosenheim

WW-Soft
Kufsteinstr. 69
8200 Rosenheim

Computer Corner
Landshuterstr. 4
8313 Vilsbiburg

Speicher-erweiterungen

Unsere Speichererweiterungen entsprechen dem neuesten Stand der Technik. Ingenieurmäßiges Schaltungsdesign, die Fertigung großer Stückzahlen auf hochmodernen Industriestraßen sowie ausgefeilte Maßnahmen zur Qualitätssicherung setzen einen hohen Qualitätsstandard, von dem auch Sie profitieren können.

Unsere Speichererweiterungen laufen in allen Rechnern. Durch gezielte Maßnahmen konnte die Stromaufnahme und die Störanfälligkeit bei knappem Bustiming deutlich reduziert werden. Dadurch können unsere Erweiterungen auch in Rechnern mit IMP-MMU (bei IMP-MMU nur gleich große Bänke möglich) oder in Atari 1040 ST problemlos betrieben werden.

Unser Angebot umfaßt eine große Palette an professionellen Speichererweiterungen, die generell in zwei Versionen lieferbar sind.

Die vollsteckbare Version ist kinderleicht einzubauen. Sämtliche Verbindungen der Speicherkarte können ohne Lötarbeiten vorgenommen werden. Voraussetzung für den Einbau dieser Version ist, daß im Rechner die MMU sowie der Videoshifter gesockelt sind. Der Steckverbinder zur MMU besitzt natürlich vergoldete Kontakte. Die teilsteckbare Version ist für Rechner mit nicht gesockelten Bauteilen (Shifter und/oder MMU) und für alle, denen das Anlöten von ca. 18 Lötverbindungen keine Probleme bereitet, gedacht.

Zum Lieferumfang jeder Speichererweiterung (bestückt) gehört eine ausführliche und bebilderte Einbauanleitung sowie ein Speichertestprogramm. Selbstverständlich wird jede unserer Speichererweiterungen vor dem Versand im Rechner stückgeprüft.

Sollte dennoch eine Frage offenbleiben, so helfen Ihnen an unserer Service-Hotline versierte Techniker gerne weiter.

Modell 1 S: Speicheraufrüstung auf 1 MB, voll steckbar, 198,00 DM.

Modell 2: Speicheraufrüstung auf 2,5 MB, teilsteckbar (beim Mega ST 2 Aufrüstung auf 4 MB möglich), 398,00 DM; dto. Leerk. 189,00 DM.

Modell 2/4: Speicheraufrüstung auf 2,5/4 MB, teilsteckbar, Speicher in zwei Stufen 2,5/4 MB aufrüstbar. Auch für Mega ST geeignet, 449,00 DM; dto. als Leerkarte 249,00 DM.

Modell 2/4 S: wie Modell 2/4, aber voll steckbar, mit vergoldeten Mikrokontakten, 498,00 DM; dto. als Leerkarte 289,00 DM.

Speichererweiterung für 1040 STE SIM-Module auf 2,5 MB 338,00 DM, auf 4 MB 676,00 DM.

Disketten-laufwerke

Diskettenlaufwerke: 3,5-Zoll- und 5,25-Zoll-Disketten-Laufwerke in vollendeter Qualität. Es werden nur die besten Materialien verwendet. Laufwerksgehäuse mit kratzfester Speziallackierung. 5,25-Zoll-Laufwerk (720 KB/1,2 MB) incl. beige Frontblende, 40/80 Track-Umschalter, Software IBM-Atari, anschlußfertig 289,- DM, Chassis Atari modifiziert 198,00 DM, 1,44-MB-Laufwerk incl. HD Interface, anschlußfertig 298,- DM, 3,5-Zoll-LW incl. beige Frontblende mit NEC FD 1037 oder TEAC FD 235 anschlußfertig 239,00 DM, Chassis 169,00 DM.

Auto-Monitor-Switchbox: A.R.S. (Automatic Resolution Selection). Das Programm wird automatisch in der richtigen Auflösung gestartet (nur TOS 1.0 und 1.2). Mit der Auto-Monitor-Switchbox können Sie über die Tastatur zwischen Monochrom und Farbmonitor umschalten oder einen Tastaturreset durchführen. Die mitgelieferte Software ist resetfest. Durch Einbinden der von uns mitgelieferten Routinen Umschaltmöglichkeit ohne RESET. Zusätzlicher BAS und Audio-Ausgang. Auto-Monitor Switchbox 59,90 DM, Auto-Monitor Switchbox Multisync 69,90 DM, weitere Modelle: von 29,90 DM bis 69,90 DM

Video Interface +: ermöglicht die Farbwiedergabe an einem Farbfernseher, Monitor oder Videorecorder mit Videoausgang (mit integrierter Auto-Monitor-Switchbox-Funktion). 159,00 DM

Neu: Echtzeit-Videodigitalisierer in 16 Graustufen (Einlesen von Videosignalen im Computer, kein Standbild erforderlich). 398,00 DM

HF-Modulator: zum Anschluß des Atari ST an jeden gewöhnlichen Farbfernseher. Der Ton wird über den Fernseher übertragen. 189,00 DM

Festplattenzubehör: wie SCSI Hostadapter, Einschaltverzögerungen, 1,2 m- DMA-Kabel etc.

STTAST II: ermöglicht den Anschluß einer beliebigen PC-(XT-) Tastatur am ST, umschaltbare Mehrfachbelegung der Tastaturbelegungen, freie Programmierbarkeit von Makros und Generieren von Start-Up-Files (mit AUTO Load), Tastaturreset, unterstützt auch PC Ditto und PC/AT Speed. 149,00 DM Set: PC Tastatur mit Mikroschalter + ST Tast II 298,00 DM

Abgesetzte Tastatur am ST: Tastaturgehäuse mit Spiralkabel, Treiberstufe, Resettaste und Joystickbuchsen eingebaut. Computertyp angeben. 109,00 DM

Towergehäuse: nur Gehäuse oder mit kundenspezifischer Bestückung ab 298,00 DM

RTS Tastaturkappen: ab 89,00 DM

Uhrmodul intern: die Bootsoftware befindet sich auf ROM's im Betriebssystem. Wichtig: Betriebssystem angeben. 119,00 DM

Floppyswitchbox: ermöglicht den Anschluß von drei Laufwerken am ST. Ausgestattet mit speziellen Treibern für 3,5 und 5,25 Laufwerke. Computertyp angeben. 89,00 DM.

Außerdem: HD-Kid — interne Beschaltung incl. Software zum Anschluß von HD-(3,5"- und 5,25"-) Laufwerken am ST. Keine Zusatzschaltung im Laufwerk erforderlich, 69,00 DM.

SCSI-Hostadapter — orig. ICD-Hostadapter, Advantage+ (mit Uhr) 298,00 DM, Advantage (ohne Uhr) 269,00 DM, Mikro 249,00 DM. Lieferung incl. DMA-Kabel, ICD-Software und Dokumentation.

Einschaltverzögerung — Ermöglicht das gleichzeitige Einschalten von Computer und Festplatte, Einbau erfolgt im Computer durch einfaches Aufstecken, Zeitverzögerung einstellbar, 49,90 DM.

Tastaturkabel Mega ST — langes Tastaturkabel (2 m) für den Mega ST 29,90 DM.

Eprombrenner — orig. MAXON Junior-Prammer, 229,00 DM, Epromkarte 128 KB incl. Gehäuse (Steckmodul), 49,90 DM.

Laserinterface II — Abschalten des Laserdrucker im Festplattenbetrieb möglich. Lieferung komplett mit Netzteil VDE, GS, 79,00 DM.

MS-DOS-Emulatoren — MS-DOS-Emulator zum internen Einbau. ATonce oder AT Speed 479,00 DM.

Fordern Sie unseren Gesamtkatalog an

ATARI Computer
Speichererweiterungen
Festplatten
Wechselplatten
Streamer
Diskettenlaufwerke
HD-Diskettenlaufwerke
HD-Kids
Laufwerkschassis
Monitorumschaltboxen
Videointerface
Videodigitalizierer
HF-Modulator
SCSI-Hostadapter
Einschaltverzögerungen
PC-Tastatur an Atari ST
Towersysteme
Towerzubehör
abgesetzte Tastatur am ST
RTS-Tastaturkappen
Echtzeituhren
Floppyswitchbox
Laufwerksgehäuse
Software (PD)
Disketten
Festplattengehäuse
Computerkabel
Mega-Tastaturkabel
Eprombrenner
Epromkarten
Laserinterface II
PC Bridge (STE)
ATonce/AT Speed
AD Speed (ICD)

```
540:    if (fsel_input(path,f_name,&exit)
541:        && (exit == 1) && f_name[0])
542:    {
543:      Biene;
544:      *(strrchr(path,'\\') + 1) = 0;
545:      strcat(path,f_name);
546:      dest = fopen(path,mode);
547:      if (dest)
548:      {
549:        if (mode[0] == 'w') make_header(dest);
550:
551:        fprintf(dest,"\nSWITCH %s = \n{",
552:                                    schaltername);
553:        for(i = 0;i< RASTER_WORTE; i++)
554:        {
555:          if (!(i % 8)) fprintf(dest,"\n\t");
556:          fprintf(dest,"%+#06x,",on_bitmap[i]);
557:        }
558:        fprintf(dest,"\n\t");
559:        for(i = 0;i< RASTER_WORTE - 1; i++)
560:        {
561:          if (!(i % 8)) fprintf(dest,"\n\t");
562:          fprintf(dest,"%+#06x,",off_bitmap[i]);
563:        }
564:        fprintf(dest,"%+#06x\n};\n",
565:                        off_bitmap[RASTER_WORTE - 1]);
566:        fclose(dest);
567:      }
568:      Pfeil;
569:    }
570:    *(strrchr(path,'\\') + 1) = 0; /* Pfadnamen
571:                                    korrigieren */
572:  }
573:  /*-------------------------*/
574:  void make_header(FILE *dest)
575:  {
576:    /* Erzeugt einen Header mit der Datenstruktur
577:       'SWITCH'. Ist nur für den ersten Schalter
578:       nötig, folgende können angefügt werden     */
579:
580:    fprintf(dest,"/*************************/\n"
581:                 "/* Schalter-Datenstruktur */\n"
582:                 "/*************************/\n");
583:    fprintf(dest,"typedef struct\n{\n\tint on[%d];"
584:                 "\n\tint off[%d];\n} SWITCH;\n\n",
585:                 RASTER_WORTE,RASTER_WORTE);
586:    fprintf(dest,"/*************************/\n"
587:                 "/*    Schalter-Bitmaps    */\n"
588:                 "/*************************/\n");
589:  }
590:  /*-------------------------*/
```

Listing 3: Der 'Button-Design'-Editor

```
 1:  /*-------------------------
 2:  |     BUT_DSGN.RSH         Resource-File          |
 3:  |                                                  |
 4:  | Dieses File wurde mit RSC2CSRC von Erik Dick     |
 5:  | erzeugt und zum Abdruck von Hand nachbe-         |
 6:  | arbeitet.                                        |
 7:  |                                                  |
 8:  -------------------------*/
 9:
10:  #include <aes.h>
11:  #include <vdi.h>
12:
13:  #define NUM_OBS 28
14:
15:  void    objc_init( void );
16:  extern  OBJECT *TREE[],rs_object[];
17:  /*-------------------------*/
18:
19:  void objc_init( void )
20:  {   int i;
21:
22:      for(i = 0; i < NUM_OBS;
23:                  rsrc_obfix(rs_object,i++));
24:  }
25:  /*------- TEDINFO-Strings -------*/
26:
27:  char    TED0_s[15] = " Button Design";
28:  #define TED0_t ""
29:  #define TED0_v ""
30:  char    TED1_s[26] = "von Hans-Jürgen Richstein";
31:  #define TED1_t ""
```

```
32:  #define TED1_v ""
33:  char    TED2_s[15] = "";
34:  #define TED2_t "_______________"
35:  #define TED2_v "XXXXXXXXXXXXXX"
36:  char    TED3_s[20] = "(c) Maxon GmbH 1990";
37:  #define TED3_t ""
38:  #define TED3_v ""
39:  #define FSTR0 "Aus"
40:  #define FSTR1 "Ein"
41:  #define FSTR2 "Test"
42:  #define FSTR3 "Schaltername"
43:  #define FSTR4 "Neuer C-Code"
44:  #define FSTR5 "C-Code anfügen"
45:  #define FSTR6 "Bitmap speichern"
46:  #define FSTR7 "Bitmap laden"
47:  #define FSTR8 "Ende"
48:  #define FSTR9 "Raster +'Ein'"
49:  #define FSTR10 "Löschen"
50:
51:  /*---------- Bitblocks ----------*/
52:
53:  int IMAGE0[16] = {0};
54:  int IMAGE1[16] = {0};
55:  int IMAGE2[16] = {0};
56:
57:  /*------- BITBLK-structures -------*/
58:
59:  BITBLK rs_bitblk[] =
60:  {   IMAGE0,0x2 ,0x10,0x0 ,0x0 ,0x1 ,
61:      IMAGE1,0x2 ,0x10,0x0 ,0x0 ,0x1 ,
62:      IMAGE2,0x2 ,0x10,0x0 ,0x0 ,0x1
63:  };
64:
65:  /*------- TEDINFO-structures -------*/
66:
67:  TEDINFO rs_tedinfo[] =
68:  {   TED0_s,TED0_t,TED0_v,IBM ,0x6 ,TE_LEFT,
69:                      0x1071,0x0 ,0x1 ,0xf ,0x1 ,
70:      TED1_s,TED1_t,TED1_v,SMALL ,0x6 ,TE_LEFT,
71:                      0x1000,0x0 ,0xffff,0x1a,0x1 ,
72:      TED2_s,TED2_t,TED2_v,IBM ,0x6 ,TE_LEFT,
73:                      0x1180,0x0 ,0xffff,0xf ,0xf ,
74:      TED3_s,TED3_t,TED3_v,SMALL ,0x6 ,TE_CNTR,
75:                      0x1180,0x0 ,0xffff,0x14,0x1
76:  };
77:
78:  /*------- Object-structures -------*/
79:
80:  OBJECT rs_object[] =
81:  {       -1, 1,27,G_BOX    ,NONE     ,OUTLINED,
82:                          0X21111L,1030,3584,42,2067,
83:           3, 2, 2,G_BOXTEXT,NONE    ,OUTLINED,
84:          (long) &rs_tedinfo[0],514,2560,1061,1793,
85:           1,-1,-1,G_TEXT   ,NONE     ,NORMAL,
86:         (long) &rs_tedinfo[1],1297,2560,1554,2560,
87:          19, 4,18,G_BOX    ,NONE     ,NORMAL,
88:                          0XFF1141L,2,2306,21,16,
89:          11, 5,10,G_BOX    ,NONE     ,NORMAL,
90:                          0XFF1101L,1,2304,19,5,
91:           6,-1,-1,G_IMAGE ,TOUCHEXIT ,OUTLINED,
92:                (long) &rs_bitblk[0],2,1,4096,4096,
93:           7,-1,-1,G_IMAGE ,TOUCHEXIT ,OUTLINED,
94:                (long) &rs_bitblk[1],2,3,4096,4096,
95:           8,-1,-1,G_STRING,NONE     ,NORMAL,
96:                               (long) FSTR0,5,1,3,1,
97:           9,-1,-1,G_STRING,NONE     ,NORMAL,
98:                               (long) FSTR1,5,3,3,1,
99:          10,-1,-1,G_IMAGE ,NONE     ,NORMAL,
100:              (long) &rs_bitblk[2],11,2,4096,4096,
101:          4,-1,-1,G_STRING,TOUCHEXIT ,OUTLINED,
102:                              (long) FSTR2,14,2,4,1,
103:         14,12,13,G_BOX    ,NONE     ,NORMAL,
104:                          0XFF1101L, 1, 6,19,  2050,
105:         13,-1,-1,G_STRING,NONE     ,NORMAL,
106:                           (long) FSTR3,515,512,12,1,
107:         11,-1,-1,G_FTEXT ,EDITABLE,NORMAL,
108:             (long) &rs_tedinfo[2],514,769,14, 1,
109:         15,-1,-1,G_BUTTON,TOUCHEXIT ,NORMAL,
110:                              (long) FSTR4,1,521,19,1,
111:         16,-1,-1,G_BUTTON,TOUCHEXIT ,NORMAL,
112:                           (long) FSTR5,1,1802,19,1,
113:         17,-1,-1,G_BUTTON,TOUCHEXIT ,NORMAL,
114:                           (long) FSTR6,1,3083,19, 1,
115:         18,-1,-1,G_BUTTON,TOUCHEXIT ,NORMAL,
116:                              (long) FSTR7,1,269,19,1,
117:          3,-1,-1,G_BUTTON,TOUCHEXIT ,NORMAL,
118:                           (long) FSTR8,1,2574,19,1,
                                                    →
```

```
119:          20,-1,-1,G_BOX    ,TOUCHEXIT ,NORMAL,
120:                          0XFF1181L,280,2306,1807,3847,
121:          27,21,26,G_BOX    ,NONE      ,NORMAL,
122:                          0XFF1141L,280,11,1807,6,
123:          22,-1,-1,G_BUTTON,TOUCHEXIT ,NORMAL,
124:                            (long) FSTR9,1,2,14,1,
125:          23,-1,-1,G_BOXCHAR ,TOUCHEXIT ,NORMAL,
126:                         (long) 0X1FF1100L,520,4,3,2049,
127:          24,-1,-1,G_BOXCHAR ,TOUCHEXIT ,NORMAL,
128:                         (long) 0X2FF1100L,1284,4,3,2049,
129:          25,-1,-1,G_BOXCHAR ,TOUCHEXIT ,NORMAL,
130:                           (long) 0X4FF1100L,1,4,3,2049,
131:          26,-1,-1,G_BOXCHAR ,TOUCHEXIT ,NORMAL,
132:                         (long) 0X3FF1100L,1803,4,3,2049,
133:          20,-1,-1,G_BUTTON,TOUCHEXIT ,NORMAL,
134:                           (long) FSTR10,1792,2048,14,1,
135:           0,-1,-1,G_BOXTEXT ,LASTOB  ,NORMAL,
136:              (long) &rs_tedinfo[3],280,2321,1807,1
137:  };
138:  /*------- TREE-Index ----------*/
139:
140:  OBJECT *TREE[] =
141:  {   (OBJECT *) &rs_object[0]
142:  };
```

Listing 4: Resource-Decompilat

```
 1:  /* BUT_DSGN.H mit den Definitionen  */
 2:
 3:  #define MAINFORM 0  /* Formula/Dialog */
 4:  #define BUTNAUS  5  /* IMAGE in Baum MAINFORM */
 5:  #define BUTNEIN  6  /* IMAGE in Baum MAINFORM */
 6:  #define AUSTEXT  7  /* STRING in Baum MAINFORM */
 7:  #define EINTEXT  8  /* STRING in  MAINFORM */
 8:  #define TESTAREA 9  /* IMAGE in  MAINFORM */
 9:  #define TEST     10 /* STRING in  MAINFORM */
10:  #define SWTCHNME 13 /* FTEXT in  MAINFORM */
11:  #define NEWCODE  14 /* BUTTON in  MAINFORM */
12:  #define APPEND   15 /* BUTTON in  MAINFORM */
13:  #define SAVEIMG  16 /* BUTTON in  MAINFORM */
14:  #define LOADIMG  17 /* BUTTON in  MAINFORM */
15:  #define ENDE     18 /* BUTTON in  MAINFORM */
16:  #define RASTER   19 /* BOX in  MAINFORM */
17:  #define ORSWITCH 21 /* BUTTON in  MAINFORM */
18:  #define MOVEUP   22 /* BOXCHAR in  MAINFORM */
19:  #define MOVEDOWN 23 /* BOXCHAR in  MAINFORM */
20:  #define MOVELEFT 24 /* BOXCHAR in  MAINFORM */
21:  #define MOVERGHT 25 /* BOXCHAR in  MAINFORM */
22:  #define CLEARALL 26 /* BUTTON in  MAINFORM */
```

Listing 5: Das Header-File

Speichererweiterung für ATARI

	520	1040	STE	ST1	ST2	STACY1
1MB	178,-	---	---	---	---	---
2MB	598,-	498,-	448,-	448,-	---	648,-
4MB	948,-	848,-	798,-	798,-	448,-	998,-

incl. Einbau und 1 Jahr Garantie !

Festplatten für ATARI

WaSy 48S / WaSy 84S 1098,-/1398,-
48MB/40ms bzw. 84MB/24ms
WaSy 110C / WaSy 210C 1898,-/2798,-
110MB/15ms bzw. 210MB/15ms mit 32KB Cache
WaSy 213M / WaSy 340M 3298,-/5198,-
213MB bzw. 340MB/15ms Maxtor, sehr schnell
WaSy 44S / Medium 1898,-/ 198,-
44 MB Wechselplatte mit Medium bzw. Medium einzeln
WaSy 44S48 2598,-
Wechselpl. + 48MB/40ms Festplatte incl. Medium
WaSy 44S110C 3398,-
44MB Wechselpl. + 110MB/15ms Festplatte incl. Medium
WaSy Stream 150 2098,-
155MB Streamer mit Kasette ca. 7MB/min. incl. Medium
WaSy Stream 150/110C 3598,-
155MB Streamer mit 110MB/15ms Festplatte incl. Medium

alle Systeme mit LACOM Software
Anschlußfertig an ATARI ST/TT
12 Monate Garantie!

Wir sind umgezogen. Bitte beachten Sie unsere neue Adresse !

ATARI ST / STACY / TT

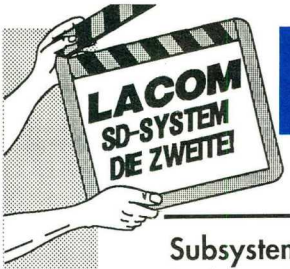
MEGA ST/1	1MB RAM, Monitor, Mouse	1448,-
MEGA ST/2	2MB RAM, Monitor, Mouse	1898,-
MEGA ST/4	4MB RAM, Monitor, Mouse	2248,-
MEGA ST/2A	wie ST/2 + AT-SPEED	2448,-
MEGA ST/2+	wie ST/2 + hyperCACHE ST+	2498,-
MEGA ST/4 030	wie ST/4 + hyperCACHE 030	4648,-
40 MB Festplatte für STACY		a.A.
Größere Festplatte für TT		a.A.

Alle ATARI ST mit TOS 1.4 und thermisch geregeltem Lüfter !
Auch andere Konfigurationen lieferbar !

hyperCACHE ST+	ca. 75% mehr Leistung	550,-
Ad-SPEED		598,-
hyperCACHE 030	TT-Power im MEGA ST	2498,-
PC/AT SPEED	Hardware-DOS Emulator	398,-/528,-
Spectre GCR	ohne ROM's	599,-
Echtzeituhr	läuft ab TOS 1.2 ohne Software	89,-
HD-Floppy-Kit	komplett mit Floppy, Geh., Netzt.	298,-
Thermische Lüfterregelung		39,-
OverScan	höhere Auflösung mit ATARI-Monitor	118,-
NEC P20	24-Nadel Drucker	948,-
NEC P60	24- Nadel Drucker	1698,-
NEC S60	Laserdrucker 1,5MB RAM	3598,-
EIZO 6500 21" S/W Monitor		3398,-

Wir führen auch Drucker, Scanner, Softwarwe, PD-Software u.v.m. Bitte fragen Sie nach !

wacker systemelektronik gmbh
Bachstr. 39
7500 Karlsruhe 21
FAX/BTX : 0721/593723
Tel.: 0721/554471



KLAPPE!

Die Erfolgsserie geht weiter!

Subsysteme für ATARI-ST-/TT-, AT- und MAC-Rechner sowie Sampler-Systeme

Festplatten von 32 MB bis 2 GB • Wechselplatten mit 44 MB • Bandlaufwerke mit 155 MB • Optische Laufwerke mit 2 x 320 MB

Durch modernste SCSI-Technik und den neuentwickelten Hostadapter „LAADAP" setzen unsere Systeme neue Maßstäbe

- Echte Parity-Option
- Hardware-Schreibschutz (per Software schaltbar)
- Bidirektionaler SCSI-Port (automatisch umschaltend)
- Zwei Atari-Rechner können zusammengeschaltet werden
- Gleichzeitiger Anschluß über DMA und SCSI möglich

Fordern Sie unser kostenloses Informationsmaterial noch heute an!

LACOM Ilias Lazaridis

Stöckmannstr. 56 • 4200 Oberhausen 1 • Tel. 02 08 / 80 50 46 • Fax 02 08 / 2 52 74

„Wußten Sie schon, daß...!"



Sie bei uns TOP-PD-Programme erhalten können, zu einem Preis, bei dem Sie sofort zugreifen sollten!

Die TOP-TEN Luxus-Pakete:

Für nur 25,- DM je Paket (Scheck/bar) erhalten Sie auf 5 2dd Disks TOP-PD-Programme portofrei incl. unseren 90-seitigen Katalog! Bei Nachnahme zzgl. 4,- DM! Ausland 30,- DM je Paket!

NEWS

Paket 1: Actiongeladene TOP-Spiele (s/w).

Paket 2: Starke Anwenderprogramme (s/w).

Paket 3: Spannende & feuerige TOP-Farbspiele (f).

Paket 4: Der richtige Einstieg für ST-Neulinge (s/w).

Paket 5: Tolle Clip-Art-Bilder in TOP-Qualität (s/w).

Paket 6: Powergeladene Midi- & Musikprogramme (s/w).

Paket 7: Erotikshow für Erwachsene → Alter! (s/w+1MB).

Paket 8: Hexereien auf dem ST. Sie werden Staunen! (1+1MB).

Paket 9: Erotik-Farb-Show für Erwachsene → Alter! (f).

Hier nun weitere Angebote:

Signum II_348.- That's Write Profi_288.- Adimens3.1_319.- Script II_249.- PKS Write_189.- STAD V13+_159.- Outline Art_349.- Soundma. II_188.- Calamus Fontedit or_189.- BTX/VTX Manager für Modem_258.- bzw. 339, für Postbox *** Drakken_71,90 *** Kick Off 2_ 63,90 *** Leisure S. Larry II_89,90

Versand: 5,- DM bei Vorauskasse und 7,- DM bei Nachnahme

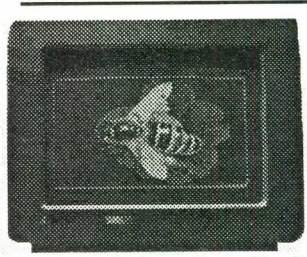
Ralf Markert
Computer & Software
Balbachtalstr. 71 * 6970 Lauda 1

Der Preisknaller: Mega Paint II 199,- DM!

Tel.: 09343/3854 (24h-Service)

Fordern Sie noch heute unseren 90-seitigen Gratiskatalog an!

SM 124 Multisync II



1 Monitor 3 Auflösungen

100 % Softwarekompatibel
Hilfsprogramme auf Disk
Beste Industriequalität
Als Bausatz lieferbar

Nach der fachgerechten Umrüstung ist Ihr Monitor SM 124 in der Lage, in allen 3 Auflösungen zu arbeiten. Die Farben werden dabei in Graustufen dargestellt. Die hohe Auflösung verliert nichts an ihrer Brillianz. Die Softwarekompatibilität wird durch diese reine Hardwarelösung nicht beeinträchtigt.

Werten Sie Ihren SM 124 auf !

SM 124 Multisync, alle 3 Auflösungen	DM 549,00
Umbau Ihres Monitors (ca. 3 Tage)	DM 249,00
Bausatz komplett bestückt*	DM 149,00
Leerplatine incl. aller Bauteile*	DM 129,00

*ausführliche Anleitung (dt.) liegt bei.

Bei Bestellung bitte Baujahr des Monitors angeben.

Preise zzgl. Porto und Verpackung DM 5,00, NN plus DM 2,00, besser V-Scheck (Ausland nur Euro-Scheck).

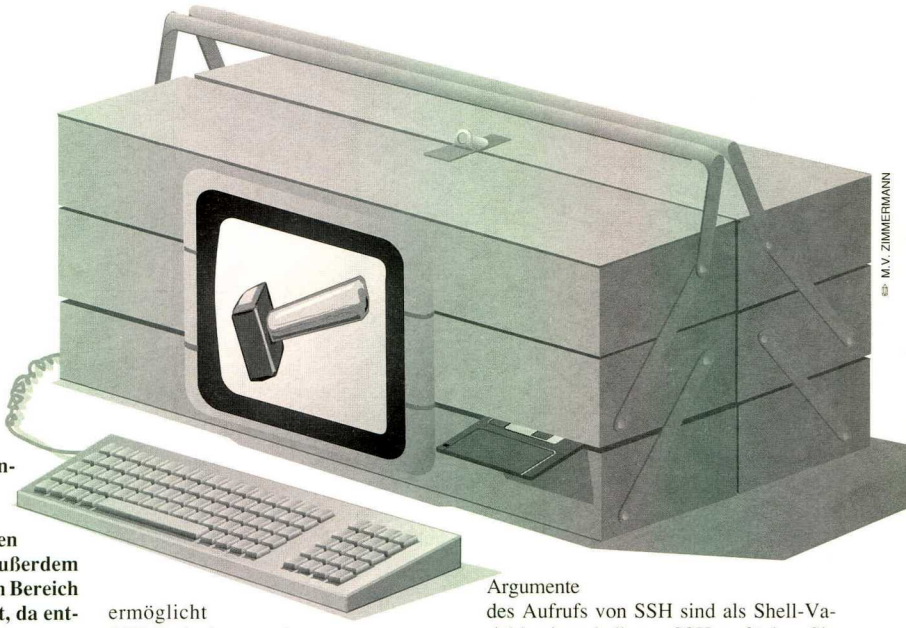
Händleranfragen erwünscht !

iks

In der Au 22
7516 Karlsbad 4
07202/7687

Programmer's Toolbox - Dateien

Teil 8: Die Shell oder das Kommando SSH



Heute ist es soweit. Das Herzstück des ersten Blocks der „Programmer's Toolbox" - die Shell - wird implementiert. Es wird dabei gezeigt, wie Shell-Variablen, Pipelining und Kommandoprozeduren realisiert werden können. Außerdem werden einige Begriffe aus dem Bereich des Übersetzerbaus eingeführt, da entsprechende Methoden bei der Erstellung der Shell benötigt werden.

Zuvor soll der Begriff des *Pipelinings* erklärt werden, da dies - laut obiger Ankündigung - eine der Fähigkeiten sein wird, die SSH besitzt. Unter Pipelining versteht man eine Verknüpfung von zwei oder mehr Kommandos, wobei jeweils die Standardausgabe des ersten Kommandos als Standardeingabe des zweiten Kommandos dient. Die Ausgabe des zweiten Kommandos ist dann wieder die Eingabe des dritten und so fort. Es entsteht gewissermaßen eine *Daten-Pipeline* durch sämtliche Kommandos. Ein Beispiel für Pipelining, wie wir es im folgenden verwenden werden, zeigt Abb.1.7. Dargestellt ist eine Pipeline, bestehend aus den beiden Kommandos *LS* und *MORE*. Auch hier erkennt man wieder deutlich eine Anleihe beim Betriebssystem UNIX.

Name
SSH - eine einfache Benutzeroberfläche

Anwendung
SSH *Argument...*

Beschreibung
SSH ist eine einfache Benutzeroberfläche (**S**imple **SH**ell). SSH erlaubt die Ausführung von TOS-Programmen, insbesondere die der PTB-Kommandos. Zusätzlich ermöglicht SSH Umlenkungen der Standarddatenkanäle, Pipelining und einfache Shell-Variablen. Die Shell-Variablen sind SSH als *Argumente* zu übergeben. Außerdem ist es unter Verwendung von Umlenkungen möglich, komplette Anweisungsfolgen in Textdateien zu speichern und als geschlossene Kommandoprozeduren abzuarbeiten. Diese Eigenschaften werden nachfolgend anhand von Beispielen dargestellt:

Beispiel
Umlenkung der Ausgabe von LS in eine Datei *Dateiname* und nachfolgendes Ansehen von *Dateiname* mit MORE:

```
LS -L > Dateiname
MORE Dateiname
```

Die Kurzfassung obiger Anweisung mit Hilfe von Pipelining:

```
LS -L | MORE
```

Das Kopieren einer Befehlsfolge in eine Datei Dateiname und deren nachfolgende Ausführung mit Hilfe eines eingeschachtelten SSH-Aufrufs. Die Ausgabe von SSH wird in *Dateiname2* abgelegt:

```
ECHO „LS -L | MORE" > Dateiname
SSH < Dateiname > Dateiname2
```

Argumente des Aufrufs von SSH sind als Shell-Variablen innerhalb von SSH verfügbar. Sie sind von $1 bis $n (n Anzahl der Argumente) durchnumeriert und können innerhalb von SSH benutzt werden. Folgende Kommandosequenz würde daher zunächst eine Datei *Dateiname* erzeugen. Beim Aufruf von SSH mit *Dateiname* als Standardeingabe würde die Shell-Variable $1 in *Dateiname* durch die Zeichenkette „*.C" (erstes Argument von SSH) ersetzt. Zusammengenommen listet das Kommando der zweiten Zeile alle C-Sourcen:

```
ECHO „LS -L $1 | MORE" > Dateiname
SSH „*.C" < Dateiname
```

Grammatik

SSH verwendet zur Interpretation von Kommandozeilen eine einfache Grammatik (s.Tab. 1).

STRING ist dabei eine Zeichenkette ohne Leerzeichen und ohne eines der folgenden Symbole: '|', '>', '<' und '$'. NUMBER ist eine positive ganze Zahl. Ferner werden leere Zeilen und solche, die mit dem Symbol '#' beginnen, ignoriert. Entsprechend können Zeilen, die mit '#' beginnen, zur Kommentierung von Kommandoprozeduren genutzt werden.

```
commandline     ::= command
                  | command „|" commandline
command         ::= commandname argumentlist stdinredirect
                                             stdoutredirect
argumentlist    ::= {empty}
                  | argument argumentlist
stdinredirect   ::= {empty}
                  | „<„ filename
stdoutredirect  ::= {empty}
                  | „>" filename
commandname, argument, filename
                ::= word
word            ::= Shell-Variable
                  | STRING
Shell-Variable  ::= „$" NUMBER
```

Tabelle 1

Beschreibung

SSH kennt insgesamt drei interne Kommandos. Es handelt sich dabei um:

```
CD    - Ändern des Arbeitsverzeichnisses
PWD   - Anzeigen des Arbeitsverzeichnisses
EXIT  - das Verlassen der Shell
```

Die Kommandos CD und PWD wurden bereits an früherer Stelle vorgestellt. EXIT hat eine sehr einfache Funktion. Es kann benutzt werden, um die Shell zu verlassen. Parameter werden dabei keine benötigt. Sind doch welche Parameter angegeben, werden sie ignoriert.

Programmierung

Die Programmierung der Shell ist in Listing 1.20 erfolgt. Entsprechend der Aufgabenstellung ist das Kommando SSH ein kleiner Interpreter für die „Sprache der Kommandozeilen". Seine Grammatik ist winzig und wurde bereits unter *Beschreibung* kurz notiert. Wir wenden uns nun der Implementierung dieses Interpreters zu.

In der Hauptfunktion *main* (Zeilen 463-484) erfolgen zunächst einige einleitende Anweisungen. Für den Fall, daß es sich beim Standardeingabekanal um eine Konsole handelt, wird eine kurze Copyright-Meldung ausgegeben, und es werden Kopien der Parameter von *main* in den globalen Variablen *gargc* und *gargv* untergebracht (Zeilen 469-474). In Folge wird eine Endlosschleife betreten (Zeilen 475-484). In ihrem Anweisungsteil erfolgt die Ausgabe eines Prompts (Zeilen 476-477) und die Kommandoeingabe. Kommentare werden gleich nachfolgend interpretiert, einfach, indem man die betreffenden Zeilen ignoriert. Für beliebige andere Kommandozeilen wird die Auswertung und Ausführung durch den Aufruf von *eval* (Zeile 482) eingeleitet.

Wie bereits gesagt: SSH ist ein Interpreter. Der Einstieg in die Interpretation geschieht mit dem Aufruf der Funktion *eval*. Unter diesem Aspekt möchte ich zunächst einige allgemeine Erläuterungen über Interpreter, Sprachen und Grammatiken geben, bevor konkret die Interpreter-Funktionen der SSH betrachtet werden. Den Ausgangspunkt bildet dabei die unter *Beschreibung* gegebene Grammatik für die „Sprache der Kommandozeilen". Genauer: Es wird der Begriff der Grammatik eingeführt und die zuvor benutzte Schreibweise erläutert.

Eine Grammatik ist ein System von *Regeln*, mit deren Hilfe eine Sprache hergeleitet werden kann. Eine Regel setzt sich aus *nichtterminalen* und *terminalen Symbolen* zusammen. Terminale Symbole sind Wörter der beschriebenen Sprache, nichtterminale ihrerseits wiederum Regeln. Sie werden, im Zuge weiterer Regelanwendungen, letztlich ebenfalls zu terminalen Symbolen entwickelt. In der unter *Beschreibung* angegebenen Grammatik kommt dabei eine Schreibweise zur Anwendung, die eine Untermenge der sogenannten *Bacchus-Naur-Form* (kurz BNF) ist. Mit dem Operator '::=' wird in BNF eine Regel definiert. Links vom Operator '::=' steht dabei der Name der Regel, auf der rechten Seite deren Produktion. Produktionen können Sequenzen von terminalen und nichtterminalen Symbolen enthalten. Zusätzlich können sie durch den Operator '|' getrennt werden. Er kennzeichnet unterschiedliche Alternativen bei der Regelanwendung.

Anmerkung: Der Operator '|' darf nicht verwechselt werden mit dem Pipe-Symbol "|". '|' ist ein Operator der BNF. "|" ist ein terminales Symbol der „Sprache der Kommandozeilen".

Mit den Festlegungen unter *Beschreibung* haben wir eine Grammatik mit den Regeln *commandline*, *command*, *argumentlist*, *stdinredirect*, *stdoutredirect*, *commandname*, *argument*, *filename*, *word* und *shellvariable* definiert. Die Grammatik besitzt die terminalen Symbole "/", "<", ">", *STRING*, "$" und *NUMBER*. *commmandline* ist das Startsymbol der Grammatik. *commandline* kann entweder zu *command* oder zu *command "|" commandline* abgeleitet werden. Soll für eine Folge von Symbolen festgestellt werden, ob sie einen Satz in der durch die Grammatik definierten Sprache darstellt, muß gezeigt werden, daß sich die Folge von Symbolen, ausgehend vom Startsymbol, mit Hilfe der Grammatik herleiten läßt. Es muß also eine Folge von Regelanwendungen angegeben werden, die den Satz bildet. Dargestellt werden derartige Herleitungen als *Ableitungsbäume*. Ausgehend vom Startsymbol (*commandline*) werden alle nichtterminalen Symbole innerhalb von Ableitungsbäumen mit den nichtterminalen und terminalen Symbolen durch „Äste" verknüpft, die von ihnen produziert werden. Die „Blätter" eines Ableitungsbaumes sind letztlich alle terminale Symbole. Ein Beispiel für einen Ableitungsbaum zeigt Abb. 1.8. Dargestellt ist eine Folge von Ableitungen (Regelanwendungen), die folgendes Kommando erkennt:

```
LS -L *.C | MORE
```

Geschrieben mit terminalen Symbolen lautet dieses Kommando:

```
STRING STRING STRING „|" STRING
```

Anmerkung: Es ist zu beachten, daß in der Abbildung einige der nonterminalen Symbole aus Platzgründen abgekürzt worden sind.

Wie man bereits am Beispiel in Abb.1.8 erkennt, sind Grammatiken in der Regel hochgradig rekursiv (die Regeln im Ableitungsbaum werden teilweise mehrfach verwendet). Entsprechend bietet es sich an, einen Interpreter mit dem Mittel der Rekursion zu formulieren. Und damit sind wir wieder zurückgekehrt zu *eval* und den übrigen Funktionen des Listings 1.20.

Zunächst legen wir die terminalen Symbole der Sprache - sie werden auch als *Tokens* bezeichnet - fest. Im Listing ist dies in den Zeilen 36-42 geschehen. Wie Sie feststellen können, sind hier die drei Token STRING, NUMBER und "$" durch das Token WORD ersetzt worden. In der Implementierung wird mithin davon ausgegangen, daß bereits bei der Erzeugung der Token eine Ersetzung der Shell-

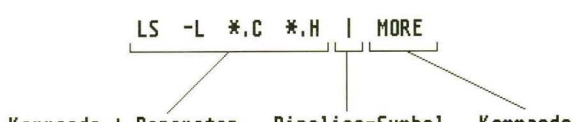


Bild 1: Beispiel für Pipelining

Wir sind Ihr starker
Atari ST Partner



Supercharger 1.4 **DM 666,--**

Kein Einlöten nötig, sondern einfach extern anschließen und fertig ist der XT / "Quasi-AT" im Lieferumfang enthalten: • **MS-DOS 4.01 • 1MB RAM • Handbuch und Toolbox.** Läuft tadellos auf dem TT.

Softwarepaket PC **DM 128,--**

Adress / Lager / Auftrag / Platten / Bücher Zeitschriften / Fakturierung mit d.Handbuch



Turbo C 2.0 Pro **DM 398,--**

inkl. Turbo Debugger/Assembler

Do you speak Turbo C?
Machen Sie mit! Wer aktuell sein will programmiert in C, der Sprache der Zukunft.

Turbo C 2.0 229,–
Turbo Debugger und Assembler 229,–



Marconi Trackerball **DM 198,--**

Die **Maus** ist **tot**, es **lebe** der **Trackball** Exaktere Cursorpositionierung, platzsparend, hohe Lebensdauer, ...einfach professioneller ! (Laut TOS 11/90 "empfehlenswert").

Mit einfacherer Mechanik:

Marconi Trackerball Lynx 98,--



Handy Scanner Typ 10 **DM 698,--**

Das Spitzenprodukt aus dem Hause Cameron! Mit hoher Auflösung (400 dpi), 16 Graustufen sowie Texterkennungs - und Grafiksoftware ein absoluter Profi unter den Scannern. 105 mm b.

Typ 3 (16 Graustufen, 200 dpi) 498,–
Typ 2 (schwarz/weiß, 200 dpi) 398,–
je 64 mm breit



Omikron Basic Compiler **DM 179,--**

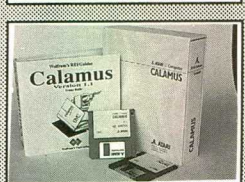
Der Hochgeschwindigkeitscompiler zum Omicron Basic.

Omikron Easybase **DM 218,--**

Einfach zu bedienende Datenbank.

Mortimer **DM 79,--**

Wohl dem, der einen Butler hat - Utility-Prg. !!



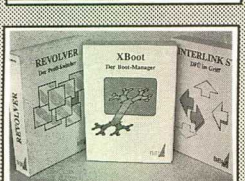
Calamus DTP **DM 698,--**

Das Spitzenprodukt im heißumkämpften Desktop-Publishing-Markt. Unheimlich leistungsstark - bei Profis längst im Einsatz.

Buch: Calamus 1.1 **DM 59,--**

Erlernen von Calamus leicht gemacht.

PKS-Edit Calamus-Texteditor **DM 148,--**



Saldo (Bela) **DM 79,--**

Preiswertes elektronisches Haushaltsbuch.

XBoot (Bela) **DM 69,--**

Äußerst praktisch für jeden Festplattenbesitzer.

New-VDI (Bela) **DM 99,--**

Softwareblitter: Machen Sie Ihrem ST Beine !!

Alles aus einer Hand ...

Software ST

Textverarbeitung

1st Word Plus 3.15 249,–
Scarabus Fonted. 100,–
That´s Write 1.5 328,–
Script2 298,–
Signum 398,–
Wordflair 239,–

CAD/Grafik

Arabesque 278,–
Arabesque Pro 368,–
CADja 998,–
Campus Art 149,–
Creator (Application) 249,–
DRAW 3.0 (Omikron) 129,–
GFA Draft Plus 348,–
Imagic (Application) 498,–
Megapaint II (Tommy) 498,–
STAD V1.3 Plus 179,–
Steve 3.2K 498,–
Technobox Drafter ... 798,–
Technobox CAD 1998,–

Calamus DTP

Outline Art 398,–
Font Editor DMC 198,–
Font Editor Didot 199,–
Calamus-Fonts:
Babble / Plub / Roca /
Yappie / Skript je 39,–
Caslo / Peking. 59,–
Aktiva / Boedet / Geodet /
Bonum / Jilly / Rund je 79,–
Repro Studio 498,–
Retouche 1198,–

Datenbanken

Adimens ST Plus 3.1 399,–
DBman 5.1 + Comp. 998,–
Masterbase 79,–
1st Adress V2.0 99,–
Superbase 249,–
Superbase Prof. 599,–
Themadat 248,–

Tabellenkalkulationen

VIP Prof. 149,–
LDW Powercalc 249,–

Buchhaltung /Fakt.

Banktransfer 498,–
Cashflow 498,–
BS Handel 498,–

fibuMAN e 398,–
fibuMAN f 768,–
fibuMAN m 968,–
Import fibuMAN 148,–
fibuSTAT 398,–

Utilities

FlexDisk 69,–
Harddisk Utility V3 69,–
Boot-iT 69,–
Copystar 3.0 169,–
Harlekin (Maxon) 129,–
HD-Sentry 139,–
HD-Accelerator 98,–
Mortimer 78,–
Neodesk 3 98,–
Turbo ST V1.8 89,–
ST-Archivar 89,–
ST-Print 69,–
ST-Plot 69,–

Midi / Musik

Cubase 1.5 798,–
Midi-Library (Omikron) . 79,–
Sampler II Maxi 8 Bit 298,–
Sampler III 16 Bit 598,–
Soundmachine II 199,–
Steinberg Twelve 99,–
Twentyfour 3.0 498,–

Lernprogramme

ST-Learn (Heim) 69,–
Geographie (Omikron) . 39,–
Learn ST plus 59,–
dto Zusatzdisks ..... je 20,–

Verschiedenes

Neu II Syntex 248,–
Syntex Demo 40,–
Sherlook 2.4 444,–
Kuma Spell 49,–
Kuma Resource II 129,–
PC-Ditto V3.96 128,–
St-Aktienstar 198,–
Reprok Büro 598,–
BTX-Manager 298,–
BTXManager DBT .. 398,–
G-Clock steckbar 79,–

Programmiersprachen

GFA EWS 2.0 49,–
GFA EWS 3.0 198,–
GFA EWS 3.5 268,–
GFA - C Konverter ... 498,–
GFA Assembler 149,–

Lattice C-Comp. 298,–
Megamax Laser-C ... 348,–
Dt. Hb. Megamax-C ... 49,–
Maxon Pascal 1.0 248,–
Megamax Modula2 .. 398,–
MCC Assembler 169,–
MCC Lisp 298,–
MCC Pascal 298,–
Omikron Basic V3.0 ... 19,90
Omikron Comp. Jun. . 99,–
Turbo - C 1.1 178,–
Mas/Bug 68K 169,–

Zubehör ST

Weide Produkte

Echtzeituhr 129,–
512KB Erweiterung . **249,–**
2/4 MB mit 2 MB best. **598,–**
4 MB mit 4 MB best. ... **898,–**

MAXON Produkte

Easytizer fertig **289,–**
Easytizer Teilesatz . 129,–
Junior Prommer fert. 229,–
Jun. Pr. Teilesatz 59,–
MGP-Gal Pr. fertig ... 229,–
dto Teilesatz 129,–
DPE Teilesatz 59,–
Zubehör Portfolio:
32 K Ramkarte 108,–
64 K Ramkarte 188,–
128 K Ramkarte 298,–
256 K Speichererw. . 398,–
Folio-Talk 98,–

Verschiedenes

Logimouse Pilot 99,–
Pal Interface 198,–
Monitorumschalter 59,–
Akustikkoppler 300 .. 278,–
...300/1200 BTX 378,–
2400 Baud Dataph. . 698,–
Atari TOS 1.4 (2/6er)198,-

ATARI-Schaltpläne

Für Rechner ..... je 29,80
Für Monitore ..... je 19,80
Für Drucker ..... je 19,80

Abdeckhauben

für 520/1040/MEGA ... 24,80
für Monitore 29,80
für MEGA & SM124 ... 39,80
für MEGA Tast/SF31414,80

Atari -PD

St / PD 2000 PD 5000
und AT Serie pro Disk. 8,--

Hardware und Preisknüller

1040 STF mit SM 124 1198,--
1040 STF mit SC1224 1498,--
1040 STE mit SM 124 1398,--
1040 STE mit SC1224 1698,--
MEGA 1 mit SM 124 1398,--
MEGA 2 mit SM 124 1798,--
MEGA 4 mit SM 124 2498,--
Portfolio 498,--
STACY 1 3400,--

Festplatten / Laufwerke / Drucker

MEGA File 30 848,--
MEGA File 60 1298,--
MEGA File 44 1898,--
ATARI CAD ROM mit Medium **998,--**
Laufwerk SF354 98,--
Laserdrucker SLM 804 2698,--
Laserdrucker SLM 605 2298,--
Amstrad Laser 512 Kb FX80 E. 1998,--

Antiviren Kit GDATA 68,--
Wordperfekt 298,--
Bolo 55,--
PKS Write 169,--
1st Word 49,--
Mono Star 28,--
k Graph 3 149,--
k Spread 3 249,--
Logistix 249,--
2nd Word 45,--
PC Speed 298,--
AT Speed 438,--
GFA Buch TOS & GEM 19.80,--

Wir über uns!!!

1. **ATARI Vertrags Händler**
2. **MARCONI Distributor in der BRD**
3. **eigene Werkstatt sehr wichtig**
4. **Laden und Versandgeschäft**
5. **eigene Entwicklungsabteilung**

Karl-Heinz Weeske
Potsdamer Ring 10
D-7150 Backnang

Kreissparkasse BK • BLZ (60250020)
74397 • Postgiro Stgt. 83326-707 •

weeske
COMPUTER-ELEKTRONIK

Zahlung per Nachnahme oder Vorauskasse. Versandkostenpauschale: Inland 7,80 DM (Ausland 19,80 DM)

Tel.: 07191-1528(29), 60076
Fax: 07191-60077

zurück an Absender 01/91

Interessiert an weiterem Info-Material ? Bitte ankreuzen !

- ◯ Hardware Atari ST
- ◯ Software + Zubehör Atari ST
- ◯ Public Domain Liste (DM 2,50)

Spezielle Info auf Anfrage !!

Vorname, Name:

Straße, Haus-Nr:

PLZ, Ort:

Telefon-Nr, Datum:

Mein Computersystem:

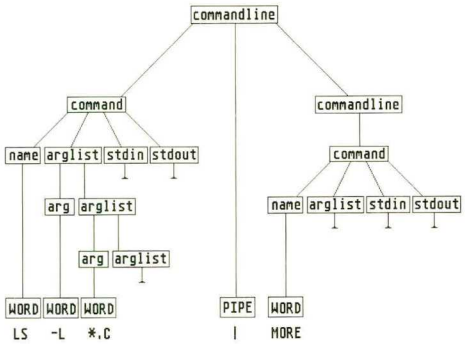


Bild 2: Ableitungsbaum einer Kommandozeile

Variablen durch Argumente von SSH stattfindet. Ferner finden sich in den Zeilen 36-42 noch Transskriptionen der terminalen Symbole "|" (PIPE), "<" (INSYM) und ">" (OUTSYM) und ein Token zur Markierung des Endes einer Kommandozeile (END). Die Generierung der Tokens wird der Funktion *nextword* (Zeilen 126-181) übertragen. Im Inneren von *nextword* befindet sich dazu eine while-Schleife, mit der die in *cline* gegebene Kommandozeile durchlaufen und entsprechend der enthaltenen Zeichen interpretiert wird. Das von *nextword* gelesene Token wird dabei in der globalen Variablen *gtoken* abgelegt. Handelt es sich bei dem Token um WORD, wird zusätzliche eine um die Shell-Variablen expandierte Version des Wertes von WORD in *gword* abgelegt. Folgende Fälle sind dabei zu unterscheiden:

Zunächst besteht die Möglichkeit des Zeilenendes. Dies wird bereits vor Betreten der Schleife abgeprüft (Zeilen 131-134). Entsprechend wird hier das Token END erzeugt und die Funktion *nextword* verlassen.

Die drei terminalen Symbole ">", "<" und "|" können innerhalb der Schleife direkt in Tokens umgesetzt werden (Zeilen 136-148). Auch hier kann die Funktion *nextword* direkt verlassen werden.

Ein (all)gemeinerer Fall liegt vor, wenn das Token WORD erkannt wird (Zeilen 152-177). Nun reicht es nicht, einen entsprechenden Wert in *gtoken* abzulegen. Es ist vielmehr zusätzlich notwendig die genauen Begrenzungen von WORD in der Kommandozeile zu bestimmen, um auch den Wert des Tokens in der Variablen *gword* ablegen zu können. Außerdem müssen auch noch eingeklammerte Bereiche einer Kommandozeile gesondert behandelt werden, da damit angezeigt wird, daß der geklammerte Bereich, unabhängig von eventuell enthaltenen Leerzeichen, ein einziges Argument bildet. All dies geschieht in den Zeilen 153-173. Es wird dabei jedoch noch nicht berücksichtigt, daß das gelesene Wort auch noch Shell-Variablen enthalten kann. Diese werden nachträglich durch einen Aufruf von *expand_gword* (Zeilen 96-124) in die globale Variable *gword* eingefügt. Innerhalb von *expand_gword* wird dazu eine Kopie von *gword* im String *erg* angelegt. Immer dann jedoch, wenn bei dem Kopiervorgang in gword eine Shell-Variable auftritt, wird statt dieser deren Wert in *erg* übernommen (Zeilen 105-113). Letztlich wird erg wieder nach *gword* zurückkopiert (Zeile 123), und die Expansion ist beendet.

Anmerkung: Die vorliegende Implementierung gibt keine Fehlermeldung für unzulässige Shell-Variablen aus. Stattdessen wird die betreffende Expansion einfach nicht vorgenommen, d.h. die Shell-Variable wird durch einen leeren String ersetzt.

Zusammengenommen läßt sich jede Kommandozeile durch eine Folge der vier Tokens WORD, PIPE, INSYM und OUTSYM, abgeschlossen mit dem Token END, darstellen. Die Funktionen *eval* (Zeilen 286-299), *commandline* (Zeilen 253-284) und *command* (Zeilen 197-251) bekommen nun die Aufgabe, eine derartige Sequenz von Tokens auf Verträglichkeit mit der Grammatik der Kommandozeilen zu untersuchen. Wie bereits angedeutet, sind diese Funktionen dabei rekursiv eingeschachtelt und es ist auch kein Zufall, daß sie die Namen der Grammatikregeln tragen, da die Überprüfung einer Regel jeweils von der gleichnamigen Funktion vorgenommen wird. In ihrer Gesamtheit bilden die Funktionen dabei einen sogenannten *Parser*. Das ist nichts anderes als eine Bezeichnung für die Funktionen, die ein Programm auf Verträglichkeit mit der zugrundeliegenden Sprachsyntax (Grammatik) checkt. Nebenbei wird in den drei Funktionen aber auch noch mehr gemacht: die Kommandozeilen, von denen wir durch die Parser-Eigenschaft der Funktionen wissen, daß sie korrekt sind, werden umgesetzt in Kommandozeilen, die auf dem ST ausführbar sind.

An dieser Stelle ist es daher angebracht, noch einmal kurz zu resümieren, welche Art von Kommandozeilen auf dem ST direkt ausführbar sind. Dies sind gerade die TTP-Programme, d.h. TOS-Programme mit der Möglichkeit, eine Kommandozeile zu übergeben. Diese darf dabei Parameter und Umlenkungen von Ein- und Ausgabe beinhalten. Da wir die Fähigkeit zur Verwendung von Shell-Variablen bereits bei der Generierung der Tokens eingebaut haben, verbleibt noch das Pipelining, welches auf die herkömmlichen Kommandozeilen abgebildet werden muß. Betrachten wir eine Pipeline folgender Art:

```
PROG_1.TTP | PROG_2.TTP |
PROG_3.TTP | PROG_4.TTP
```

Hieraus wird ein Folge von Kommandoaufrufen generiert, die ohne die Verwendung von Pipelines auskommen, nichtsdestotrotz aber die gleiche Funktionalität besitzen wie der vorhergehende Aufruf:

```
PROG_1.TTP > 00000000.TMP
PROG_2.TTP < 00000000.TMP >
             00000001.TMP
PROG_3.TTP < 00000001.TMP >
             00000002.TMP
PROG_4.TTP < 00000002.TMP
RM.TTP *.TMP
```

Mit Hilfe von temporären Dateien werden dabei Kommandozeilen erzeugt, die direkt in TOS ausführbar sind. Hierzu ist nur der Kommandoname von der nachfolgenden Kommandozeile abzutrennen und ein *Pexec*-Aufruf mit den erhaltenen Parametern durchzuführen.

Innerhalb der Funktionen des Parsers werden „nebenbei", aus den gelesenen Informationen, Kommandozeilen der obigen Art generiert (Zeilen 191, 211+212, 240+241, 262+263 und 267). Eine Hilfsfunktion - *tempfile*, Zeilen 183-187 - generiert dazu die Namen für die temporären Dateien. An Programmstellen, wo jeweils eine komplette ST-Kommandozeile (Der Begriff "ST-Kommandozeile" wird hier im Sinne von "einer für den ST direkt verständlichen Kommandozeile" verstanden.) generiert worden ist, erfolgt ein *execute*-Aufruf (Zeilen 264 und 271). *execute* (Zeilen 414-461) übernimmt die Ausführung dieser Kommandozeile. Zunächst ist diese hierzu in Kommandoname (*cmd*) und Parameterzeile (*cmdline*) aufzuteilen (Zeilen 424-436). Weiterhin wird die Parameterzeile noch in einen Pascal-String übersetzt (Zeile 437, String *pcmdline*), damit der Parameter in *Pexec* verwendet werden kann. Nun ist zu entscheiden, ob ein internes oder ein externes Kommando vorliegt. Liegt ein internes

vor, wird die weitere Ausführung von der Funktion *intexecute* (Zeilen 361-412) übernommen. Ansonsten wird in *execute* verblieben. Hier sind nun noch Vorkehrungen zu treffen für den Fall, daß dem Kommandonamen die Typbezeichnung TTP fehlt. In diesem Fall wird die Endung TTP automatisch angehängt (Zeilen 446-447). Weiterhin wird zunächst versucht, das Kommando im Arbeitsverzeichnis auszuführen (Zeile 448). Schlägt dieser Versuch fehl, wird noch versucht, das Kommando im Verzeichnis *PATH* auszuführen (Zeilen 450-458). *PATH* ist eine in der Zeile 25 definierte Konstante, die dazu dient, das Verzeichnis anzugeben, das die PTB-Kommandos enthält.

Zurück zur Funktion *intexecute*. Wie bereits gesagt, wird von *intexecute* die Ausführung der internen Kommandos übernommen. Entsprechend finden sich oberhalb von *intexecute* noch einmal die Vereinbarungen der Kommandos CD und PWD. Sie werden von *intexecute* aufgerufen. Der größte Teil des Anweisungsteils von *intexecute* ist jedoch nicht damit beschäftigt, die Funktionsaufrufe zu gewährleisten, dies geschieht ganz nebenbei am Ende der Funktion (Zeilen 402-409). Nein, die vorangehenden Anweisungen wandeln vielmehr die Kommandozeile so um, wie es das Betriebssystem beim Aufruf eines Programms durchführt. D.h. es werden die Parameter *argc* und *argv* konstruiert, um eine möglichst leichte Einbindung weiterer interner Kommandos zu gewährleisten (Zeilen 370-401). Auf diese Weise bleibt es dem Leser belassen, das Kommando SSH um weitere (interne) Kommandos zu bereichern. Sie werden einfach eingehängt, indem die if-Anweisungen in den Zeilen 439-441 und in 402-406 entsprechend ergänzt werden, und indem das betreffende Kommando (unter Streichung der Funktion *main* innerhalb des jeweiligen Kommandos) in Listing 1.20 aufgenommen wird.

Vorausschau

Mit dieser Anregung ist die Vorstellung der Shell abgeschlossen. Damit endet auch der erste thematische Block der „Programmer's Toolbox". Innerhalb der weiteren Blocks wird nichts mehr an der Shell geändert, stattdessen werden lediglich noch eine Reihe externer Kommandos erstellt. Eine detailliertere Vorausschau finden Sie im nächsten Serienteil. Für heute soll nur soviel verraten werden: Es geht um die Verarbeitung von Text.

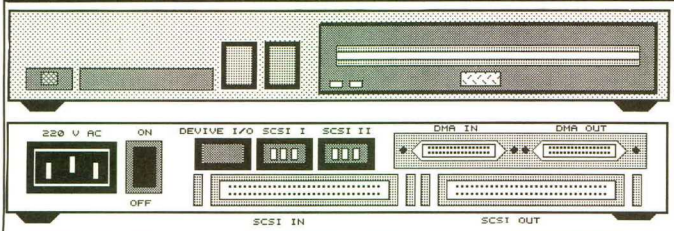
Dirk Brockhaus

```
 1:  /*
 2:   * Listing 1.20, Datei : ssh.c
 3:   * Programm            : SSH - Eine einfache Shell
 4:   * Modifikationsdatum   : 12-Dez-89
 5:   * Abhängigkeiten       : stdio.h, osbind.h,
                               string.h,
 6:   *                       ctype.h, local.h, atom.h
 7:   */
 8:
 9:  #include <stdio.h>
10:  #include <osbind.h>
11:  #include <string.h>
12:  #include <ctype.h>
13:  #include "local.h"
14:  #include "atom.h"
15:
16:  /*
17:   * Konstante     : PATH
18:   *
19:   * Aufgabe       :
20:   *
21:   * Enthält das Verzeichnis in dem zusätzlich zum
22:   * aktuellen Verzeichnis gesucht wird.
23:   */
24:
25:  #define PATH "C:\\BIN\\"
26:
27:  /*
28:   * Datentyp      : TOKEN
29:   *
30:   * Aufgabe       :
31:   *
32:   * Aufzählungstyp für die lexikalische Analyse der
33:   * SSH-Kommandos.
34:   */
35:
36:  typedef enum {
37:      WORD,
38:      PIPE,
39:      INSYM,
40:      OUTSYM,
41:      END
42:  } TOKEN;
43:
44:  /*
45:   * Variablen     : gword, ginfile, rest, cmdline
46:   *                 gtoken, remove, tempcnt gargc,
47:   *                 gargv
48:   *
49:   * Aufgabe       :
50:   *
```

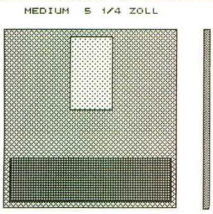
```
51:   * Einige globale Variablen, die während der
        Analyse
52:   * und Interpretation von SSH-Kommandos benötigt
53:   * werden.
54:   */
55:
56:  static char    gword[255],
57:                 ginfile[15],
58:                 *rest,
59:                 cmdline[255];
60:  static TOKEN   gtoken;
61:  static BOOLEAN remove;
62:  static short   tempcnt,
63:                 gargc;
64:  static char    **gargv;
65:
66:  /*
67:   * Funktionen    : expand_gword, nextword,
                        tempfile,
68:   *                 wordlist, command, commandline,
69:   *                 eval
70:   *
71:   * Parameter     : expand_gword();
72:   *                 newstart = nextword(oldstart);
73:   *                 tempfile(filename);
74:   *                 wordlist();
75:   *                 ok = command(infile);
76:   *                 ok = commandline(infile);
77:   *                 eval(str);
78:   *                 char    *newstart,
79:   *                         *oldstart,
80:   *                         *str,
81:   *                         *filename;
82:   *                 BOOLEAN ok,
83:   *                         infile;
84:   *
85:   * Aufgabe       :
86:   *
87:   * Die nachfolgenden Funktionen bilden das
        Kernstück
88:   * bei der Interpretation von SSH-Kommandos.
89:   * <expand_gword> und <nextword> nehmen die
90:   * lexikalische Analyse vor. Die übrigen
        Funktionen setzen
91:   * die SSH-Kommandos in ST-Kommandozeilen um.
92:   * Dabei wird auch gleichzeitig eine korrekte
93:   * Syntax sichergestellt.
94:   */
95:
96:  void expand_gword()
97:  {   char erg[128],                                →
```

```
 98:             work[2],
 99:             *gptr = &gword[0];
100:     int  argno;
101:
102:     strcpy(erg, "");
103:     strcpy(work, " ");
104:     while (gptr[0] != 0) {
105:         if (gptr[0] == '$') {
106:             gptr++;
107:             argno = 0;
108:             while (gptr[0] != 0 &&
109:                    isdigit(gptr[0])) {
110:                 argno = argno * 10 +
111:                         todigit(gptr[0]);
112:                 gptr++;
113:             }
114:             if (argno >= 1 && argno < gargc)
115:                 strcat(erg, gargv[argno]);
116:         }
117:         else {
118:             work[0] = gptr[0];
119:             strcat(erg, work);
120:             gptr++;
121:         }
122:     }
123:     strcpy(gword, erg);
124: }
125:
126: char *nextword(cline)
127: char *cline;
128: {   char  *work = cline;
129:     short i;
130:
131:     if (cline == NULL || strlen(cline) == 0) {
132:         gtoken = END;
133:         return(NULL);
134:     }
135:     while (work != 0) {
136:         if (work[0] == '<') {
137:             gtoken = INSYM;
138:             strcpy(gword, "<");
139:             return(++work);
140:         } else if (work[0] == '>') {
141:             gtoken = OUTSYM;
142:             strcpy(gword, ">");
143:             return(++work);
144:         } else if (work[0] == '|') {
145:             gtoken = PIPE;
146:             strcpy(gword, "|");
147:             return(++work);
148:         } else if (work[0] == ' ' ||
149:                    work[0] == '\t')
150:             ++work;
151:         else {
152:             gtoken = WORD;
153:             i = 0;
154:             if (work[0] == '\"') {
155:                 do {
156:                     gword[i] = work[i];
157:                     i++;
158:                 } while(work[i] != 0 &&
159:                         work[i] != '\"');
160:                 gword[i] = '\"';
161:                 i++;
162:             }
163:             else
164:                 do {
165:                     gword[i] = work[i];
166:                     i++;
167:                 } while(work[i] != 0 &&
168:                         work[i] != ' ' &&
169:                         work[i] != '\t' &&
170:                         work[i] != '>' &&
171:                         work[i] != '<' &&
172:                         work[i] != '|');
173:             gword[i] = 0;
174:             expand_gword();
175:             if (work[i] == 0)
176:                 return(NULL);
177:             return(&work[i]);
178:         }
179:     }
180:     return(work);
181: }
182:
183: void tempfile(filename)
184: char *filename;
185: {   remove = TRUE;
186:     sprintf(filename, "%08d.TMP", tempcnt++);
187: }
188:
189: void wordlist()
190: {   if (gtoken == WORD) {
191:         sprintf(cmdline, "%s %s", cmdline,
                    gword);
192:         rest = nextword(rest);
193:         wordlist();
194:     }
195: }
196:
197: BOOLEAN command(infile)
198: BOOLEAN infile;
199: {   char    *work;
200:     BOOLEAN outfile = FALSE;
201:
202:     if (gtoken != WORD) {
203:         fprintf(stderr, "ssh: WORD expected\n");
204:         return(FALSE);
205:     }
206:     else {
207:         sprintf(cmdline, "%s %s", cmdline,
                    gword);
208:         rest = nextword(rest);
209:         wordlist();
210:         if (infile)
211:             sprintf(cmdline, "%s %s", cmdline,
212:                     ginfile);
213:         if (gtoken == INSYM) {
214:             if (infile) {
215:                 fprintf(stderr,
216:      "ssh: Only one stdin-Redirection is
                allowed\n");
217:                 return(FALSE);
218:             }
219:             infile = TRUE;
220:             rest = nextword(rest);
221:             if (gtoken == WORD) {
222:                 sprintf(cmdline, "%s < %s",
                            cmdline,gword);
224:                 rest = nextword(rest);
225:             }
226:             else {
227:                 fprintf(stderr, "ssh: Missing
                            WORD\n");
228:                 return(FALSE);
229:             }
230:         }
231:         if (gtoken == OUTSYM) {
232:             if (outfile) {
233:                 fprintf(stderr,
234:      "ssh: Only one stout-Redirection is
                allowed\n");
235:                 return(FALSE);
236:             }
237:             outfile = TRUE;
238:             rest = nextword(rest);
239:             if (gtoken == WORD) {
240:                 sprintf(cmdline, "%s > %s",
                            cmdline,gword);
242:                 rest = nextword(rest);
243:             }
244:             else {
245:                 fprintf(stderr, "ssh: Missing
                            WORD\n");
246:                 return(FALSE);
247:             }
248:         }
249:         return(TRUE);
250:     }
251: }
252:
253: BOOLEAN commandline(infile)
254: BOOLEAN infile;
255: {   char *work,
256:          filename[15];
257:     void execute();
258:
259:     if (command(infile)) {
260:         if (gtoken == PIPE) {
261:             tempfile(filename);
262:             sprintf(cmdline, "%s > %s", cmdline,
263:                     filename);
264:             execute(cmdline);
265:             strcpy(cmdline, "");                    →
266:             rest = nextword(rest);
267:             sprintf(ginfile, "< %s", filename);
```

DIE EINZIGARTIGE ALTERNATIVE ZUR WECHSELPLATTE



* ICD ADVANTAGE- ADAPTER * ANSCHLUSS
* WIE HARDDISC * GEHT MIT TOS 1.0 BIS 1.6
* 100% ATARI- UND AHDI- 3.XX- KOMPATIBEL
* 256 MB/PARTITION MAX (THEOR.)
* 14 PARTITIONEN MAXIMAL * BOOTPART. FREI WAEHLBAR * KOMPATIBEL ZU: VORTEX, PC-, AT-SPEED, SUPERCHARGER, PC-DITTO, SPECTRE, ALADIN U.A. * HARDWARE-SCHREIBSCHUTZ
* SOFORT BOOT-FAEHIG * SPURENCACHING 16 KB * EINZIGARTIGE FEHLERKORREKTUR UND SICHERHEIT * SEHR HOHE LEBENSDAUER VON MEDIUM UND GERAET * EINFACHSTE BEDIENUNG * DEUTSCHES HANDBUCH
1 JAHR GARANTIE * RUECKGABE - RECHT ****



DATEN	BESCHREIBUNG
250-300 KB/S	UEBERTRAGUNGSRATE DURCHSCHNITTLICH
20,05 / 24 MB	SPEICHERKAPAZITAET FORMATIERT/UNFORMATIERT PRO DISC
55-65 MS	MITTLERE ZUGRIFFSZEIT
JA	MEDIA-DISC-CHANGE PER ESCAPE-TASTE
JA - 0 UND 1	DMA-ADRESSE VON HINTEN EINSTELLBAR
JA	DMA-BUS GEPUFFERT IN/OUT
OPTION	SCSI-BUS HERAUSGEFUEHRT
GEHAEUSE	ATARI-MEGA-DESIGN MIT ABGERUNDETEN KANTEN - GRAU
NETZTEIL	VDE-SCHALTNETZTEIL, REICHT AUS FUER ZWEI SYSTEME
INTERFACE	ICD-HOSTADAPTER, KOMPL. SCSI-BEFEHLSSATZ NACH ANSI
ECHTZEITUHR	JE NACH WUNSCH OHNE AUFPREIS DMA-ADRESSE 6
SOFTWARE	KOMPL. SOFTWARE, GENAUSO WIE FUER FESTPLATTEN ANBEI
EMPFINDLICHK.	WESENTL. UNEMPFINDLICHER GEGEN SCHOCK, STAUB ALS HARDDISC
LEBENSDAUER	10-30 MAL HOEHER ALS BEI SEHR GUTEN FESTPLATTEN
LUEFTER	KEIN LUEFTER VORH, EXTREM LEISES LAUFWERK
ERROR CORREKT.	AUTOM. FEHLERKORREKTUR+ERKENNUNG, SPUREN-CHACHING

GRUNDAUSTATTUNG : SCSI-SCHALTER
AUSBAUSTUFE 1 SCSI-BUS extern
AUSBAUSTUFE 2 SCSI-BUS in-out
AUSBAUSTUFE 1 : +100 DM, 2: +200 DM

SIEHE ST_COMP.11-90
Seite 60 TEST !
DM : 1898.- MEDIUM: 79 DM
MIT VERBATIM-LAUFWERK
AT - SPEED : 449 DM V: 2.22

ATARI TT (32 MHz)
mit VGA-MULTISYNC, ab 4MB HAUPTSPEICHER:

OHNE MONITOR UND OHNE FESTPLATTE
SOFORT LIEFERBAR!
DAFÜR EINBAU NACH WUNSCH:
FEST- ODER WECHSELPLATTE!

ATARI-PAKETE: SM 124 + MEGAFILE 30 + WORD PERFECT:
MEGA-1- PAKET: 1998 DM | MEGA-1 +SM124 solo :1.398 DM | SM 194 Bigscreen: 3698DM
MEGA-2- PAKET: 2198 DM | MEGA-2 +SM124 solo :1.598 DM

ST-3 720 KB DISKETTENLAUFWERK (TEAK) 199 DM
ST-5 360,720,1.2 MB 5.25"Laufw. (TEAK) 299 DM
Komplett incl Netzteil, Gehäuse, Kabel ect

FISCHER COMPUTER GOETHE-7 6101 FR.-CRUMBACH 06164-4601 FAX: 3748

SCSI FESTPLATTEN NOCH MODERNER UND ZUVERLÄSSIGER

SUPER AUSSTATTUNG ALLE GERATE WERDEN KOMPLETT ANSCHLUSSFERTIG AUSGELIEFERT !!
MEGA-ST-STAHLBL GEH GRAU, NETZKABEL 2m, DMA - KABEL, HANDBUCH, KURZANLEITUNG, DISKETTEN
ICD-ADAPTER ADVANTAGE ICD-TREIBER-UND APPLICATION-SOFTWARE
100 % ATARI-ST --- ATARI-TT KOMPATIBEL AHDI 3.xx
MS-DOS-KOMPATIBEL PC-SPEED, AT-SPEED, SUPERCHARGER, PC-DITTO, IBM-KOMP
SPECTRE, ALADIN, OS 9, RTOS, MINIX SCSI-BUS EXTERN UND DMA_ADR-SCHALTER extra!
VORTEX, LASERDRUCKER, SCANNER-- keine Probleme !!! NETZWERKFÄHIG !!!
LÄUFT UNTER TOS 1.0 bis 2.0 mit allen Rechnern.
SCSI-ADRESSE : DMA: 0-7
DMA GEPUFFERT IN - OUT AUTOPARK HARDWAREMÄSSIG
64 kB CACHE_SPEICHER 64 kB SOFTWAREMASSIG, 32 kB HARDWAREMASS
AUTOBOOTFÄHIG VON ALLEN PARTITIONEN, BOOT FREI WAHLBAR - ACCESSORYSTEUERUNG (GEM)
14 PARTITIONEN UNTER TOS, PARTITIONEN EINZELN PARTITIONIERBAR (TOS,DOS..)
KONSTANTGEREGELTER LÜFTER EINZIGARTIGE RETRY - VERIFY - FUNKTION
SUPERLEISE -- NEUESTE LAUFWERKE EINBAU EINES ZWEITEN CHASSIS - JA !
ALLE PLATTEN HABEN INTERLEAVE 1-1

MHD 50	48 MB (SEAGATE ST 157N) 28 ms, 620kB-s 3.5 zoll	998 DM
MHD 80	85 MB (SEAGATE ST 296N) 28 ms, 550kB-s 5.25 zoll	1.098 DM
MHD 81	84 MB (S.-IMPRIMIS 1096 N) 24 ms, 770kB-s 3.5 zoll	1.248 DM
MHD 140	140MB ("-") NUR 15 ms, 1020KB-s, 32kB CACHE 3.5 zoll	1.998 DM
MHD 170	170MB ("-") NUR 15 ms, 1200KB-s, 64kB CACHE 3.5 zoll	2.398 DM
MHD 210	210MB ("-") NUR 15 ms, 1250KB-s, 64kB CACHE 3.5 zoll	2.545 DM

SCSI-BUS: 50 DM
DMA_SCHALTER:
AUFPREIS: 25 DM

neu QUANTUM 1 zoll 17 ms, 1440 kB-s, 64+64 kB CACHE, prog bar
besser und schneller als die alten Quantum-Platten

MHD 52 PRO	52 MB	1.498 DM
MHD 105 PRO	105 MB	1.898 DM

DIE PRO-SERIE WIRD MIT Backup-Programm plus Optimizer ausgeliefert
Alle Platten sofort lieferbar
2 Jahre Garantie !!

Michael Fischer • Computersysteme
Goethestr. 7 • 6101 Fränkisch-Crumbach • Tel. 0 61 64 / 46 01

```
268:                return(commandline(TRUE));
269:            }
270:            else if (gtoken == END) {
271:                execute(cmdline);
272:                strcpy(cmdline, "");
273:                return(TRUE);
274:            }
275:            else {
276:            fprintf(stderr, "ssh: End-of-line
                        expected\n");
277:            return(FALSE);
278:            }
279:        }
280:        else {
281:            fprintf(stderr, "ssh: Execution
                        fails\n");
282:            return(FALSE);
283:        }
284:    }
285:
286:    void eval(cline)
287:    char *cline;
288:    {   void execute();
289:
290:        rest = nextword(cline);
291:        tempcnt = 0;
292:        remove = FALSE;
293:        strcpy(cmdline, "");
294:        commandline(FALSE);
295:        if (remove) {
296:            strcpy(cmdline, " rm.ttp *.TMP");
297:            execute(cmdline);
298:        }
299:    }
300:
301:    /*
302:     * Funktionen    : cd, pwd, intexecute, execute
303:     *
304:     * Parameter     : ok = cd(argc, argv);
305:     *                 ok = pwd(argc, argv);
306:     *                 ok = intexecute(cmd, cmdline);
307:     *                 execute(cline);
308:     *                 BOOLEAN ok;
309:     *                 short   argc;
310:     *                 char    **argv,
311:     *                         *cmd,
312:     *                         *cmdline,
313:     *                         *cline;
314:     *
315:     * Aufgabe       :
316:     *
317:     * Die nachfolgenden Funktionen übernehmen die
318:     * Ausführung von internen und externen
          Kommandos.
319:     * Die drei Kommandos <cd>, <pwd> und <exit>
320:     * sind hier intern realisiert. Alle übrigen
321:     * Kommandos werden nachgeladen. Die Funktion
322:     * <intexecute> dient der Ausführung interner
323:     * Kommandos; die Funktion <execute> führt
324:     * externe Kommandos aus, dabei sortiert sie
325:     * zuvor die internen Kommandosaus und leitet
          ihre Interpretation ein.
326:     */
327:
328:    BOOLEAN cd(argc, argv)
329:    short argc;
330:    char  *argv[];
331:    {   if (argc == 2) {
332:            convupper(argv[1]);
333:            if (!acd(argv[1])) {
334:                fprintf(stderr,
335:                        "cd: can't find
                                directory\n");
336:                return(FALSE);
337:            }
338:        }
339:        else {
340:            fprintf(stderr,
341:                    "cd: need exactly one
                            parameter\n");
342:            return(FALSE);
343:        }
344:        return(TRUE);
345:    }
346:
347:    BOOLEAN pwd(argc, argv)
348:    short argc;
349:    char  *argv[];
```

```
350:    {   char actdir[100];
351:
352:        if (argc == 1) {
353:            apwd(actdir);
354:            printf("%s\n", actdir);
355:            return(TRUE);
356:        }
357:        fprintf(stderr, "pwd: no parameter
                    expected\n");
358:        return(FALSE);
359:    }
360:
361:    BOOLEAN intexecute(cmd, cmdline)
362:    char *cmd, *cmdline;
363:    {   short   argc = 1,
364:                i;
365:        char    **argv,
366:                *work;
367:        BOOLEAN first,
368:                ok;
369:
370:        work = cmdline;
371:        first = TRUE;
372:        while (work[0] != 0) {
373:            if (work[0] != ' ') {
374:                if (first) {
375:                    first = FALSE;
376:                    argc++;
377:                }
378:            }
379:            else
380:                first = TRUE;
381:            work++;
382:        }
383:        argv = (char **)malloc(4 * (argc - 1));
384:        argv[0] = "";
385:        work = cmdline;
386:        first = TRUE;
387:        i = 1;
388:        while (work[0] != 0) {
389:            if (work[0] != ' ') {
390:                if (first) {
391:                    first = FALSE;
392:                    argv[i] = work;
393:                    i++;
394:                }
395:            }
396:            else {
397:                first = TRUE;
398:                work[0] = 0;
399:            }
400:            work++;
401:        }
402:        if (strcmp(cmd, "CD") == 0)
403:            ok = cd(argc, argv);
404:        else if (strcmp(cmd, "PWD") == 0)
405:            ok = pwd(argc, argv);
406:        else if (strcmp(cmd, "EXIT") == 0)
407:            exit(0);
408:        else
409:            ok = FALSE;
410:        free(argv);
411:        return(ok);
412:    }
413:
414:    void execute(cline)
415:    char *cline;
416:    {   char cmd       [128],
417:             cmdline   [128],
418:             pcmdline  [128],
419:             directory[128],
420:             filename  [128],
421:             *work;
422:        long erg;
423:
424:        while (cline[0] == ' ')
425:            cline++;
426:        work = cline;
427:        while (work[0] != 0 && work[0] != ' ')
428:            work++;
429:        if (work[0] == 0)
430:            strcpy(cmdline, "");
431:        else
432:            strcpy(cmdline, &work[1]);
433:        if (work[0] == ' ');
434:            work[0] = 0;
435:        strcpy(cmd, cline);
436:        convupper(cmd);
```

→

```
437:        ctop(cmdline, pcmdline);
438:        pathsplit(cmd, directory, filename);
439:        if (strcmp(cmd, "CD") == 0 ||
440:            strcmp(cmd, "PWD") == 0 ||
441:            strcmp(cmd, "EXIT") == 0) {
442:            if (!intexecute(cmd, cmdline))
443:              fprintf(stderr, "ssh: Execution
                          fails.\n");
444:            return;
445:        }
446:        else if (onlyalpha(filename))
447:            strcat(cmd, ".TTP");
448:        erg = Pexec(0, cmd, pcmdline, "");
449:        if (erg < 0) {
450:            pathsplit(cmd, directory, filename);
451:            if (strcmp(directory, "") == 0) {
452:                strcpy(cmd, PATH);
453:                strcat(cmd, filename);
454:                erg = Pexec(0, cmd, pcmdline, "");
455:                if (erg < 0)
456:                    fprintf(stderr,
457:                            "ssh: Execution fails.
                                  %s %ld\n",
458:                            "Return value is", erg);
459:            }
```

```
460:        }
461:    }
462:
463:    void main(argc, argv)
464:    short argc;
465:    char  **argv;
466:    {   char cmdline[127],
467:             *work;
468:
469:        if (isatty(STDIN)) {
470:            printf("SSH - Simple SHell\n");
471:            printf("(c) 12/89 by D.Brockhaus\n");
472:        }
473:        gargc = argc;
474:        gargv = argv;
475:        do {
476:            if (isatty(STDIN))
477:                printf("$ ");
478:            if (gets(cmdline) == NULL)
479:                exit(0);
480:            if (!(strcmp(cmdline, "") == 0 ||
481:                  cmdline[0] == '#'))
482:                eval(cmdline);
483:        } while (TRUE);
484:    }
```


STÜCKLISTEN verwalten mit K-FAKT ST (Fakturierung) — FAKTURIERUNG * HOBBY BUCHFÜHRUNG * LERNEN

COMPTABLE ST (Buchführung) DM 198.00
Ein Buchführungsprogramm für Geschäft und Privat. 500 Konten (DATEV), 10 Steuersätze, bis zu 10000 Buchungen. Steuersätze und Privatanteilsätze können bereits im Kontenrahmen vorgegeben werden. Auswertungen: Saldenliste, Kontenblätter, Journal, Kassenbuch, Gewinn/Verlust-Rechnung, Umsatzsteuerdaten jeweils für einen beliebigen Monat, ein beliebiges Quartal oder das ganze Jahr! Alle Ausgaben auf Bildschirm, Drucker oder Datei. Buchungsmemo für 25 Buchungen (einfach durch Anklicken abrufbar). Universelle Druckeranpassung. Incl. 50,– seitigem, reichhaltig illustriertem Handbuch (mit Ausdruckbeispielen) im Ringordner! DEMO-DISKETTE DM 20,–

K-FIBU ST (Kleine Finanzbuchhaltung) DM 398.00
Leistungen wie COMPTABLE ST, zusätzlich Bilanz, erweiterte Umsatzsteuerdaten-Auswertung, und Modulschnittstelle für Buchungsimport (z.B. von K-FAKT ST). DEMO-DISKETTE DM 20,–; HANDBUCH VORAB DM 50,– (Handbuch wird angerechnet)

K-FAKT ST (Professionelles Fakturierungssystem) DM 398.00
Adress-, Artikel und Stücklistenverwaltung. Umsatz/Absatzstatistik, frei wählbare Indexfelder. Erstellung von Angebot, Auftr.-Best., Lieferschein, Rechnung, Storno, Gutschrift, Mahnung, Anfrage, Bestellung, Eingangsbeleg, Aufmaßberechnungen. Alle Arten von Listen und Formularen. Anbindung an 1st Word Plus. Besteuerung nach Artikel oder Kunde, somit auch für Steuersystem in der Schweiz geeignet! Universeller Druckertreiber, an alle Drucker anzupassen. Incl. ausführlichem, über 100 Seiten starken Handbuch mit zahlreichen Illustrationen. DEMO-VERSION DM 20,– HANDBUCH VORAB DM 50,– (wird angerechnet !)

ST-MATHETRAINER II (Lernprogramm) DM 59.00
Für 1.–6. Schuljahr. 1x1 und Mischrechnen mit wählbaren Höchstzahlenwerten. Umrechnung von Gewichten und Längenmaßen, im Schwierigkeitsgrad durch eingebaute Editierfunktion frei an den Lernbedarf bzw. den Schulbuchstoff anzupassen. Mit Benotung und Protokoll. Voll in GEM eingebunden! Incl. deutscher Bedienungsanleitung.

ST-RECHTSCHREIBEN II (Lernprogramm) DM 59.00
Für 1.–6. Schuljahr. Wörter in Sätze einfügen, Singular und Plural, Kommata setzen. Im Schwierigkeitsgrad durch eingebaute Editierfunktion frei an den Lernbedarf bzw. Schulbuchstoff anzupassen. Mit Benotung und Protokoll. Voll in GEM eingebunden! Incl. deutscher Bedienungsanleitung.

TKC-VIDEO (Verwaltung von Videocassetten) DM 79.00
Verwaltet bis zu 5000 Videofilme pro Datei. 10 Felder für Titel, Spielzeit, Zählwerk, Darsteller, Verleihen an, Bemerkungen, usw. Umfangreiche Such- u. Selektierfunktionen. Ausdruck von Listen + Etiketten (frei über ASCII-Maske zu gestalten). Voll unter GEM, deutsche Bedienungsanleitung.

TKC-MUSICBOX (Musiktitelverwaltung) DM 79.00
Verwaltet bis zu 5000 CDs, LPs oder MCs. Suche nach Einzeltiteln, Ausdruck von Haupttitelübersicht u. Gesamttitelliste mit MC-Nummern und Etiketten. Voll unter GEM, incl. deutscher Bedienungsanleitung.

TKC-TRAINER (Trainingsprogramm) DM 99.00
Sehr hoher Lernerfolg durch Karteikastenprinzip. Für Sprachen, Chemie, Physik, etc. Voll unter GEM, leicht auch für Schüler zu bedienen. Incl. deutschem Handbuch.

TKC-BANKMANAGER (mit Sammler-Ausdruck) DM 99.00

ST-GIRO PLUS DM 49.00
Überweisungsdruckprogramme für den geschäftlichen (häufigen) und privaten Gebrauch. Auch für Schecks + Lastschriften. Ausdruck über ASCII-Masken an alle Formulare anzupassen!

ST-VOKABELTRAINER (Trainingsprogramm) DM 49.00
Vokabeltraining für Englisch, Franzosisch, Italienisch, Spanisch und zeichenkompatible Sprachen. 100 Vokabeln pro Datei, voll unter GEM.

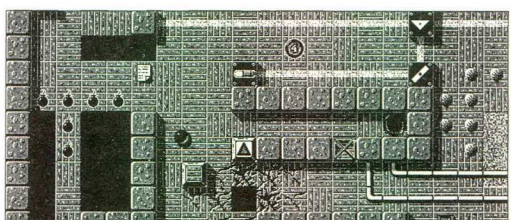
TK COMPUTER-TECHNIK Thomas Kaschadt
Hard- & Software - Entwicklung & Vertrieb

Bischofsheimer Straße 17 * Postfach 60 * 24-Stunden-Auftragsannahme per Anrufbeantworter!
D-6097 Trebur-Astheim * FAX 06147-3555
Fernruf : (06147) 3550 * Btx. 06147-3555

In Ihrem Atari schlummert eine Welt aus Bits und Bytes, von der Sie bisher vielleicht noch nichts gewußt haben. Nun aber ist diese Welt bedroht und benötigt dringend Ihre Hilfe! Finden und öffnen Sie die lebensnotwendigen OXYD-Steine. Für diese schwere Aufgabe brauchen Sie aber nicht nur Mut und Geschick, sondern auch einen ausgeprägten Forschergeist, um die vielen Rätsel dieser geheimnisvollen digitalen Welt zu entschlüsseln.

OXYD © 1990
das Spiel in Ihrem ST*

100 Ein-Spieler-Landschaften !! PLUS !! 100 Zwei-Spieler-Landschaften



Verbinden Sie zwei Rechner über MIDI und spielen Sie mit Ihren Freunden zusammen in den riesigen Labyrinthen.

Außerdem: 20KHz-6-Kanal-Titelsound, einfache Maussteuerung, brilliante Digitalsound-Effekte, BIC-Taste, MIDI-Modus, 72Hz Animation, massig Spezialeffekte, über 150 verschiedene Steinarten, über 100 verschiedene Objektarten, über 90 verschiedene Bodenplatten, ca. 150 Millionen Quadratpixel Spielfläche, deutsche Texte

OXYD-Dongleware-Diskette nur **5 DM****
OXYD-Buch (zum Spielen aller Levels notwendig) nur **50 DM*****

DIGITAL ART Meinolf Schneider * Im Spitzerfeld 30 * 6903 Neckargemünd
Telefonische Bestellung: 06223/8740

*nur mit monochromen Monitor und mind. 1 MegaByte Speicher, ST, STE, Mega & TT
**zzgl. Versandkosten, incl. 14% Mwst. ***zzgl. Versandkosten, incl. 7% Mwst.

SciGraph 2.0

BILD DER ZAHL

SciGraph 2.0 (AtariST/TT) kostet: 599 DM



Hendrik Haase Computersysteme
präsentiert:

Atari-Computer

Atari 1040 STE	Preissenkung
Atari Mega ST 1	Preissenkung
Atari Mega ST 2	Preissenkung
Atari Mega ST 4	Preissenkung
Atari Megafile 30 Festplatte	
Atari Megafile 60 Festplatte	1240,- DM
Wechselplatte 44	Preissenkung
Vortex Datajet 30 Festplatte	1100,- DM
Vortex Datajet 60 Festplatte	1690,- DM
Star Drucker LC 24/10	668,- DM
Epson LQ 550	698,- DM
Fujitsu DL 1100	898,- DM
HP Deskjet 500	1550,- DM
Atari Laserdrucker	Preissenkung
Graustufen Multisync	550,- DM
Mitsubishi Multisync	1298,- DM

Stacy-Computer und TT-Computer
in unterschiedlichen Versionen
TT-Vorführgeräte auf Anfrage

Bestellungen und Informationen bei:

Hendrik Haase Computersysteme

Wiedfeldtstraße 77 • D-4300 Essen 1

Telefon 0201 - 422575 • Fax 0201 - 410421

C S H Ingenieurbüro
für angewandte Computertechnik
Schillerring 19
D-8751 Großwallstadt/Main

Hardwarebeschleuniger:

TURBO 16V2.0
16MHz; 32 KB Cache
CMOS SMD Technik
incl. TURBO ST-Programm **DM 598,00**

TURBO 030
24...32...50MHz !!!
32 KB Cache
eigenes TOS, Fast RAM optional
CMOS SMD Technik
incl. MC68882 Coprozessor **ab DM 2498,00**

Computer (16MHz-Version)
z.B. **ATARI MEGA 2-16** **DM 2498,00**

Fest- u. Wechselplatten:
SCSI-Laufwerke in MEGA Gehäuse mit **SUPRA**
ProHos Adapter incl. Echtzeituhr, SCSI-Bus u.a.

Festplatten
30 ... 380 MB **ab DM 798,00**

Wechselplatte
SQ555 **44MB** incl. 1 Medium **DM 1698,00**
anschlußfertig für ATARI ST Computer u.a.

Modem
Das Modem verfügt z.Zt. über keine ZZF-Zulassung. Der Betrieb am öffentlichen Netz der DBP Telekom ist daher nach §15 FernmGes. unter Strafe gestellt.
SUPRA 2400
300...2400Baud,Hayeskomp. **ab DM 228,00**

INFO anfordern oder ☎ 06022 - 24405

MODULA MARZIPAN

Sweet Secrets №4

Felder fast grenzenlos

In dieser Folge beschäftigen wir uns wieder mit einer Datenstruktur, verwenden dabei opaque Typen und rechnen maschinennah mit Speicheradressen.

Hat man Daten vorliegen, die in sehr regelmäßiger Form dargestellt werden sollen, greift man zu einem ARRAY. Dabei läßt sich durch geschachtelte ARRAYs oder durch Felder, die aus RECORDs bestehen, eine beachtliche Komplexität der Darstellung erreichen.

Allerdings kann man bei der Verwendung von Feldern an Grenzen stoßen, die einerseits im Compiler begründet sind, andererseits ihre prinzipiell statische Eigenschaft betreffen.

Die meisten Modula-Compiler haben bestimmte Limitierungen, insbesondere in der Größe des für ein Modul verfügbaren Speicherplatzes für statische Variablen. Auf dem Atari ST legt der Prozessor MC 68000 eine solche Grenze bei 32 kBytes nahe.

ARRAYs werden intern angesprochen, indem man zur Startadresse einen Offset addiert, der sich aus Index*Elementgröße ergibt. Als Beispiel nehmen wir ein Feld *a*, das aus 1000 INTEGER-Elementen besteht und bei (der fiktiven) Adresse 2000 beginnt. INTEGER belege dabei genau 2 Bytes.

Der Code für einen Zugriff auf das Element *a[10]* muß also den Index nehmen und ihn mit der Elementgröße Zwei multiplizieren. Es ergibt sich der Offset 20, der für einen Zugriff zur Startadresse 2000 addiert werden muß. Der MC 68000 bietet nun einen speziellen Modus für einen solchen Zugriff. Dabei gilt allerdings die Einschränkung, daß der Offset nur von -32768 bis +32767 laufen darf, da für den Offset-Wert nur 16 Bits erlaubt sind.

Damit könnte ein solches Feld maximal 64 kBytes belegen. Tatsächlich aber benutzen die meisten Compiler nur positive Offsets und beschränken sich so auf eine maximale Feldgröße von 32 kBytes. Eine Ausnahme war übrigens die letzte Version des TDI-Compilers, der eine 32-Bit-Arithmetik für Feld-Indizes verwendete.

Aus diesen Eigenschaften der Code-Erzeugung ergeben sich also Einschränkungen, die in der Sprache selber nicht gegeben sind und auf anderen Rechnern vielleicht auch gar nicht auftauchen. Existiert eine solche Schranke aber, gibt es leider keinen Weg, sie zu umgehen.

Dynamische Größe

Da der Sprachkern von Modula eine solche Umgehung nicht ermöglicht, muß ein kleines Modul geschrieben werden, mit dem auch sehr große ARRAYs möglich sind. Listings 2 und 3 zeigen die Definitions- und Implementationsmodule *Large Arrays*, die für die Arbeit mit LPR entworfen wurden.

Die Idee für große Felder ist einfach: Man besorgt sich zur Laufzeit einen genügend großen Speicherbereich und stellt Prozeduren zum Lesen und Schreiben eines Feldelements bereit. Letztere vollziehen die oben beschriebene Berechnung der Speicheradresse eines Feldelements und können so zwischen einem Argument und dem Feldspeicher kopieren.

Im Definitionsmodul wird *Array* als opaquer Typ exportiert. Streng nach der Sprachdefinition sind solche Typen auf POINTER beschränkt, deren Inhalt daher initial dynamisch eingerichtet werden muß. Zur Deklaration einer Variablen gehört also ein Aufruf einer Initialisierungsprozedur.

In *LargeArrays* ist dies *AllocateArray*. Dessen einfache Funktion zeigt ein Blick in das Implementationsmodul. *Array* wird definiert als ein Zeiger auf *ArrayHeader*. In diesem Record findet sich ein Feld für die Größe der Feldelemente und die Adresse des für das Feld benutzten Speicherbereichs.

AllocateArray richtet zunächst dynamisch den Feldkopf ein und berechnet dann aus den übergebenen Parametern den Platzbedarf für das gewünschte Feld. Mit einem weiteren *Storage.ALLOCATE* wird dieser vom Betriebssystem angefordert und im Feldkopf vermerkt. Nun sind die *Array*-Variable initialisiert und der Feldspeicher eingerichtet.

Die Anzahl der Elemente entspricht übrigens nicht dem höchsten Index, da deren untere Grenze immer bei Null liegt. Bei *LargeArrays* - und auch bei dem noch folgenden *DynamicArrays* - laufen die Indizes von 0 bis Elementanzahl-1.

Das Lesen eines Feldelements ist einfach. Als Parameter erhält Get die Array-Variable, also einen Zeiger, einen Index und einen VAR-Parameter beliebiger Größe. Durch die Verwendung von AR-

RAY OF BYTE zusammen mit der Angabe über die Größe der Feldelemente bei der Initialisierung kann unser *Array*-Typ Elemente beliebiger Größe speichern, also vom BOOLEAN bis zum LONGREAL und darüber hinaus zu Verbundtypen.

Aus dem Index und der Elementgröße sowie der Anfangsadresse des Feldspeichers läßt sich einfach die Speicheradresse des ersten Bytes eines Elements errechnen. In dem Modul findet sie sich in der Zeiger-Variablen *Byte* wieder, wobei diverse VAL-Benutzungen nötig sind, um den Compiler zu überlisten. Immerhin rechnen wir hier mit Speicheradressen - ein überaus systemnahes Verfahren, das in dieser Form nicht portabel sein muß.

In einer kleinen Schleife kann *Get* nun die Bytes des Elements aus dem Argument kopieren und somit auslesen. Das Schreiben eines Feldelements mit *Put* geschieht identisch, nur daß die Kopierrichtung umgekehrt ist.

Da dynamisch Speicher angefordert wurde, muß er spätestens vor Verlassen eines Programms auch wieder freigegeben werden. In *LargeArrays* leistet dies die Prozedur *DeleteArray*, die mit zwei *Storage.DEALLOCATE*-Aufrufen den Platz für den Feldspeicher und den Feldkopf an das Betriebssystem zurückgibt. Übrigens sollten die Felder nicht erst beim Verlassen des Programms freigegeben werden, sondern so früh wie möglich, um den Speicher nicht allzusehr zu belasten.

In einigen Modula-Systemen muß man bei *Storage.DEALLOCATE* die Größe des freizugebenden Speichers angeben. In diesem Fall sollte man in einem zusätzlichen Eintrag des Feldkopfes die Anzahl der Elemente vermerken, um die Speicherbelegung zu errechnen.

Insgesamt also kein größeres Modul; für das Thema unserer Serie, die kleinen Tricksereien, ist aber die Berechnung von Speicheradressen und deren Modula-konforme Formulierung mit VAL interessant. Bild 1 zeigt nochmals die Speichersituation bei einem eingerichteten Feld.

Sparsame Größe

Die schiere Größe der nun möglichen Felder kann aber nicht unbedingt der Weisheit letzter Schluß sein. Nehmen wir an, daß ein Feld mit 131072 Elementen eingerichtet wurde. Wer etwas Umgang mit Zweierpotenzen hat, erkennt vielleicht, daß bei zwei Bytes pro Element hier 256 kBytes Speicher belegt wird. Soll das Feld aus LONGREALs mit acht Bytes bestehen, ist auf einen Schlag ein komplettes Megabyte belegt - für einige STs zuviel.

Es kann auch Anwendungen geben, bei denen nur ein Bruchteil des Feldes mit Werten beschrieben ist, wo aber aus irgendwelchen Umständen die Verwendung einer Feldstruktur mit Indizes Vorteile bringt. Es wäre dann völlig ineffizient, den Speicher so zu blockieren.

Daher bietet das Modul *DynamicArrays* in den Listings 3 und 4 Abhilfe, indem nur die Elemente Speicherplatz belegen, in denen tatsächlich Werte stehen.

Auf den ersten Blick sieht die Definition des Feldkopfes im Implementierungsmodul der in *LargeArrays* sehr ähnlich. Aber: *ArrayMemory* ist hier nicht als Zeiger auf die eigentlichen Daten gedacht, sondern auf ein Feld von Zeigern, worin für jedes Element des eigentlichen Datenfeldes ein POINTER vorgesehen ist. Ist dieser gleich NIL, ist auch das Feld leer, ansonsten zeigt er auf ihn. Die Initialisierungs-, Lese- und Schreibroutinen sorgen dafür, daß Speicherplatz nur dann allokiert wird, wenn ein Wert in ein Feldelement geschrieben wird.

Mit *AllocateArray* wird ein dynamisches Feld eingerichtet. Zunächst wird - weil der Typ *Array* opaque ist - der Feldkopf mit *Storage.ALLOCATE* eingerichtet und die Anzahl der gewünschten Feldelemente vermerkt. Es folgt nun die Allokation des Zeigerfelds und die Initialisierung aller Zeiger auf NIL. Die dabei verwendeten Tricks zur Adreßrechnung kennen Sie schon aus *LargeArrays*.

Schauen wir uns nun Get an. Zunächst errechnet es wiederum die Adresse des gesuchten Zeigers im Zeigerfeld. An dieser Stelle folgt nun der entscheidende Unterschied zu *LargeArrays*. Nur wenn der Zeiger (in *p^*) ungleich NIL ist, wurde vorher ein Wert abgelegt und kann kopiert werden.

In unserer Implementierung werden die Werte aber nicht einfach abgelegt, dann müßte man in der Initialisierung auch einen Parameter für die Größe der Elemente übergeben. Die Feldelemente bei *DynamicArrays* können unterschiedlichen Platz belegen! Damit dieses Feature funktioniert, muß zusätzlich zu den Daten deren Größe gespeichert werden. Wir legen fest,

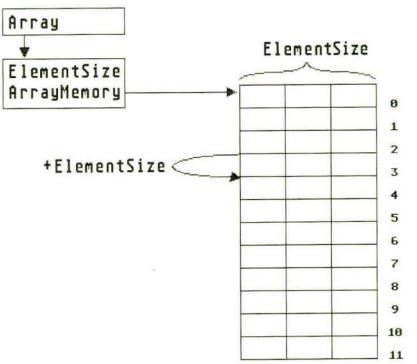


Bild 1: Die Speicherorganisation eines Felds mit LargeArrays

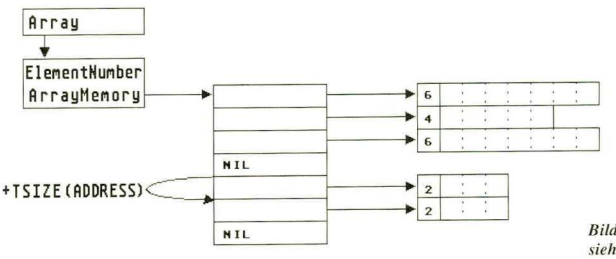


Bild 2: Und so sieht's im Speicher bei DynamicArrays aus

daß bei einem Feldelement, auf das ein Feldzeiger ungleich NIL zeigt, zunächst ein CARDINAL-Wert die Anzahl der Daten-Bytes minus 1 enthält. Danach erst folgen die Daten.

Get berücksichtigt dies so, daß der Feldzeiger zunächst nach *c* kopiert wird, das als Zeiger auf ein CARDINAL deklariert ist. Die tatsächliche Speicheradresse der Daten läßt sich einfach durch Addieren der Größe eines CARDINALs ermitteln und wird in *b* abgelegt. Der Rest ist einfach: Wie bei *LargeArrays* kopiert eine kleine Schleife die Daten-Bytes. Dabei läuft die Schleife aber nicht bis zu einem für alle Feldelemente geltenden *Element*

GESTALTEN SIE MIT CALAMUS® ?

BUSINESS !

Briefbögen
Visitenkarten
Aufkleber
Nutzeneinbindungen
jede Menge Formulare
mit Passermarken
Falz– u. Positionsmarken
und vieles mehr

ARTWORKS business

Mit dem neuen Business–Gestaltungspaket von ARTWORKS COLLECTION steht Ihnen jetzt dieses und noch vieles mehr copyright-frei und fertig gestaltet zur freien Verfügung!

Gebrauchsfertig angelegt in über 80 CDK–Dokumenten. Dazu im umfangreichen Handbuch mit vielen zusätzlichen Informationen zu Gestaltung und Typografie, Layout und Druckvorlagenerstellung mehrfarbig abgebildet!
Geliefert wird im stabilen Schuber. Für 398.– DM.

* Calamus ist eingetragenes Warenzeichen der Firma DMC

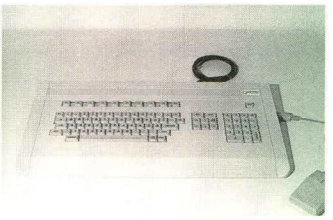
INFO...

ARTLINE
Pipinstraße 4
479 Paderborn
Tel.: 05251/ 28 23 92
Fax: 28 23 91

ARTWORKS business

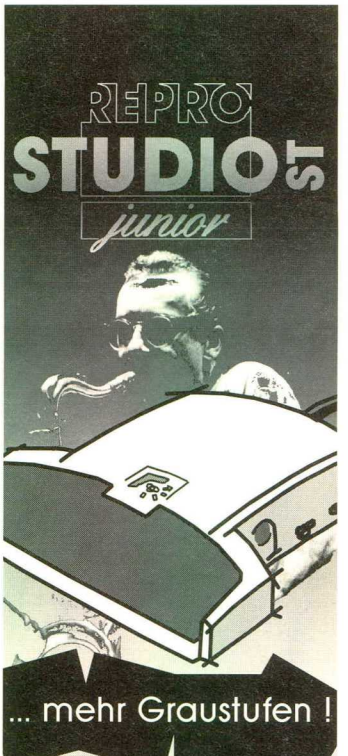
Professionelle Tastatur für ST

- mit einem Mouse-Anschluß/Joystick (optional 2. Joystickanschluß)



- Tastatur-Prozessor läßt sich zurücksetzen (Tastenkombination)
- praktisches sowie formschönes Design in ergonomisch günstiger Form.

ASTON GmbH
Ruhrorter Straße 9
4200 Oberhausen 1
Tel.: (0208) 80 18 32 o. 80 72 19
Fax: (0208) 80 87 03



... mehr Graustufen !

Logitech SCANMAN™ und REPRO STUDIO ST junior

... mehr als nur ein Scanner mit Scansoftware! Der Einstieg in die digitale Reprographie

Upgrade auf Normal-Version oder Pro-Version jederzeit möglich

Software: Scannen direkt 4 Fenster 4 Raster Wandeln S/W in Halbton

Bildgröße nur durch Speicher begrenzt

256 Graustufen Blockfunktionen Grauverfahren Freischneiden Retuschieren Gestalten

Masken Bildverknüpfung Airbrush Rastern Helligkeit Kontrast Filter

Scanner: 100 bis 400 DPI 36 Graustufen oder neu: bis 256 ECHTE Graustufen

Treiber für andere Handscanner lieferbar!

105 mm breit - durch "two-pass" Scanverfahren auch breitere Scans möglich

Druckertreiber für 9- und 24-Nadler sowie Laserdrucker (HPLjet, ATARI) inclusive

Scanman+ Software 39K Grauscanner auf Anfrage! Software allein 248.-

Genau das habe ich gesucht!
Bitte schicken Sie mir umgehend:

___ REPRO STUDIO ST *junior* mit☐ ohne☐ Scanman+ zum Einzelpreis von _____ DM
___ Infos 256 Grauscanner PC/ST

Mein Scanner: _____
(bei Best. ohne Scanner)

Name: _____

Straße: _____

Ort: _____

Unterschrift: _____

Ich bezahle per:
Nachnahme ☐ (zzgl. Porto+Verp.)
Vorkasse ☐

Ausschneiden und senden an:
Trade iT, Richard Römann
6112 Groß–Zimmern, Jahnstraße 18
Tel. 06071–41089

©9/90 Trade iT, Erstellt mit REPRO STUDIO ST und CALAMUS®

Signum! Script Tools / Fonts

PUBLIC DOMAIN

5 DS–DISK PAKET VÖLLIG ÜBERARBEITET !!!
Der ATARI-MESSE-Hit!

Das neue PD-Paket! Mit neuer Software und Fonts! Für tolle Einladungen, Gruß/Glückwunschkarten, Werbung...!

SNAPFONT Mischung aus "Scrabus" und "Headline". Aus Grafiken Signum-Fonts für 9/24-Nadeldrucker u. Laser PI gestalten u. große Überschriften in Signum/Script knappen.

HOUDINI *Update!* Das ultimative PD-Signum-Tool! Signum-Fonts sind jetzt um 90°/180°/270° Grad drehbar u. als neue Fonts abspeicherbar! Im Update zu 1.10 neu überarbeitet u. verbessert! (Doc.)

BIG FONT & TURNFONT Große Überschriften aus Signum-Fonts als Grafik in Signum, 45 u. 90 Grad drehbar! Spitze auch für Grafik- und DTP.

FUNKTIONSTASTEN Papierschablone für F-Tasten als SDO zum Ausschneiden. Mit Blanko-SDO und fertiges Signum-SDO!

SIG-TO-GEM Signumgrafik in GIS/GEM umwandeln! Schnell u. Wirklich!

CHSETSFI/SUCHSETS Font Such-PRG's.

SIG SHELL Verbindung wichtiger PRG's, incl. SIG-Grafik-Konvert.

SIGHELP Menütexte für SCRCOP.ACC

URKUNDE SDO/Bilder einer Urkunde.

MASSTAB Zeilenlineal in cm-Einteilung als Signum-Bild. Direktes Positionieren ohne schwierige Zeil-Einteilung.

VORLAGE SDO für Sig.-Standard-Parameter

LINEAL–24 Eine kpl. neue Version eines cm-Rasters! Sehr umfangreich und noch genauer.

SIGNUMI-CLIPART Ca. 100 neue kleine Grafiken für viele Festtiechen. Ob für Einladungen, Briefkopfe, ...Gruß-/Glückwunschkarten-, neu und einmalig bearbeitet!

>> PD-Signum-Fonts direkt vom Autor freigegeben, << nur wegen Platzmangel extrem verkleinert

24-NADELDRUCKER-FONTS (extrem verkleinert)

9-NADELDRUCKER-FONTS (9 auf Disk!)

Das Exklusiv-Paket mit mit 5 vollen DS
• 1 PD-Katalog
gibt's cash&carry für
DM 39,- (-,5.- Porto u. Verpack.) •

PUBLIC DOMAIN EDITION
PD-EXPRESS

JÖRG RANGNOW SOFTWARE
ITTLINGER STR 45 · 7570 EPPINGEN 3 · TEL 07262/5131

• (Vorauskasse als V.-Scheck; nur Inland • DM 2.- bei NN !)

Public-Domain Software

Hier stellen wir Ihnen unsere neuesten ACTION - Pakete vor. Die Pakete werden alle drei Monate erneuert und wieder mit aktuellster PD-Soft bespielt!

Jedes ACTION - Paket besteht aus 3 doppelseitigen PD-Disketten.

Jedes Paket kostet nur DM 10,--

Ausgabe: 12/90 bis (einschl.) 2/91

GAS.-01/ ***Monochromspiele - Action*** (3 Disks, DM 10,--)
Darkness is for ever: ein Adventure, daß Sie viele Stunden in seinen Bann ziehen wir. **Rennhund:** sportliches Managerspiel für 1 bis 8 Teilnehmer. **Quizmaster 0.5:** nettes Ratespiel für 1 bis 2 Spieler. **Knack den Tresor:** sehr aufwendiges Spiel, bei dem Sie 40 alte Bilder erstehen müssen. **Chicago:** Glücksspiel. **Play Me:** sehr schöne Tetris-Variante. **Hinem:** Memoryspiel mit 198 Karten. **Cave Miner II:** super Boulder-Dash-Variante. **Fly A Jet:** Flugsimualtor einer Boing 727; mit vielen Funktionen. **Segelflug:** Segelflug-Simulator. **Memory:** "normale" Memory-Variante.☐

GAF.-02/ ***Farbspiele - Action*** (3 Disks, DM 10,--)
Persifal: spannendes Abenteuerspiel. **Megamix 1.2:** Memoryspiel. **TriDo 1.5:** super Geschicklichkeitsspiel. **Xenotron 2.3:** schnelles Weltraum-Ballerspiel. **Crisslefridge:** geniales Boulder-Dash Spiel mit super Grafik. **Two for Two:** Panzerspiel, das eine schnelle Reaktion und gute Nerven voraussetzt. **Double Bounce:** Tennisspiel für 2 Sportler. **Gilbert:** Memo-Variante. **Deathstar:** Kampf gegen Monster und vieles mehr. **Damonold:** abwechslungsreiches Ballerspiel mit digit. Stimmen. **Chymer:** Adventure.☐

ANW.-01/ ***Anwender - Action*** (3 Disks, DM 10,--)
Beropress 1.0: Druckt Poster, Grußkarten etc. Verarbeitet sogar Signum-Zeichensätze. **Formular 1.1:** Schreibprog. zum Ausfüllen von Formularen. **Stundenplan. Bildklau:** lädt Bilder ein und verarbeitet diese in Data-Zeilen für GfA-Basic. **Disk Manager:** archiviert Diskettenінhalte. **HoudiNI :** Tool für Signum. **Getfish - Tab:** verwaltet bel. Sportligen. **Adress 5.0:** Adreßverwaltung.☐

MUG.-01/ ***Musik und Grafik - Action*** (3 Disks, DM 10,--)
Van Gogh III: eines der besten Mal/Zeichenprogramme. **Click & Beep:** belegt die Tastatur mit digit. Sounds. **Butterfly Artist:** ausgefallenes Malprogramm. **Freedrum 2.5:** Sound/Trommelmaschine mit digit. Effekten. **ST-Design:** leistungsfähiges Zeichenprog.☐

CLP.-01/ ***Clip Art - Action*** (3 Disks, DM 10,--)
die besten Clips im "PAC"-Format (Stad).☐

SIG.-01/ ***Signum Zeichensätze - Action*** (3 Disks, DM 10,--)
ca. 30 verschiedene Signum (Script) Zeichensätze (für 9 und 24 Nadeldrucker), sowie verschiedene Tools.☐

Fordern Sie auch bitte unseren Gesamtkatalog (gedruckt) für DM 3,-- an.
Ach: Bei uns kostet jede PD-Disk nur DM 4,--

COMPUTER-SERVICE KOHLER
Don-Carlos-Str. 33B, D-7000 Stuttgart 80
Telefon: 07 11 / 6 78 73 92 (24h-Bestellservice)

Size, sondern bis zur jeweiligen Größe in *c^*. Zum Ablegen eines Feldelements mit *Put* muß das Schema berücksichtigt werden. Nach Ermittlung des Feldzeigers kann zunächst getestet werden, ob schon ein Element abgelegt ist. Da die Feldelemente keine feste Größe haben, muß dann überprüft werden, ob das neue Element vielleicht größer oder kleiner ist. Trifft dies zu, wird der vorher belegte Speicherbereich freigegeben und der Feldzeiger auf NIL zurückgesetzt.

War kein Element vorhanden oder wurde es gelöscht, muß Speicherplatz für das neue Element eingerichtet werden. Ein *Storage.ALLOCATE*-Aufruf fordert entsprechend viele Bytes vom Betriebssystem an. Deren Anzahl ergibt sich aus dem Platzbedarf des Längeneintrags plus der Größe des übergebenen Datums. Ersten Wert erhalten wir durch TSIZE(CARDINAL), letzteren durch HIGH(Data)+1. Zur Erinnerung: Bei einem offenen Feld liefert HIGH die Anzahl der Elemente von Null an gezählt.

Ist der Speicher eingerichtet, muß nur noch die Größe des Feldelements vermerkt werden, was wieder durch einen Zugriff über *c^* geschieht. Der Rest von *Put* beschränkt sich auf das Kopieren der Bytes des neuen Feldelements.

Wir gingen anfangs davon aus, daß Speicher knapp ist. Daher ist es sinnvoll, auch eine Routine zum Löschen eines Feldelements anzubieten. *Delete* leistet dies, indem einfach der Feldzeiger ermittelt und dieser an *Storage.DEALLOCATE* übergeben wird.

Abschließend muß es natürlich wieder eine Routine zum Löschen des gesamten Feldes geben. Im Gegensatz zu *Large Arrays* ist dies ein zweistufiger Vorgang, da zunächst alle vorhandenen Feldelemente einzeln freizugeben sind. Mit etwas Adressenberechnung geht dies in einer einfachen REPEAT-Schleife. Der Rest ist die Deallokation des Zeigerfelds und des Feldkopfes.

Bild 2 zeigt die Speicherstruktur, die *DynamicArrays* benutzt. Systemnah ist hierbei besonders der Zugriff auf den jeweiligen Längeneintrag der Feldelemente, der durch geschicktes Kopieren von Zeigern völlig Modula-konform wird.

Zwei Anmerkungen lassen sich abschließend anfügen. Da das Zeigerfeld immer eingerichtet wird, benötigt ein Feld mit 4000 Elementen immer 16000 Bytes (bei vier Bytes pro Adresse), auch wenn kein einziges Element gespeichert ist. Zu umgehen wäre dieser Nachteil nur durch die Verwendung einer Liste für die Zeiger. Dann aber könnte man ganz auf die Array-Struktur verzichten und gleich eine Liste verwenden.

Werden sehr viele Speicheroperationen durch wiederholtes Löschen und Neueinrichten von Feldelementen ausgeführt, fraktioniert der Speicher leicht. Das heißt, daß durch die Aktionen viele kleine Speicherbereiche reserviert und dazwischen viele kleine Lücken sind, vielleicht insgesamt genügend Speicher für ein neues Feldelement vorhanden ist, aber nicht zusammenhängend. Es hängt von der Speicherverwaltung Ihres Modula-Systems ab, ob dadurch Probleme entstehen - dennoch sollten Sie diesen Fakt bei der Programmierung im Kopf behalten.

RT

```
 1:  DEFINITION MODULE LargeArrays;
 2:
 3:  (* By Robert Tolksdorf 10/90 *)
 4:
 5:  FROM SYSTEM IMPORT BYTE;
 6:
 7:  TYPE Array;
 8:
 9:  PROCEDURE AllocateArray(VAR a:Array;
10:                          ElementSize : CARDINAL;
11:                          ElementNumber :
                              LONGCARD);
12:
13:  PROCEDURE Get(a:Array; Index:LONGCARD; VAR
      Data:ARRAY OF BYTE);
14:
15:  PROCEDURE Put(a:Array; Index:LONGCARD;
      Data:ARRAY OF BYTE);
16:
17:  PROCEDURE DeleteArray(VAR a:Array);
18:
19:  END LargeArrays.
20:
```

Listing 1: Definitionsmodul LargeArrays

```
 1:  IMPLEMENTATION MODULE LargeArrays;
 2:
 3:  (* By Robert Tolksdorf 10/90 *)
 4:
 5:  FROM SYSTEM IMPORT BYTE, ADDRESS, TSIZE, VAL;
 6:  IMPORT Storage;
 7:
 8:  TYPE ArrayHeader = RECORD
 9:                     ElementSize : CARDINAL;
10:  (* Falls bei Storage.DEALLOCATE eine
      Groessenangabe noetig ist,
11:     dann muesste hier noch ein Eintrag fuer die
         Elementanzahl
12:     hinzu, der in AllocateArray gesetzt werden
         muesste *)
13:                     ArrayMemory : ADDRESS;
14:                   END;
15:       Array       = POINTER TO ArrayHeader;
16:
17:  PROCEDURE AllocateArray(VAR a:Array;
18:                          ElementSize : CARDINAL;
19:                          ElementNumber :
                              LONGCARD);
20:  BEGIN
21:    (* Feldkopf allokieren *)
22:    Storage.ALLOCATE(a,TSIZE(ArrayHeader));
23:    (* ElementgröPe vermerken *)
24:    a^.ElementSize:=ElementSize;
25:    (* Hier eventuelle auch Elementanzahl
        vermerken *)
26:    (* Feldspeicher allokieren *)
27:    Storage.ALLOCATE(a^.ArrayMemory,
28:                     VAL(LONGCARD,ElementSize)*
                          ElementNumber);
29:  END AllocateArray;
30:
31:  PROCEDURE Get(a:Array; Index:LONGCARD; VAR
      Data:ARRAY OF BYTE);
32:  VAR Byte   : POINTER TO BYTE;
33:      i      : CARDINAL;
34:  BEGIN
35:    (* Elementoffset aus Index und Elementgroesse
        errechnen
36:       und zur Basisadresse addieren *)
37:    Byte:=VAL(ADDRESS,Index*VAL(LONGCARD,
        a^.ElementSize)+
38:                     VAL(LONGCARD,
                          a^.ArrayMemory));
39:    (* Byteweise alle Daten kopieren *)
40:    FOR i:=0 TO a^.ElementSize-1 DO
41:      Data[i]:=Byte^;
42:      (* Adresse ein Byte weitersetzen *)
43:      Byte:=VAL(ADDRESS,VAL(LONGCARD,Byte)+1D);
44:    END;
45:  END Get;
46:
47:  PROCEDURE Put(a:Array; Index:LONGCARD;
      Data:ARRAY OF BYTE);
48:  VAR Byte   : POINTER TO BYTE;
```

→

Die MikroDisk Festplatte ist so klein, daß Sie sie glatt übersehen. (Tip: Info anfordern!)

ST oder AT, das ist hier die Frage ...

EXTRAS

NEU: Laser-NR/LCD Kit
Der Atari Laser wird zum ddd-LASER/8+, also geräuschlos in den Druckpausen, sowie LC-Display für die Blattzählung. Steckfertig. Einbau OHNE Löten!
nur DM 99,-

Schluß mit dem Lärm. NR-Kit für alle Megafile Festplatten.
nur DM 49,-

NEU: Leiser Lüfter für alle Mega-Rechner nur DM 39,-

Das ddd HD-Modul. -Original-. Mit Software nur DM 59,-

NEU: Einschaltverzögerung für alle ST's. nur DM 49,-

HD-Diskettenstationen

HD-Diskettenstationen ? Wozu ? - Obwohl die meisten schon wissen warum, möchten wir die Gründe kurz nennen: 1.) doppelt so viel Speicherplatz pro Diskette; 2.) doppelt so schnelle Datenübertragung; 3.) IBM-Diskettenformate können gelesen werden (mit PC- oder AT-SPEED); 4.) sehr günstiges Speichermedium !!! 5.) voll kompatibel zu 720KB Disketten (also normales Arbeiten wie bisher; 6.) sehr hochwertige Qualität zum günstigen Preis. Um die HD-Option zu nutzen, wird das HD-Modul benötigt.

3.5" HD-Station zum Einbau incl. Anleitung	DM 169,-
3.5" wie vor mit ddd HD-Modul	DM 222,-
3.5" externe HD-Station anschlußfertig	DM 244,-
3.5" wie vor mit HD-Modul	DM 294,-
5.25" HD-Station zum "Einbau" incl. Anleitung	DM 177,-
5.25" wie vor mit ddd HD-Modul	DM 229,-
5.25" externe HD-Station anschlußfertig	DM 285,-
5.25" wie vor mit HD-Modul	DM 333,-

COMPUTER:

Wir halten die Preise klein

wie Atari Mega ST mit abgesetzter Tastatur, Echtzeituhr, Blitter, Monitor 70 Hz (SM124), Maus, Basic usw.; zusätzliche schnelle serielle Schnittstelle für Steueraufgaben und Softwarepaket. Selbstverständlich mit voller Garantie.

M1, mit Monitor, komplett:	nur noch 1244,-		M1-PC, mit Monitor, komplett:	nur noch DM 1644,-
M2, mit Monitor, komplett:	nur noch 1694,-	dto. mit PC-SPEED:	M2-PC, mit Monitor, komplett:	nur noch DM 2094,-
M4, mit Monitor, komplett:	nur noch 2111,-		M4-PC, mit Monitor, komplett:	nur noch DM 2511,-

wie vor, jedoch zusätzlich AT-SPEED eingebaut oder mit HyperCache+ 16MHz eingebaut. Bei voller Garantie !

M1-AT, mit Monitor, komplett:	nur noch 1744,-	M1-16MHz, mit Monitor, komplett:	nur noch DM 1844,-
M2-AT, mit Monitor, komplett:	nur noch 2194,-	M2-16MHz, mit Monitor, komplett:	nur noch DM 2294,-
M4-AT, mit Monitor, komplett:	nur noch 2611,-	M4-16MHz, mit Monitor, komplett:	nur noch DM 2711,-

Alle Rechner wahlweise auch gleich mit HD-Laufwerk 1.44MB (incl. HD-Modul) eingebaut: (Aufpreis DM 199,-) oder mit MegaScreen eingebaut (Aufpreis DM 298,-). Alle erweiterten Rechner haben bei uns weiterhin volle Garantie !

Bei über 1000 Produkten allein im ST-Bereich können wir hier nur einige aufführen. Fragen Sie bitte an, wenn Sie etwas suchen, z.B.:

DRUCKER MONITORE LASER FESTPLATTEN ERWEITERUNGEN SOFTWARE u.v.m.

AT-Ecke

PLOTTER

DIN A3 Plotter für AT und ST. 8 Stifte/Farben. 450mm/sec schnell. Anschließbar an parallele oder serielle Schnittstelle. HPGL kompatibel. Kleiner Preis, aber hochwertige Qualität. DM 1.333,-

OVERHEAD-

Display. Ideal für Schulen. So kommt das Computerbild per Overheadprojektor an die Wand. Leichter Anschluß, einfache Bedienung. Kleiner Preis, aber hochwertige Qualität. DM 1.333,-

80486

80286er, 80386SX, 80386er und 80486 Rechner nach Ihren Wünschen zusammengestellt. Wir machen das günstige Angebot. (Bsp.: 80486 mit 4MB RAM, Festplatte, Monitor, DOS usw. kompl. ab DM 5980,-)

Sie wählen das Gehäuse, das Mainboard, die Diskettenstation, den Speicherausbau, die Grafikkarte, die Schnittstellen, die Festplatte, den Controller, den Monitor, das Betriebssystem, die Maus.

IHREN Rechner bekommen Sie bei uns. Wir beraten Sie gerne.

80286

80286-16MHz (LM)
1 MByte RAM, aufrüstbar
1*par, 2* ser. Schnittstellen
3.5" oder 5.25" Laufwerk
DOS-Betriebssystem
102er Tastatur, deutsch
32 MByte Festplatte
S/W Monitor
DM 1777,-

wie vor, jedoch:
80286-20MHz (LM)
2 MByte RAM, aufrüstbar
44 MByte Festplatte
DM 1994,-

VGA-Aufrüstung: + DM 144,-

Komplettsystem zum erstellen einer Anzeige wie dieser (Rechner, Monitor, Laserdrucker, Software) bei uns schon für unter DM 5000,-. Info anfordern!

Versand per NN, europaweit und Direktverkauf in Hannover

Öffnungszeiten: MO. - FR. von 10 - 18 Uhr durchgehend
Samstag und Sonntag geschlossen.

Es gelten unsere Geschäftsbedingungen

Rufen sie doch mal an

```
49:       i       : CARDINAL;
50:   BEGIN
51:     (* Elementoffset aus Index und Elementgroesse
        errechnen
52:        und zur Basisadresse addieren *)
53:     Byte:=VAL(ADDRESS,Index*VAL(LONGCARD,
        a^.ElementSize)+
54:                          VAL(LONGCARD,
                              a^.ArrayMemory));
55:     (* Byteweise alle Daten kopieren *)
56:     FOR i:=0 TO a^.ElementSize-1 DO
57:       Byte^:=Data[i];
58:       (* Adresse ein Byte weitersetzen *)
59:       Byte:=VAL(ADDRESS,VAL(LONGCARD,Byte)+1D);
60:     END;
61:   END Put;
62:
63:   PROCEDURE DeleteArray(VAR a:Array);
64:   BEGIN
65:     (* Wenn Feld ueberhaupt vorhanden ... *)
66:     IF a#NIL THEN
67:       (* ... und Speicher allokiert ... *)
68:       IF a^.ArrayMemory#NIL THEN
69:         (* ... Feldspeicher deallokieren *)
70:         Storage.DEALLOCATE(a^.ArrayMemory)
71:       END;
72:       (* Feldkopf freigeben *)
73:       Storage.DEALLOCATE(a);
74:       (* sicherheitshalber auf NIL *)
75:       a:=NIL;
76:     END;
77:   END DeleteArray;
78:
79:   END LargeArrays.
80:
```

Listing 2: Implementationsmodul LargeArrays

```
 1:  DEFINITION MODULE DynamicArrays;
 2:
 3:  (* By Robert Tolksdorf 10/90 *)
 4:
 5:  FROM SYSTEM IMPORT BYTE;
 6:
 7:  TYPE Array;
 8:
 9:  PROCEDURE AllocateArray(VAR a:Array;
10:                          ElementNumber:LONGCARD);
11:
12:  PROCEDURE Get(a: Array; Index:LONGCARD;
13:              VAR Data : ARRAY OF BYTE);
14:
15:  PROCEDURE Put(a: Array; Index:LONGCARD;
16:              Data : ARRAY OF BYTE);
17:
18:  PROCEDURE Delete(a:Array; Index:LONGCARD);
19:
20:  PROCEDURE DeleteArray(VAR a:Array);
21:
22:  END DynamicArrays.
23:
```

Listing 3: Definitionsmodul DynamicArrays

```
 1:  IMPLEMENTATION MODULE DynamicArrays;
 2:
 3:  (* By Robert Tolksdorf 10/90 *)
 4:
 5:  FROM SYSTEM IMPORT BYTE, ADDRESS, VAL, TSIZE;
 6:  IMPORT Storage;
 7:
 8:  TYPE ArrayHeader = RECORD
 9:                       ElementNumber : LONGCARD;
10:                       ArrayMemory   : ADDRESS;
11:                     END;
12:       Array       = POINTER TO ArrayHeader;
13:
14:  PROCEDURE AllocateArray(VAR a:Array;
```

```
15:                          ElementNumber:LONGCARD);
16:  VAR p:POINTER TO ADDRESS;
17:      i:LONGCARD;
18:  BEGIN
19:    (* Feldkopf allozieren *)
20:    Storage.ALLOCATE(a,TSIZE(ArrayHeader));
21:    (* Elementanzahl vermerken *)
22:    a^.ElementNumber:=ElementNumber;
23:    (* Speicher für die Elementzeiger allozieren *)
24:    Storage.ALLOCATE(a^.ArrayMemory,
25:                     ElementNumber*VAL(LONGCARD,
                          TSIZE(ADDRESS)));
26:    (* Zeiger auf den ersten Elementzeiger setzen *
        )
27:    p:=VAL(ADDRESS,a^.ArrayMemory);
28:    i:=0;
29:    (* Alle Elementzeiger auf NIL setzen *)
30:    REPEAT
31:      p^:=NIL;
32:      (* nächster Elementzeiger ist eine ADDRESS-
          Breite weiter *)
33:      p:=VAL(ADDRESS,VAL(LONGCARD,p)+VAL(LONGCARD,
          TSIZE(ADDRESS)));
34:      i:=i+1D;
35:    UNTIL i=ElementNumber; (* i laeuft von 0 bis
        ElementNumber-1 *)
36:  END AllocateArray;
37:
38:  PROCEDURE Get(a: Array; Index:LONGCARD;
39:               VAR Data : ARRAY OF BYTE);
40:  VAR p:POINTER TO ADDRESS;
41:      c:POINTER TO CARDINAL;
42:      b:POINTER TO BYTE;
43:      i:CARDINAL;
44:  BEGIN
45:    (* Zeiger auf gewünschten Elementzeiger
        einrichten *)
46:    p:=VAL(ADDRESS,VAL(LONGCARD,a^.ArrayMemory)+
47:                   Index*VAL(LONGCARD,
                        TSIZE(ADDRESS)));
48:    (* Wenn es ein Element gibt ... *)
49:    IF p^#NIL THEN
50:      (* Die Elementgroesse steht am Anfang ... *)
51:      c:=p^;
52:      (* und die Daten dahinter *)
53:      b:=VAL(ADDRESS,VAL(LONGCARD,p^)+VAL(LONGCARD,
          TSIZE(CARDINAL)));
54:      (* Element Byte-weise kopieren *)
55:      FOR i:=0 TO c^ DO
56:        Data[i]:=b^;
57:        (* Zeiger ein Byte weiter setzen *)
58:        b:=VAL(ADDRESS,VAL(LONGCARD,b)+1D);
59:      END;
60:    END;
61:  END Get;
62:
63:  PROCEDURE Put(a: Array; Index:LONGCARD;
64:               Data : ARRAY OF BYTE);
65:  VAR p:POINTER TO ADDRESS;
66:      c:POINTER TO CARDINAL;
67:      b:POINTER TO BYTE;
68:      i:CARDINAL;
69:  BEGIN
70:    (* Zeiger auf gewünschten Feldzeiger setzen *)
71:    p:=VAL(ADDRESS,VAL(LONGCARD,a^.ArrayMemory)+
72:                   Index*VAL(LONGCARD,
                        TSIZE(ADDRESS)));
73:    (* Wenn es schon ein Element gibt ... *)
74:    IF p^#NIL THEN
75:      (* ... Groesse testen und ... *)
76:      c:=p^;
77:      (* ... wenn nicht wie gewuenscht, den alten
          Eintrag loeschen *)
78:      IF c^#VAL(CARDINAL,HIGH(Data)) THEN
79:        Storage.DEALLOCATE(p^);
80:        p^:=NIL;
81:      END;
82:    END;
83:    (* Wenn kein Element vorhanden oder vorher
        gelöscht ... *)
84:    IF p^=NIL THEN
85:      (* ... Speicher einrichten fuer
          Elementgroesse und Daten *)
86:      Storage.ALLOCATE(p^,TSIZE(CARDINAL)+
          HIGH(Data)+1);
87:      c:=p^;
88:      (* Groesse vermerken *)                    →
```

```
 89:       c^:=HIGH(Data);
 90:     END;
 91:     (* Zeiger auf erstes Datenbyte *)
 92:     b:=VAL(ADDRESS,VAL(LONGCARD,p^)+VAL(LONGCARD,
          TSIZE(CARDINAL)));
 93:     (* Byte-weise kopieren *)
 94:     FOR i:=0 TO c^ DO
 95:       b^:=Data[i];
 96:       (* Zeiger ein Byte weitersetzen *)
 97:       b:=VAL(ADDRESS,VAL(LONGCARD,b)+1D);
 98:     END;
 99:   END Put;
100:
101:   PROCEDURE Delete(a:Array; Index:LONGCARD);
102:   VAR p:POINTER TO ADDRESS;
103:   BEGIN
104:     (* Zeiger auf gewuenschten Elementzeiger
          setzen *)
105:     p:=VAL(ADDRESS,VAL(LONGCARD,a^.ArrayMemory)+
106:                    Index*VAL(LONGCARD,
                        TSIZE(ADDRESS)));
107:     (* Falls Element vorhanden *)
108:     IF p^#NIL THEN
109:       (* ... loeschen und Elementzeiger auf NIL *)
110:       Storage.DEALLOCATE(p^);
111:       p^:=NIL;
112:     END;
113:   END Delete;
114:
115:   PROCEDURE DeleteArray(VAR a:Array);
116:   VAR p:POINTER TO ADDRESS;
117:       i:LONGCARD;
118:   BEGIN
119:     (* Wenn es das Feld ueberhaupt gibt ... *)
120:     IF p#NIL THEN
121:       (* p auf den ersten Elementzeiger setzen *)
122:       p:=VAL(ADDRESS,a^.ArrayMemory);
123:       i:=0;
124:       (* alle Elemente loeschen *)
125:       REPEAT
126:         IF p^#NIL THEN
127:           Storage.DEALLOCATE(p^);
128:         END;
129:         (* naechster Elementzeiger steht eine
             Adress-Breite dahinter *)
130:         p:=VAL(ADDRESS,VAL(LONGCARD,p)+
              VAL(LONGCARD,TSIZE(ADDRESS)));
131:         i:=i+1D;
132:       UNTIL i=a^.ElementNumber; (* von 0 bis
            ElementNumber-1 *)
133:       (* Speicher fuer Elementzeiger freigeben *)
134:       Storage.DEALLOCATE(a^.ArrayMemory);
135:       (* Feldkopf freigeben *)
136:       Storage.DEALLOCATE(a);
137:       (* sicherheitshalber auf NIL setzen *)
138:       a:=NIL;
139:     END;
140:   END DeleteArray;
141:
142:   END DynamicArrays.
```

Listing 4: Implementationsmodul DynamicArrays

```
 1:  ...
 2:  FROM SYSTEM IMPORT BYTE, WORD, VAL ;
 3:  ...
 4:
 5:  TYPE WordByte = RECORD CASE : BOOLEAN OF
 6:                               FALSE : w: WORD;
 7:                             | TRUE  : b: ARRAY[0..1]
                                OF BYTE;
 8:                    END END;
 9:
10:  ...
11:
12:  PROCEDURE Low(w:WORD):WORD;
13:  VAR t: WordByte;
14:  BEGIN
15:    t.w:=w;
16:    RETURN VAL(CARDINAL,t.b[1])
17:  END Low;
18:
19:  PROCEDURE High(w:WORD):WORD;
20:  VAR t: WordByte;
21:  BEGIN
22:    t.w:=w;
23:    RETURN VAL(CARDINAL,t.b[0])
24:  END High;
25:
26:  ...
```

Dieses Listing wurde in der dritten Folge leider vergessen.

Die Megafile zum Micropreis!

Megafile 30 jetzt nur noch DM 698,–

Atari 520 STE 798,–	**Speichererweiterungen:**
Atari STE 1 MB 888,–	ST/STE auf 1 MB ..99,–
Atari STE 2 MB 1188,–	STE auf 2 MB 298,–
Atari STE 4 MB ...1388,–	ST auf 2,5 MB 448,–
Atari Mega ST 1 998,–	STE auf 4 MB 598,–
Atari Mega ST 2 ...1698,–	ST auf 4 MB 748,–
Atari Mega ST 2	
4 MB 1998,–	Atari Portfolio 398,–
Atari SM 124 298,–	1st Word plus 148,–
Atari SC 1224 548,–	**Word Perfect 148,–**
AT-Speed 444,–	Adimens ST plus....248,–
Supercharger 666,–	Aditalk ST plus 248,–
Atari Megafile 60 ..1198,–	Turbo C Pro 398,–
Star LC 24-10 698,–	Power Pack 99,–
Turbo 16 598,–	Public Domain 5,–



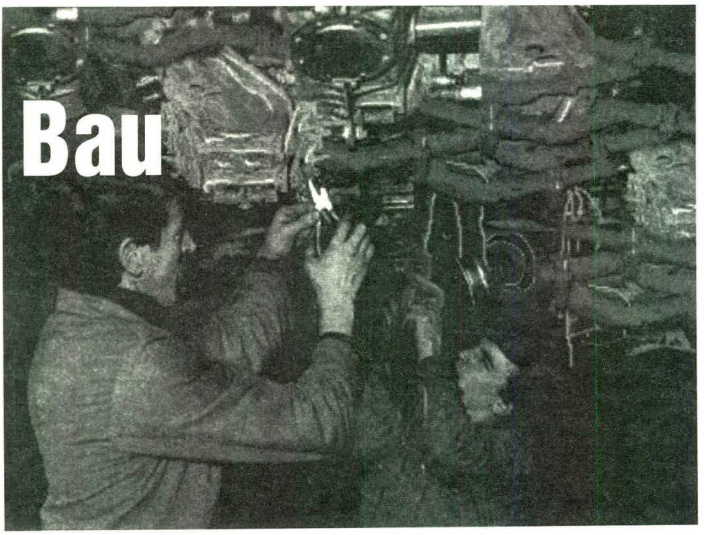
WITTICH COMPUTER GMBH

VERSANDZENTRALE
Tulpenstr. 16 • 8423 Abensberg
Telefon und Fax (09443) 453

LADENVERKAUF
Luitpoldstraße 2 • 8400 Regensburg
Tel. (0941) 562530 Fax (0941) 562510

24 Stunden Bestellannahme • Telefonische Beratung 10.00 bis 20.00 Uhr

Compiler-Bau



Teil 2

Bevor man einen Compiler für eine höhere Programmiersprache schreibt, muß man sich erst einmal mit der Repräsentation der einzelnen Hochsprachenobjekte in der Zielsprache beschäftigen. Für uns heißt das, daß wir uns nach dem Überblick, den wir aus der ersten Folge gewonnen haben, nicht gleich in die Details der Compiler-Programmierung stürzen. Stattdessen werden wir uns zuerst die Laufzeitumgebung des Compilats ansehen. Erst danach beschäfigen wir uns mit der ersten und meist auch einfachsten Phase eines Compilers, der lexikalischen Analyse.

In der ersten Folge haben wir gesehen, wie ein Compiler den Quellcode Stück für Stück zerlegt, analysiert und dann aus der gewonnen Information ein äquivalentes, also funktionsgleiches Programm in der Zielsprache konstruiert. Damit dieses Programm korrekt erzeugt werden kann, muß sich der Compiler-Bauer vorher mit der Zielmaschine auseinandersetzen. Er muß sich dabei sowohl das Betriebssystem als auch die Zielsprache und die von ihr zur Verfügung gestellten Objekte ansehen. Interessante Fragen sind zum Beispiel: Wie kann man Speicherplatz vom Betriebssystem anfordern? Wie kann man eine Zeichenkette in der Zielsprache darstellen? Wie kann man den Mechanismus eines Prozeduraufrufs implementieren?

Zum Beispiel kann eine Zeichenkette durch das ASCII-Zeichen mit der Nummer Null abgeschlossen werden, oder das erste Byte gibt die Länge der Zeichenkette an. Die zweite Möglickeit hat den Vorteil, daß die Berechnung der Länge einer Zeichenkette sehr schnell geht, aber auch den Nachteil, daß keine Zeichenketten mit mehr als 255 Bytes Länge dargestellt werden können. Die Implementierung von Prozeduraufrufen ist sehr viel kritischer, da man hierbei viele verschiedene Probleme berücksichtigen muß. So zum Beispiel den Zugriff auf Variablen in äußeren Sichtbarkeitsbereichen, Parameterübergabe, Anlegen von lokalen Variablen und eventuell auch die Implementierung von Prozedurvariablen. Nachdem wir uns mit solchen und ähnlichen Fragen beschäftigt haben, nehmen wir die lexikalische Analyse etwas genauer unter die Lupe. Das letzte Mal haben wir die Aufgaben dieser ersten Phase eines Compilers schon kennengelernt. Die gesamte überflüssige Information wie Kommentare und Formatierung wird vom Scanner entfernt, und die einzelnen Bausteine einer Programmiersprache wie Bezeichner, Schlüsselwörter und Sonderzeichen werden zu kleinen Päckchen, den Tokens, zusammengefaßt. Wie man einen Scanner entwirft und implementiert, wollen wir uns dieses Mal ansehen. Am Schluß dieser Folge schauen wir uns die Vorgehensweise, die wir beim Lösen des Problems Scanner benutzt haben, mal genauer an.

Repräsentanten

Eine höhere Programmiersprache stellt dem Benutzer in der Regel eine Zahl von vordeklarierten Datentypen (elementare Datentypen) und Datentypkonstruktoren (zusammengesetzte Datentypen) zur Verfügung. Zu den grundlegendsten Datentypen gehören wohl ganze Zahlen (INTEGER), einzelne Zeichen (CHAR) und oft auch ein boolscher Typ (BOOLEAN). Datentypkonstruktoren sind dagegen zum Beispiel Zeiger, Felder (Arrays) und Verbunde (Records). Diese Konstruktoren erlauben es, aus den schon vorhandenen Datentypen neue zu konstruieren, daher auch der Name.

Die Darstellung der vordeklarierten Datentypen in der Zielsprache, also zum Beispiel in 68000-Assembler, ist meist recht einfach. Oft stellt die Zielsprache auch gleichartige Typen zur Verfügung. Zum Beispiel kann der Modula-2-Datentyp *INTEGER* oder *int* von C direkt als 16- oder 32-Bit-binärcodierte Zahl dargestellt werden. Die 68000 stellt bei dieser Repräsentation auch gleich die grundlegendsten Operationen wie Addition, Subtraktion usw. zur Verfügung. Ähnlich einfach fällt die Wahl bei der Repräsentation für den Datentyp *CHAR* bzw. *char*. Er wird einfach als 8-Bit-Zahl im ASCII-Format dargestellt.

Schwieriger wird die Angelegenheit oft bei Typen, die vom Benutzer mit Hilfe der Datentypkonstruktoren deklariert wurden. Die Darstellung von Zeigertypen ist meist durch den Zielcode vorgegeben, aber ein Array kann schon die ersten Probleme aufwerfen. Eindimensionale Arrays werden in der Regel durch eine Aneinanderreihung der Array-Elemente implementiert. Bei einem zweidimensionalen Array

gilt es zuerst zu entscheiden, ob es in row-major- oder column-major-Ordnung abgelegt wird, d.h. ob man es als eine Reihe aus Spalten oder eine Spalte aus Reihen repräsentiert. Diese Entscheidung wird allerdings manchmal von der Sprachdefinition der Quellsprache vorweggenommen. Modula-2 fordert zum Beispiel row-major-Arrays, da die Schreibweise *ARRAY Index1, Index2 OF ElemType* äquivalent zu *ARRAY Index1 OF ARRAY Index2 OF ElemType* ist. Eine weitere zusammengesetzte Datenstruktur ist das Record. Es besteht in der Regel aus mehreren Feldern, die alle einen anderen Typ besitzen können und normalerweise direkt hintereinander im Speicher abgelegt werden. Der vom Compiler erzeugte Code adressiert die einzelnen Komponenten per indirekter Adressierung mit Offset zur Startadresse des Records. Zur Darstellung der einzelnen Elemente eines Modula-2-Mengentyps (Set) werden in der Regel sogar die einzelnen Bits der Maschine herangezogen. In einer Menge von 0 bis 31 (*SET OF [0..31]*) wird das Element Nummer x zum Beispiel durch das x-te Bit eines Maschinenwortes dargestellt. Eini-

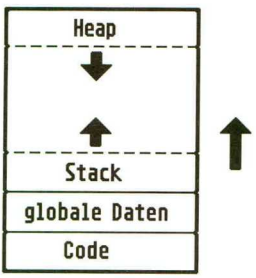


Abb. 1: Mögliche Speicherbelegung eines kompletten Programms

ge Modula-2-Compiler begrenzen die von ihnen übersetzte Sprache derart, daß sie keine Mengen von Aufzählungstypen mit mehr als 32 Elementen zulassen, da die 68000 keine Befehle zur Manipulierung von mehr als 32 Bit großen Bit-Feldern zur Verfügung stellt. Dies ermöglicht erst die 68020. Trotzdem sollte ein Compiler solche Operationen durch Routinen in der Laufzeitbibliothek zur Verfügung stellen.

Sobald ein Programm gestartet wird, bekommt es vom Betriebssystem eine gewissen Menge Speicher zur Verfügung gestellt. In diesem Speicherbereich müssen in der Regel der Code, die globalen (statischen) Variablen und der Laufzeit-Stack des Programmes untergebracht werden. Normalerweise besteht noch die Möglichkeit, während des Ablaufs des Programmes Speicher dynamisch vom Betriebssystem anzufordern. All dieser



Speicherplatz muß sinnvoll verwaltet und nach Gebrauch wieder freigegeben werden. Zu der Verwaltung gehört auch, daß bei einem Aufruf einer Prozedur oder Funktion deren lokale Variablen angelegt und nach deren Beendigung wieder freigegeben werden. Außerdem müssen meist noch Parameter vom Aufrufer an die Prozedur bzw. Funktion übergeben werden. Damit dies alles reibungslos und möglichst noch effizient vonstatten geht, ist es wichtig, daß man sich ein genaues Bild über die Speicherbelegung des erzeugten Programmes macht. Dabei ist auch darauf zu achten, wie die einzelnen Phasen eines Prozeduraufrufs oder einer Rückkehr von einer Prozedur die Speicherbelegung verändern.

Abb. 1 zeigt eine typische Speicheraufteilung. Der Pfeil zeigt die Richtung der aufsteigenden Adressen an. Zuunterst steht der Code, also die compilierte Form des Programmes, der direkt von den globalen Daten gefolgt wird. Beide besitzen eine feste Größe. Die Lage der einzelnen Prozeduren im Codebereich und der einzelnen Variablen und evtl. auch Konstanten innerhalb der globalen Daten wird vom Compiler festgelegt und während der Laufzeit nicht mehr verändert. Deshalb bezeichnet man sie auch als statisch. Anders sieht dies auf dem Stack und im Heap aus. Bei jedem Prozeduraufruf und zum kurzzeitigen Zwischenspeichern von Werten werden Daten auf den Stack geschrieben und dieser somit nach oben hin erweitert. Spätestens beim Verlassen einer Prozedur werden die innerhalb der Prozedur auf den Stack gelegten Daten wieder abgeräumt. Da Prozeduren in den meisten Programmiersprachen nur in der umgekehrten Reihenfolge verlassen werden können, in der sie aufgerufen wurden, werden die Daten auch in der umgekehrten Reihenfolge vom Stack genommen. Somit ist die Verwaltung des Stacks recht einfach. Es gibt einen Stackpointer, der auf auf das oberste Stackelement zeigt. Sobald eine Prozedur aufgerufen wird, wird der Stackpointer entsprechend dem Speicherbedarf der Prozedur inkrementiert. Beim Verlassen der Prozedur wird der Stackpointer wieder auf seinen alten Wert zurückgesetzt. Somit ist aller Speicherplatz oberhalb des Stackpointers frei und der unterhalb belegt. Die Verwaltung des Heaps ist dagegen um einiges komplizierter. Hier wird der Speicherplatz meist in einer nicht vorhersehbaren Reihenfolge angefordert und wieder freigegeben. Aus diesem und anderen Gründen liegt die Verwaltung des Heaps meist beim Betriebssystem und nicht bei dem compilierten Programm. Sie ist für uns also zweitrangig. Wichtig ist nur, daß es eine Betriebssystemfunktion zum Anfordern und eine zum Freigeben eines Speicherblocks möglichst beliebiger Größe gibt.

In der Abbildung wachsen Heap und Stack aufeinander zu; das ist recht günstig, da der Speicherbedarf beider Bereiche von Programmlauf zu Programmlauf stark schwanken kann und der Speicher auf diese Weise möglichst optimal ausgenutzt wird. Leider ist es auf manchen Maschinen und unter manchen Betriebssystemen (zum Beispiel auch auf dem ST) nicht möglich, den Speicherplatz in dieser Weise aufzuteilen. Bei Verwendung einer 680x0 muß der Stack zum Beispiel genau in der umgekehrten Richtung, also von oben nach unten, wachsen, da sich der Prozessor beim Aufruf von Unterprogrammen auf diese Konvention verläßt. Da das TOS eine Vertauschung von Stack und Heap nicht ohne Einschränkungen zuläßt, reserviert man für den Stack notgedrungen einen Bereich festgelegter Größe und läßt ihn in diesem Bereich von oben nach unten wachsen. Für Programme, die nicht stark rekursiv sind, bedeutet das meist auch keine wesentliche Einschränkung.

Lokales

Nachdem wir uns die Speicheraufteilung eines kompletten Programmes angesehen haben, wollen wir als nächstes den Bereich betrachten, dessen Verwaltung in der Regel komplett vom Compiler organisiert wird. Auf dem Stack werden, wie wir oben gesehen haben, die lokalen Daten der einzelnen Prozeduraufrufe abgelegt, dazu gehören auch solche Dinge wie die Rücksprungadresse und ein Zeiger auf die lokalen Daten des Aufrufers. Ein solches Paket lokaler Daten für einen einzelnen Prozeduraufruf bezeichnet man als ein Activation-Record. Da die Struktur eines Activation-Records etwas komplizierter als die der gesamten Speicheraufteilung ist, zeigt Abb. 2 nicht nur den prinzipiellen Aufbau eines Activation-Records, sondern einen Schnappschuß des Stacks bei der Abarbeitung des folgenden Programmfragments. Auch hier zeigt der große Pfeil wieder in die Richtung aufsteigender Adressen.

```
PROCEDURE a;
      VAR i: INTEGER;

      PROCEDURE b (x: INTEGER);
            VAR j: INTEGER;
            BEGIN
                  j := 1;
                  i := x;
            END b;

      PROCEDURE c;
            BEGIN
                  b (3);
            END c;

      BEGIN
            i := 10;
            c;
      END a;
```

Der Schnappschuß erfolgte vor dem Abarbeiten der Zuweisung *i := x*. Es sind also alle drei Prozeduren aktiv, und sie wurden in der Reihenfolge *a*, *c*, *b* aufgerufen. Wie man anhand des Activation-Records der Prozedur *c* sieht, besteht ein solches Record mindestens aus drei Einträgen: Der Rückkehradresse, dem Zeiger auf den dynamischen und dem auf den statischen Vorgänger. Den Unterschied zwischen statischem und dynamischem Vorgänger sieht man deutlich an Prozedur *b*. Ihr statischer Vorgänger ist die textuell umgebende Prozedur, also *a*. Der dynamische Vorgänger ist aber *c*, also die Prozedur, von der *b* aufgerufen wurde. Auf den ersten Blick erscheinen diese beiden Zeiger möglicherweise etwas willkürlich gewählt. In einer Programmiersprache, die die Deklaration lokaler Prozeduren zuläßt, sind sie aber dringend notwendig, wie wir im folgenden sehen werden. Sobald die Abarbeitung einer Prozedur beendet ist, muß diese ihr gesamtes Activation-Record erst einmal abräumen, bevor sie zu ihrem Aufrufer zurückkehren kann. Um dies möglichst einfach zu bewerkstelligen, besitzt jede Prozedur einen Link-Pointer. Durch ein einfaches Laden des Stack-Pointers mit dem Link wird das gesamte Activation-Record freigegeben. Bevor eine neu aufgerufene Prozedur ihren eigenen Link-Pointer berechnet, muß sie den des Aufrufers allerdings auf den Stack retten, um ihn beim Rücksprung wiederherstellen zu können. Einen solchen geretteten Link-Pointer nennt man Zeiger auf den dynamischen Vorgänger. Da die Prozedur dieses Retten genau in dem Moment durchführt, in dem der Stack-Pointer den Wert hat, den der Link annehmen soll, ergibt es sich, daß der Link einer Prozedur auf den dynamischen Vorgänger, also den geretteten Link des Aufrufers, zeigt. Diese Verkettung der Zeiger auf den dynamischen Vorgänger wird in Abb. 2 deutlich. Die Verwaltung der Link-Pointer und das Allozieren bzw. Freigeben der Activation-Records läßt sich auf einer 680x0 übrigens sehr einfach durch die beiden Befehle *link* und *unlink* realisieren.

Offen bleibt, wozu der Zeiger auf den statischen Vorgänger nötig ist. Die Prozedur *b* greift in ihrer Zuweisung *i := x* auf die Variable *i* zu. Da diese in der äußeren Prozedur *a* deklariert ist, liegt sie, wie aus der Abbildung sofort ersichtlich ist, auch im Activation-Record von *a*. Bevor wir uns mit diesem Problem beschäftigen, schauen wir uns allerdings erst einmal an, wie *b* auf seine eigenen lokalen Variablen zugreift. Auch dazu findet der Link-Pointer Verwendung. Alle lokalen Variablen liegen oberhalb der Adresse, die durch den Link-Pointer referiert wird. Da die Anzahl und Größe der lokalen Variablen bei der Übersetzung vom Compiler ermittelt werden können, besitzt jede lokale Variable eine vom Compiler berechenbare Distanz zu der vom Link-Pointer referierten Adresse. Die Variable *j* der Prozedur *b* hat zum Beispiel einen Abstand von acht Bytes, da sowohl der Zeiger auf den statischen als auch der auf den dynamischen Vorgänger genau vier Bytes lang sind. Die meisten Mikroprozessoren besitzen Adressierungsarten, die es erlauben, zusätzlich zu einem Register einen Offset anzugeben. Dadurch kann man auf die lokalen Variablen direkt zugreifen, ohne daß die jeweilige Distanz vorher umständlich zum Wert des Link-Pointers addiert werden muß. Der Zeiger auf den statischen Vorgänger enthält, wie aus der Abbildung ersichtlich ist, den Link-Pointer der im Quelltext textuell umgebenden Prozedur. Soll nun also in der Prozedur *b* auf die Variable *i* zugegriffen werden, geschieht der Zugriff nicht direkt mit einem Offset über den Link-Pointer. Stattdessen wird über den Link-Pointer der Zeiger auf den statischen Vorgänger geladen. Da dies der Link-Pointer der textuell umgebenden Prozedur, also *a*, ist, kann man auf *i* jetzt mit diesem geladenen Zeiger und dem normalen, d.h. auch in *a* gültigen, Offset zugreifen.

Soll ein Zugriff auf eine Variable einer textuell noch weiter außenliegenden Prozedur stattfinden, wird nach dem Laden des Zeigers auf den statischen Vorgänger nicht gleich auf die Variable zugegriffen, sondern stattdessen der jetzt neu erreichbare Zeiger auf den statischen Vorgänger der Prozedur *a* geladen. Dies Spiel läßt sich natürlich beliebig fortsetzen. Unklar ist noch, woher eine Prozedur den Zeiger auf ihren statischen Vorgänger bekommt. Glücklicherweise kennt den aber der Aufrufer der Prozedur, so daß er beim Aufruf mit übergeben werden kann. Wer möchte, kann sich als kleine Übung anhand der Sichtbarkeitsregeln von Pascal oder Modula-2 einmal klarmachen, daß der Aufrufer immer in der Lage ist, diesen Zeiger zu ermitteln. Dabei sollte einem dann auch klar werden, daß immer das auf dem Stack oberste Activation-Record einer bestimmten Prozedur als statischer Vorgänger für den Prozeduraufruf einer zu dieser Prozedur lokalen Prozedur herhalten muß. Der einzige noch nicht besprochene Punkt ist die Übergabe von Parametern. Direkt vor dem Aufruf einer anderen Prozedur legt der Aufrufer alle Parameter auf den Stack. Da zwischen dem Link-Pointer einer Prozedur und den an sie übergebenen Parametern somit nur die Rückkehradresse steht, besitzen also auch die Parameter eine feste, wenn auch negative Distanz zum Link-Pointer und sind somit über ihn erreichbar.

Sobald eine Prozedur verlassen wird, gibt sie den größten Teil ihres Activation-Records durch das schon beschriebene Kopieren des Link-Pointers in den Stack-Pointer frei. Anschließend lädt sie den Zeiger auf den dynamischen Vorgänger vom Stack, um den Link-Pointer des Aufrufers wiederherzustellen. Bevor sie endgültig zurückkehren kann, muß sie die Rückkehradresse vom Stack retten und anschließend alle übergebenen Parameter vom Stack entfernen. Letzteres geschieht durch ein simples Dekrementieren des Stack-Pointers und ist kein Problem, da

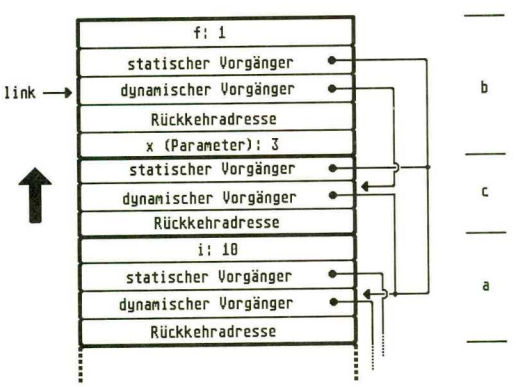
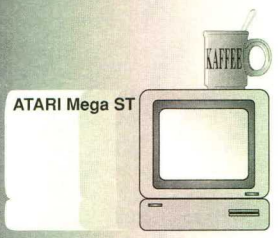


Abb. 2: Schnappschuß des Laufzeit-Stacks (Activation-Records)

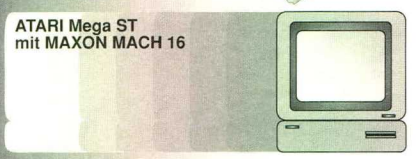
TURBOPOWER

für den ATARI ST



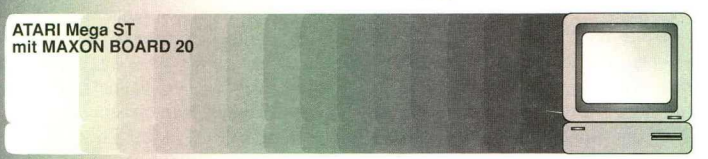
System Performance Index 1,0





System Performance Index 1,85





System Performance Index 3,6

Sorry, aber in Zukunft werden Sie auf die Kaffeepause verzichten müssen !



DER ATARI ST

Wie jeder Computer erreicht auch der ATARI ST mit steigenden Ansprüchen der Anwender und wachsender Komplexität der Software irgendwann die Grenze seiner Leistungsfähigkeit. Spätestens, wenn die Produktivität des Anwenders durch sein Werkzeug gebremst wird, ist es Zeit, aufzurüsten. Für alle ATARI ST-Profis stehen mit den neuen MAXON-Beschleunigerkarten zwei Lösungen zur Verfügung, die ihren Rechner in neue Leistungsdimensionen vorstoßen lassen.

MAXON MACH 16

Mit bestechenden Leistungsmerkmalen wartet diese Beschleunigerkarte für den 260ST, 520ST, 520ST+, 1040ST sowie alle Mega ST-Modelle auf: Ein mit 16 MHz getakteter Prozessor MC 68000 bringt Ihren ATARI ST in Verbindung mit 16 kByte schnellem (0 Waitstates) Cache-Memory auf Trab. Bei höchster Kompatibilität zu bestehenden Anwendungen wird eine durchschnittliche Beschleunigung der gesamten Systemleistung um den Faktor 1,85 erreicht. Zusätzlich bietet die MACH 16 einen Steckplatz für einen optionalen mathematischen Coprozessor 68881, der mit der entsprechenden Software das Rechnen mit Fließkommazahlen um den Faktor 15 beschleunigen kann. Damit bietet die MACH 16 eine optimale und zukunftssichere Möglichkeit, mit dem ATARI ST für wenig Geld in neue Leistungsbereiche vorzustoßen.

Unverbindliche Preisempfehlung DM 695,-
Bestell-Nr. 900820

MAXON BOARD 20

Mit dem MAXON BOARD 20 vollzieht der ATARI ST den Leistungssprung zur echten 32Bit-Workstation. Durch seine überzeugenden technischen Eckdaten - Prozessor MC 68020 mit 16 MHz Taktrate, 32 kByte Cache-Memory mit 32 Bit Busbreite, optimierte Cache-Verwaltung sowie höchste Kompatibilität durch das in zwei ROMs enthaltene TOS 1.6 - markiert es den Schritt zu einer neuen Rechnergeneration. Aufgerüstet mit dem MAXON BOARD 20 wird die Arbeitsgeschwindigkeit des ATARI ST im Praxisbetrieb auf 360% und mehr beschleunigt. Schon heute voll ausgerichtet auf die hohen Anforderungen einer kommenden Software-Generation, stellt das MAXON BOARD 20 damit ein Muß für alle Anwender dar, die ihren Rechner auch in der Zukunft professionell einsetzen wollen.

Unverbindliche Preisempfehlung DM 1895,-
Bestell-Nr. 900830

© M.V. ZIMMERMANN

MAXON Computer • Schwalbacher Str. 52 • 6236 Eschborn • Tel.:06196/481811 • FAX:06196/41885
Mit Einbau: Eickmann Computer • l.d.Römerstadt 249 • 6000 Frankfurt • Tel.:069/763409 • FAX:069/7681971

der Compiler anhand der Prozedurdeklaration die Anzahl der Bytes bestimmen kann, die von den Parametern belegt werden. Nachdem nun, sofern notwendig, noch ein Rückgabewert auf den Stack oder in ein Register geladen wurde, steht dem Rücksprung mit Hilfe der Rückkehradresse nichts mehr im Weg.

In Programmiersprachen wie C, die keine lokalen Prozeduren zulassen, kann der Zeiger auf den statischen Vorgänger natürlich wegfallen. Außerdem ist in C der Aufrufer für das Abräumen der Parameter zuständig, da es hier erlaubt ist, Funktionen zu deklarieren, die mit einer unterschiedlichen Anzahl von Parametern aufgerufen werden können. Ein Beispiel ist die Systemfunktion *printf*.

Zum Abschluß dieses Themas soll noch einmal erwähnt werden, daß der Stack-Aufbau bei Verwendung einer 680x0 als Zielprozessor genau andersherum ist. Dort würde das Activation-Record unserer Beispielprozedur *a* also obenauf liegen, darunter *c* und dann erst *b*. Damit erhalten die Offsets zum Adressieren der lokalen Variablen und der Parameter natürlich ein anderes Vorzeichen.

Päckchen packen

Nach diesem Exkurs in die Tiefen der Zielmaschine gönnen wir uns einen etwas leichter verdaulichen Stoff. Wie wir schon im ersten Teil dieser Serie gesehen haben, ist es die Hauptaufgabe des Scanners, alle für den weiteren Übersetzungsvorgang unwesentliche Information aus dem Quelltext zu entfernen. Dabei kann man grob zwei Arten von überflüssiger Information unterscheiden: Information, die nur dem Menschen zugute kommt und somit vollständig eliminiert werden muß. Dies sind zum Beispiel alle Kommentare und die Formatierung des Programms. Die zweite Art ist aufgeblähte Information wie die Schlüsselwörter und die oft beliebig langen Bezeichnernamen. Im Gegensatz zu den Kommentaren sind die Bezeichnernamen nicht völlig überflüssig, da sie dazu dienen, die verschiedenen Prozeduren, Variablen usw. voneinander zu unterscheiden. Ordnet man jedem in einem Programm vorkommenden Bezeichner eine eindeutige Zahl zu, kann man im weiteren statt des ursprünglichen Namens diese Zahl verwenden und spart dadurch sowohl Speicherplatz als auch Verarbeitungszeit. Erinnern wir uns an das Beispiel aus der letzten Folge. Hier hat der Scanner die Quelltextzeile *tail := tail MOD pipeMax + 1* in die folgende Token-Folge umgewandelt: <id> <becomes> <id> <mod> <id> <plus> <intconst>. Allerdings sind wir jetzt offensichtlich einen



Schritt zu weit gegangen. Sowohl für *tail* als auch für *pipeMax* steht in der Token-Folge das Token <id>, außerdem ist der Wert der Konstanten <intconst> völlig verlorengegangen. Um diese für den weiteren Übersetzungsvorgang notwendigen Informationen zu erhalten, werden alle Tokens, bei denen es nicht nur auf die Tokens ankommt, mit einem Attribut versehen. Unsere Token-Folge sieht dann wie folgt aus: <id>[1] <becomes> <id>[1] <mod> <id>[2] <plus> <intconst>[1]. Sowohl die Bezeichner-Tokens als auch das Token für die Konstante sind nun mit genügend Information versehen, um dieses Codestück zu übersetzen. Die Werte für die Attribute der Bezeichner werden, wie schon in der ersten Folge beschrieben, von der Bezeichnertabelle festgelegt. Für uns ist hier nur wichtig, daß gleich geschriebene Bezeichner auch das gleiche Attribut erhalten.

Nachdem wir uns klargemacht haben, was wir von dem Scanner erwarten, ist es an der Zeit, uns um die Programmierung Gedanken zu machen. Doch vorher wollen wir uns kurz noch einmal den Platz des Scanners in einem Compiler anschauen.

Als erste Phase bekommt der Scanner seine Eingabe direkt von der Quelldatei. Lediglich ein Präprozessor kann ihm noch vorangestellt sein. Seine Ausgabe gibt der Scanner in Form einer Token-Folge direkt an den Parser weiter. Außerdem muß der Scanner noch Zugriff zur Fehlerbehandlung, der Bezeichner- und der Konstantentabelle haben. Da der Scanner die einzige Phase eines Compilers ist, der den Quellcode liest und ihn Zeichen für Zeichen analysieren muß, ist die Geschwindigkeit des Scanners, trotz seiner recht einfachen Aufgabe, sehr wichtig für die Gesamtgeschwindigkeit des Compilers.

Ein wenig Theorie

Die zentrale Prozedur des Scanners wollen wir hier *NextToken* nennen. Wann immer sie aufgerufen wird, liefert sie das nächste Token des Quellprogramms und liest dazu soviel Quelltext wie nötig.

Die nächstliegende Methode, die Prozedur zu implementieren, ist wohl etwa der folgenden Bauart: Lies das nächste Zeichen; wenn es ein Leerzeichen ist, dann lies noch ein Zeichen. Wenn es ein Doppelpunkt ist, dann schau, ob darauf ein Gleichheitszeichen folgt. Wenn ja, dann liefere das Token für ":=" (<becomes>), wenn nein, dann ... . Dieses Verfahren ist nicht nur umständlich und unübersichtlich, sondern es führt meistens auch noch zu einem recht langsamen Scanner. Glücklicherweise hält ein Teilbereich der theoretischen Informatik einen recht einfachen Algorithmus für uns bereit, der nicht nur ein übersichtliches Programmieren des Scanners möglich macht, sondern auch das Maximum an Geschwindigkeit bietet.

Wir wollen uns jetzt natürlich nicht in die Abgründe vieler Sätze und Definitionen begeben, sondern lediglich soviel Nutzen aus der Theorie formaler Sprachen ziehen wie nötig, um unseren Scanner zu bauen.

Automaten

Für den Scanner stellt sich der Quellcode als eine kontinuierliche Folge von ASCII-Zeichen dar. Einen Teil dieser Zeichen, zum Beispiel Kommentare, kann er komplett ignorieren. Aus dem Rest muß er bestimmt Zeichenkombinationen herausfischen und dann das Token für die gelesene Zeichenkombination an den Parser liefern. Eine solche Zeichenkombination ist in Modula-2 zum Beispiel das reservierte Wort *PROCEDURE* oder auch das Sonderzeichen ":". Wie wir weiter oben gesehen haben, kommt dabei manchmal erschwerend hinzu, daß das Token ein Attribut besitzt, wie das zum Beispiel bei Bezeichnern der Fall ist.

Wir können uns den Scanner jetzt als Automaten vorstellen, der einen internen Zustand besitzt. Der Zustand, den der Automat bei der Initialisierung erhält, ist sein Startzustand. Sobald der Automat jetzt ein Zeichen liest, wechselt er von einem Zustand in einen anderen. Daraufhin liest er wieder ein Zeichen und wechselt abhängig vom gelesenen Zeichen und dem aktuellen Zustand in einen neuen Zustand. Dieses Spiel setzt sich fort, bis das Ende der Folge von Eingabezeichen erreicht ist; für unseren Scanner bedeutet dies das Ende des Quelltextes.

Bis jetzt ist die ganze Angelegenheit allerdings noch sehr witzlos, da wir immer noch keine Tokens, geschweige denn Attribute erzeugt haben. Deshalb führen wir neben dem Startzustand noch einen ganzen Satz besonders ausgezeichneter Zustände ein, die als Finalzustände bezeichnet werden. Der Automat kommt in einen solchen Finalzustand, wenn er eine Buchstabenfolge erkannt hat, die in ein Token umgewandelt werden kann.

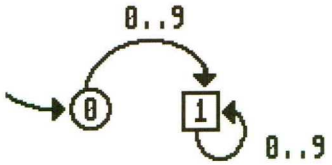


Abb. 3: Endlicher Automat für natürliche Zahlen

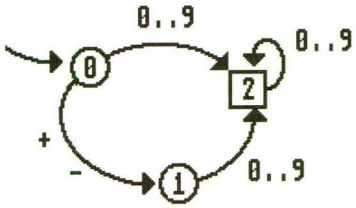


Abb. 4: Endlicher Automat für ganze Zahlen

Um das ganze etwas anschaulicher zu gestalten, sehen wir uns jetzt erst mal ein Beispiel an. In Abb. 3 ist das Zustandsübergangsdiagramm eines solchen Automaten abgebildet. Jeder Kreis und jedes Quadrat mit einer Zahl repräsentiert einen Zustand des Automaten, dabei gibt die Zahl die Nummer des Zustandes an. In unserem Beispiel ist Nummer 0 der Startzustand. Erkennen kann man das an dem Pfeil, der von keinem Zustand ausgeht und im Zustand 0 endet. Alle Finalzustände werden durch Quadrate dargestellt, in unserem Beispiel ist dies also nur der Zustand Nummer 1. Jeder Pfeil stellt einen möglichen Zustandswechsel des Automaten dar und ist mit den Eingabezeichen beschriftet, die diesen Zustandswechsel auslösen. Da ein solcher Automat nur endlich viele verschiedene Zustände einnehmen kann, nennt man ihn einen endlichen Automaten. Unser Beispielautomat besteht sogar nur aus zwei Zuständen.

Interessant ist nun, für welche Eingabesequenzen der abgebildete Automat seinen Finalzustand erreicht. Dies ist offensichtlich bei einer einzelnen "7" der Fall, aber auch jede beliebige andere Ziffer sorgt für einen Zustandswechsel von Zustand 0 auf Zustand 1. Was passiert, wenn wir dem Automaten im Startzustand die Zeichenfolge "82" vorsetzen? Die "8" sorgt für ein Wechsel von 0 auf 1, und die "2" wechselt von Zustand 1 in 1, bewirkt also keine echte Änderung des Zustands. Somit ist der Automat auch nach dem Akzeptieren der Folge "82" im Finalzustand 1. Offensichtlich gilt dies für alle mindestens ein Zeichen langen Folgen, die nur aus Ziffern bestehen. Der Automat akzeptiert also alle Zeichenfolgen, die eine natürliche Zahl repräsentieren.

In Abb. 4 wurde unser Beispielautomat so erweitert, daß er optional noch ein Vorzeichen akzeptiert. Er besteht aus drei Zuständen, und diesmal ist Nummer 3 der Finalzustand.

Mit den endlichen Automaten steht uns also ein recht übersichtlicher Mechanismus zur Verfügung, um die Menge aller zu akzeptierenden Zeichenfolgen zu beschreiben. Wie wir später sehen werden, läßt sich ein endlicher Automat auch sehr einfach und effizient programmieren. Leider können wir in der bisher beschriebenen Form noch keinen kompletten Scanner als endlichen Automaten darstellen. Warum das so ist und wie wir Abhilfe schaffen können, werden wir gleich sehen. Vorher wollen wir uns allerdings noch einen etwas komplizierteren endlichen Automaten ansehen. Dieser soll nicht nur ganze Zahlen, sondern auch Fließkommazahlen akzeptieren. Dazu sehen wir uns zuerst einen Automaten an, der nur Fließkommazahlen akzeptiert, und verschmelzen diesen dann mit dem aus Abb. 4.

Verschmelzung

Der Automat für Fließkommazahlen ist in Abb. 5 dargestellt. Er besteht immerhin aus sieben verschiedenen Zuständen und hat zwei Finalzustände. Er endet im Finalzustand 4, wenn er eine Fließkommazahl ohne Exponenten akzeptiert hat, und mit Exponenten im Zustand 7. Startet man den Automaten zum Beispiel mit der Zeichenfolge *2.34E+7*, durchläuft er nacheinander die einzelnen Zustände 0, 2, 1, 4, 4, 5, 6 und 7.

Hängen wir die beiden Automaten aus Abb. 4 und 5 einfach im Startzustand zusammen, erhalten wir den Automaten, der in Abb.6 dargestellt ist. Dieser Automat hat leider einen großen Nachteil. Läßt man ihn zum Beispiel das Zeichen "0" im Startzustand akzeptieren, kann er sowohl in Zustand 9 als auch in Zustand 2 übergehen. Er würde also das eine Mal einen Finalzustand erreichen und das andere Mal nicht. Einen solchen Automaten nennt man indeterministisch, da er sich bei gleicher Ausgangssituation zufällig für zwei verschiedene Wege entscheiden kann. Da wir in unseren Scanner keinen Zufallszahlengenerator einbauen möchten, ist ein solcher Automat für uns uninteressant. Glücklicherweise existiert zu jedem indeterministischen ein deterministischer endlicher Automat (DEA), der genau dieselben Zeichenfolgen akzeptiert. Leider

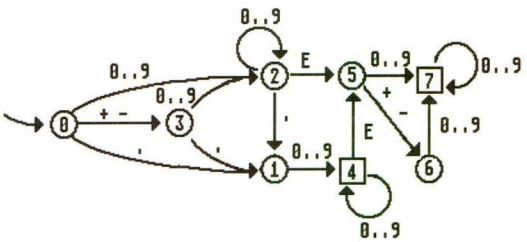


Abb. 5: Endlicher Automat für Fließkommazahlen

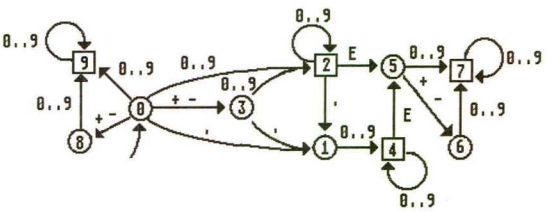


Abb. 6: Indeterministischer Automat für Fließkomma- und ganze Zahlen

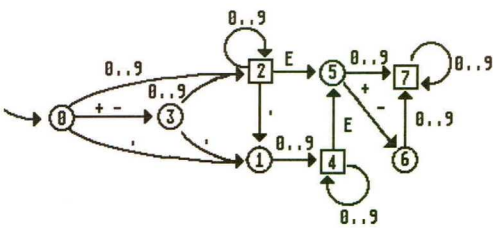


Abb. 7: Deterministischer Automat für Fließkomma- und ganze Zahlen

ist der Algorithmus, der eine solche Umwandlung für jeden beliebigen Automaten durchführt, nicht ganz einfach, so daß er den Rahmen dieses Artikels sprengen würde.

Für die meisten Probleme, die beim Bau von Scannern auftreten, ist er auch gar nicht notwendig, weil man den zugehörigen DEA auch leicht ohne Algorithmus findet. In unserem Beispiel entspricht das Zusammenspiel der drei Zustände 0, 8 und 9 genau dem der Zustände 0, 3 und 2, nur daß 2 kein Finalzustand ist. Dies ist auch sehr einleuchtend, da 0, 3 und 2 das Akzeptieren der ganzen Zahl, die vor dem Punkt oder "E" einer Fließkommazahl steht, beschreiben. Wenn wir nun eine Verschmelzung dieser Zustände durchführen, erhalten wir den DEA in Abb. 7. Dieser endet im Finalzustand 2, sobald er eine ganze Zahl akzeptiert hat.

Muster

Bevor wir einen DEA zur Implementierung eines Scanners benutzen können, müssen wir allerdings immer noch etwas Denkarbeit leisten. Ein DEA bekommt normalerweise eine Zeichenfolge im Startzustand vorgesetzt; und wenn er sich nach dem Akzeptieren der kompletten Folge in einem Finalzustand befindet, hatte die Zeichenfolge die richtige Form. In einem Scanner wollen wir aber innerhalb einer sehr langen Folge von Zeichen alle Vorkommen von reservierten Wörtern, erlaubten Sonderzeichen, Bezeichnern usw. der zu verarbeitenden Programmiersprache finden. Wir wissen dabei nicht vorher, wo diese anfangen und aufhören.
Indem wir den DEA derart erweitern, daß er auch Kommentare und Folgen von Leerzeichen, Zeilenumbrüchen und ähnlichem akzeptiert und dabei jeweils in einem Finalzustand landet, haben wir das Überlesen von Kommentaren und Formatierung auch mit dem Mechanismus der DEAs gelöst. Allerdings liefert der Scanner kein Token an den Parser, sobald er in einem solchen Finalzustand ist, sondern versucht sofort, die Zeichenkette für das nächste Token zu lesen.

Wenn wir einen solchen DEA in einem Scanner benutzen wollen, gilt es jetzt nur noch ein sehr wichtiges Problem zu lösen. Sobald der DEA aus Abb. 7 zum Beispiel die Folge *3.1* liest, befindet er sich in dem Finalzustand 4 und kann das Token für eine Fließkommazahl an den Parser übergeben. Es könnte aber durchaus sein, daß direkt danach die Folge *41* kommt. Statt nach dem Token für die Fließkommazahl nun noch ein Token für eine ganze Zahl zu liefern, ist wohl die natürlichere Interpretation, eine Fließkommazahl nach dem



kompletten Akzeptieren von *3.141* zu melden. Noch schlimmer wird es bei der Analyse von Modula-2-Deklarationen. Hier kann die Zeichenfolge *3.* zum Beispiel am Anfang von *3.1* und *3..4* vorkommen. Im ersten Fall ist *3.* der Anfang einer Fließkommazahl, und im zweiten Fall ist *3* eine ganze Zahl und ".." ein Sonderzeichen, das bei der Deklaration von Unterbereichstypen verwendet wird. Aus dieser Misere hilft uns das Prinzip des längsten Musters (principle of the longest match). Es besagt, daß wir solange wie möglich Zeichen akzeptieren und uns dabei immer den letzten Finalzustand merken, den wir erreicht haben. Sobald es nicht mehr weitergeht, melden wir das zum letzten erreichten Finalzustand gehörige Token. Leider ist es dazu auch nötig, die Zeichen, die seit dem letzten Finalzustand gelesen wurden, in die zu akzeptierende Zeichenfolge zurückzustellen. Da es sowieso nicht sinnvoll ist, jedes Zeichen der Quelldatei einzeln vom Betriebssystem zu holen, sondern besser ganze Blöcke gelesen werden, ist dies aber nicht sehr schlimm.

Natürlich kann es jetzt auch passieren, daß kein Zeichen mehr akzeptiert werden kann, obwohl noch nie ein Finalzustand erreicht wurde. Dies bedeutet, daß eine Zeichenfolge gelesen wurde, zu der es kein Token gibt. Eine solche Zeichenfolge kann aber nur in einer fehlerhaften Eingabedatei stehen und führt somit zu einem Übersetzungsfehler der Bauart *Unerlaubtes Zeichen*. Genau genommen müßte man im Zustandsübergangsdiagramm eines DEAs noch einen weiteren Finalzustand einführen, und von jedem anderen Zustand führt ein Pfeil, der mit allen Eingabezeichen versehen ist, für die das Verhalten sonst nicht definiert ist, zu diesem speziellen Finalzustand, der auch als Fehlerzustand bezeichnet wird. Der Übersicht halber läßt man diesen Fehlerzustand im Zustandsübergangsdiagramm aber meistens weg.

Jetzt können wir unseren DEA endlich zum Bau eines Scanners verwenden. Da die Eingabe sowieso gepuffert werden muß, kann der Scanner sich auch die aktuelle Pufferposition merken, sobald er im Startzustand ist. Hat er zum Beispiel einen Bezeichner akzeptiert, steht die Zeichenfolge des Bezeichners genau zwischen der im Startzustand gemerkten und der aktuellen Pufferposition. Diese Zeichenfolge wird an die Bezeichnertabelle (siehe erste Folge) übergeben, und die von ihr für den Bezeichner gelieferte Zahl wird als Attribut mit dem Token an den Parser übergeben.

Gesteuert

Nachdem wir nun das komplette Akzeptierverhalten eines Scanners mit einem DEA beschreiben können, stellt sich die Frage, wie wir einen DEA in ein lauffähiges Programm umsetzen können. Dafür gibt es prinzipiell zwei Möglichkeiten. Die erste nutzt die Tatsache, daß ein DEA auch mit einer Zustandsübergangstabelle beschrieben werden kann. Eine solche Tabelle hat für jeden Zustand des DEA eine Zeile und für jedes mögliche Eingabesymbol eine Spalte. Der Zustand, der erreicht wird, sobald der Automat in einem Zustand *q* ein Eingabezeichen *t* akzeptiert, wird im Kreuzungspunkt der Zeile von *q* und der Spalte von t notiert. Eine solche Tabelle kann man zum Beipiel in einem zweidimensionalen Array ablegen und den Zustandsübergang dann mit einer Anweisung der Art *CurrState := StateTable [CurrState, CurrChar]* realisieren, wobei die Zustandsübergangstabelle in *StateTable*, der aktuelle Zustand des Automaten in *CurrState* und das aktuelle Eingabezeichen in *CurrChar* gespeichert sind. Nach jedem Zustandsübergang wird dann überprüft, ob der aktuelle Zustand ein Finalzustand ist. Wenn ja, wird er als letzter erreichter Finalzustand gemerkt. Sobald ein Übergang in den oben erwähnten Fehlerzustand stattfindet, wird die zum letzten gemerkten Finalzustand gehörige Aktion ausgeführt. Beim Erkennen von Kommentaren oder ähnlichem ist dies nur ein Rücksetzen in den Startzustand, sonst wird in der Regel ein dem letzten Finalzustand entsprechendes Token an den Parser geliefert, wobei vorher evetuell noch ein Attribut berechnet wird. Dies kann zum Beispiel das schon erwähnte Eintragen eines Bezeichners in die Bezeichnertabelle sein. Wurde bis dahin allerdings noch kein Finalzustand erreicht, wird eine Fehlermeldung ausgelöst. Nach diesem Verfahren implementierte Scanner heißen tabellengesteuert.

Ausprogrammiert

Eine weitere Möglichkeit ist, den gesamten Scanner als große Case-Anweisung (für C-ler: entspricht einem switch) zu realisieren. Dabei existiert für jeden möglichen Zustand des DEA ein Case-Fall.

PASCAL Bücher,
für Einsteiger und Fortgeschrittene

MERKMALE Band I: Einsteiger Buch
Haben Sie genug vom Spaghetti-Code unübersichtlicher Basic-Programme? Hat Ihr Monitor Kopfschmerzen vom Bomben fehleranfälliger C-Programme? Dann sollten Sie Pascal kennenlernen!
Pascal ermöglicht:
- klar gegliederte und strukturierte Programme
- flexible und selbstdefinierte Datentypen

ST-PASCAL PLUS ermöglicht darüber hinaus:
- einfachen Zugriff auf sämtliche Routinen des Betriebssystems
- komfortable Nutzung der graphischen Benutzeroberfläche GEM

In zwei Bänden machen wir Sie vom Einsteiger zum Pascal Profi.

INHALT Band I:
- das Buch beginnt bei Null und setzrt keine Vorkenntnisse voraus. Es stellt eine umfassende und leichtverständliche Einführung dar. Übungsaufgaben am Ende jedes Kapitels bieten dem Leser die Möglichkeit, die gewonnenen Erkenntnisse in die Praxis umzusetzen. Vergleiche zu Basic erleichtern es den ehemaligen Basic-Programmieren umzusteigen. Besondere Beachtung erfährt das von CCD vertriebene ST-PASCAL PLUS System.

HARDCOVER
inclusive Programmdiskette
Bestell Nr. B-439
ISBN 3-923250-81-9

DM 59,--



MERKMALE:
- fängt bei Null an und führt Sie Schritt für Schritt in die Programmiersprache Pascal ein
- die Beispiele wurden so gewählt daß sie:
+in der Problemstellung auch für Nichtmathematiker nachvollziehbar ist
+in der Praxis immer wieder auftauchen
+die besonderen Möglichkeiten von Pascal verdeutlichen
+die besonderen Möglickeiten des Computers auch im nichtmathematischen Bereich verdeutlicht
+der Leser wird vertraut gemacht mit den wichtigen Programmstrukturen
. Konstanten, Datentypen und Variable
. Kontollstukturen, Bedingungen und Verzweigungen
. komplexe Datenstrukturen
. Sortieralgorithmen, binäres Suchen
. Zeiger, Listen, Bäume
. dynamische Datenstrukturen



Das Buch orientiert sich zwar am für ATARI konzipierten ST Pascal Plus, berücksichtigt aber auch USCD-Pascal und Turbo Pascal
Die fast nebenbeizu erarbeitenden Programme wurden so gestaltet, daß sie praktisch einsetzbat sind und je nach Anforderung leicht modifiziert werden können.

HARDCOVER
über 390 Seiten
Bestell Nr. B-447
ISBN 3-923250-89-4

DM 49,--



MERKMALE Band II: Profi Buch
Wenn Sie:
- Pascal zwar in den Grundzügen beherrschen, aber höher hinaus wollen
- proffessionelle Programme unter GEM erstellen wollen
- die speziellen Eigenschaften von ST-PASCAL-PLUS kennenlernen wollen,
- einen Lehrgang in höheren Programmiersprachen machen wollen,
- einfach das letzte aus Ihrem ST herausholen wollen,
dann sollten Sie dieses Buch lesen !!!

INHALT Band II:
- Höheres Programmieren
zirkuläre Listen, binäre Bäume, Stacks, Heaps, und Queues, einfach und doppelt verkettete Listen, auswertung mathematischer Terme etc.
- ST-Spezial
Module und Bibliotheken, externe Routinen nutzen, der Supervisor, Peek und Poke, VT-52 Emulation, eigene Bibliotheken erstellen
- Der Zugriff aufs Betriebssystem
BIOS, XBIOS, GEMDOS, Joystickabfrage, das Diskettenformat des ST, Disketten selber formatiern, IKBD-Commands
- Kommunikation mit ASSEMBLER und C
Zusammenarbeit mit anderen Sprachen, Parameterübergabe, Supexec und Pexec
- Grafik/GEM
Line A-Grafik, Zugriff auf GEM, Alertboxen, Dialoge, Fileselector, RSC-Files, Windows, das begrenzte Multitasking in GEM, GEM intern, Mausprogrammierung, Events und Messages, Pulldown-Menüs, Desk-Accessories

HARDCOVER
inclusive Programmdiskette
Bestell Nr. B-444
ISBN 3-923250-96-7

DM 59,--

BESTELL - COUPON

Heim-Verlag

Heidelberger Landstr.194
6100 Darmstadt-Eberstadt
Telefon 06151 - 56057
Telefax 06151 - 56059

Bitte senden Sie mir

____ Pascal auf dem ATARI Band I a 59,-- DM
____ Pascal auf dem Atari Band II a 59,-- DM
____ Pascal für Nichtmathematiker a 49,-- DM

zzgl. Versandk. DM 6,--
(Ausland DM 10,--)

unabhängig von der bestellten Stückzahl

Name, Vorname _____
Straße _____
PLZ, Ort _____
oder benutzen Sie eingeheftete Bestellkarte

in Österreich:
RRR EDV GmbH
Dr. Stumpf Str. 118
A-6020 Insbruck

in der Schweiz:
Data Trade AG
Landstr.1
CH-5415 Rieden-Baden

Der Else- bzw. Default-Fall der Case-Anweisung wird als Übergang in den Fehlerzustand behandelt. Lösen alle Eingabesymbole bis auf eines einen Übergang in den Fehlerzustand aus, ist eine If-Anweisung allerdings günstiger. Wurde die Auswahl bezüglich aktuellem Zustand und aktuellem Eingabesymbol getroffen, wird der aktuelle Zustand auf den Folgezustand gesetzt, und falls dies ein Finalzustand ist, wird er als letzter erreichter Finalzustand gemerkt.

Diese zweite Möglichkeit, einen DEA zu implementieren, ist bei einer Programmierung von Hand natürlich sehr viel übersichtlicher als ein tabellengesteuerter Scanner. Wird der Scanner aber automatisch von einem Werkzeug wie LEX (siehe erste Folge) erzeugt, ist es einfacher, einen tabellengesteuerten Scanner zu erzeugen. Außerdem bieten sich bei einer Tabelle noch verschiedene Komprimierungsverfahren an, mit denen man Speicherplatz sparen kann. Wie sich ein Scanner automatisch erzeugen läßt, sehen wir uns gleich noch etwas genauer an.

Für beide Implementierungsverfahren bietet sich noch eine recht wesentliche Optimierung an. In dem Zustandsübergangsdiagramm eines DEA gibt es meist sehr viele Finalzustände, die unabhänig vom nächsten Eingabezeichen immer in den Fehlerzustand übergehen. In diesem Fall ist es natürlich purer Unfug, sich diesen Finalzustand als letzten erreichten Finalzustand zu merken, nur um im nächsten Schritt festzustellen, daß der DEA in den Fehlerzustand übergeht, um daraufhin zu besagtem Finalzustand zurückzukehren. Da das nächste Eingabezeichen hier für das Verhalten des DEA keine Rolle spielt, kann die zu dem Finalzustand gehörige Aktion sofort zur Ausführung kommen, der Finalzustand muß also auch nicht gemerkt, und es müssen keine Eingabezeichen in die zu akzeptierende Zeichenfolge zurückgestellt werden.

Automatisch

Wie versprochen, wollen wir uns jetzt kurz ansehen, welche Mittel zur automatischen Erzeugung von Scannern bereitstehen. Der Scanner-Generator LEX fand in der ersten Folge schon Erwähnung. Er gehört zum Betriebssystem UNIX und hat wohl schon vielen Programmierern das Leben erleichtert. Inzwischen gibt es schon mehrere Weiterentwicklungen von LEX, wie FLEX (Fast LEX) oder REX (Regular EXpressions), die LEX sowohl in Geschwindigkeit und Speicherbedarf des erzeugten Scanners als auch in der Generierungszeit um Längen schlagen. Einem solchen Generator kann man natürlich kaum eine Grafik des Zustandsübergangsdiagramms eingeben. Auch die Beschreibung des DEAs mit Hilfe der Zustandsübergangstabelle wäre zu umständlich und bei einem etwas größeren Scanner auch viel zu fehleranfällig. Stattdessen füttert man ihn mit einer Beschreibung der Zeichenfolgen, die zu einem Finalzustand führen. Eine solche Beschreibung besteht allerdings nicht aus der Aufzählung aller möglicher Zeichenfolgen, dies wären bei Bezeichnern auch unendlich viele, sondern wird mit Hilfe regulärer Ausdrücken vorgenommen. Diese sind wie folgt definiert: Jeder Buchstabe ist ein regulärer Ausdruck und steht für die Zeichenfolge, die nur aus diesem einen Buchstaben besteht. "X | Y" ist ein regulärer Ausdruck und steht für "X oder Y". "XY" ist ein regulärer Ausdruck und steht für "erst X, dann Y". "X*" ist ein regulärer Ausruck und steht für die leere Zeichenfolge, "X", "XX", "XXX" usw. "X+" ist ein regulärer Ausdruck und steht für "X", "XX", "XXX" usw. Dabei dürfen Klammern natürlich wie gewohnt verwendet werden. Der Automat aus Abb. 3 entspricht etwa dem regulären Ausdruck "('0'|'1'|'2'|'3'|'4'|'5'|'6'|'7'|'8'|'9')+" oder kürzer "('0'|...|'9')+". Der Fließkommazahlenautomat aus Abb. 4 entspricht "(['+'|'-'] ('0'|...|'9')+ 'E' ['+'|'-'] ('0'|...|'9')+)|['+'|'-'][('0'|...|'9')+]'.'('0'|...|'9')+ ['E' ['+'|'-'] ('0'|...|'9')+]". Dies ist auf den ersten Blick nicht gerade besonders übersichtlich. Sobald man sich an diese Form der Darstellung etwas gewöhnt hat und "('0'|...|'9')" noch durch *Digit* abkürzt, steht es dem Zustandsübergangsdiagramm nicht sehr nach. In jedem der genannten Scanner-Generatoren schreibt man hinter einen solchen regulären Ausdruck jeweils ein kurzes Programmstück, das die Aktion enthält, die ausgeführt werden soll, sobald im Quelltext eine Zeichenfolge erkannt wurde, auf die der reguläre Ausdruck paßt. Meist ist dies nur die Rückgabe eines Tokens an den Parser. Das Verschmelzen der einzelnen Automaten, Reduzierung der Anzahl der Zustände, Erzeugung der Zustandsübergangstabelle und Generierung des Scanners als C- oder Modula-2-Programm wird von den Scanner-Generatoren nun automatisch durchgeführt.

Wen die Algorithmen zur Erzeugung des DEA aus den regulären Ausdrücken und zur Reduktion der Zustände näher interessieren, kann beides in [1] nachlesen. Beide basieren auf der Theorie formaler Sprachen, sind beweisbar korrekt und nicht gerade leicht verdaulich.

Und wie geht's weiter?

Das war's für dieses Mal. Abgesehen von dem compilertechnischen Wissen enthält der Abschnitt über den Bau der lexikalischen Analyse auch ein gutes Beispiel für das prinzipielle Vorgehen in der Informatik. Man hat ein Problem (hier den Bau eines Scanners), dessen direkte Lösung zu unübersichtlich ist. Also sucht man sich ein abstraktes Modell des Problems (hier den DEA mit dem Prinzip des längsten Musters), das in ähnlicher Form schon möglichst lange als gut untersucht gilt (hier die DEAs, die schon erforscht wurden, als niemand an Compiler gedacht hat). Nun sucht man ein möglichst effizientes Implementationsschema für das Modell (hier den tabellengesteuerten und den ausprogrammierten Scanner). Um ein Programm zu schreiben, in dem das so behandelte Problem auftritt, führt man zuerst einen kompletten Entwurf auf der Basis des abstrakten Modells durch. Erst anschließend wird dieses mit Hilfe des Implementationsschemas in ein lauffähiges Programm umgesetzt und dann gegebenenfalls noch von Hand optimiert. Soll ein derart implementiertes Programm geändert werden, führt man alle Änderungen erst am Modell durch und baut sie dann in das Programm ein. Dadurch erspart man sich oft einen ganzen Haufen von Sackgassen, erhöht die Zuverlässigkeit der Software und erleichtert die Fehlersuche. Fast umsonst liefert uns diese Vorgehensweise auch eine Methode zur Erstellung von Werkzeugen, die die Umsetzung des Modells in ein lauffähiges Programm übernehmen.

In der nächsten Folge werden wir ein weiteres Beispiel für diese Methode des Programmentwurfs kennenlernen, denn auch die Funktion eines Parsers läßt sich durch einen Automaten beschreiben. Allerdings werden wir uns dann von den endlichen Automaten trennen und zu den Kellerautomaten übergehen müssen.

Manuel Chakravarty

Literatur:
[1] Aho/Sethi/Ullman: „Compilers: Principles, Techniques and Tools", Addison-Wesley
[2] Waite/Goos: „Compiler Construction", Springer

Datenstrukturen & Algorithmen

in Omikron.BASIC und Modula 2



Teil 2

Die wesentlichen Operationen auf lineare Listen sind Suchen, Einfügen, Löschen von Elementen. Ihre Realisierung ist abhängig vom Listentyp: Wir betrachten Keller, Schlangen und geordnete Listen.

Der *Keller* (Stapel, engl. 'stack') verwaltet die Elemente nach dem LIFO-Prinzip (Last In First Out). Das zuletzt gespeicherte Element wird zuerst gelesen. Vergleichbar ist der Keller mit einem Bücherstapel. Ein neues Buch legen Sie oben auf den Stapel. Damit Ihr Turm nicht umfällt, entfernen Sie immer das oberste Buch - streng nach der biblischen Regel: "Die Ersten werden die Letzten sein".

Die *Schlange* (engl. 'queue') organisiert die Elemente nach dem FIFO-Prinzip (First In First Out). Neue Element werden an das Ende der Schlange angehängt. Gelöscht wird immer das Element am Anfang. Eine Warteschlange vor einer Telefonzelle verdeutlicht diese Verwaltung: Wer sich zuerst anstellt, darf zuerst telefonieren.

Die *geordnete Liste* verwaltet die Elemente nach einer bestimmten Ordnung - sie ist sortiert. Beispielsweise speichert eine geordnete Zahlenliste die Zahlen nach ihrer Größe bezüglich der Kleiner-Relation '<'. Für Wortlisten bietet sich die lexikographischen Ordnung an. Auch Texte speichert man meist in geordneten Listen. Die Listenelemente sind Textzeilen, Sätze oder Absätze. Die zugrundeliegende Ordnung folgt aus der Semantik des Textes, das heißt dem Textinhalt.

Das *Schlüsselfeld* ist die Komponente der Datenstruktur, nach welcher die Liste geordnet ist. Eine geordnete Liste mit Personendaten speichern Sie beispielsweise nach dem Nachnamen oder nach dem Alter geordnet. Beides gleichzeitig geht nicht, ohne die Linearität zu verletzen.

Keller und Schlange sind nach dem zeitlichen Auftreten der Elemente geordnet, nicht jedoch nach dem Wert eines Schlüsselfeldes. Wir bezeichnen sie deshalb als ungeordnet. Die Einfüge- und Löschposition ist fest vorgegeben - entweder am Listenkopf oder -ende.

Bei geordneten Listen hängt sowohl die Einfüge- als auch die Löschposition von der Ordnung ab. Beispielsweise fügen Sie 'Meier' zwischen 'Kaiser' und 'Nolte' ein, aber nicht vor 'Adams'. Bei Textlisten gibt der Text-Cursor diese Position an.

Diese drei Listentypen sind Beispiele, an denen sich die Operationen gut demonstrieren lassen. In der Praxis verwendet man oft zusammengesetzte Listentypen. Beispielsweise die Verwaltung von Dateien in einem Druckerpuffer: Ein Druckerpuffer ist ein klassisches Beispiel für eine Schlange - bei manchem Drucker ist die Bezeichnung *Warteschlange* passender. In einem guten Druckerpuffer können Sie eine noch nicht ausgedruckte Datei aus der Warteliste löschen. Dies geht nicht, wenn Sie sich streng an die Regeln der Schlange halten: Einfügen am Ende und Löschen am Anfang. Sie benötigen zusätzlich die Operation *Suchen mit anschließendem Löschen*.

Ein weiteres Beispiel ist ein Keller, in dem jedes Element höchstens einmal enthalten ist. Man darf nur dann ein Element einfügen, wenn dieses noch nicht enthalten ist. Die Operation lautet dann: *Einfügen am Kopf mit vorheriger Suche*.

Die folgenden Beispiele beziehen sich auf den Datentyp *Wortliste*. Diesen beschreibt der Deklarationsteil (Anfang) von Listing 3. Ein Element der Wortliste besteht aus einem Wort, welches wir im Feld *Wl_Wort$()* speichern, und einem Cursor auf den Nachfolger in *Wl_Next%()*. Bitte beachten Sie, daß wir nur eine Infokomponente berücksichtigen. *Wl_Wort$* können Sie für Ihre Anwendungen durch beliebig viele Infokomponenten ersetzen.

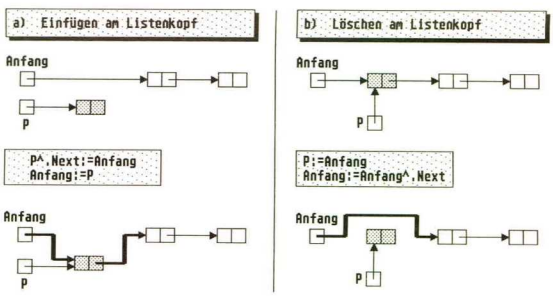


Bild 5a + b

Keller als Listenimplementation

Lineare Listen eignen sich sehr gut zur Realisierung von Kellern. Die zugehörigen Listenoperationen sind Löschen und Einfügen am Listenkopf. Bild 5a stellt die notwendigen Zeigeroperationen dar, wobei die betroffenen Zeiger hervorgehoben sind. Die Befehle sind in Modula 2-Syntax beschrieben. Diese ist anschaulicher als die entsprechende OB-Schreibweise.

Wir fügen das Element, auf das *P* zeigt, am Listenkopf ein. Zuvor besorgen wir mit *Allocate* Speicherplatz für das Element. Zwei Schritte sind notwendig: Zuerst lassen wir den Nachfolger des neuen Elements auf den Listenanfang zeigen. Diese Operation lautet in Modula 2 *P^.Next:=Anfang*. In OB schreiben wir *Wl_Next%(P%)=Anfang%*. Anschließend machen wir mit *Anfang:=P* das neue Element zum Listenanfang. In OB verbiegen wir diesen Zeiger mit *Anfang% = P%*. Bild 6 demonstriert die Cursor-Realisierung am Beispiel der Wortliste 'Karl' -> 'Paul' -> 'Anna'. Nach dem Einfügen lautet die Liste: 'Hans' -> 'Karl' -> 'Paul' -> 'Anna'. Die betroffenen Zeiger sind hervorgehoben.

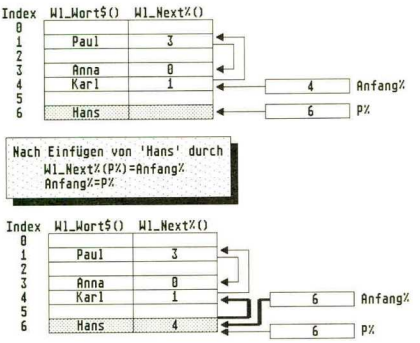


Bild 6: Cursor-Realisierung

Die Prozedur *Ll_Einfuegen_Am_Kopf(R Anfang%,R Ende%,P%)* aus Listing 2 realisiert die Verzeigerung in einer linearen Liste. Sie ist Dank der Cursor *Anfang%* und *Ende%* universell einsetzbar. Die Prozedur kümmert sich nicht um die Beschaffung des Speicherplatzes. Dadurch ist sie unabhängig von den Infokomponenten der zugrundeliegenden Datenstruktur.

Die Prozeduren *Speichern_In_Keller(Wort$)* und *Lesen_Aus_Keller(R Wort$)* in Listing 3 arbeiten auf einer höheren Abstraktionsebene. Sie sind eine Implementierung des abstrakten Datentyps *Keller von Wörtern*. Der einzige Parameter ist das zu speichernde oder zu lesende Wort. Die Prozeduren besorgen die Speicherverwaltung, bearbeiten die Infokomponenten, verbiegen die Zeiger und behandeln Fehler. Die Zeigerbehandlung können Sie in diesen beiden Prozeduren ersetzen durch Aufrufen der Prozeduren *Ll_Einfuegen_Am_Kopf(Keller_Anfang%, Dummy%, P%)* beziehungsweise *LL_Kopf_Loeschen (Keller_Anfang%,Dummy%)* aus Listing 2. *Dummy%* ist eine Dummy-Variable. Das ist ein Platzhalter für einen Parameter, den wir nicht berücksichtigen.

Bild 5b zeigt das Löschen am Listenkopf. Dieses ist genauso einfach wie das Einfügen: Zuerst merken wir uns mit *P:=Anfang* einen Zeiger *P* auf den Anfang der Liste. Dann verbiegen wir den Zeiger auf den Listenanfang mit *Anfang:=Anfang^.Next* auf das zweite Element. Den Speicherplatz für den ehemaligen Listenkopf, darauf zeigt *P*, geben wir mit Deallocate frei. In OB lauten die beiden Zeiger-Befehle: *P%=Anfang%: Anfang%=Wl_Next%(Anfang%)*.

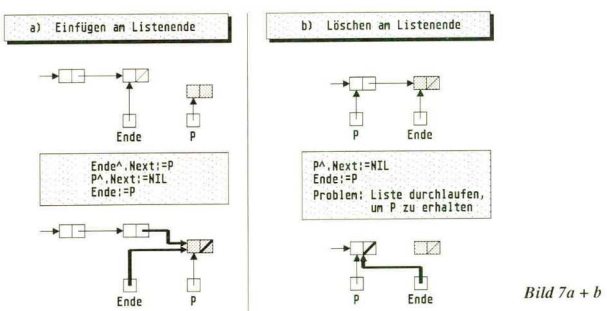


Bild 7a + b

Bitte beachten Sie eine Unschönheit: *Ll_Einfuegen_Am_Kopf()* kümmert sich nicht um die Beschaffung des Speicherplatzes. Dagegen gibt *Ll_Kopf_Loeschen()* mit *Deallocate_Wl()* Speicherplatz frei. Dies ist ein Tribut an die Unabhängigkeit von den Infokomponenten. Die nächsthöhere Abstraktionsebene ist bereits die Keller-Implementierung, bei der diese Unschönheit wegfällt.

Bei fast allen Listenoperationen ist die Reihenfolge der einzelnen Zeigeroperationen wesentlich. Angenommen Sie vertauschen beim Einfügen die Befehle: Der Listenanfang zeigt dann auf das neue Element, bevor dieses auf den Rest der Liste verweist. Die ehemalige Liste ist unerreichbar - Sie finden Ihre Daten im Speicher nicht mehr.

Schlange als Listenimplementation

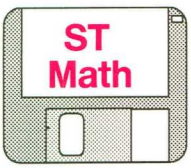
Auch zur Realisation einer Schlange bietet sich eine lineare Liste an. Ein Element lesen wir am Anfang der Schlange; am Ende fügen wir ein neues ein. Das Lesen am Listenkopf haben wir bereits im Abschnitt über Keller beschrieben. Einziger Unterschied: Eine Schlange benötigt einen Zeiger auf das Listenende. Falls Sie das letzte Element der Schlange lesen, wird dieser Zeiger *NIL*. Dies demonstrieren die Prozeduren *Lesen_Aus_Schlange(R Wort$)* in Listing 3 und *Ll_Kopf_Loeschen (R Anfang%,R Ende%)* in Listing 2.

Bild 7a verdeutlicht die Schritte, die am Listenende einfügen. *P* zeige auf das einzufügende Element. *Ende^.Next:=P* hängt das neue Element an die Liste. *P^.Next:=NIL* markiert das neue Listenende und *Ende:=P* läßt den Pointer auf das Listenende auf das neue Element zeigen. In OB lauten diese drei Befehle: *Wl_Next%(Ende%)=P%: Wl_Next%(P%)=0: Ende%=P%*. Die Prozeduren *Ll_Anhaengen(R Anfang%,R Ende%,P%)* aus Listing 2 und *Speichern_In_Schlange(Wort$)* aus Listing 3 enthalten die Implementierung.

Löschen des letzten Listenelementes

Eine weitere wichtige Operation ist das Löschen des letzten Listenelementes. Leider ist dies nicht so einfach wie die bislang beschriebenen Operationen. Bild 7b demonstriert dies. *Ende* zeigt auf das zu löschende Element. Nach dem Löschen muß der *Next*-Zeiger des vorletzten Elementes NIL sein. Dazu benötigen wir einen Zeiger auf das vorletzte Element. Wie finden wir diesen? Wir beginnen am Listenanfang und durchsuchen die Liste solange, bis wir

Nützliche SOFTWARE für Schüler, Studenten und techn. Interessierte



Das Programm für Symbolische Algebra und Analysis in der Version 2..3
Leistungsumfang: + rechnet ohne Rundungsfehler/ + verarbeitet symbolische Ausdrücke wie 2x + 3y = 5z / + löst Gleichungen nach beliebigen Variablen auf / + beherrscht Grenzwerte, Differential- und Integralrechnung / + ideal für Matrizenrechnung / + einfache und komfortable Bedienung / + Wurzeln, Potenzen mit rationalen Exponeneten / +Quadratwurzel / + Quadratische Gleichungen / + Ganze Zahlen / + rationale Zahlen u.v.m.
- ST-Math ist für Schüler ab der 8. Klasse bis zu den ersten Studiensemestern geeignet. ST-Math ist aber auch für Lehrer und Ingenieure bei ihren Berechnungen eine große Hilfe

DM 98,--

Programm für die grafische Präsentation und die Auswertung von Daten
Umfangreiche Optionen und eine eigens entwickelte Benutzeroberfläche ermöglichen die mathematische Bearbeitung und das Verändern und Einstellen aller vom Programm verwendeten Daten und Parameter, sowie die flexible Gestaltung und Beschriftung der mit den Daten erstellten Diagrammen. Das WYSIWYG Prinzip gestattet dem Anwender, ein auf dem Bildschirm dargestelltes

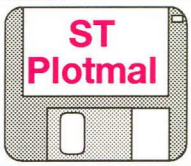


Diagramm voll auf seine Bedürfnisse anzupassen, um es dann auf Nadel-, Laserdrucker oder Plotter auszugeben.
Merkmale: Hilfe-Funktionen/umfangreiche Pufferfunktionen/250 Datensätze und 9999 Wertepaare/Diagramme/Grafische Symbole und Linien/ Beschriftung/ Text- und Wertepaar Editor/Auswerte Funktionen

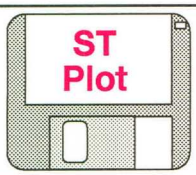
DM 298,--

Life Simulationssoftware auf ihrem ATARI ST
Im Gegensatz zu herkömmlichen Simulationssystemen gibt es eine neue Idee mit Namen CAM. Celluar Automata Machines arbeiten auf rein graphischer Ebene. Heutzutage können Sie komplexe Naturwissenschaftliche Probleme lösen. Anwendungsgebiete von Pro-Cam sind die Simulation von chemischen Reaktionen und physikalischen Prozessen. Ebenso können Fragestellungen der Ökonomie, der Soziologie und der Biologie behandelt werden.
Eigenschaften: mausgesteuerte Oberfläche/hohe Geschwindigkeit (bis zu 1 Generation pro Sec. bei einer Auflösung von 256x160 Pixeln, da vollständig in Assembler programmiert/Filmoption

Beispiele für Chemie, Biologie, Physik, Geographie usw. vorhanden

DM 189,--



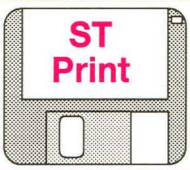


Kurvendiskussions- und Funktionsplottersoftware
- ist ein Programm, das mehr kann als nur Funktionen zeichnen. Es bestimmt die richtigen Ableitungen und damit werden Null-, Extrem- und Wendestellen berechnet. Definitionsbereich und Periode werden festgestellt/zeichnet die Schaubilder von bis zu 3 Funktionen und deren erste und zweite Ableitung auf einen Bildschirm/Komplett GEM-gesteuert/läuft in mittlerer und hoher Auflösung/hohe Rechengeschwindigkeit/Ausschnitte der Funktion können vergrößert und verkleinert werden/einfache Bedienung, sie geben die Funktion ein und den Rest erledigt das Programm/Komfortable Zeichnung und Beschriftung der Achsen und Skalierung

sämtliche Funktionen eines naturwissenschaftlichen Rechners sind vorhanden

DM 69,--

Multiaccessory (5 Programme in einem)
Durch seine Vielseitigkeit und einfache Bedienung erleichtert ST-Print die tägliche Arbeit mit dem Computer und dem Drucker
Resetfeste RAM-Disk: einstellbare Größe von 32-4000 KB/Laufwerk C-P verwendbar
Druckerspooler: Größe einstellbar 2-510 KB/abschaltbar/auch für Hardcopys/optimerte Programmierung
Virenkiller: für Bootsektorviren/als ACC oder PRG

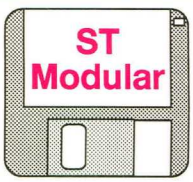


Druckervoreinstellung: Einstellung des Druckers vom Desktop aus/viele Einstellmöglichkeiten/Komfortabel
Druckerzeichenkonverter: Anpassung aller Zeichen des ATARI an den verwendeten Drucker
Hardcopyroutine: für alle 9-18-24 Nadeldrucker und HP-kompatible Laserdrucker/verschiedene Auflösungen/kann an jeden Drucker angepaßt werden

DM 69,--

Das MSM-2 ist ein vollständiges, professionelles Modula-2 Entwicklungssystem
Optimierter Linker zur Erzeugung kompakter Programme und Accessories / Inline-Assembler mit Zugriffsmöglichkeiten auf Modula-Variablen / Projekt Unterstützung / Debugging auf Modula oder Maschinencode-Ebene (auch beides gemischt) / Komplettes Monitorprogramm mit Tracer ist integriert 7 Umfangreiche Modulbibliothek mit Quelltexten / keine Nachladezeiten, da alle nötigen Komponenten im Speicher gehalten werden / in Assembler programmiert, dadurch werden nur ca. 80 KB belegt / extrem schneller Compiler (20.000 Zeilen pro Minute) der direkt vom Editor aus per Tastendruck aufgerufen werden kann und beim auftreten von syntaktischen Fehlern den Editor automatisch aufruft / bis zu 48 Module können gleichzeitig im Speicher gehalten werden, dadurch treten beim Starten eines Programms und beim Übersetzen keine Modul-Ladezeiten auf

DM 298,--



Preise sind unverbindlich empfohlene Verkaufspreise

BESTELL - COUPON

Heim Verlag

Heidelberger Landstr.194
6100 Darmstadt-Eberstadt
Tel.: 0 61 51 / 5 60 57-58
Fax: 0 61 51 / 5 60 59

Bitte senden Sie mir _____

_____

zuzüglich 6,-- DM Versandkosten (Ausland DM 10,--)

Name, Vorname _____

Straße _____

PLZ, Ort _____

oder benutzen Sie eingeheftete Bestellkarte

unabhängig von der bestellten Stückzahl

in Österreich:
RRR EDV GmbH
Dr. Stumpf Str. 118

A-6020 Innsbruck

in der Schweiz:
Data Trade AG
Landstr.1

CH-5415 Rieden-Baden

ein Element finden, dessen *Next*-Zeiger auf das Listenende zeigt. Dies ist der Zeiger *P* aus Bild 7b.

Das Löschen ist nun simpel: Zuerst geben wir den Speicherplatz für das letzte Listenelement frei. *P^.Next:=NIL* markiert das vorletzte Element als Listenende. *Ende:=P* biegt den Pointer auf das Listenende um. Falls *Ende* anschließend *NIL* ist, muß auch *Anfang=NIL* gesetzt werden. Die Liste ist dann leer. In OB lauten die Befehle:

```
Deallocate_Wl(Ende%)
Wl_Next%(P%)=0
Ende%=P%'
```

Der Algorithmus ist auch korrekt, wenn Sie ihn auf eine leere Liste anwenden. Das heißt, *Ende%* ist gleich Null. Dabei darf *Deallocate(0)* keine Fehler erzeugen.

Für die Suche nach dem vorletzten Element benutzen wir eine WHILE-Schleife:

```
P%=Anfang%
WHILE Wl_Next%(P%)<>Ende%
  P%=Wl_Next%(P%)
WEND
```

Die Suche funktioniert nicht, wenn die Liste nur ein Element enthält, das heißt *Anfang%* ist gleich *Ende%*. Wir behelfen uns mit einem Trick: Wir beginnen die Suche nicht mit dem Listenanfang, sondern mit einem leeren Listenelement. Dieses stellen wir vor die Liste. Dadurch zeigt der Next-Zeiger des leeren Listenelementes auf den eigentlichen Listenanfang. Enthält die Liste nur ein Element, ist der Nachfolger von *P%* gleich *Ende%*. Die Schleife endet sofort.

Bei Cursor-Realisierungen bietet sich das nullte Feldelement als leeres Listenelement an. Wie oben bereits erwähnt, sind alle Komponenten des nullten Elementes als undefiniert anzusehen. Wir dürfen *Wl_Next%(0)* einen Wert zuweisen, ohne Schwierigkeiten mit anderen Routinen zu bekommen.

Wir lassen das nullte Element auf den Listenanfang zeigen und starten die Suche mit dem nullten Element. Enthält die Liste nur ein Element, terminiert die Schleife sofort. *P%* ist dann Null, und *Wl_Next%(P%)* zeigt auf das Listenende:

```
Wl_Next%(0)=Anfang%
P%=0
WHILE Wl_Next%(P%)<>Ende%
  P%=Wl_Next%(P%)
WEND
```

Suchen in linearen Listen

Eine sehr wichtige Grundoperation ist die Suche. Oben haben wir einen Zeiger auf das vorletzte Element bestimmt - das Element an einer bestimmten Position. Nun suchen wir ein bestimmtes Element in einer linearen Liste. Wir testen, ob es in der Liste enthalten ist. Meist interessiert ein Zeiger auf dieses oder ein Nachbarelement.

Beispielsweise suchen wir in unserer Personenliste den Herrn Meier. Wir fragen nach dem Element in unseren Personendaten mit dem sogenannten Schlüsselwert 'Meier' im Schlüsselfeld *Personen_Name$()*. Entsprechend suchen wir den Schlüsselwert '5' im Schlüsselfeld *Zl_Zahl%()*.

Wir starten die Suche nach einem Schlüsselwert X am Listenanfang. Wir durchlaufen die Liste nun solange, bis X gefunden wurde oder das Listenende erreicht ist. Den Namen 'Meier' suchen wir folgendermaßen:

```
X$="Meier" ' gesuchter Schlüsselwert
P%=Anfang% ' Suche am Listenanfang
           beginnen
WHILE P%<>0 AND
Personen_Name$(P%)<>X$
  P%=Personen_Next%(P%)  ' P% soll
                         auf den
                         Nachfolger
                         zeigen
WEND
```

Enthält *P%* nach Beendigung der Schleife den Wert Null, so wurde der Schlüsselwert 'Meier' nicht gefunden - das Listenende hat abgebrochen. Andernfalls ist *P%* ein Cursor auf die zugehörige Datenstruktur.

Die WHILE-Schleife läßt sich mit einem Trick beschleunigen: Wir sorgen dafür, daß auf jeden Fall ein Element mit dem gesuchten Schlüssel gefunden wird. Dann ist die Bedingung *P%<>0* überflüssig. Die Schleife unterbricht immer wegen des gefundenen Elementes und nie aufgrund des Listenendes.

Der Trick: Wir hängen ein Element mit dem gesuchten Schlüsselwert an das Ende der Liste. Dieses Element ist ein sogenannter *Wächter* ('Marke', engl. 'sentinel'). Ist *P* nach Beendigung der Schleife gleich dem Zeiger auf den Wächter, ist das Element nicht enthalten - der Suchalgorithmus ist auf den Wächter aufgelaufen.

Die Verwendung eines Wächters bietet sich bei Cursor-Realisierungen von linearen Listen an: Die Infokomponenten des Elementes mit dem Index Null sind unbenutzt. Zudem 'zeigt' das Listenende aufgrund des *NIL*-Cursors Null auf dieses nullte Element. Wir schreiben daher den gesuchten Schlüsselwert in das nullte Element des Schlüsselfeldes. Ist das Element nicht enthalten, so ist *P%* nach der Suche gleich Null.

Bei Pointer-Realisierungen muß man zusätzlichen Speicherplatz für den Wächter anfordern und diesen an die Liste hängen. Dies ist wesentlich aufwendiger.

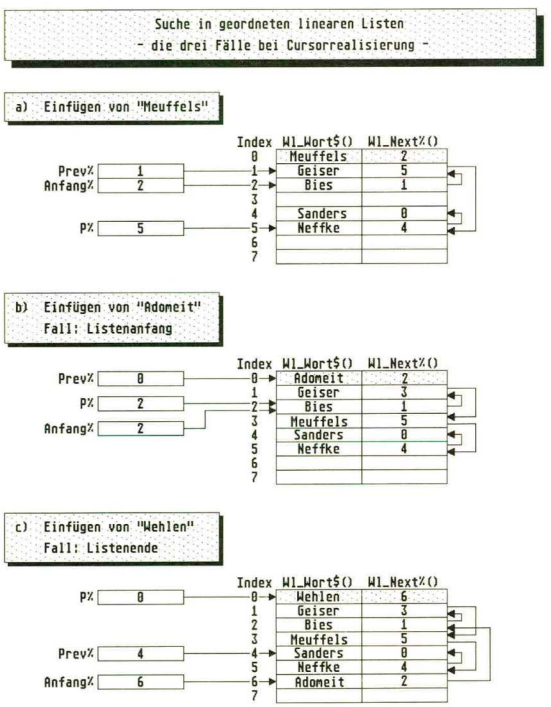


Bild 8a + b + c

```
X$="Meier" ' gesuchter Schlüsselwert
P%=Anfang% ' Suche am Listenanfang
             beginnen
Personen_Name$(0)=X$ ' Wächter ans
                       Listenende
                       hängen
WHILE Personen_Name$(P%)<>X$
  P%=Personen_Next%(P%) ' P% soll
                          auf den
                          Nachfolger
                          zeigen
WEND
' Auswertung
IF P%<>0 THEN ' gesuchter Schlüssel-
                wert gefunden: P%
                zeigt a. das Element
   PRINT Personen_Name$(P%),
         Personen_Vorname$(P%),
   PRINT Personen_Alter%(P%),
         Personen_Telnr%L(P%)
ELSE
   ' nicht gefunden, da P% auf den
     Wächter aufgelaufen ist
   PRINT "Eine Person ";X$;" ist
          nicht gespeichert."
ENDIF
```

Geordnete lineare Listen

In geordneten Listen ist das Einfügen und Löschen komplizierter als bei den beiden bislang beschriebenen Listentypen. Wir benötigen einen Zeiger auf die Einfüge- oder Löschposition. In einigen Fällen ist dieser bekannt. Beispielsweise zeigt in einer Textverarbeitung die Schreibmarke (Text-Cursor) auf die aktuelle Position im Text.

Wir beschäftigen uns im folgenden mit Listen, in denen dieser Zeiger unbekannt ist. Dies sind zum Beispiel Listen mit Datenbankcharakter. Die Elemente seien nach der Ordnung '<' sortiert, und jedes Element soll höchstens einmal vorkommen.

Suchen und Einfügen...

...in eine geordnete lineare Liste - Das Einfügen von Elementen in eine geordnete Liste ist in zwei Teile gegliedert: Zuerst suchen wir die Einfügestelle. Wenn das Element nicht in der Liste enthalten ist, fügen wir es ein.

Für die Suche modifizieren wir den obigen Algorithmus. Oben haben wir einen Zeiger auf das Element mit dem Schlüsselwert berechnet. Die Schleife endet, sobald der Wert des Schlüsselfeldes eines Elements gleich dem Schlüsselwert ist.

In geordneten Listen verschärfen wir die Bedingung: Die Schleife terminiert, wenn der Wert im Schlüsselfeld eines Elements größer oder gleich dem Schlüsselwert ist. Ist das einzufügende Element nicht enthalten, bricht die Schleife beim nächstgrößeren Element ab.

Dazu ein Beispiel: Wir fügen 'Meuffels' in die Liste

'Bies' -> 'Geiser' -> 'Neffke' -> 'Sanders'

ein. Dazu durchläuft der Zeiger *P* solange die Liste, bis *P* auf ein Element zeigt, dessen Wert im Schlüsselfeld größer als 'Meuffels' ist. Diese Schleife endet im Beispiel, sobald *P* auf 'Neffke' zeigt.

Leider hilft uns *P* nicht weiter. Wir können 'Meuffels' nicht vor 'Neffke' einfügen, denn unsere Liste besitzt nur Zeiger auf den Nachfolger. Stattdessen fügen wir 'Meuffels' hinter 'Geiser' ein. Wie bekommen wir den Zeiger auf 'Geiser'? Abhilfe schafft ein zweiter Laufzeiger - auf den Vorgänger. Dieser Zeiger *Prev* läuft immer ein Element hinter dem Suchzeiger *P* her. Dies erreichen wir durch die Anweisungen *Prev:=P; P:=P^.Next* in der Suchschleife. Bild 8a demonstriert die Situation nach der Suche.

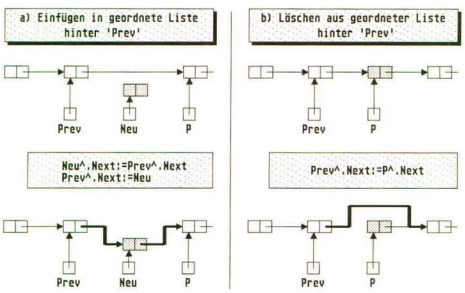


Bild 9a + b

Der Suchalgorithmus nutzt die beiden oben beschriebenen Tricks: Der Wächter markiert das Ende der Suche. Das leere Element am Listenanfang vermeidet die Sonderbehandlung, falls wir am Listenkopf einfügen. *Wl_Wort$(0)* enthält den Wächter, markiert also das Listenende. In *Wl_Next%(0)* speichern wir den Zeiger auf den Listenanfang. Das heißt, das nullte Element ist gleichzeitig Listenanfang und -ende. Bild 8b zeigt den Fall, daß wir vor dem ersten Element einfügen. In Bild 8c ist die Suche auf den Wächter aufgelaufen. Das neue Element hängen wir an die Liste.

Die Prozedur *Liste_Einfuegen(Wort$)* in Listing 3 demonstriert die Suche mit anschließendem Einfügen. Bild 9a veranschaulicht die Zeigeroperationen. Das Element, auf das *Neu* zeigt, fügen wir zwischen den Elementen ein, auf die *Prev* und *P* zeigen. *Neu^.Next:=Prev^.Next* hängt die Liste ab *P* an das neue Element. *Prev^.Next:=Neu* bindet das neue Element ein. In OB lauten diese Befehle: *Wl_Next%(Neu%)=Wl_Next%(Prev%)*: *Wl_Next%(Prev%)=Neu%*.

Sie sehen, daß wir unseren Zeiger *P* für das Einfügen nicht benötigen. *P* ist gleich *Prev^.Next*. Im Programm verwenden wir daher *P* beim Einfügen als Zeiger auf das neue Element. Dies erspart uns eine zusätzliche Variable für den Zeiger *Neu*.

In den Fällen aus den Bildern 8b und 8c sind jeweils die Zeiger auf Listenanfang und -ende zu aktualisieren.

Suchen und Löschen...

... in einer geordneten linearen Liste - Beim Löschen in einer geordneten linearen Liste suchen wir zuerst die Löschposition wie beim Einfügen. Nach der Suche können drei Fälle eintreten: Angenommen, der Laufzeiger *P%* zeigt auf ein Element mit einem größeren Schlüsselwert als dem gesuchten. Dann ist das zu löschende Element nicht in der Liste enthalten - das Löschen erübrigt sich. Ist *P%* gleich Null, ist der Algorithmus auf den Wächter aufgelaufen. Auch dann brauchen wir nichts zu entfernen. Andernfalls löschen wir das Element, auf das *P%* zeigt, denn dieses hat den gesuchten Schlüsselwert. Bild 9b zeigt, daß dazu ein Befehl genügt: *Prev^.Next:=P^.Next* beziehungsweise *Wl_Next%(Prev%)=Wl_Next%(P%)*. Falls wir den Listenanfang oder das Listenende löschen, biegen wir die entsprechenden Zeiger um.

Realisierung von Allocate und Deallocate

Zum Schluß beschreiben wir die Realisierung der Freispeicherverwaltung bei einer Cursor-Realisierung. Wir implementieren *Allocate_xxx(R P%)* und *Deallocate_xxx(P%)*. In Sprachen, die Pointer unterstützen, bearbeitet das Programmsystem die Freispeicherverwaltung.

Den freien Speicherplatz verwalten wir in einer Liste: der Freispeicherliste. Ein Element ist gleichzeitig Cursor auf das nächste freie Element. Wir speichern die Freispeicherliste allein in den *Next*-Komponenten unserer Listenstruktur. Wir berücksichtigen keine expliziten Infokomponenten.

DER MAXON GAL-PROMMER



Das Entwicklungssystem für Logikschaltungen

Der **MAXON GAL-Prommer 16/20** ist das praktische und leistungsstarke Programmiergerät für die gängigen GAL-Typen 16V8 und 20V8. Möchten auch Sie diese modernen Bausteine nutzen, um Ihre Schaltungen schnell, preiswert und flexibel aufzubauen? Mit dem MGP 16/20 kein Problem, denn im ausführlichen Bedienungshandbuch wird auch erklärt, wie herkömmliche Logik (NOR-, NAND-, NOT-, ...-Gatter) in ein GAL programmiert wird und wie diese Bausteine arbeiten.

Neben der enormen Platz- und Stromersparnis bleibt eine einmal aufgebaute Schaltung flexibel, denn ein GAL ist in wenigen Sekunden umprogrammiert und kann somit neuen Anforderungen angepaßt werden. Einige typische Einsatzgebiete eines GALs sind Adreßdekoder, Zustandsautomaten, logische Gatter oder die **PAL-Simulation**.

Zusammen mit dem MGP wird in der neuesten Version ein vollständiger TwoPass Logic-Compiler ausgeliefert. Dieser Compiler übersetzt logische Gleichungen für die GALs 16V8 und 20V8 sowie deren A-Typen in die JEDEC-Dateien, die in das GAL programmiert werden. Dabei führt der Compiler optional eine Optimierung der Logikgleichungen nach Quine-McCluskey durch. Zusammen mit dieser Software-Erweiterung stellt der MGP ein komplettes Entwicklungssystem für logische Schaltungen am ATARI ST dar.

Easytizer - Der Videodigitizer ohne Geheimnisse

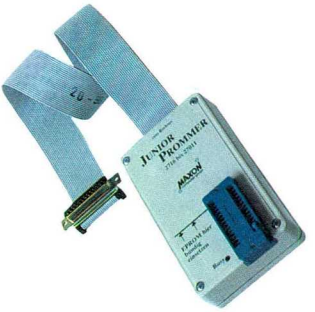
Mit dem **Easytizer** (Bestell-Nr.: 871000) können Sie beliebige Videosignale von einer Schwarzweiß- oder Farbkamera, einem Videorekorder oder direkt vom Fernsehgerät (mit Composite-Video-Ausgang) digitalisieren und somit auf dem Bildschirm Ihres ATARI ST sichtbar machen. Der Easytizer wird am Modul-Port des ST angeschlossen. Die Auflösung beträgt 800x600 Bildpunkte, so daß in mittlerer Auflösung 640x200 Bildpunkte in vier Graustufen dargestellt werden können. In dieser Betriebsart werden 12,5 Bilder in der Sekunde wiedergegeben, im hochauflösenden Modus 640x400 Bildpunkte in Schwarzweiß. Einige Besondere Merkmale des Easytizers:

Software vollständig in Assembler • Von STAD und Sympatic Paint ansteuerbar • Ein beliebiger Bildausschnitt kann in ein mit der Maus wählbares Format vergrößert und verkleinert werden • Spiegeln eines Bildes in horizontaler und vertikaler Ebene • Animation mit beliebig vielen Bildern möglich, nur durch die Kapazität des Rechners begrenzt (beim Mega ST4 über 100 Bilder) • eingebauter Druckertreiber für die mittlere Auflösung für NEC P6/P7 und EPSON oder Kompatible • wahlweise automatische oder manuelle Helligkeitseinstellung • Schnappschuß

Erhältlich ist der Easytizer als Fertiggerät (komplett aufgebaut und geprüft, inklusive Diskette mit der Easytizer-Software und Bedienungsanleitung) oder als Teilesatz (Platine, programmiertes GAL 16V8, Quarzoszillatormodul 32 MHz, Diskette und Bedienungsanleitung).

MAXON Computer GmbH
Schwalbacher Straße 52 • 6236 Eschborn
Telefon 06196/481811 • FAX 06196/41885

DER MAXON JUNIOR PROMMER



Das universelle EPROM-Programmiergerät

Der **Junior Prommer** (Bestell-Nr.: 880310) programmiert alle gängigen EPROM-Typen (und deren CMOS-Typen), angefangen vom 2716 (2 kByte) bis zum modernen 27011 (1 MBit). Mit dem optionalen Mega-Modul-Adaptersockel (Bestell-Nr.: 880313) lassen sich sogar die neuesten 32pol. EPROM-Typen 27010, 27020, 27040 und 27080 brennen. Aber nicht nur EPROMs, sondern auch einige ROM- und EEPROM-Typen lassen sich lesen bzw. programmieren.

Die komfortable Software bietet mannigfaltige Optionen, wie z.B. Zerlegung eines 16-Bit-Words in High- und Low-Byte, 5 Programmieralgorithmen für hohe Datensicherheit sowie einen funktionellenHex-/ASCII-Monitor mit vielen Edierfunktionen. Neben der Treiber-Software werden eine RAM-Disk, ein Programm zum Erstellen von EPROM-Karten, Lese- und Programmierroutinen sowie der Schaltplan mitgeliefert.

Optionaler Adaptersockel für MegaBit-EPROMs erhältlich!

Übersicht der mit dem Junior Prommer programmierbaren Speichertypen

Typ			
Int.ID.			
EPROM'S			
2716	25V	2732	25V
2732A	21V	2764	21V
2764A	12V	27128	21V
27128A	12V	27256	12V
27256	21V	27512	12V
27513	12V	27011	12V
Mit MEGA-Adapter			
✓ 27010	12V	27020	12V
27040	12V	27080	12V
ROM'S			
4732		4764	
47128		47256	
EEPROM'S			
X2804A		X2816A	
X2864A		X28256A	

DPE - Der universelle elektronische Drucker-Port-Expander

Der **DPE** ist eine vielseitige und funktionelle Erweiterungsschaltung für den Drucker-Port des ATARI ST. Welcher Besitzer von mehr als einem Zubehörgerät, das den Drucker-Port belegt, hat nicht schon einmal das dauernde Umstecken sattgehabt? Mit dem DPE ist damit Schluß! Der DPE ermöglicht es, elektronisch zwischen 3 verschiedenen Geräten umzuschalten. Für Geräte, die ihre Stromversorgung über den Joystickport beziehen, bietet der DPE zwei zusätzliche Joystick-Buchsen. Da der DPE über eigene Treiberbausteine verfügt, können die angeschlossenen Geräte problemlos auch über größere Entfernungen angesteuert werden.

Erhältlich ist der DPE als Bausatz (doppelseitige Platine, Stückliste sowie Bauanleitung) unter der Bestell-Nummer 900800.

AUSFÜHRUNGEN UND PREISE:

• MGP-GAL-Programmiergerät	DM	229,00
• MGP-Platine, -Software und -Gehäuse	DM	129,00
• MGP-Update auf Version 2.0 mit Logic Compiler	DM	35,00
• Junior Prommer ST (Fertiggerät wie beschrieben)	DM	229,00
• Adaptersockel Mega Modul f. Junior Prommer	DM	99,00
• Junior Prommer-Leerplatine und -Software (o. Bauteile)	DM	59,00
• Junior Prommer-Leergehäuse (gebohrt und bedruckt)	DM	39,90
• ROM-Karte 128 kByte, bietet max. 4 EPROMs Platz (fertigbestückt o. EPROMs)	DM	58,00
• Easytizer (Fertiggerät)	DM	289,00
• Easytizer (Teilesatz wie oben beschrieben)	DM	129,00
• DPE-Platine und Bauanleitung	DM	49,00

Versandkosten:
Inland DM 7,50
Ausland DM10,00
Auslandsbestellungen **nur** gegen Vorauskasse
Nachnahme zuzgl. DM 4,00 Nachnahmegebühr.

In Bild 4 (siehe 1. Folge) ist *Wl_Fsp_L%* der Cursor auf den Anfang der Freispeicherliste. Das Feld *Wl* ist für sechzehn Elemente dimensio–niert. Zehn Elemente sind belegt. Das Null-Element ist weder frei noch besetzt. Die Freispeicherliste enthält die fünf Elemente mit den Indizes sechs, fünf, dreizehn, zehn und drei. Die Reihenfolge hängt davon ab, wann welche Elemente freigegeben worden sind.

Sie markieren das Element, auf das *P%* zeigt, als frei, indem Sie *P%* in die Freispeicherliste aufnehmen. Entfernen Sie *P%* aus der Freispeicherliste, gilt das Element als benutzt, auf das *P%* zeigt.

Die Freispeicherliste realisiert man als Keller. Es ist unwichtig, welches Element Allocate liefert. Elemente entnehmen und speichern wir daher am Listenkopf. Auf diesen zeigt der Cursor *xxx_Fsp_L%*.

Allocate_xxx(R P%) weist dem Cursor *P%* den Index des ersten Elements der Freispeicherliste zu und entfernt dieses aus der Liste. Falls die Freispeicherliste leer ist, das heißt *xxx_Fsp_L%* gleich Null, redimensioniert Allocate die Felder und aktualisiert die Freispeicherliste. Falls keine Redimensionierung möglich ist, erhält *P%* den Wert Null ('NIL'). Listing 3 enthält den zugehörigen Programmtext.

Allocate_xxx() löscht am Listenkopf der Freispeicherliste. Wir demonstrieren dies am Beispiel aus Bild 4. Mit *Allocate_Wl (Neues%)* fordern wir Speicherplatz für ein neues Listenelement an. Im Cursor *Neues%* bekommen wir den alten Wert von *Wl_Fsp_L%* zurückgegeben. *Neues%* ist nach dem Aufruf also gleich Sechs. Die Freispeicherliste beginnt nach dem Aufruf bei Element fünf.

Die Freispeicherliste ist eine Menge, denn jeder Eintrag kommt höchstens einmal vor. Ein Element darf nur dann als frei markiert werden, wenn es noch nicht in der Freispeicherliste enthalten ist. Die Implementierung von *Deallocate_xxx (P%)* sucht daher zuerst den Cursor *P%* in der Freispeicherliste. Ist er nicht enthalten, dann fügt sie ihn am Kopf ein. Dadurch wird das Element mit dem Index *P%* als frei markiert.

Betrachten wir die Suche nach einem Cursor *P%* in der Freispeicherliste. Prinzipiell benutzen wir obigen Suchalgorithmus. Die *Next*-Komponente ist sowohl Info- als auch Verwaltungskomponente. Wir können daher nicht beide Tricks zur Vereinfachung der Schleife anwenden. Die Implementierung aus Listing 3 speichert in *Wl_Next%(0)* den Wächter *P%*. Den Fall, daß *P%* das erste freie Element ist, fangen wir mit der Abfrage *IF P%= Wl_Fsp_L% THEN* ab.

Bewertung des Zeigerkonzeptes

Zeiger sind ein sehr mächtiges Programmierwerkzeug. Viele Operationen werden durch sie unabhängig von den Daten. Betrachten wir beispielsweise eine Textverarbeitung ohne Zeigernutzung. Die Zeilen sollen in einem Feld von Strings der Reihe nach gespeichert sein - die Zeilennummer entspricht dem Feldindex. Sie haben einen 1000zeiligen Text und wollen vor der ersten Zeile eine Zeile einfügen. Die Lösung ist sehr unbequem: Sie kopieren alle 1000 Zeilen in das nachfolgende Feldelement. Die 1000ste Zeile kopieren Sie an die Stelle 1001 und zum Schluß die erste Zeile an die zweite Stelle im Feld. Anschließend speichern Sie die neue Zeile im ersten Feldelement. Das Löschen der ersten Zeile ist genauso aufwendig. Wieder bewegen Sie 1000 Zeilen.

Mit Zeigernutzung arbeiten Sie effizienter. Aus dem Text machen Sie eine doppelt verkettete lineare Liste. Für jede Zeile speichern Sie zusätzlich je einen Zeiger auf die vorherige und die nachfolgende Zeile. Die Reihenfolge der Zeilen folgt allein aus der Verzeigerung. Es ist unwichtig, wo die Zeilen im Speicher stehen. Eine neue Zeile schreiben Sie irgendwo in das Feld und biegen nur die Zeiger um.

Sämtliche Operationen, die die Reihenfolge der Zeilen betreffen, realisieren Sie ausschließlich über die Zeiger. Die Textzeilen bleiben unberührt. Kurzum: die Operationen erfolgen auf die Zeiger - nicht auf die Daten.

In der nächsten Ausgabe beschreiben wir Radixsort - einen Sortieralgorithmus auf Basis von linearen Listen.

Sven Krüppel

```
'*********************************************
'*                                           *
'* Operationen auf ungeordnete lineare Listen
   im Feld Wl. Diese Routinen                *
'* betreffen nur die Verzeigerung.           *
   Infokomponenten werden nicht berührt.     *
'* (!) Die Prozeduren enthalten keine        *
   Mechanismen gegen fehlerhafte Übergabe-   *
'* parameter.                                *
'*                                           *
'*********************************************

DEF PROC Ll_Anhaengen(R Anfang%,R Ende%,P%)
  'Hängt das Element, auf das P% zeigt, an das
   Ende der Liste an, die durch
  'Anfang% und Ende% charakterisiert ist.
  'Um unabhängig von den Infokomponenten zu
   bleiben, muß der Speicherplatz
  'für das Element vor dem Aufruf mit
   Allocate_Wl(P%) angefordert worden
   sein.
  IF Anfang%=0 THEN Anfang%=P%' Liste war
                                bislang leer
  Wl_Next%(P%)=0'     Ende der verlängerten
                      Liste markieren
  Wl_Next%(Ende%)=P%' neues Element anhängen
  Ende%=P%'           Cursor auf das Listenende
                      umbiegen
RETURN 'Ll_Anhaengen

DEF PROC Ll_Einfuegen_Am_Kopf(R Anfang%,R Ende%,
                              P%)
  'Fügt das Element, auf das P% zeigt, am Kopf
   der durch Anfang% und Ende%
  'charakterisierten Liste ein.
  'Um unabhängig von den Infokomponenten zu
   bleiben, muß der Speicherplatz
  'für das Element vor dem Aufruf mit
   Allocate_Wl(P%) angefordert worden
   sein.
  IF Ende%=0 THEN Ende%=P%' Liste war bislang
                            leer
  Wl_Next%(P%)=Anfang%
  Anfang%=P%
RETURN 'Ll_Einfuegen_Am_Kopf

DEF PROC Ll_Kopf_Loeschen(R Anfang%,R Ende%)
  'Das erste Listenelement wird gelöscht.
  LOCAL P%
  P%=Anfang%' Listenkopf zwischenspeichern
  Anfang%=Wl_Next%(Anfang%)' Listenkopf
                             neusetzen
  Deallocate_Wl(P%)' Speicherplatz freigeben
  IF Ende%=P% THEN Ende%=0'  Liste wird leer
RETURN 'Ll_kopf_loeschen

DEF PROC Ll_Ende_Loeschen(R Anfang%,R Ende%)
  'Das letzte Listenelement wird gelöscht.
  LOCAL P%
  ' Liste durchlaufen, um das vorletzte Element
    zu suchen
  Wl_Next%(0)=Anfang%' Trick: starte mit leerem
                              nullten Element
  P%=0
  WHILE Wl_Next%(P%)<>Ende%
    P%=Wl_Next%(P%)                    →
```

```
48:     WEND
49:     'P% zeigt nun auf den Vorgänger von Ende
50:     Deallocate_Wl(Ende%)' Freigabe des
                              Speicherplatzes
51:     Wl_Next%(P%)=0' neues Listenende markieren
52:     Ende%=P%' Zeiger auf neues Listenende
53:     IF Ende%=0 THEN Anfang%=0' Liste war leer
         oder enthielt nur noch ein Element
54:   RETURN 'Ll_Ende_loeschen
```

```
 1:  '*********************************************
 2:  '*                                           *
 3:  '* Implementierung des abstrakten Datentyps
 4:     'lineare Liste' zur Speicherung v. Worten. *
 5:  '*                                           *
 6:  '* Sven Krüppel, 18.9.90  (c) MAXON Computer  *
 7:  '*                                           *
 8:  '*********************************************
 9:  '* Dieses Programm verwaltet vier Typen von
        linearen Listen und demonstriert          *
10:  '* die zugehörigen Grundoperationen:         *
11:  '*                                           *
12:  '* Die Liste 'Keller' wird nach dem LIFO-
        Prinzip bearbeitet. (Stack)               *
13:  '* Die Liste 'Schlange' wird nach dem FIFO-
        Prinzip bearbeitet. (Queue)               *
14:  '* Die Liste 'Liste' ist eine geordnete
        lineare Liste.                            *
15:  '* Die Freispeicherliste ist eine Menge, d.h.
        eine lineare Liste in der                 *
16:  '* keine doppelten Elemente vorkommen.
        Die Elemente der Freispeicher-            *
17:  '* liste sind Cursor (keine Wörter)          *
18:  '*                                           *
19:  '* Die Listen speichern jeweils ein Wort
        (String). Wir nennen sie Wortlisten.      *
20:  '* Aufbau des Datentyps Wortliste: (Modula 2-
                                        Schreibweise)*
21:  '*    TYPE WortPtr   = POINTER TO Wortliste   *
22:  '*         Wortliste = RECORD                 *
23:  '*                       Wort : String;      *
24:  '*                       Next : WortPtr      *
25:  '*                     END;                  *
26:  '*                                           *
27:  '* Die drei Wortlisten speichern wir in
        OMIKRON.BASIC in den Feldern              *
28:  '* Wl_Wort$() und Wl_Next%() mittels Cursor-
        Realisierung. Die Freispeicher-           *
29:  '* liste arbeitet nur mit dem Feld Wl_Next%.  *
30:  '*********************************************
31:
32:  '***** Globale Variablen *****
33:  Wl_Groesse%=0' Groesse des Feldes zur
                    Speicherung der Wortliste. Der
34:                ' Wert entspricht dem größten
                    zugreifbaren Index, da das
                    nullte-Feldelement nicht
35:                ' mitgezählt ist.
36:  Wl_Fsp_L%=0'   Cursor auf den Anfang der
                    Freispeicherliste
37:  Keller_Anfang%=0'   Cursor auf den Anfang der
                         Liste 'Keller'
38:  Schlange_Anfang%=0' Cursor auf die Leseposition
                         der Liste 'Schlange'
39:  Schlange_Ende%=0'   Cursor auf die
                         Schreibposition der Liste
                         'Schlange'
40:  Liste_Anfang%=0'    Cursor auf den Anfang der
                         geordneten Liste 'Liste'
41:  Liste_Ende%=0'      Cursor auf das Ende der
                         geordneten Liste 'Liste'
42:
43:  '***** Hauptprogramm *****
44:  Dimensioniere_Wl(10,Fehler%)' Feld zur
                                   Speicherung der
                                   Wortlisten erst
                                   einmal
45:                               ' für 10 Elemente
                                   initialisieren.
46:  IF Fehler% THEN PRINT "Zu wenig Speicherplatz":
      END
47:
48:  CLS
```

```
49:  PRINT "Test der Listen zur Speicherung von
           Worten": PRINT
50:  PRINT "Befehle:  SK : Speichern in Keller"
51:  PRINT "          LK : Lesen aus Keller"
52:  PRINT "          SS : Speichern in Schlange"
53:  PRINT "          LS : Lesen aus Schlange"
54:  PRINT "          SL : Speichern in Liste"
55:  PRINT "          LL : Lesen aus Liste"
56:  PRINT "          Q  : Programmende"
57:  PRINT
58:  REPEAT
59:    INPUT "Befehl :";A$:A$= UPPER$(A$)
60:    IF A$="SK" THEN INPUT "Wort:";W$:
         Speichern_In_Keller(W$)
61:    IF A$="LK" THEN Lesen_Aus_Keller(W$):
         PRINT "gelesenes Wort :";W$
62:    IF A$="SS" THEN INPUT "Wort:";W$:
         Speichern_In_Schlange(W$)
63:    IF A$="LS" THEN Lesen_Aus_Schlange(W$):
         PRINT "gelesenes Wort :";W$
64:    IF A$="SL" THEN INPUT "Wort:";W$:
         Liste_Einfuegen(W$)
65:    IF A$="LL" THEN INPUT "Wort:";W$:
         Liste_Loeschen(W$)
66:    Drucke_Wl
67:    PRINT "'Keller'    :";:
         Drucke_Liste(Keller_Anfang%)
68:    PRINT "'Schlange' :";:
         Drucke_Liste(Schlange_Anfang%)
69:    PRINT "'Liste'     :";:
         Drucke_Liste(Liste_Anfang%)
70:    PRINT
71:  UNTIL A$="Q"
72:  END
73:
74:
75:  DEF PROC Drucke_Liste(Anfang%)
76:    ' Druckt die Liste ab Index Anfang%
77:    IF Anfang%=0 THEN PRINT "<leere Liste>";
78:    WHILE Anfang%<>0
79:      PRINT Wl_Wort$(Anfang%);" ";
80:      Anfang%=Wl_Next%(Anfang%)
81:    WEND
82:    PRINT
83:  RETURN ' Drucke_Liste
84:
85:
86:  '*********************************************
87:  '*                                           *
88:  '*  Bearbeitung der Liste 'Keller' als Stapel
        nach dem LIFO-Prinzip                     *
89:  '*  (LIFO = Last In First Out)                *
90:  '*  Operationen: Einfügen und Löschen am
                     Listenkopf                   *
91:  '*                                           *
92:  '*********************************************
93:
94:  '***** Element in 'Keller' speichern *****
95:
96:  DEF PROC Speichern_In_Keller(Wort$)
97:    LOCAL P%' Cursor auf Listenelement
98:
99:    Allocate_Wl(P%)' Speicherplatz für neues
                        Element anfordern
100:   IF P%<>0 THEN 'kein Fehler
101:      Wl_Wort$(P%)=Wort$' Infokomponente
                              eintragen
102:      Wl_Next%(P%)=Keller_Anfang%' Einfügen am
                                       Listenkopf
103:      Keller_Anfang%=P%
104:   ELSE
105:      PRINT "Zu wenig Speicherplatz"
106:   ENDIF
107: RETURN ' Speichern_In_Keller
108:
109: '***** Element aus 'Keller' lesen *****
110:
111: DEF PROC Lesen_Aus_Keller(R Wort$)
112:   LOCAL P%' Cursor auf Listenelement
113:   Wort$=""
114:   IF Keller_Anfang%<>0 THEN ' Keller noch nicht
                                  leer
115:      Wort$=Wl_Wort$(Keller_Anfang%)'
                                 Infokomponente lesen
116:      P%=Keller_Anfang%'     Löschen
           am Listenkopf
117:      Keller_Anfang%=Wl_Next%(Keller_Anfang%) →
```

COMPEDO

SPEZIALFARBBÄNDER GmbH i.G.

Für 90% aller Matrixdrucker in Rot, Gelb, Blau, Pink und Schwarz, oder als 4-Farbband für Colordrucker erhältlich

IHR COMPUTERAUSDRUCK VOM NORMALPAPIER ZUM AUFBÜGELN AUF TEXTILIEN MIT COMPEDO SPEZIAL-FARBBÄNDER

Jetzt auch auf Keramik, Glas, Alu, Metall u. a. Werkstoffen aufdrucken!

Anwendung
- Gegenstand lackieren
- Transfer-Ausdruck mit Klebeband aufkleben
- 15 min. einbrennen (z.B. im Backofen)
- Ausdruck entfernen - Fertig !

Die Entscheidung für das Creative

- Bügeln auf T-Shirts, Jacken, Regenschirme, Kissen etc.
- waschecht - ideal für Werbung
- Lebensdauer wie normales Markenfarband

Normalmarkenfarbbänder zu Superpreisen! z. B.:

CITIZEN SWIFT	34,90	STARLC 10	33,90	CITIZEN SWIFT	9,10	STAR LC 10	7,80
EPSON LQ 500/800	35,90	STAR LC 10 4-COLOR	46,90	EPSON LQ 500/800	10,90	STAR LC 10 4-COLOR	15,70
EPSON LX 80/90	31,90	STAR LC 24 - 10	36,80	EPSON LX 80/90	8,50	STAR LC 24 - 10	11,30
EPSON LQ 2550 4-COLOR	49,90	STAR NL 10 / NB 24 - 10	35,90	EPSON LQ 2550 4-COLOR	24,50	STAR NL 10 / NB 24 - 10	9,10
COMM. MPS 802	36,90	NEC P2200	37,90	COMM. MPS 802	9,20	NEC P2200	12,00
COMM. MPS 803	36,80	NEC P6 + / P7 +	39,90	COMM. MPS 803	9,30	NEC P6 + / P7 +	12,70
COMM. MPS 1500 4-COLOR	49,00	NEC P6 + / P7 + 4-COLOR	59,90	COMM. MPS 1500 4-COLOR	18,95	NEC P6 + / P7 + 4-COLOR	28,40
COMM. MPS 1224 4-COLOR	47,80	NEC P2 / P6	37,50	COMM. MPS 1224 4-COLOR	18,50	NEC P2 / P6	10,20
SEIKOSHA SP	35,90	NEC P* / P6 4-COLOR	59,90	SEIKOSHA SP	12,10	NEC P2 / P6 4-COLOR	28,40
PRÄSIDENT 63 XX	29,90	PANASONIC KXP 10 80/90	36,90	PRÄSIDENT 63 XX	7,90	PANASONIC KXP 10 80/90	10,70
OKI ML 390	36,70	PANASONIC KXP 1124	38,90	OKI ML 390	10,40	PANASONIC KXP 1124	11,70
OKI ML 292 4-COLOR	59,90	APPLE IMAGEWRITER	36,90	OKI ML 292 4-COLOR	29,20	APPLE IMAGEWRITER	8,90

Weitere Preise auf Anfrage – Alle Preise in DM inkl. MwSt.



COMPEDO
Postfach 1352 5860 Iserlohn
Tel.: 02371/41071-72
Fax: 02371/41075

Komplettsysteme für Textildruck mit Verkaufskonzept und Betreuung für Existenzgründer!
★ ★ ★ Rufen Sie an! ★ ★ ★

Normalfarbbänder, auch in Rot, Gelb, Blau, Grün und Braun gegen geringen Aufpreis lagermäßig lieferbar.

Lackset...17.90
(Speziallack, Pinsel, hitzefestes Klebeband und Abroller)

Weiteres Zubehör für den Transferdruck: T-Shirts, Kissenbezüge, Filzposter, Kalender und Puzzles zum bedrucken, auf Anfrage.

Versandpauschale DM 6,- Nachnahme o. Vorkasse (Ausland). Händlerkonditionen auf Anfrage!

Textverarbeitung am ST

Interessieren Sie sich für That's Write? Wir haben das Programm mehrere Wochen lang in der Praxis getestet und einen ausführlichen

That's Write Erfahrungsbericht

verfaßt. Diesen können Sie unter dem Stichwort That's Write Erfahrungsbericht kostenlos bei uns anfordern.

ALTEX Textsysteme
Dipl.-Ing. Georg Altmann Tel.: 0871/78496
Dekan Simburger Str. 13 8300 Ergolding

VIDEO ED8

Video-Schnittsteuerung für 8 mm Camkorder mit dem Atari.

Neue Version 1.5

DM 528, –

Weitere Informationen erhalten Sie bei:

Creative Video
Am Schwegelweiher 2
8551 Hemhofen
Telefon 09195 / 2728
Fax 09195 / 8718

EDISONSTRASSE 9A · 2800 BREMEN 33
TELEFON 0421 / 27 27 29 · FAX 0421 / 27 07 71

CALAMUS IN
FOTO SATZ QUALITÄT

WIR BELICHTEN IHRE CALAMUS-DOKUMENTE IN 1270 DPI (635, 2540 DPI) AUF FOTOSATZFILM UND -PAPIER.

WILHELM **kuhlmann**
DTP + SATZSERVICE

Modems

BEST 1-2-3 * 300, 1200, 1200/75 Bit/s	278,-
BEST 2400 L * 300, 1200, 2400 Bit/s	288,-
BEST 2400 PLUS * 300, 1200, 1200/75, 2400 Bit/s	398,-
BEST 2400 EC MNP 5 * 300, 1200, 2400 Bit/s	498,-
BEST 2448 LF 300, 1200, 2400 Bit/s, 4800 Bit/s Send-Fax	348,-
SUPREME 9624 1200, 2400 Bit/s 9600 Bit/s Send/Receive-Fax inklusive Software ST-Fax send	678,-
GVC 9600 V.42/V.32/MNP 1200, 2400,9600 Bit/s, V.32, V.42, MNP 5	1698,-

Umrüstung BEST 2400 L auf 2448 LF mit Software ST-FAX 150,-

Auf BEST-Modems 12 Monate Garantie
* Diese Modems mit deutschem Handbuch

Anschluß der Modems am Netz der DBP Telekom ist strafbar!

Btx/Vtx mit dem *ATARI*
MultiTerm pro

Btx/Vtx-Darstellung mit Graustufen und bis zu 32/4096 Farben
Mit Automatischem Makro Generator und Programmiersprache MPL

Bildschirmtext

Wir setzen neue Maßstäbe !

An Modem V.24 158,- • An D-BT03 236,-

Schweiz: *tribatech ag* Tel: 062-260222

STFAX

Fax mit dem ATARI ST

Neu !

Endlich kann der ST faxen !
Telefax-Versand an jedes Fax-Gerät.
Empfangs-Option für Modem Supreme 9624 in Vorbereitung.
Einbinden von Grafiken in Telefaxe.
Darstellen der Telefaxe auf dem Bildschirm.
Kopf- und Fußzeilen mit Grafik.
Telefonbuch zum komfortablen Versenden.
Lauffähig auf Großbildschirm und TT !
Fax Modem zum Betrieb erforderlich.

ST FAX send Software nur 98,-
ST-FAX send und BEST 2448 LF 398,-
ST-FAX send und Supreme 9624 678,-
Schweiz: EDV-Dienstleistungen, Tel: 01/784 89 47

TKR

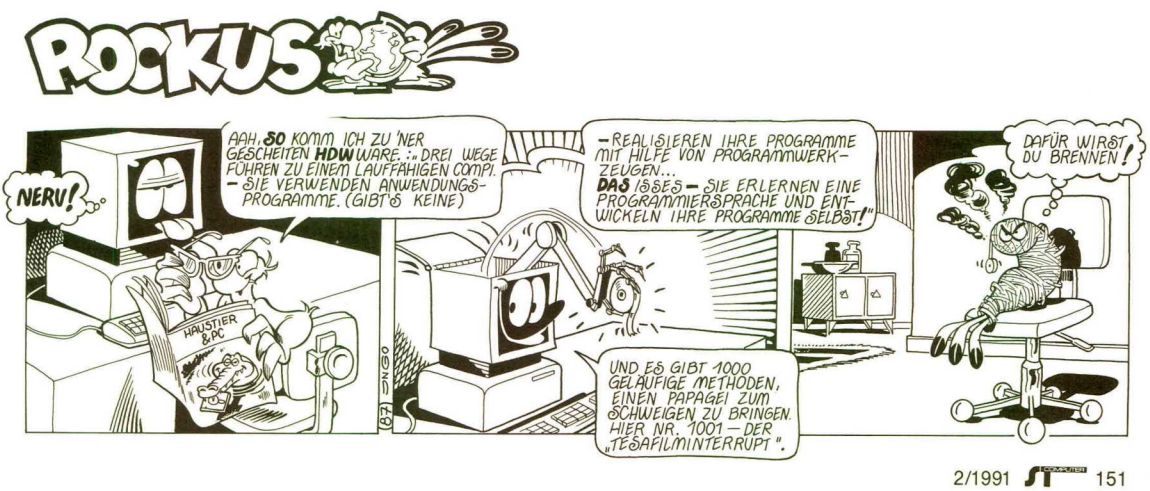
Projensdorfer Str. 14 • 2300 Kiel 1
Tel: 0431 - 33 78 81 • Fax: 0431 - 3 59 84
Btx: * TKR #

Händleranfragen erwünscht !

```
118:        Deallocate_Wl(P%)' Speicherplatz für
                               Element zurückgeben
119:      ENDIF
120:    RETURN ' Lesen_Aus_Keller
121:
122:
123:    '*********************************************
124:    '*                                           *
125:    '*  Bearbeitung der Liste 'Schlange' nach dem
            FIFO-Prinzip                             *
126:    '*  (FIFO = First In First Out)               *
127:    '*  Operationen: Einfügen am Listenende,
                         Löschen am Listenkopf       *
128:    '*                                           *
129:    '*********************************************
130:
131:    '***** Element in 'Schlange' speichern *****
132:
133:    DEF PROC Speichern_In_Schlange(Wort$)
134:      LOCAL P%' Cursor auf Listenelement
135:
136:      Allocate_Wl(P%)' Speicherplatz für neues
                           Element anfordern
137:      IF P%<>0 THEN 'kein Fehler
138:        Wl_Wort$(P%)=Wort$' Infokomponente
                               eintragen
139:        Wl_Next%(P%)=0' Einfügen am Listenende:
                            neues Element als Ende
                            markieren
140:        Wl_Next%(Schlange_Ende%)=P%
141:        Schlange_Ende%=P%
142:        IF Schlange_Anfang%=0 THEN
              Schlange_Anfang%=P%' Schlange war
                                   leer
143:      ELSE
144:        PRINT "Zu wenig Speicherplatz"
145:      ENDIF
146:    RETURN ' Speichern_In_Schlange
147:
148:    '***** Element aus 'Schlange' lesen *****
149:
150:    DEF PROC Lesen_Aus_Schlange(R Wort$)
151:      LOCAL P%' Cursor auf Listenelement
152:      Wort$=""
153:      IF Schlange_Anfang%<>0 THEN ' Schlange noch
                                       nicht leer
154:        Wort$=Wl_Wort$(Schlange_Anfang%)'
                                 Infokomponente lesen
155:        P%=Schlange_Anfang%'  Löschen a. Listenkopf
156:        Schlange_Anfang%=Wl_Next%(Schlange_Anfang%)
157:        Deallocate_Wl(P%)' Speicherplatz für
                               Element zurückgeben
158:        IF P%=Schlange_Ende% THEN
              Schlange_Ende%=0' Schlange wurde
                                geleert
159:      ENDIF
160:    RETURN ' Lesen_Aus_Schlange
161:
162:
163:    '*********************************************
164:    '*                                           *
165:    '*  Bearbeitung der geordneten linearen Liste
            'Liste'                                  *
166:    '*  Operationen: Suchen, Einfügen, Löschen   *
167:    '*                                           *
168:    '*********************************************
169:
170:    '***** Wort in 'Liste' suchen und einfügen,
              falls es nicht vorhanden ist *****
171:
172:    DEF PROC Liste_Einfuegen(Wort$)
173:      LOCAL P%,Prev%' Cursor
174:      ' Suche initialisieren
175:      Wl_Next%(0)=Liste_Anfang%' leeren Listenkopf
                                     simulieren
176:      Wl_Wort$(0)=Wort$' Wächter anhängen
177:      Prev%=0
178:      P%=Liste_Anfang%
179:      ' Suchen
180:      WHILE Wl_Wort$(P%)<Wort$' Liste durchlaufen
181:        Prev%=P%
182:        P%=Wl_Next%(P%)
183:      WEND
184:      ' P% zeigt nun auf ein Element mit
            Wort$>=Wl_Wort$(P%)
185:      ' Außerdem gilt: Wl_Next%(Prev%)=P%, auch
            wenn Prev%=P%=0 (leere Liste)
186:      IF Wl_Wort$(P%)>Wort$ OR P%=0 THEN
187:          'Wort noch nicht vorhanden, also vor (!)
              P% in die Liste einfügen
188:          Allocate_Wl(P%)' P% wird Zeiger auf das
                               neue Element
189:          Wl_Wort$(P%)=Wort$'
                    Informationskomponenten kopieren
190:          Wl_Next%(P%)=Wl_Next%(Prev%)' Nachfolger
                                            umbiegen
191:          Wl_Next%(Prev%)=P%
192:          IF Prev%=0 THEN Liste_Anfang%=P%' Es wurde
                              vor dem ersten Element
                              eingefügt
193:          IF Liste_Ende%=Prev% THEN Liste_Ende%=P%
                    ' Es wurde ans Listenende angehängt,
194:                ' denn der Wächter hat die Suche
                      beendet
195:      ENDIF
196:    RETURN 'Liste_Einfuegen
197:
198:
199:    '***** Wort in 'Liste' suchen und Eintrag
          löschen, falls er existiert *****
200:
201:    DEF PROC Liste_Loeschen(Wort$)
202:      LOCAL P%,Prev%' Cursor
203:      ' Suche initialisieren
204:      Wl_Next%(0)=Liste_Anfang%' leeren Listenkopf
                                     simulieren
205:      Wl_Wort$(0)=Wort$' Wächter anhängen
206:      Prev%=0
207:      P%=Liste_Anfang%
208:      ' Suchen
209:      WHILE Wl_Wort$(P%)<Wort$' Liste durchlaufen
210:        Prev%=P%
211:        P%=Wl_Next%(P%)
212:      WEND
213:      ' P% zeigt nun auf ein Element mit
            Wort$>=Wl_Wort$(P%)
214:      ' Außerdem gilt: Wl_Next%(Prev%)=P%, auch
            wenn Prev%=P%=0 (leere Liste)
215:      IF P%<>0 AND Wl_Wort$(P%)=Wort$ THEN
216:      ' Wort vorhanden, also löschen
            IF P%=Liste_Anfang% THEN
              Liste_Anfang%=Wl_Next%(P%)' Anfang
                                          löschen
217:          Wl_Next%(Prev%)=Wl_Next%(P%)
          ' Vorgänger von P% zeigt auf Nachfolger von P%
218:          Deallocate_Wl(P%)
219:          IF P%=Liste_Ende% THEN Liste_Ende%=Prev%
          ' Listenende wurde gelöscht
220:      ENDIF
221:    RETURN 'Liste_Loeschen
222:
223:
224:    '*********************************************
225:    '*                                           *
226:    '*  Bearbeitung des Feldes Wl zur Speicherung
            der Listen incl.                         *
227:    '*  Freispeicherverwaltung                   *
228:    '*                                           *
229:    '*********************************************
230:
231:
232:    DEF PROC Drucke_Wl
233:      ' Druckt die Listen inclusive Cursor
234:      LOCAL I%
235:      PRINT TAB (38);"Index"; TAB
            (45);"Wl_Wort$()"; TAB
            (60);"Wl_Next%()"
236:      FOR I%=0 TO Wl_Groesse%
237:        IF I%<>0 THEN
238:          IF I%=Wl_Fsp_L% THEN PRINT "Anfang
                                  Freispeicherliste";
239:          IF I%=Liste_Anfang% THEN PRINT "Anfang
                                  'Liste'  ";
240:          IF I%=Liste_Ende% THEN PRINT "Ende
                                  'Liste'  ";
241:          IF I%=Schlange_Anfang% THEN PRINT
                                  "Anfang 'Schlange'  ";
242:          IF I%=Schlange_Ende% THEN PRINT "Ende
                                  'Schlange'  ";
243:          IF I%=Keller_Anfang% THEN PRINT "Anfang
                                  'Keller'";
244:        ENDIF
245:        PRINT TAB (40);I%; TAB (45);Wl_Wort$(I%);
              TAB (60);Wl_Next%(I%)
```

→

```
246:      NEXT I%
247:      PRINT
248:    RETURN
249:
250:
251:    '***** (Re-)dimensionieren des Feldes mit der
         Datenstruktur Wortliste *****
252:
253:    DEF PROC Dimensioniere_Wl(Anzahl_Neue%,
                                  R Fehler%)
254:      ' Anzahl_Neue% gibt die Anzahl zusätzlicher
          (!) Einträge an.
255:      ' Rückgabewert : 0 = kein Fehler, -1 = Fehler
                           aufgetreten
256:      LOCAL I%,Fehler%
257:      Fehler%=0
258:      ' Felder dimensionieren.
259:      ON ERROR GOTO Dim_Error
260:      DIM Wl_Wort$(Wl_Groesse%+Anzahl_Neue%)
          ' Feld mit der Komponente 'Wort'
261:      DIM Wl_Next%(Wl_Groesse%+Anzahl_Neue%)
          ' Feld mit dem Cursor auf den Nachfolger
262:      ON ERROR GOTO 0
263:      IF NOT Fehler% THEN ' kein Fehler beim
                              Dimensionieren aufgetreten
264:         'Freispeicherliste generieren, d.h. die
           neuen Einträge als frei markieren
265:         Wl_Fsp_L%=Wl_Groesse%+1' Neuer Anfang der
                                      Freispeicherliste
266:         Wl_Groesse%=Wl_Groesse%+Anzahl_Neue%
267:         FOR I%=Wl_Fsp_L% TO Wl_Groesse%-1
268:           Wl_Next%(I%)=I%+1
269:         NEXT I%
270:         Wl_Next%(Wl_Groesse%)=0
             ' Ende der Freispeicherliste markieren
271:      ENDIF
272:    RETURN
273:  -Dim_Error
274:      Fehler%=-1
275:      RESUME NEXT
276:
277:
278:    '***** Einen Cursor auf einen freien Eintrag im
         Feld wl liefern *****
279:
280:    DEF PROC Allocate_Wl(R P%)
281:      ' P enthält nach dem Aufruf den Cursor auf
          ein freies Element
282:      ' P=0, falls kein Speicherplatz mehr
283:      LOCAL Fehler%=0
284:      IF Wl_Fsp_L%=0 THEN ' keine freien Einträge
                              mehr vorhanden
285:         Dimensioniere_Wl(10,Fehler%)' Feld um 10
                                Einträge vergrößern
286:      ENDIF
287:      IF Fehler% THEN
288:         P%=0' kein Speicherplatz mehr vorhanden
289:      ELSE
290:         ' Das erste freie Element entfernen wir
              mit der Operation
291:         ' Löschen am Listenkopf aus der
              Freispeicherliste.
292:         P%=Wl_Fsp_L%' P% zeigt auf das erste freie
                           Element
293:         Wl_Fsp_L%=Wl_Next%(Wl_Fsp_L%)
             ' Das Kopfelement der Freispeicherliste
294:         ' zeigt auf seinen Nachfolger
295:      ENDIF
296:    RETURN ' Allocate_Wl
297:
298:
299:    '***** Ein Feldelement als frei markieren *****
300:
301:    DEF PROC Deallocate_Wl(P%)
302:      ' P ist der Cursor auf das freizugebende
          Element
303:      ' (!) Die Felder werden nicht durch
          Redimensionierung verkleinert
304:      ' Ein Feldelement darf nur als frei markiert
          werden, wenn es nicht schon
305:      ' frei ist => P% muß in der Freispeicherliste
          gesucht werden. Nur wenn
306:      ' P% nicht vorhanden ist, darf dieser Cursor
          aufgenommen werden.
307:
308:      IF P%<>0 THEN
309:         ' Suche nach P% im Feld Wl_Next%()
310:         LOCAL Hilf%' Hilfscursor, der die
                          Freispeicherliste durchläuft
311:         IF P%=Wl_Fsp_L% THEN 'P% ist der Anfang
                                   der Freispeicherliste
312:            Hilf%=1
313:         ELSE
314:            Hilf%=Wl_Fsp_L%' Suche am Listenanfang
                                beginnen
315:            Wl_Next%(0)=P%' Wächter ans Ende der
                               Freispeicherliste hängen
316:            WHILE Wl_Next%(Hilf%)<>P%' Suchfeld:
                 Wl_Next%()
317:              Hilf%=Wl_Next%(Hilf%)
318:            WEND
319:         ENDIF
320:         IF Hilf%=0 THEN 'P% ist noch nicht als
                             frei markiert
321:           ' P% wird durch Einfügen am Listenkopf
                in die Freispeicherliste eingetragen
322:           Wl_Next%(P%)=Wl_Fsp_L%
               ' der Nachfolger von P% zeigt auf den
                 Kopf der Freispeicherliste
323:           Wl_Fsp_L%=P%
               ' der Kopf der Freispeicherliste wird
                 nun umgebogen
324:         ENDIF
325:      ENDIF
326:    RETURN ' Deallocate_Wl
```



Hausverwaltung

Erprobte Branchenlösung für:
Gewerbe-, Miet-, Mischobjekte
gesetzl. Nebenkostenabrechnung
Netzwerkversion verfügbar!

kleine Version	398,00 DM
bis 40.000 ME	1698,00 DM
+ Nebenkostenabrechnung	298,00 DM

IDEE Individuelle Computer-Lösungen GmbH
Waidmannstraße 12; 2000 Hamburg 50
Tel.: 040/85 50 66; Fax: 040/ 850 18 58

ATARI ST

Anwendersoftware		Spieleso ftware	
CCD		Star Trash	50,-
ST Pascal + V2.0x	220,-	Gunship	79,-
Tempus Editor 2.1x	100,-	Indiana Jones (Adv.)	69.-
Tempus Word	a.A.	Oil Imperium	53,-
Assembler Tutorial	90,-	Populous	69,-
GFA		Rick Dangerous	69,-
GFA Basic 3.5 (I.+C.)	240,-	RVF Honda	69,-
GFA Assembler	140,-	Sleeping Gods Lie	69,-
Omikron		Spherical	53,-
Omikron Basic Comp.	170,-	Zak McKraken	69,-
Mortimer, Utility	75,-		
Appl. Syst.		Diskbox 3,5" 80er	19,-
Signum!2	418,-	Supercharger 1MB	720,-
Scarabus	90,-	Coprozessor f.Sc.	285.-
Signum Revers Acc.	90,-	Traktrix	80,-
Protos	64,-	Approximationsprg.	
Fontdisketten verfügbar		für sämtl. Fkttypen	

Porto: Vorkasse 4,- Nachnahme 7,- DM

Computerversand G. Thobe
Pf. 1303 - W-4570 Quakenbrück
Tel.: (05431) 5251

Roman Modern

Komplette und umfangreiche Schriftfamilie für **Signum** oder **Script** in fünf harmonisch aufeinander abgestimmten Schriftschnitten für 24-Nadel- oder Laser-Drucker:

Roman Modern Regular
Roman Modern Bold
Roman Modern Italic
Roman Modern Bold Italic
ROMAN MODERN CAPS

Jeder Schnitt liegt in sieben Größen vor (6, 8, 10, 11, 12, 14, 16pt), verfügt über Ligaturen, Sonderzeichen und einen Grundbestand an akzentuierten Buchstaben — je Schnitt und Größe mehr als 170 Zeichen.

Als Vorlage diente die TEX-Schriftfamilie CMR, so daß nun auch Signum-Anwender eine ähnlich einzigartige typografische Ausgabequalität erzielen können.

Roman Modern für **Signum**	130,- DM
Roman Modern für **Script**	100,- DM

(speziell angepaßt, ohne 16pt Fonts!)

Gegen Verrechnungsscheck oder per Nachnahme, zzgl. 5,- DM Versandkosten bei *

Detaillierte Informationen und Schriftproben gegen 2,- DM Rück-Porto (in Briefmarken) bei *

* H.Schlicht, Ketzendorfer Weg 4H,
2104 Hamburg 92, Tel.: 040 / 7 01 64 92

SCSI-Festplatten

180 MB 3 1/2" Fujitsu Laufw. superleise
64KB Cache, 19 ms Sonderpr. 2298,-DM

80 MB 3 1/2" Quantum Laufw. superleise
64KB Cache, 19 ms Sonderpr. 1448,-DM

44 MB Wechselplatte orig. SyQuest Laufw.
28ms, incl. Medium Sonderpr. 1598,-DM

Alle Platten kompl. anschlußf. im modernen Slimline-Gehäuse (orig. ATARI-Gehäuse 88,-DM Aufpreis). Vorber. für zweite Platte. DMA-Adresse von außen einstellbar. SCSI-Adapter 100% AHDI kompatibel. Autopark. 2 Jhr. Garantie. usw.

Speicherer weiterungen und Tower-Umbauten auf Anfrage.

EDV PARTNER HORN
Leipzigerstr. 34 6301 Pohlheim 1
Telefon: 06403/67680

SCANNER

für Atari ST an den Druckern: NEC P2200,P6,P7, EPSON FX80, FX85, RX80, STAR NL10, LC10 und am STAR LC 24-10.

Scannen Sie mit festen Sitz des Scankopfes.

- RS 232-Anschluß. Der empfindlichere Modulport bleibt frei. Es sind **keine** Lötarbeiten erforderlich.
- Das **bidirektionale (!)** Scannen bei den Epson Druckern und beim Star LC10 halbiert Ihre Scanzeiten.
- Assembler-Scanroutinen garantieren Präzision.
- Einstellbar: Scankontrast, Scanparameter, Zoomfaktor.
- Grafikformate (monochrom): Screen/Doodle, Degas und .IMG Format für den Datelexport.

SCANNER (anschlußfertig) DM 298,- per NN.

Dipl.-Ing. Gerhard Porada, Dürrlewangstr. 27
7000 Stuttgart 80, Tel.: 0711/ 74 47 75

Professionelle Schön-Schrift

mit **Signum** und **Laser-/24-Nadeldrucker** (fast alle Fonts auch für **script** geeignet)

- Modernes, optimal lesbares Schriftbild
- in den Größen 8, 10, 12 und 15 Punkt,
- neu: 6 (normal) sowie 20 Punkt (fett),
- einschließlich unproportionaler Ziffern
- normal und fett, und dazu zwei schlau
- ausgetüftelte „SONDER"-Zeichensätze

mit griechischem Alphabet (A, α, B, β, Γ, γ, Δ, δ, ..., Ω, ω), römischen Zahlen (I, II, III, IV, ... X, ... MCMLXXXXI), fremdsprachigen (Å, å, Ç, ç, Ø, ø, Œ, œ, ñ, ¿), kaufmännischen (£, ‰, ®, ©, ℗) und mathem. Zeichen (∞, ∫, ≡, ≠, ≢, <, ≦, ≧, >, ~, ≈, ≡, ±, ×, ÷, ∢ usw.) und vielem mehr (♂, ♀, @, ©, ①, ... ⑨, ⑩, ... ⑨⑨, ½, ⅞, →, ←, ↑, ↓) für die unterschiedlichsten Einsatzbereiche.

Ausführliche Info mit Schriftprobe
■ für 3,- DM in Briefmarken anzufordern bei:
Walter Schön, Berg-am-Laim-Str. 133 A,
8000 München 80, Tel. (0 89) 4 36 22 31.

Neu für Tabellenköpfe: senkrecht laufende 10-Punkt-Schrift

PD SONDERANGEBOT PD

Alle ST-Computer, Erotik, Grafik-Sound-Demos, PD-Abo, eigene Serien . . . . .

10 Disketten	35 DM
20 Disketten	59 DM
50 Disketten	125 DM

Katalogdiskette gratis
großes Soft- und Hardwareangebot, viele PD's

SW-HAUSHALT Haushaltsbuchführung, sehr einfache Bedienung, GEM, S/W ..... 69 DM
SW-VIDEO Videofilmverwaltung, komfortables Handling, GEM, S/W ..... 59 DM
Info kostenlos, Demo je 10 DM incl. Versand

Versandkosten bei VK 4 DM, bei NN 6 DM
Lieferung innerhalb 48 Stunden

SW-SOFTWARE
Soft- und Hardwarevertrieb
Beethovenstraße 10, 7938 Oberdischingen
Tel. 07305/8325

K&L Datentechnik
Bahnhofstraße 11
3551 Bad Endbach
Tel. (02776) 8145
graffiti

Graffiti-Shell/Paint: (349.-)
Neu
.Über 400 Funktionen, mehrere Seiten.
.Pixelbearbeitung in 3 verschiedene Zoommodi.
Näheres siehe ST-Magazin 5/90 und 8/90.

Lavadraw Plus: (99.-)
Das Zeichenprogramm für den kleinen Geldbeutel.

Graffiti-Konvert: (89.-)
Neu
.Druckerspooler
.Snap:ACC
.Ramdisk auch resetfest
.Konvert (nur für Laser optimal einsetzbar).
Ermöglicht das Laden von ganzen Seiten aus vielen bekannten Textverarbeitungs-, DTP-, CAD-, und Zeichenprogrammen.

Alle Lavadraw-Besitzer, die das Programm **vor** dem 01.10.90 gekauft haben, können Graffiti zu einem Preis von 150,-DM direkt von K&L erhalten. Dazu müssen Sie das Komplette Lavadraw-Paket an uns Senden.

Unverbindlich empfohlene Verkaufspreise
Infos erhalten Sie nur gegen DM 3,- in Briefmarken.

Telefonhotline Montags von 16:00 bis 19:00 Uhr

Public Domain Versand
Anton Peter Maassen
Am Lindenplatz 17 *W-4040 Neuss 1*

Jede Atari ST Diskette kostet
nur 4.-DM und das auf
einer Fuji Film Diskette

Alle bekannten Serien sind
Lieferbar. Auch der neue Hit
die Cobra Serie

Gratis Liste anfordern !!!!

Bismarckstr.84 | 1000 Berlin-12

midisystems Geerdes

Tel: 030 - 31 67 79 Fax: 030 - 3 12 18 26

MIDIBOX ™ Multitimbral Expander: 8 Instrumente + drums

16-stimmig, 99 Sounds, D/A: 16bit, S/N: 80dB !

Das komplette Profi-Orchester: **499.-** DM

Band In A Box

Generiert nach Akkord-Eingabe [C-F-G7-Am...]: Piano + Bass + Drums und spielt in 24 Stilarten von Rock bis Reggae (+Midifiles) **198.-** DM

1stTRACK Professional MIDI SEQUENCER

Midifiles, Recording, Playback, Profi Tools für Realtime, Schnitt & Nachbearbeitung **158.-** DM

MIDIMUSIC Collection

700 Profi-Titel von Rock bis Klassik je: **35.-** DM

Atari-ST Speichererweiterungen

(inklusive Einbau und Versand)

260-ST 520-ST	auf 1,0 MByte auf 2,5 MByte auf 4,0 MByte	180,- DM 420,- DM 800,- DM
520-ST[M]	auf 2,5 MByte auf 4,0 MByte	470,- DM 800,- DM
1040-ST[F] MEGA-ST	auf 3,0 MByte auf 4,0 MByte	470,- DM 800,- DM
MEGA-ST	auf 4,0 MByte	400,- DM

hyperCACHE+ inklusive Einbau	575,- DM
AT Speed inklusive Einbau	575,- DM
Autoswitch-OverScan inkl. Einbau	160,- DM
TOS 1.4 („Rainbow-TOS")	195,- DM
Megabit-Chip Siemens HYB511000	10,- DM
SIM 1M*8 (für ST[E])	125,- DM

Christian Rupp
Am Kronwerk 9 W-6740 Landau
☎ 06341/84993

ATARI ST Astrol. Kosmogramm

Auf Namen, Geb.Zeit+Ort (Koordinaten) werden errechnet: Sternzeit, Aszendent, MC, 11 Objekt-Positionen, Radianten, Aspekte im Tierkreis (Planeten,Sonne,Mond Mondknoten), Koch/Schaeck-Häuser - Minutengenau mit Sommerzeiten u. Einlesung vieler Ortskoordinaten * Allgem. Persönlichk.Analyse m.Ideal-Partner-Skala,Horoskop-Diagramm - Schirm-/Drucker 3DIN4 S. DM 75.-

BIOKURVEN ATARI ST

Wissensch.Trendbestimmung d.Körper-Seele-Geist-Rhythmik, auf Schirm monatlich vor+zurück, Drucker beliebig lang m.Tagesanalyse und krit.Zeiten DM 56.-

ATARI ST Kalorien-Polizei

Auf pers.Daten erfolgen Bedarfsrechnung Vergleich m. eingegebenem Verzehr in Eiweiß+Fett+Kohlenhydraten - Ideal-/Uber-/Untergew.Best. - Vitalstoffe+Gehalte - Tätigk.+Verbrauch - Aufst.v.Diätplänen DM 56.-

Casino-ROULETT ATARI ST

Mit Schnellsimulation, Chancentest, Häufigkeitsanalyse, Kassenführung.Setzen m.Maus a.Tischgrafik 68.-

ATARI ST VEREIN

System von 7 PRG: Grunddaten-Editor, Mitgliederdatei m.Beitragsübers., Listen, Etiketten, Rundschrb.-Eindruck - Mahnung - Lieferanten-Bestellung -Freunde-u. Turniergegner- +Termine-Datei *Möglichk.wie vor- Inventar/-tür - Kasse m.Belegdruck + Protokoll auf Disk und Drucker - Einnahme-/Ausgabe-Bilanz DM 196.-

Globaler Sternenhimmel ATARI ST

Zeigt den aktuellen Sternenhimmel für Zeit+Ort nach Eingabe - Klick auf Stern gibt Namen+Daten aus - Planeten, Sterne, Sternbilder blinkend/verbunden - Teleskop zeigt vergrößerte Himmelsausschnitte - Wandern simuliert geogr./zeitliche Schnellbewegung DM 89.-

Programme für alle ST Modelle - Exzellent in Struktur, Grafik, Sound

Alle in Deutsch, S/W und Farbe

ATARI® Midi Center **MCS** Epson, Technics, Korg Vertrags Händler

MEGA ST, SM124, MEGAFILE 30.
COMPLET SET AB DM 1998,-
DAZU WORD PERFECT 4.1

**NUR SO LANG DER VORRAT REICHT

TT PREISSENKUNG

		Midi	
ST MEGA 1, Maus, TOS 1.4, SM 124	DM 1248,-		
ST MEGA 2, Maus, TOS 1.4	DM 1598,-		
ST MEGA 2 mit 4 MB	DM 1998,-		
Megafile 30	DM 698,-		
SM 124	Ab. DM 225,-		
1040STFM, TOS 1.4	DM 998,-		
SPEICHERERWEITERUNG 260, 520, 1040, Mega 1 u. 2		Kecy board	DM
2 MB Erweiterung (2,5)	DM 398,-		
4 MB Erweiterung	DM 698,-		
2/4 Karte mit 2 MB	DM 435,-		
2 MB Leerkarte	DM 98,-		
4 MB Leerkarte	DM 179,-		
STE auf 2 MB	DM 260,-		
STE auf 4 MB	DM 520,-		

Midi u. Computer Systeme Tel. 0231/759468-Fax759467
Baroper Bahnhof 9 4600 Dortmund 50
Mo.-Fr. 11.00 - 13.00 /15.00 - 18.30 Sa. 11.00 - 14.00

ATARI ST PD SERVICE

5,25" ATARI Laufwerk ..... 229,- DM
3,5" ATARI Laufwerk ..... 189,- DM

Wir führen alle PD Serien im Atari ST Bereich

NEU eingetroffen Atari PD aus den USA

Ca. 1000 PD auf Lager, pro Disk ..... 1,80 DM
Disketten von Euch ..... 0,50 DM

3.5" Diskbox für 88 Disketten ..... 12,95 DM
5,25" Diskbox für 100 Disketten ..... 12,95 DM
3,5" Disketten NoName 2DD, 10 St. ..... 11,95 DM
5,25" Disketten NoName 2D, 10 St. ..... 5,40 DM

CTN
EDV Anlagen GbR
Westwall 4
4270 Dorsten
Tel.: 02362-42991 + 42925
Fax: 02362-42263
BTX: 02362-64510

ATARI ST Registrierkasse

ST+Drucker - Beleg Schmal-o.Normaldruck,auch für Beleg-Drucker - Protokoll auf Disk, ausdruckbar - Leistungen/Artikel von Disk o.Hand - Firmendaten - Werbeslogan - Kassenstand - Kassiermarke DM 146.-

GESCHÄFT ATARI ST

Editor f. Formular-, Adressen-, Artikel- +Dienstleistungsdateien - Angebot/Vorschlag, Auftr.Bestätigung, Auftrag/Bestellung, Rechnung, Lieferschr., Mahnung - Eingabe Hand o.Datei - Durchrechnung u. Menge Preis, Aufschlag/Rabatt, MwSteuer, Skonto usw. - Verpackung-/Versand-Angaben - Editor für Textfeld - Kein Datenverbund mit Lager-/Finanz-Buchhaltung DM 196.-

ATARI ST Inventur, Fibu-gerecht

Kontinuierl. Lager-Bestandsverwaltung m.Bild-Moment-u./o.Listenauswertung - Tages- bis Jahres-Neuinventur d. Streichen/Andern/Hinzufügen - Gruppenauszuge nach Code - Bis 3000 Positionen/Datei DM 116.-

Provisionsabrechnung ATARI ST

Editor f.Vertreter-, Kunden- u.Firmen-Dateien - Eingabe von Hand/Datei - Prov.Satz -99,99% -Stornos/Spesen - Endbetrag m.o.MWSteuer - Ausdruck DM 116.-

ATARI ST TYPIST

Der **ST**+Drucker als Elektronik-Schreibmaschine - Ausdruck zeilenweise - 15 Zeilen Bildschirm-Display - Korrektur - Je nach Drucker bis zu 30 Schriften - Ablage auf Disk - Kopie-Ausdruck - Super ! DM 86.-

Etikettendruck ATARI ST

Druckt Auflagen von 40 gängigen Lochrand-Haftetiketten-Formaten - Texteingabe in jeweils passende Bildschirmmaske - Ablage auf Disk für jederzeitige Neuauflage - Schriftenwahl n.Drucker-Handbuch DM 89.-

ATARI ST BACKGAMMON

Das Strategie+Glück-Spiel - Bestechende Grafik - In Schwarz/Weiß und Farbe - Ausf. Anleitung DM 58.-

Textverarbeitung/DTP

That's Write Profi	259,-
That's Write Postscript	459,-
That's Adress	159,-
Publishing Partner Master	689,-
Reading Partner	899,-
Script	169,-
Script II	289,-
Signum!2	369,-
Tempus Word	489,-

Programmiersprachen

ST Pascal Plus	219,-
Hisoft FTL-Modula-2	269,-
SPC Modula 2	319,-
Megamax Modula	349,-
Laser C inkl. Debugger	329,-
GFA-Basic EWS 3.5	229,-
GFA-Basic EWS 3.0	179,-
GFA-Basic Konv. nach C	449,-
Turbo C 2.0 Pro	369,-
Maxon Pascal	249,-

Die schnellste Textsuche auf dem Atari ST

TAS - TEXTSEARCH bis zu 7 MB pro Minute *39.-*

Emulatoren und Beschleuniger

	für	ST	STE	MEGA
AT-Speed	80286, 8 Mhz	429,-	489,-	489,-
hyperCACHE ST-	68000, 16 Mhz	499,-	☎	559,-
Maxon Mach 16	68000, 16 Mhz	649,-	☎	709,-
Makro Turbo 16	68000, 16 Mhz	589,-	649,-	649,-
Maxon Board 20	68020, 16 Mhz			1859,-
hyperCACHE 030	68030, 25 Mhz, angepasstes TOS			2399,-
Turbo 030/24	68030, 24 Mhz, angepasstes TOS, FPU 68882			2459,-
Turbo 030/32	68030, 32 Mhz, angepasstes TOS, FPU 68882			3759,-
Turbo 030/50	68030, 50 Mhz, angepasstes TOS, FPU 68882			4459,-

Superschnelle Festplatten

ICD-Hostadapter mit Echtzeituhr, 3 1/2 Zoll Laufwerke von Quantum

AHS-52L	52 MB, 950 KB/s, 17ms	1128,-
AHS-105L	105 MB, 1050 KB/s, 17ms	1628,-
AHS-2000	210 MB, 1100 KB/s, 10ms	2548,-

SCSI-Einbaufestplatten für TT, sehr leise, Hardware-Cache

Conner 105 MB, 3Watt	1298,-	Quantum 105 MB, 6W	1198,-
Conner 210 MB, 3Watt	2129,-	Quantum 210 MB, 13W	2179,-
NEC P60 A4, 24N.	1389,-	NEC P70 A3, 24N	1699,-

Preise in DM inkl. Mwst. Versand per NN zzgl Versandkosten

Torsten Anders Software
Hohenstaufenallee 35
5100 Aachen
Tel.: 0241 - 74246

WOBO-Service
Willi B. Werk

MegaPlot Der Werteplotter	189,--
Signum! Zwei	a.A.
STAD 1.3+	151,--
Script, Script II	a.A.
ScripTrenn	50,--
That's Write Profi	269,--
TEMPUS-WORD	489,--
WORDFLAIR	196,--
CALAMUS	691,--
Publ. Partner Mast.	666,--
GFA-BASIC 3.5	216,--
GFA-ASSEMBLER	119,--
ST Pascal Plus	199,--
LDW Power Calc	206,--
Btx/Vtx-Man. o.I.	245,--
Anti Viren Kit 3.x	79,--
Quick ST II	56,--
XBoot	63,--
OXYD das Buch	45,--
* Spiele zu Superpreisen *	

Adimens ST plus 3.1	309,--
1ST-BASE	169,--
EasyBase	a.A.
I.D.A.	329,--
TiM I	119,--
1ST fibuMAN	126,--
fibuMAN e	309,--
TEMPUS V.2.xx	99,--
Diskus	112,--
Harlekin	112,--
Mortimer/Mort.plus	a.A.
NeoDesk 3	83,--
STar Designer	136,--
Hard&Soft Herb. Prod.	a.A.
Speichererweit. 4MB	579,--
That's a Mouse *neu*	83,--
ReproSt.j.+Scanman	509,--
Emulatoren: nur neueste Versionen!	
vortex ATonce	435,--
AT-Speed	429,--
PC-Speed	259,--
SpeedBridge	68,--
Supercharger V1.4	698,--

Alle Preise in DM, inkl. der gesetzl. MWSt. Vorbehaltlich Irrtümer und Preisänderungen. Bei Vorkasse 2% Skonto, zuzügl. DM 5,50 Versandkostenanteil; bei Nachnahme kein Skonto, zuzügl. DM 9,50 Versandkostenanteil. Kein Ladenverkauf! Dies ist nur ein kleiner Ausschnitt aus unserer Preisliste, die wir Ihnen gerne kostenlos übersenden.

Sielwall 87, D-2800 Bremen 1 - Tel. 0421/75116

ATARI ST GELD

30 Routinen für Umgang mit Geld: Anlage - Vermögensbildung - Rentensparen - Rendite - Kredite - Lasten - Zinsen - Hypothek - Laufzeit - Amortisation - Raten - Gleitklausel - Nominal/Effektiv Zins - Akonto+Restverzinsung - Diskont - Konvertierung - kpl.Tilgungspläne Bild/Druck DM 96.-

DATEIVERWALTUNG ATARI ST

Datenfelder von je 8 Zeilen a 33 Zeichen, je Datei max.3000 - Suchcode von max.33 Zeichen, mit jedem mehr die Zielgruppe eingengend - Optionen : Code, Nummer, alle, Blatt vor/zurück, Streichen, Ändern(zeilenweise), Hinzufügen - Druck: 80-Zeichen-/Blockliste, Seitenvorschub, Etiketten, Datenfeld-Maske - Gezielte Aufgaben, superschnell- Übersichtlich, bedienerfreundlich, mausgesteuert

Adressen	66.-	**Galerie**	116.-
Bibliothek	116.-	**Lager**	116.-
Briefmarken	116.-	**Personal**	116.-
Diskothek	76.-	**Stammbaum**	116.-
Exponate	116.-	**Videothek**	76.-

DEFIN DATA ZUM SELBSTDEFINIEREN DER ERFASSTEN DATEI-DATEN DM 146.-

Versandkosten pro Sendung : Nachnahme DM 5.70, Ausland DM 10.70 Vorkasse DM 3.-
Liste gegen adressierten Freiumschlag DIN A5/DM1.-
Händler sehr erwünscht.

IDEE-SOFT

I. DINKLER
Am Schneiderhaus 7
Tel. 02932/32947 FAX 3 26 54 D-5760 ARNSBERG 1

Der SteuerStar '90

Lohn- u. Einkommensteuer 90

50 ,- DM/Update 30 DM

für alle ATARI-ST sw/col

Test: ST-Magazin 2/89 :

"Der Steuerstar... nimmt ohne Zweifel einen sicheren Platz in der Reihe der Spitzensoftware für den ST ein."

Dipl. Finanzwirt J. Höfer

Grunewald 2a

5272 Wipperfürth

Tel. o2192/3368

SOFTHANSA

zuverlässig - preiswert - schnell (24-Stunden-Service)

HARDWARE

AT-Speed	420,–	RTS-Tastaturkappen ab	120,–
PC-Bridge	50,–	2 MB-Speichererweit.	355,–
AT-Speed Buch	45,–	4 MB-Speichererweit.	590,–
SPECTRE GCR	550,–	Handyscanner 400 dpi	450,–
HD-Interface	55,–	GAL-o. Junior-Prommer	195,–

AutoSwitch OVERSCAN 100,–

SOFTWARE

Tempus Word ... a.A.	APiSoft Tools	1st Card	255,–
Signum2	(SDO merge...) ☎	SciGraph 2	500,–
incl. PD	345,– I.D.A.	320,– ST Pascal+	200,–
That's Address	155,– BASiCHART	175,– Tempus	
That's Pixel	125,– Turbo C Pro	350,– Editor	100,–
Turbo C 2.0	200,– MathLib	110,– ST Statistik	300,–
Maxon Pascal	220,– Diskus	110,– 1st fibuMAN	120,–
Neodesk III	80,– Harlekin	115,– fibuMAN e	305,–
That's Write	270,– Script II	245,– fibuMAN f	610,–

8000 München 90 * Untersbergstraße 22

☎ 24 Std. Bestellannahme 089 / 6 97 22 06 ☎

Weitere Artikel auf Anfrage. Preisliste anfordern ! Vorführung von Programmen nach Vereinbarung. Preisänderungen und Irrtum vorbehalten. Lieferung per Nachnahme od. Vorkasse.

HARDWARE

Supercharger 1 MB		695,–
VORTEX ATonce	-VGA-	425,–
AT Speed	**-VGA-**	**425,–**
NEC P20		795,–
NEC P60		**1375,–**
HP Deskjet 500		**1375,–**
Star XB 24-10		995,–
ICD ADSpeed 16 MHz		595,–
MAXON Mach 16		645,–
14" EIZO Monitor 9060 SZ		1595,–
21" EIZO Monitor 6500		**2995,–**
20" EIZO Monitor 9400i		4995,–
Matrix Monitor + Karte		a.A.
Scanner Epson GT 6000		**3995,–**
Portfolio		395,–
Mega ST 1		995,–

Computersysteme

WAVE 6300 Gießen Südanlage 20 Tel 0641 / 72357 Fax 72371

NEU **Lex-o-Thek**

Das Modul-Lexikon für den ATARI ST

'Lex-o-Thek': das Grundprogramm DM 49.-

Ein notwendiges Accessory, mit dem mehrere Module bedient werden können. Voll mit der Maus oder der Tastatur bedienbar.

Modul 1: '3rd Word': das Synonymenlexikon DM 89.-

Nach Eingabe eines Suchbegriffs in das '3rd Word'-Fenster werden Blöcke sinnverwandter Begriffe angezeigt. Nach jedem angezeigten Begriff kann weiter verzweigt werden.

Modul 2: 'Herz-Schmerz', das Reimlexikon DM 69.-

'Herz-Schmerz' ist nicht nur für Dichter und Denker interessant. Sie bekommen zu jedem eingegebenen Wort hunderte Reimwörter angezeigt. So kann sich auch der Hobbydichter zu jedem Anlaß einen passenden Reim machen.

Modul 3: 'Bonmot', die Zitatedatenbank DM 69.-

'Bonmot' enthält eine Vielzahl klassischer und moderner Zitate, Sprichwörter, Bonmots, Spontansprüche, Definitionen, Bibelsprüche und Bauernregeln. Die Zitate werden durch ausführliche Sachregister, Stichwortregister und Autorenregister erschlossen.

Komplettpaket-Preis (Ersparnis: 27,- DM) DM 249.-

Preise zuzügl. Versandkosten: Vorkasse 5.- DM, NN 8.- DM

RR-SOFT Reinhard Rückemann Grundstrasse 63 5600 Wuppertal 22 02 02 / 64 03 89

Festplatten

PROTAR profile

20 MB	795,–	30 MB	945,–
40 MB	1195,–	40 MB DC	1295,–
60 MB	1495,–	**80 MB**	**1595,–**
80 MB DC	2195,–	160 MB DC	2795,–
44 MB **Wechselplatte**		incl.Medium	**1645,–**
60 MB und 150 MB **Streamer**			a.A.

VORTEX Datajet

30 MB	1095,–	**48 MB !!!**	**1195,–**
60 MB	1595,–	90–180 MB	a.A.

Fest- Wechselplatten Kombinationen a.A.

Hard & Soft, FSE Festplatten **a.A.**

Auf Wunsch: Festplatten m. PD Software (MAXON PD 140 – 380) 1 MB nur 2,–

Cartridge 44MB (f. Wechselplatten) **215,–**

Computersysteme

WAVE 6300 Gießen Südanlage 20 Tel 0641 / 72357 Fax 72371

Friedliche Aufrüstung

TUNE UP 16: **16 MHz**

schnellerer Bildaufbau, höhere Rechenleistung, doppelt schneller Zugriff aufs Betriebssystem

399,– DM INCL. EINBAU

TUNE UP für alle ATARI® ST / MEGA® ST

VARIO-RAM Speichererweiterungen

2,5 MB voll steckbar, nachrüstbar auf 4 MB 498,– DM

4 MB voll steckbar, ohne Löten einzubauen 698,– DM

AT-SPEED: MSDOS-Emulator 498,– DM

Einbaukosten: VARIO-RAM 75,– DM AT-SPEED 50,– DM

4 MB + 16 MHz **999,–** DM

Wozu brauchen Sie noch den TT?? INCL. EINBAU

Vorkasse + 5 DM / Nachnahme + 7 DM Versandkosten

ST PRAXIS *Rückemann Soft & Tronic*

Grundstrasse 63, 5600 Wuppertal 22

TEL: 02 02 / 64 03 89 FAX: 64 65 63

SOFTWARE

That's Write 1.5	265,–	Adimens 3.1+	295,–
Tempus Word	485,–	Adimens 2.3	125,–
Signum2!	a.A.	Aditalk 3.0 +	295,–
Wordflair	225,–	Themadat 4.0	215,–
Cypress	a.A.	Calamus	665,–
Script 2	a.A.	Calamus S, SL	a.A.
Convector	225,–	Outline Art	335,–
STAD 1.3+	a.A.	Didot	185,–
Megapaint II prof.	695,–	ImS Cranach	555,–
Arabesque	215,–	Cadja	895,–
Arabesque Pro	325,–	Technob. Drafter	745,–
MAXON Pascal	245,–	fibuMAN f	695,–
3K Software	a.A.	C.A.S.H. Softw.	a.A.

Turbo C 2.0 Pro 365,–

Word Perfect 195,–

Adimens 2.3 125,–

Computersysteme

WAVE 6300 Gießen Südanlage 20 Tel 0641 / 72357 Fax 72371

CSR-MODEMS CSR-MODEMS CSR-MODEMS

MODEMS

der neuen Generation

CSR MODEM

- V.24-Kabel
- TAE F-Stecker
- Handbuch deutsch
- Drews BTX-Decoder
- 18 Monate Garantie



	Postzulassung Mit ZZF	Ohne*
CSR 2400 300, 1200, 2400 bps	599,–	298,–
CSR 2400 PLUS 300, 1200, 2400 bps + BTX-Norm 1200/75	649,–	348,–
CSR 2400 MNP 5 300, 1200, 2400 bps Datenkomprimierung u.Fehlerkorrektur	749,–	448,–
CSR 2400 MNP 5 PLUS 300, 1200, 2400 bps + BTX-Norm 1200/75 Datenkomprimierung u.Fehlerkorrektur	799,–	498,–
CSR 9600 MNP 5 PLUS 300 – 9600, 1200/75 19200 eff. bps	a.A.	1398,–
CSR 2400 FAX + FAX mit 9600 bps	a.A.	649,–

* Anschluß ohne ZZF ist strafbar!

CSR

Breslauer Str. 19

D-3575 Kirchhain

TEL: (0 64 22) 34 38

FAX: (0 64 22) 75 22

BTX: * C S R #

CSR-MODEMS CSR-MODEMS CSR-MODEMS

Computer & Electronic & Zubehör HERGES

Obere Rischbachstraße 88 • 6670 St. Ingbert

Telefon (06894) 383178 / Telefax (06894) 382855

Atari-Computer + Zubehör:

Mega-1, kpl. (o. Monitor), 1.0-MB RAM	DM	1228,–
Mega1wie oben, jed. mit 2-/ 2.5-MB RAM	DM	1578,–
Mega-2, kpl. (ohne Monitor), 2.0-MB RAM	DM	1578,–
Mega-2, kpl. (ohne Monitor), 4.0-MB RAM	DM	1948,–
PC-Speed, incl.Buch „PC/AT Gewußt wie"	DM	378,–
AT-Speed, incl.Buch „PC/AT Gewußt wie"	DM	478,–
Buch einzeln „PC/AT Speed-Gewußt wie"	DM	48,–
Aufpr. PC/AT Speed-Einbau in Neurechner	DM	+50,–
Atari Monochrom-Monitor Typ SM-124	DM	298,–
Wechselcartridge SQ-400 für MF-44	DM	258,–
Sonstige Atari-Produkte, Zubehör usw		a.A.

Achtung!

Wir liefern auch fast alle IC's/Chips, Dioden, Transistoren, Spannungsregler, Kondensatoren, Widerstände, Kabel/Litzen, usw.! Kleinteile nur in Verpackungseinheiten. Bitte anfragen!

Ramerweiterung + Dyn.Ram's + Gal's + Eproms:

DRAM 1 MB* 1 Typ S1 1000-70 ns	DM	13,95
2 MB RAM-Erweiterung (0 MB bestückt)	DM	128,–
2 MB RAM-Erweiterung (2 MB bestückt)	DM	328,–
2 MB Bausatz, komplett (ohne Ram's)	DM	58,–
Nur Leerplatine, incl. Bestückungsanleitung	DM	58,–
Gal 16V8-25L (Sockel für 16V8 DM -,35)	DM	4,–
Gal 20V8-25L (Sockel für 20V8 DM -,40)	DM	6,–
E-Prom 27C256-150 ns Progr.Spr. 12,5 V	DM	8,90
E-Prom 27C256-200 ns, Progr.Spr. 12,5 V	DM	6,80
E-Prom 27C512-150 ns, Progr.Spr. 12,5 V	DM	12,80
E-Preom 27C512-200 ns, Progr.Spr. 12,5 V	DM	10,80
E-Prom 27C010-150 ns, Progr.Spr. 12,5 V	DM	24,80
E-Prom 27C010-200 ns, Progr.Sp. 12,5 V	DM	22,80

IC-Sockel LC-Low Cost / PZ-Präz. / SM=schmal:

64pol PZ	5,50	48pol PZ	2,95	40pol LC	0,75
28pol LC	0,45	24pol LC	0,42	24pol SM	0,40
18pol LC	0,25	14pol LC	0,20	08pol LC	0,15
40pol PZ	2,25	32pol LC	0,55
20pol LC	0,35	18pol LC	0,30
PZ-Kont.	0,07	PZ-Sfkt	0,10

Atari-/ Universal-/ -Netzteile:

Schaltnetzteil-Platine, anschlußfertig aufgebaut und geprüft, kurzschlußfest, VDE 0804/6806 TÜV Rheinland, Abmessungen: 160 mm * 100 mm * 45 mm, Typ: + 5V/5 Amp., + 12 V/2,5 Amp., -12 V/0,5 Amp. DM 76,- Original Schaltnetzteil für Atari-/-Computer-/-Festplatten a.A.

Wir liefern auch Telefone, Haussprechanlagen, Anrufbeantworter! Bitte kostenlose Komplettliste anfordern! Lieferung erfolgt zuzüglich Porto/Verpackung, alle Angebote freibleibend! *** Gegen Terminvereinbarung auch Selbstabholung ***

HL Computer

Rechnersysteme
Zubehör
Beratung

ATARI-ST-Speichererweiterungen incl. Einbau/Garantie:

alle ST (mit 512 KB)	auf 1 MB	99 DM
	auf 2,5 MB	375 DM
	auf 4 MB	649 DM
alle ST's (mit 1 MB)	auf 3 MB	395 DM
	auf 4 MB	649 DM
MEGA 2 (mit 2 MB)	auf 4 MB	350 DM

Platinen 2 MB o. RAM's 89 DM, bestückt m. RAM's 275 DM
RAM's: 511000 Stck nur 11 DM / 41256 Stck nur 4.40 DM
Festplatten anschlußfertig für ATARI-ST ab 929 DM
z.B. SCSI-Speed-Drives, mit SCSI- u. DMA-Port, Echtzeituhr umfangreicher Software, dt. Handbuch, s. leise, 1 Jahr Gar.
49 MB für 1199 DM 85 MB für 1349 MB

HP-Deskjet 500 Tintenstrahldrucker	1399 DM
Scanman Handyscanner incl. Repro-Studio	540 DM
AT-Speed oder AT-Once incl. Einbau	499 DM

Jemand billiger ???
Die Preise könnten sich geändert haben !
Rufen Sie uns doch mal an !!!

HL-Computer Horst Lehmann Tel.: 06693/674
Birkenstr.11 3585 NEUENTAL 1
24-h-Bestellannahme Tel. Beratung nach 17 h

NEU! Ultimate-PCB

IC1

Professionelles Platinenlayout
Layout-Editor mit Autorouter

- vektororientiertes CAD
- WYSIWYG-Darstellung
- Platinengröße bis 1.6 x 1.6 m²
- 20 Layer, Multilayerplatinen bis 11 Lagen
- Auflösung 1/1000 Zoll
- Feinleitertechnik
- beidseitige SMD-Bestückung
- 4 Lötaugenformen, Außen- und Bohrdurchmesser frei wählbar
- beliebige Leiterbahnbreiten
- Zoll- oder mm-Raster
- stufenloses Zoom
- umfangreiche Bauteilbibliotheken, leicht erweiterbar
- Gummibandtechnik

für Atari ST, 1 MB, SM124, 720KB-LW

- Netzlistenübernahme aus Schaltplanprogramm möglich
- Luftlinien-, Signalverwaltung
- Online Design-Rule-Check
- integr. Mehrpaß-Autorouter
- Vorzugsrichtungen, 45/90°
- autom. Durchkontaktierung
- variable Route-Bereiche
- Sperrflächen
- Treiber für alle gängigen 9- und 24-Nadeldrucker
- Treiber für HPGL-Plotter, Gerber-Fotoplotter, Excellon-Bohrautomat
- Ausgabe von Platinenlayout, Bestückungsplan, Lötstoppmaske, Bohrplan
- Ausgabemaßstab 0.1 bis 10
- deutsches Handbuch

HK-DATENTECHNIK
Dipl.-Ing. H. Kahlert
Heerstraße. 44
4047 Dormagen 11

Einführungspreis DM 149.-
Demodiskette DM 10.-
Info kostenlos ! zzgl. Porto + Verpackung

HASCS
FANTASY AND ADVENTURE

HASCS - THE GAME CREATOR

"HASCS ist ein erstaunlich mächtiger Spielegenerator. Er ermöglicht die Erstellung anspruchsvoller Rollenspiele, bei denen die Interaktion mit dem Benutzer über Grafik und Maus abläuft."

Urteil des Computermagazines XEST 7/90, Seite 55-56

"Sollten alle Eingaben Ihren Wünschen entsprechen, können Sie stolz sein, denn das Endresultat kann sich bei HASCS sehen lassen. ... Im wesentlichen kann man HASCS - THE GAME CREATOR uneingeschränkt empfehlen, da der Editor vor allem durch seine Einfachheit besticht." nur

Das meinte die Zeitschrift PD-JOURNAL 1/91, Seite 62-64 49,--

HASCS - Expedition nach Vartoria

"Die Neugier, welche Ereignisse und Orte nun bevorstehen, schiebt einem vorzeitigen Spielabbruch einen wirksamen Riegel vor." nur

So das Computermagazin XEST 7/90, Seite 57 24,90

HASCS läuft nur in der hohen Auflösung, doppelseitiges Laufwerk ist erforderlich. Bezahlung: Bar, Scheck oder per Nachnahme. Alle Preise zzgl. DM 3,-- Porto. Beide Programme als ein Paket: DM 69,--. Bei Fragen fordern Sie bitte unsere kostenlose Produktinformation an.

COMPUTER-SERVICE KOHLER
Don-Carlos-Str. 33B □ D-7000 Stuttgart 80
Tel.: 07 11 / 6 78 73 92 (24h-Service)

ATARI ST Anwendungssoftware

ST-AUFTRAG DM 398.–
Integriertes Paket zur Auftragsabwicklung für Hersteller, Händler, Vertreter, Kleingewerbetreibende und Dienstleistungsbetriebe. ST-AUFTRAG bietet einen gewaltigen Funktionsumfang, von dem Sie bei anderen Programmen nur träumen können, und dies bei einfachster Handhabung. Aber am besten überprüfen Sie das selbst, indem Sie gleich eine Demodiskette (DM 20.– nur gegen Vorauskasse) anfordern.

ST-BOOKKEEPER DM 198.–
Komfortables Buchführungsprogramm der neusten Generation für Gewerbe und Privathaushalt. Freie Kontenwahl (mit Angabe der Ust-Sätze und Privatanteile). Ausgaben-Journal, Saldenliste, Kontenblätter, GuV-Rechnung, Ust-Voranmeldung usw. Deutsches Handbuch.

ST-ÜBERWEISUNGSDRUCK DM 45.–
Bedruckt alle Arten von Überweisungsträgern, Schecks, Zahlkarten usw. Universelle Druckeranpassung, eigene Formulare können mit einem Texteditor selbst erstellt bzw. angepaßt werden. Automatische Umwandlung des Betrages in ein Zahlwort. Die Eingaben sind speicherbar.

DATENBANKANWENDUNGEN DM 69.–
9 fertig benutzbare Anwendungen für Adimens ST, u.a. Termine/Adressen, Lager, Videothek, Schülernoten, Bibliothek, Zeitschriftenartikel, KFZ-Vermittlung.

ST-SCHREIBMASCHINE DM 59.–
Mit diesem Programm arbeitet Ihr Atari ST mit Matrix- (oder Typenraddrucker wie eine Schreibmaschine. Sehr gut geeignet zum Ausfüllen von amtlichen Formularen und zum Schreiben von kurzen Briefen. Ausdruck entweder Zeichen für Zeichen oder über editierbares Display. 10 Flieskettasten definierbar. Druckeranpassung. Gedruckte deutsche Anleitung.

ST-ETIKETT DM 59.–
Bedruckt alle Arten von Etiketten (bis zu 10 Bahnen nebeneinander). Ausgabebild kann vom Anwender frei definiert werden (mit 1st Word Plus). Mit Seriennummerngenerator und Wiederholfunktion.

ST-TRAINER MATHEMATIK DM 59.–
Lern- und Trainingsprogramm für Schüler (1.–6. Schuljahr). Abfrage der vier Grundrechenarten, Bruchrechnen, Kürzen, Erweitern, Potenzieren, Radizieren, Längen-, Flächen-, Raum- und Gewichtsmaße (Umrechnungen). Die Abfrage- und Benotungsbereiche sind frei einstellbar. Protokolldruck optional. Mit deutschem Handbuch.

AS-HAUSHALT (Haushaltsbuchführung) DM 89.–
ST-FIRMENBUCH (Buchführung für Gewerbetreibende) DM 149.–

Gesamtkatalog kostenlos! Versand nur gegen Vorauskasse (V-Scheck, keine zusätzlichen Versandkosten) oder per Nachnahme (DM 6.50 Versandkosten). Demodisketten nur gegen Vorauskasse.

AS DATENTECHNIK
AS-DATENTECHNIK
Mainzer Str. 69
D-6096 RAUNHEIM
Telefon: 06142/2 26 77
Fax/Btx: 06142/2 33 79

Schulmeister ST

Atari ST (Mega ST) 500 K Ram, sw - Monitor Die Noten- und Klassenverwaltung mit Pfiff. Ein flexibles, bewährtes Konzept für Lämpels aller Schulstufen. Lassen Sie Ihren Rechner die tägliche Routinearbeit erledigen damit Sie sich Ihren pädagogischen Aufgaben widmen können. Auch für die Schweiz geeignet !

Ausführliche Information mit Freiumschlag anfordern bei:

M.Heber-Knobloch
Auf der Stelle 27
7032 Sindelfingen



ST-AUFTRAG

Die integrierte Geschäftssoftware für den ATARI ST



Was ist ST-AUFTRAG?
ST-AUFTRAG ist ein Programm zur Auftragsabwicklung für Hersteller, Händler, Vertreter, Kleingewerbetreibende und Dienstleistungsgewerbetreibende, soweit ihre Produkt- und Dienstleistungspalette klar gegliedert ist. Das Prinzip: Durch Verknüpfung einer Adresse aus einem Adressenstamm mit Artikeln aus einem Artikelstamm wird ein Fakturadokument erstellt (z.B. Rechnung). Bei der Erstellung werden verschiedene Daten beeinflußt bzw. erzeugt (offene Rechnung, Artikelumsatz, Kundenumsatz, Lagerumschlag, Geschäftsumsatz usw.).

Leistungsübersicht in Stichworten:
- komplette Artikelverwaltung (max. 65535) mit Langtextbezeichnung (12 Zeilen à 60 Zeichen), Stücklistenverwaltung, 5 VK-Preise mit Mengenstaffeln und Rabattsätzen, Umsatzstatistik, diverse Sortier- und Selektionsverfahren. Ausgabe als Liste oder Etiketten.
- komplette Adressenverwaltung (max. 65535) mit Umsatzstatistik, 5 zeiliger Info, Kundenrabatt, Kredit, Zahlungsfrist usw. Ausgabe als Liste, Etiketten oder Serienbrief.
- Auftragsabwicklung beliebiger Angebote, Bestellungen, Auftragsbestätigungen, Lieferscheine, Rechnungen, Gutschriften, Warenrücknahmen und Sammelrechnungen. Fakturierung netto oder brutto. Dokument-Makros beschleunigen die Fakturierung. 10 Versandarten. Lieferanschrift extra.
- Sonstiges: Offene-Posten-Liste, Bestellvorschlagsliste, autom. Mahnwesen, Lagerüberwachung, alle Aufgaben auf Drucker, Bildschirm oder Datei, 16 frei editierbare Liefer/Versandarten, Nachnahmegebührenkarten bedrucken, Kurzbrief, Import/Export, Schnittstelle zu Buchführung.

Alle Ausgaben werden über Formulare gesteuert, die vom Anwender individuell verändert werden können (mit 1st Word Plus). Alle Schriftattribute und Schrifttreiben von 1st Word Plus verwendbar. Druckeranpassung über CFG-File.

Benötigte Hardware:
Monochrommonitor, doppelseitiges Diskettenlaufwerk, ATARI ST oder MEGA ST mit mindestens 1 MB RAM. Festplatte empfehlenswert.

PREISE:
ST-AUFTRAG (mit Handbuch im Ringordner) DM 398.–
Demoversion DM 20.–
Handbuch (nur lieferbar mit Demoversion) DM 50.–

Die Preise für Demoversion und Handbuch werden bei Kauf voll angerechnet!!! Versand nur per Vorauskasse (keine Versandkosten) oder Nachnahme (DM 5.– Versandkosten)! Demo und Handbuch nur per Vorauskasse!

AS DATENTECHNIK
AS-DATENTECHNIK
Mainzer Str. 69
D-6096 RAUNHEIM
Telefon: 06142/2 26 77
Fax/Btx: 06142/2 33 79

50 Spiele auf 10 doppelseitigen PD-Disketten

--- Game-Paket ---

für nur DM 30,--

Hier kommen Spiele-Freaks (Ballern, Adventure, Fantasy, Managen, Strategie, Sport, Denken, Hüpfen, Quiz uvm...) auf ihre Kosten. Für nur DM 30,-- erhalten Sie 10 doppelseitige PD-Disks mit insgesamt 50 Spielen! Ein Farb- und Monochromemulator haben wir gleich noch mitkopiert...

...Rennhund, Quizmaster, XPlus, Wurm, Hinem, Zyklop, Kniffel, Revenge, Face's Revenge, Blocke, Khan, Xenotron, Chicago, Transmitter GmbH, Poolbillard, Laser Chess, Risky, Droid III, Pferd, New Tris, Roadblocks, Borsenfieber, Crystel Cave 2.2, Schiffe, Toledo Salamanca, Economy, WZCS, SSH, Cromo, Anno 1700, Bauer, Mirakel, Imperium, Kolonial, Play me, Space Fighter 2, Mirror, Motodrom, Tangran, Up & Down, Othello, Cubus, Pente, The Lords of Fate, Lost Civilisation, Ripcord, Bridge it, Castle Combat, Clowns, Space Ball und Meteor... Bez.: Scheck, Bar oder per Nachn.

Fordern Sie auch bitte unseren Gesamtkatalog (gedruckt) für DM 3,-- an.
Ach: Bei uns kostet jede PD-Disk nur DM 4,--

Solche "Schlager" gibt es nur bei...
COMPUTER-SERVICE KOHLER
Don-Carlos-Str. 33B, D-7000 Stuttgart 80
Telefon: 07 11 / 6 78 73 92 (24h-Service)

2,5 Megabyte
für Atari 260/520/1040ST und Mega1/2.

- Bausatz mit 2-seitiger Platine (Lötstoplack)
- Sockel mit gedrehten, vergoldeten Kontakten und Kondensatoren
- Kompletter Kabelsatz
- 10-seitige Einbauanleitung für jeden Typ.
- Auch für SMD-MMU's, 3MB möglich.

ab DM 89,–

Versand: DM 5,–, NN: DM 7,50. Einbau incl. RAMs günstig zu Tagespreisen. Fordern Sie ausführlichere, kostenlosen Infos an.

THOMAS HEIER
SPEICHERERWEITERUNG

Gorch-Fock-Straße 33 • 2000 Schenefeld
Tel.: 040 / 83 93 10 00 – 01

Einkommen-/Lohnsteuer 1990

Direkt vom Steuerfachmann. Berechnet alles. Komfortable Eingaben, jederzeit korrigierbar, aussagekräftige Ausgabe mit Hinweisen auf Steuervergünstigungen, Datenabspeicherung, Alternative Berechnungen, Berlinpräferenz, § 10e! 54-seitige ausführ. Broschüre. **Ausdruck in die Steuererklärung.** (Mantel, N, V, KSO)

Vorgestellt als Entdeckung des Monats in PC Praxis 1/91

Für Atari ST mono nur 99 DM

Demo-Disk 10 DM · Info gg. Porto bei

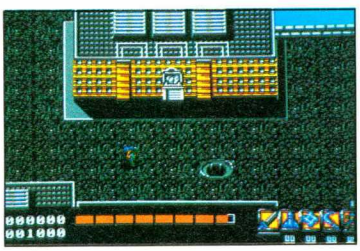
Dipl. Finanzwirt Uwe Olufs
Bachstr. 70k · 5216 Niederkassel 2
Tel.: 02208/4815 FAX/BTX 022084815

Hallo Freaks,

Viel Neues gab es in diesem Monat nicht. Die Hersteller scheinen alle Spiele schnell noch vor Weihnachten auf den Markt geworfen zu haben, so daß jetzt nicht mehr viel übrig bleibt. Natürlich gab's einige neue Spiele, doch die übliche Flut setzte nicht ein.



Die Turtles, anscheinend der neue Mega-Kult in Amerika, sind auch nach Deutschland geschwappt. Das allerdings mit einer riesigen Flutwelle, denn nach dem Film, der Fernsehserie, den Rubbelbildern, Plüschtieren und Comics gibt's jetzt auch die Turtles als Spiel. Und, wer hätte es gedacht, die Packung enthält Aufkleber, Tätowierbilder und Postkarten. Natürlich auch eine Diskette, auf der das Spiel „Teenage Mutant Hero Turtles" enthalten ist. Sie sind unterwegs, um die schöne April O'Neill zu befreien, die von Shredder, einem üblen Unhold, entführt wurde. Schön und gut, nur wie läuft es ab? Als die Schildkröten Leonardo, Raphael, Michelangelo und Donatello laufen Sie durch die Stadt oder „die schöne Seite des Abwasserkanals" und eliminieren Ihre Gegner - das bei einem mäßigen Sound und einer nicht gerade umwerfenden Grafik. Die Turtles erinnern etwas an die üblichen Plattformspiele, sind aber anders konzipiert. Insgesamt kann das Game eigentlich nur absoluten Turtles-Fans empfohlen werden. Ansonsten: Finger weg!



Turtles: Hängen im Schacht

Bei S.T.U.N. Runner fliegen Sie mit einem Raumschiff durch enge Kanäle - und das mit einer atemberaubenden Geschwindigkeit von über 900 Meilen pro Stunde. Das sollte man sich nicht so leicht vorstellen, denn wenn man den falschen Weg auf der Strecke fährt, wird das Raumschiff langsamer und Sie bekommen Zeitprobleme. Richtig gelesen, zusätzlich fliegen Sie auch noch gegen die Zeit. Damit Sie auch den richtigen Weg finden, sind im ersten Level Sterne verteilt, die die optimale Strecke anzeigen. In späteren Levels muß man dann allerdings den Weg selbst finden - dann wird es auch zur reinen Routine. Gegner gibt's dabei auch. Einmal die Feuertaste drücken, und sie sind mit einem Mini-Laser, der eher nach einer Pinzette aussieht, zerschossen. Das ganze ist recht schnell aus Vektorgrafik aufgebaut und macht auch einige Tage lang Spaß, doch irgendwann wird's dann auch langweilig.

Soweit für diesen Monat. Stürzen Sie sich ins Vergnügen und lassen Sie sich nicht unterkriegen! Viel Spaß beim Zocken.

MP



S.T.U.N. Runner: Erst Spaß, dann Lähmung

Pick 'n Pile

8

Grafik

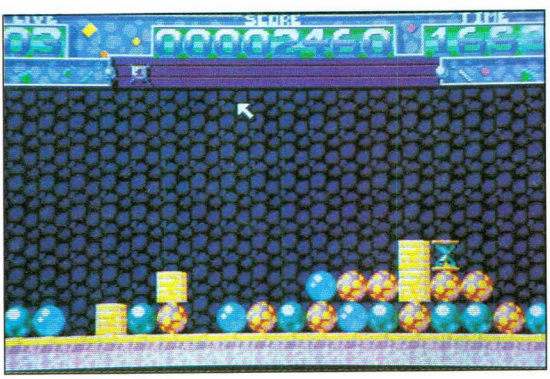
Sound

Motivation

Die einfachsten Spielprinzipien führen meistens zu den besten Ergebnissen. Das fing an beim ersten Ballerspiel (man denke an „Fort Apocalypse") und führte über Plattformspiele („Space Taxi") bis hin zu Tetris, dem erfolgreichsten Spiel der Welt. Zwar weniger erfolgreich, doch auch sehr spielenswert ist „Pick 'n Pile". Kommen Sie mit in die gute Kinderstube!

Woran denken Sie, wenn Sie klingelnde Kassen, Geldgeklimper und Uhrengeticke hören? Richtig, an einen kassierenden Uhrenhändler. Was das mit Pick 'n Pile zu tun hat? Nichts. Oder besser: fast nichts. Denn genau das ist die digitalisierte Sound-Unterstützung des Spiels, wenn der ebenfalls digitalisierte Vorspann verklungen ist. Dafür übrigens ein Lob, denn der Sound klingt wirklich sehr gut und ist von einem echten Geklingel nicht zu unterscheiden.

Natürlich ist das noch nicht das ganze Spiel, das wäre ja langweilig. Konkret geht es darum, Kugeln gleicher Farbe aufeinanderfallen zu lassen - dann verschwinden sie von der Spielfläche. Nun gibt es allerdings ab und zu auch eine ungerade Anzahl von Kugeln. Dafür sind die Joker da, die auf jede Kugel fallen können und dabei sogar Extra-Punkte erzielen.



„Wie stellt er sich das denn vor?" werden Sie fragen. Recht so. Gesteuert wird Pick 'n Pile per Maus. Sie müssen lediglich auf die zu verschiebende Kugel klicken und sich aussuchen, von wo sie nach unten fallen soll (sie fällt natürlich senkrecht nach unten). Trifft sie, wie gesagt, auf eine gleichfarbige Kugel, verschwinden beide, wenn nicht, stapeln sich die Kugeln auf. Welchen der farbigen Bälle Sie nehmen, ist ganz egal. Er darf sogar unter anderen begraben sein, selbige rutschen dann einfach um eine Position nach unten.

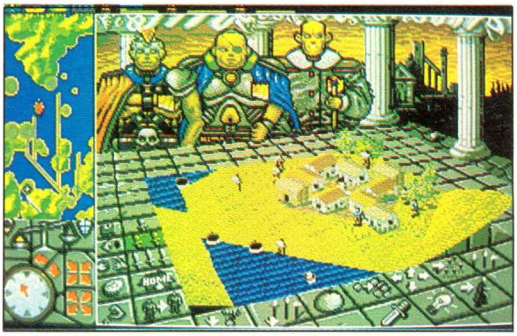
Was wäre ein solches Spiel, wenn man es nicht gegen die Zeit spielen müßte? Die Uhr tickt unermüdlich vor sich hin und weist Sie in Ihre Schranken. Hier ist also nicht nur eine perfekte Reaktion gefragt, sondern zusätzlich noch Schnelligkeit. Wer hier nicht rechtzeitig fertig wird, hat Pech gehabt und darf wieder von vorne anfangen.

So einfach sich das Spielprinzip auch anhören mag, es macht doch einigen Spaß, Pick 'n Pile zu spielen. Man könnte das Game als eine Mischung aus vielen unterschiedlichen Spielen bezeichnen, allesamt jedoch Denkspiele. Für die Strategen unter Ihnen ist Pick 'n Pile ein Spiel, das auf keinen Fall in der Diskettenbox fehlen darf!

MP

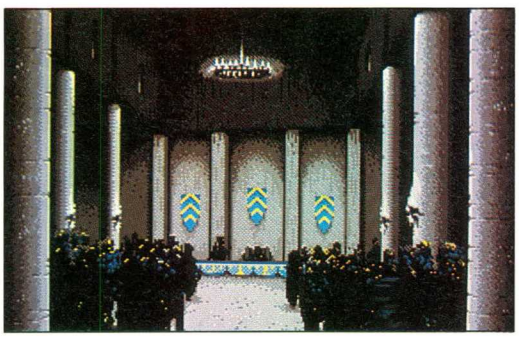
Powermonger

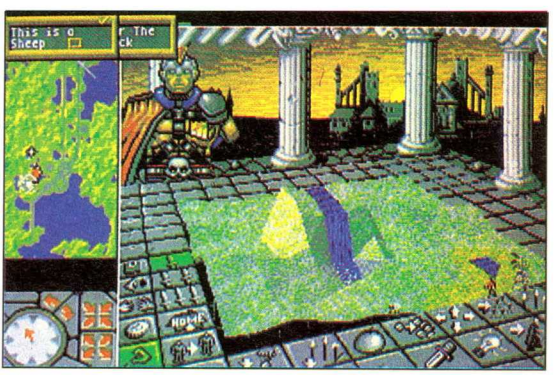




Die Götter müssen nicht verrückt sein, sie müssen lediglich gut spielen können. Zu dieser Einsicht kamen wir schon bei „Populous", dem kolossalen Machtspiel zwischen himmlischen Instanzen. Es steckte soviel Phantasie und Ideenreichtum in diesem Strategiespiel, daß es monatelang die Hitlisten anführte. Die Fortsetzung „Populous II" ist erst im Sommer 1991 fertig. Aber jetzt kommt erst einmal „Powermonger", ebenfalls vom Spitzen-Programmiererteam „Bullfrog". Wie „Electronic Arts" verlauten ließ, finden sich darin all die vielen tollen Ideen, die nicht mehr in „Populous" unterzubringen waren. Wie dort kann man auch in „Powermonger" zu zweit und per Modem spielen. Auch läßt sich das Tempo des Spielablaufs variieren. Trotz aller Verwandschaft hat „Powermonger" einen ganz eigenen Charakter. Es gibt sogar sowas wie ein zufallsgesteuertes „Königreich-Construction-Set", für den Fall, daß man die vorgegebenen Territorien lässig in die Tasche gesteckt hat. Und dann gibt es natürlich eine neue Story. Bei „Powermonger" ist der Kunde wirklich König - und zwar König des Inselreiches Miremer. Er hat bloß nichts mehr davon, seit ein Erdbeben das Eiland völlig verwüstet hat. Nur noch Sorgen! Wohin mit dem verschreckten Volk? Da es in der Spielwelt noch 195 weitere Siedlungsräume gibt, gehen König und Soldaten auf die Suche nach dem geeigneten Land. Dummerweise sind all diese Gegenden bereits besiedelt. Aufgabe des Herrschers ist es folglich, die Ländereien in Besitz zu nehmen. Als noch unerfahrener Eroberer muß man zunächst die zirka 30seitige Anleitung studieren und dann eine vorgegebene kleine Insel einnehmen. Anschließend herrscht Eroberungsfreiheit: jedes interessante Inselchen oder Siedlungsland wird in einem Auswahlmenü einfach angeklickt und im Spielverlauf nach strategischen Gesichtspunkten vereinnahmt. Ein guter Feldherr braucht eine gute Landkarte. Deshalb ist dauernd ein Kartenfeld eingeblendet, auf dem man zwischen unterschiedlichen Darstellungsweisen hin- und herschaltet. Eine besteht in reiner Landschaftsgrafik, eine andere in einer Siedlungs- und Wegekarte. Anhand einer dritten ist erkennbar, wo eßbare Tiere leben und wo sich Gemüse und Früchte pflücken lassen. Mit einer Zoom-Funktion läßt sich der jeweilige Gegenstand des Interesses vergrößern und genau ansehen. Von solch raffinierten Features strotzt das Spiel nur so. Deshalb auch die gehörige Anzahl Icons und Symbole, mit denen man steuert, identifiziert, kämpft usw. Will man beispielsweise Genaueres über eine neuentdeckte Pflanze erfahren, klickt man das „?"- Symbol an und erhält reichlich Antwort. Und wenn ein Feldzug fällig ist, klickt man das Invent-Icon, und sofort erfinden und schmieden die Handwerker geniale Kampfmaschinen und Waffen. Alles untersteht dem Befehl des Spielers. Er kann sogar die Laune seines Captains, des Ranghöchsten aller Untergebenen, einstellen. Dazu wird das Schwerter-Icon benutzt. Ein Schwert stimmt den Captain friedlich, bei zwei Schwertern wirkt er recht autoritär und bei dreien brennt dem Guten leicht die Sicherung durch. Des Captains Stimmung muß der Spieler wie ein strategisches Instrument einsetzen. Geht es darum, in einem fremden Dorf Speis' und Trank zu schnorren, wird ein bedrohlicher Captain das Verlangte erhalten, sollte sich dort aber nicht wieder blicken lassen. Auf die freundliche Tour bekommt er nur wenig Nahrung, wird aber bei seinem nächsten Besuch willkommen geheißen. Wie man sieht, kann man ein Land auch friedlich erobern. Der König hat bereits dann gewonnen, wenn er die Hälfte der Bevölkerung hinter sich hat. Merke: satte Untertanen sind wohlgesonnene Gefolgsleute. Herausforderung in diesem Spiel ist es, solche goldenen Regeln zu erkennen und zu befolgen. „Powermonger" fasziniert, weil hier eine Phantasiewelt durch Gesetze und stimmige Details lebendig wird. Man muß die Zeichen nur richtig deuten, zum Beispiel den Sound. Blökende Schafen sind in Hungerzeiten ein Schicksalswink. Schließlich gibt Schafsfleisch einen guten Braten. Manches Geräusch dient als Warnung: Heftiges Schnaufen des Captains bedeutet meist eine Fehlentscheidung. „Powermonger" ist ein Spiel für Leute mit Köpfchen. Entsprechend komplex und vielschichtig geriet deshalb die Icon-Steuerung. Im Kleinen wie im Großen hat „Bullfrog" ein bißchen Schöpfer gespielt. Die Crew wird dies auch weiterhin tun; sie arbeitet bereits an einer Powermonger-Landschaftsdiskette mit neuen Inseln. „Powermonger" ist schon jetzt ein Klassiker unter den Strategiespielen. Es gehört zu den besten Games der letzten Monate und darf einfach in keiner Software-Sammlung fehlen.

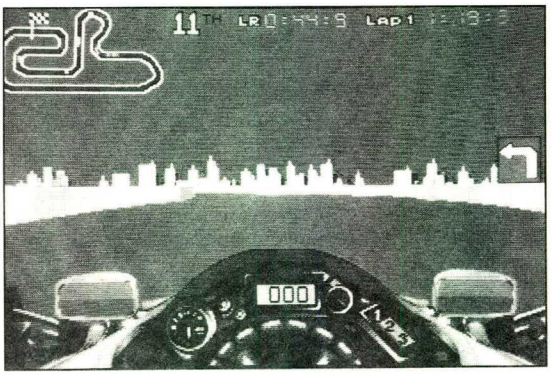
CBO





Vektor Championship Run

2
Grafik
Sound
Motivation

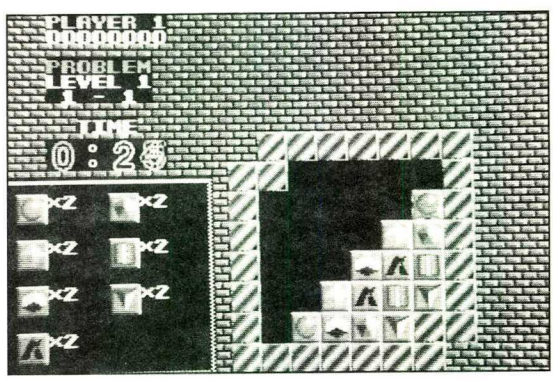


Fahrsimulationen aller Art erfreuen sich nach wie vor ungebrochener Beliebtheit. Es verwundert deshalb auch kaum, daß so manches kleine Software-Haus sich eine Scheibe vom gewinnträchtigen Kuchen abschneiden möchte. Vektor Championship Run vom Mini-Label Impulze ist einer dieser zahlreichen Trittbrettfahrer und ein schlechter noch dazu. Am Steuer eines Formel-1-Wagens nimmt der Spieler an einer Weltmeisterschaftssaison teil. Um sich für das jeweils nächste der insgesamt sechs Rennen zu qualifizieren, muß mindestens ein Platz unter den ersten Vier belegt werden. Kein leichtes Unterfangen, nimmt man die Fahrt doch stets als zehnter und damit letzter des Feldes auf. Der Rennverlauf wird aus der Perspektive des Fahrers in ausgefüllter Vektorgrafik dargestellt. Die links oben in den Bildschirm eingeblendete Übersichtskarte erleichtert die Orientierung, scharfe Kurven werden zusätzlich durch Schilder angezeigt. Das Armaturenbrett beinhaltet eine Geschwindigkeitsangabe in Meilen und zwei kleine Rückspiegel, die herannahende Mitstreiter einfangen. Gesteuert wird mit der Maus, die Gangschaltung übernimmt auf Wunsch der Computer. Jeder Kurs kann beliebig oft trainiert werden, um sich mit den Gegebenheiten vertraut zu machen. „Championship Run" beweist einmal mehr, daß schnelle Grafik allein noch lange kein gutes Spielgefühl ergeben muß. Zwar werden Biegungen, Hindernisse und gegnerische Fahrzeuge in tadelloser Geschwindigkeit und ausreichendem Detail herangezoomt, doch macht die übersensible Steuerung die Kontrolle des Wagens zur Qual. Selbst im niederen Drehzahlbereich bricht das Vehikel bei der kleinsten Mausbewegung aus, weswegen man vorsichtshalber die meiste Zeit mit zirka 100 Stundenkilometern über die Piste düst. Den Bit-Konkurrenten ist dieses Problem merkwürdigerweise nicht vertraut: Ständig wird man von hinten angefahren und überholt. Mit etwas Übung mag man sich an die Handhabung des Programms gewöhnen, unverzeihlich aber ist der Mangel an Optionen. Boxenstops, Zwei-Spieler-Modus oder Zeitlupenwiederholungen bleiben Hits wie „Lotus Esprit Turbo Challenge" oder „Indianapolis 500" vorbehalten. Addiert man dazu die nach schlecht digitalisiertem Rasenmäher klingenden Sound-Effekte, ergibt sich eine einzige große Mangelerscheinung, die nach wenigen Minuten der digitale Kolbenfresser ereilt. Den Programmierern sollte man vorsichtshalber den Computer abnehmen.

CBO

Puzznic

8
Grafik
Sound
Motivation



Seit es Tetris gibt, scheinen alle Computerspieler auf dem Denkspiel-Trip zu sein. Wer ab und zu in der Spielhalle vorbeischaut, hat sicher auch schon mal Puzznic gesehen. Das Warten auf den Clone dieses Spiels hat ein Ende, denn ab sofort ist es erhältlich.

Worum geht's? Ganz einfach. In einem (je nach Level) mehr oder weniger verwinkelten Raum, der durch Steine begrenzt wird, gilt es, passende Puzzlesteine zusammenzuschieben. Immer wenn zwei oder mehr Steine des gleichen Musters neben- oder übereinander liegen, verschwinden sie. Sind alle Steine abgebaut, ist der Level geschafft und es geht in den nächsten. Das ist eigentlich schon das ganze Spielprinzip.

„Das kann doch nicht so schwierig sein", denkt sich der geneigte Spieler, und fällt nach dem dritten Level gehörig auf die Nase. Bei Puzznic gibt es verschiedene Hindernisse, die einem das Leben ziemlich schwer machen. So ist beispielsweise manchmal eine ungerade Anzahl von Steinen vorhanden. Das Problem dabei liegt auf der Hand: Wenn zwei gleiche Steine aufeinandertreffen, kann man den dritten nicht mehr verwerten. So gilt es also, geeignete Vorsprünge zu finden, in denen diese Aufgabe gelöst werden kann. Ein anderes Problem sind etwa sich bewegende Steine - entweder waagerecht oder senkrecht. Wer mit denen nicht umgehen kann, ist ziemlich schnell verloren.

Das allein wäre noch keine Schwierigkeit, wenn man nicht auch noch gegen die Zeit spielen müßte. Und die ist ziemlich kurz bemessen, manchmal zu kurz, um richtig nachdenken zu können. Dabei stellt sich dann auch ein Nachteil des Spiels heraus: Der Spielstand läßt sich nicht abspeichern, dadurch ist es leider nicht möglich, in dem Level fortzufahren, in dem man aufgehört hat - und es sind immerhin 144 verschiedene Spielstufen, die zu bewältigen sind. Dadurch kann das Hocharbeiten in den nächsten Level zum Geduldsspiel werden. Wie gut, daß Puzznic eine Pausenfunktion besitzt, die man nötigenfalls auch eine Nacht lang stehenlassen kann!

Keine Frage: Freunde von Strategie- und Geduldsspielen werden um Puzznic nicht herumkommen, es spielt sich exzellent. Die Grafik könnte zwar noch etwas verbessert werden, genauso wie der Sound, doch das Spielprinzip und die Motivation sind unschlagbar. Puzznic bedeutet: schlaflose Nächte, in denen man krampfhaft überlegt, wie man diesen verflixten Level lösen kann. Ist ein weiterer Kommentar notwendig?

MP

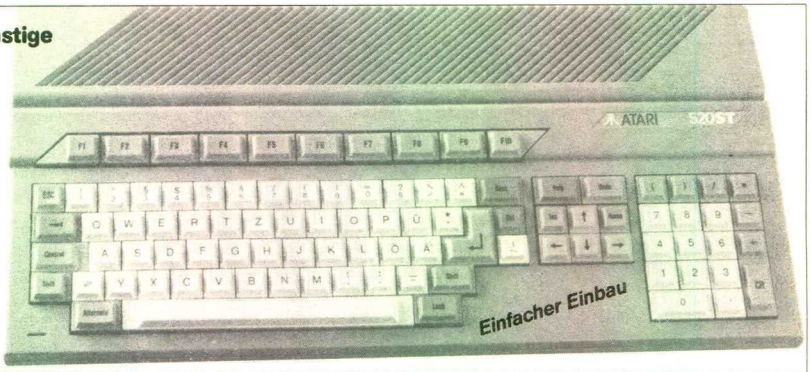
Komfortable und preisgünstige Umrüstung mit hohem Bedien-Komfort und optimalem Design

Farblich abgesetzte Flachtastatur

Farbe grau/weiß

Verstärkung des Tastendruckes durch Federnsatz

Preis DM komplett:
Baureihe ST 139,–
MEGA ST 130,–



AIS - Elektronik Postfach 64 · 7533 Tiefenbronn · ☎ (0 72 34) 69 15 + 52 32 · Fax 55 74

CIS Lohn & Gehalt 2

JETZT UMSTELLEN

Die professionelle Abrechnungs-Software für Ihren ST.

Unser CISYSTEM:



CIS-Lohn & Gehalt ist das derzeit einzige durch die Spitzenverbände der Sozialversicherer zum automatisierten DEVO-Meldeverfahren zugelassene Programm für Atari-ST-Computer. *Auszug aus dem Sachregister:* **Schnittstellen zu TIM II, ST-Fibu, fibuMan, Berlin-Abrechnung, Baulohn, Kassenlisten, Meldewesen, Steuerberechnung, modulares Konzept, Urlaubskonto, Sonderauswertungen, betr. Altersversorgung, VL, ZVK, Urlaubskasse ...**

CIS - Es ist Ihre Entscheidung, wieviel Arbeit Sie sich durch unsere Produkte sparen.

Ciechowski Computer Innovations **Ober-Saulheimer Straße 18, D-6501 Wörrstadt, Telefon (06732) 7354 oder 5018**

H plus das MODULA-2 TOOL

- Atari ST/TT
- MINIX
- RTOS-UH
- OS/9
- UNIX V
- Sun 3
- HP-UX
- MUNIX

HMplus, das blitzschnelle Modula-2 TOOLKIT mit automatischem MAKE und integriertem Linker für (fast) alle mc680x0 und mc68881/2 Computer

mosys, Jürgen Reiß, Petrinistr.34, 87 Würzburg ☎ (0931) 28 11 93

AB COMPUTER GmbH ATARI Beratung - Service
5000 Köln 41 Sülz Mommsenstr. 72 Ecke Gleuelerstraße

Ihr Fachhändler in Köln für Atari / XT / AT **Tel.: 02 21/43 01 442, Fax 46 65 15**
Wir bieten Ihnen noch Beratung und Service für Ihren Computer

SCSI-Festplatten > 580 KB/s	
20 MB 40 ms SCSI	748,-
40 MB 28 ms SCSI	999,-
40 MB 19 ms SCSI	1200,-
44 MB 25 ms Wechselplatte	1600,-
60 MB 40 ms SCSI	1450,-
80 MB 12 ms SCSI Quant.	1650,-
105 MB 12 ms SCSI Quant.	1848,-

ST Mega 1/SM 124 mit 1 MB	1400,-
ST Mega 2/SM 124, Maus	1800,-
ST Mega mit 4 MB, Maus	2200,-
ST Mega 2, 30 MB, WordPerfect	2198,-
ST Mega 4, 30 MB, WordPerfect	2598,-
ST Mega 4, 16 MHz, NEU Preis auf Anfrage	
Desktop-Anlage ST 4MB, 30 MB, ATARI Laser, Calamus	5800,-

Einige Artikel haben Lieferzeit. Anfragen!

PC Speed für den ST Version 1.4	298,-
PC Speed mit Einbau in ST, 24 St.	350,-
ST. Laufw. o. Bus 3.5 anschlußf.	239,-
ST Laufwerk40/80 5.25 Zoll TEAC	279,-
TEAC-Lw. roh, für Einbau in ST 1040	180,-
NEC Lw. roh für Einbau in ST 1040	190,-
ST Laufwerk, roh 3.5 TEAC 1.44 MB	180,-

NEU: AT Emulator von Vortex 80286	
Einbau wie bei PC Speed	450,-
mit Einbau in ST nur	490,-
Einbau innerhalb 24 Stunden	
VGA-Auflösung, komplett lieferbar	
NEU: AT Speed-Emulator von Sack	
80286 im Angebot nur	480,-

Speichererweiterung für Ihren ATARI, alle Modelle	
Speicherkarte 2 MB / 2.5 MB mit 2 MB bestückt	450,-
Speicherkarte 4 MB / 2 MB bestückt, teilsteckbar	450,-
Speicherkarte 4 MB / 4 MB bestückt	700,-
NEU: Erweiterung voll steckb. 4 MB-Chips	
Test CT 1/91 Super klein 2 MB	548,-

Gleiche Erweiterung 4 MB	798,-
Speicherkarte 512 KB auf 1 MB steckbar	198,-
Drucker	
NEC PT 60 A4	1498,-
Panasonic 1123	650,-
Citizen 24 Nadeln	848,-
Citizen SD124 24 N.	600,-
HP Deskjet	1398,-
Laser SLM 605	2200,-

Eizo Monitor 9060 SZ 14-Zoll	1550,-
14 Zoll Mon.	999,-
Multisync S/W	548,-
Monitor Kabel	69,-
Switchbox 2 Mon. an ST mit Softw.	45,-
HF Modulator	198,-
ST Tastatur Geh.	120,-
ST Uhr intern	95,-
Adimens 3.0	398,-
WordPerfect Text	148,-
Mega Paint Prof.	798,-
1st Word	180,-
Signum2 Text	400,-
Tempus 2.06	119,-
Tempus Word	798,-

Script Text	169,-
Script Text 2	280,-
Freesoftware aus ST	
10 Stk. nur	50,-
Freesoft einzeln	6,-
Über 800 PD Disk-Info anfordern gegen	5,-
Mega Paint 2	450,-
Calamus	798,-
(Font nach Wahl)	
Fax Modem 2400/4800	398,-
mit Fax Software ST	
Modern Discovery 2400/1200/300	278,-
Die Inbetriebnahme der Modems am öffentlichen Postnetz der BRD ist verboten und unter Strafe gestellt.	

Atari/Star/Schneider/Panasonic sind eingetragene Warenzeichen. Wir liefern für Ihre Firma die richtige Soft/Hardware/Beratung und Aufstellung. Faktura für XT/AT PC Komplettsystem mit Einweisung Info im Laden. Öffnungszeiten 10.00-13.00 Uhr, 14.00-18.00 Uhr Samst. 10.00-14.00 Uhr

---

Bei uns werben bringt

GEWINN

Sprechen Sie mit uns.
Heim Verlag ☎ 06151/56057

Toyota Celica GT Rallye

7
Grafik
Sound
Motivation

„Nichts ist unmöglich - Toyooo-taa!" tönt die Fernsehwerbung. Was zumindest spielerisch alles machbar ist, zeigt Gremlins' neueste Rallyesimulation. Man lenkt seinen Toyota Celica mit Joystick oder Maus, schaltet selbst oder läßt die Automatik ran: Je nachdem, was man für eine Empfindlichkeit eingestellt hat, reagiert der Celica sensibel oder stur auf Steuerbewegungen. Witzige und schwierige Variante: der Punkt „Reverse Steering" im Hauptmenü. Dabei verhält sich der Wagen entgegengesetzt seiner Steuerung: Reißt man das Steuer nach rechts, zieht der Toyota nach links ab. Echt witzig! Etwas hilfreicher als dieses Feature ist der Co-Pilot. Man kann ihn selber instruieren, was folgendermaßen geschieht: im Bildschirmfenster scrollt der gewählte Rallyekurs, der gezoomt und aus der Vogelperspektive gezeigt wird, wobei man sich schwierige Passagen aussucht und hier Wortkommandos eingibt. Während des Rennens wird der Beifahrer beispielsweise vor einer brenzligen Rechtskurve „turn right" ausrufen. Wie wichtig ein Co-Pilot ist, merkt man, wenn man sich auf dessen computergesteuerte Version verläßt. Der nimmt es nämlich nicht so genau mit dem richtigen Timing. Bereits auf den drei Übungspisten erfährt man dann, daß ein Crash gegen Bäume oder Häuser satte 20 Strafpunkte kostet. In diesen Ländern werden anschließend auf je zehn Parcours die entscheidenden Rennen gefahren. Nur wenn man bei einer Zehn-Strecken-Tour unter den ersten 20 durchs Ziel rauscht, darf man im nächsten Land an den Start. Die ausgefüllte Vektorgrafik bietet neben Rallyespaß auch noch landschaftliche Eindrücke. Hügel, Waldgebiete und Regen prägen die britische Insel, Sandstürme und Steppen charakterisieren Mexiko. Im winterlichen Finnland behindern Schneestürme, Matsch und Fahrbahnglätte den flotten Celica. Da Gremlin bei schlechter Witterung realitätsverliebt auch die Sicht entsprechend mies werden ließ, mußten gerechterweise auch Scheibenwischer eingebaut werden. Ein Druck auf die F1-Taste setzt die Wischblätter in Bewegung. Was es da außerdem noch an Schaltern, Hebeln und Knöpfen gibt, sieht man im Cockpit, das wie in „Test Drive" aus der Sicht des Fahrers gezeigt wird; sogar die Hände sind beim Lenken zu sehen. Wirklichkeitsnähe und gute Spielbarkeit sind in „Toyota Celica GT Rallye" Trumpf. Schade nur, daß die Codeabfrage so nervig ist und sich keine gegnerischen Rallyefahrer auf den Pisten tummeln.



CBO

Panza Kick Boxing

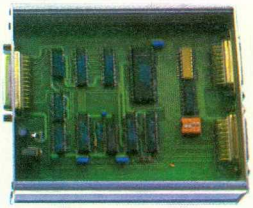
8
Grafik
Sound
Motivation

Kickboxing, das ist Boxen mit Faust und Fuß für harte Männer. Während sich abendländische Boxer nach strengen Regeln der Kunst mit weichen Handschuhen bearbeiten, langen die fernöstlichen Kollegen ordentlich hin. Im Zweispieler-Modus von „Panza Kick Boxing" geht es Mann gegen Mann. Stellvertretend für willige Gegner bietet das Programm acht computergesteuerte Kickboxer. Portraits stellen die potentiellen Gegner vor. Vom schlaffen Softi bis zur stahlharten Kampfmaschine sind alle Typen vertreten. Den eigenen Fighter steuert man per Joystick oder Tastatur. Da dieses Kickboxing-Spiel beachtliche 55 Schlag- und Fußtechniken drauf hat, wird es notwendig, sich Kombinationen für Angriff und Verteidigung zusammenzustellen. Jedes dieser sogenannten Patterns kann maximal 13 Schläge und Tritte enthalten. Für jeden Kampf lassen sich neue Kampfmuster festlegen. Zu allererst jedoch empfiehlt sich das Fitneßtraining mit Gewichtheben, Seilhüpfen und Sandsackboxen. Wie fit die Kickboxer sind, verrät ein Screen im Hauptmenü. Viele Trainingspunkte garantieren Ausdauer und Erfolg, wenn es ernst wird. Und das wird es, wenn man die Anzahl der Kampfrunden bestimmt und in der ersten Runde den computergesteuerten Schlaffi ausgetrickst hat. Danach gibt es kein Pardon mehr: die Faust aufs Auge, den Fuß in den Magen, den Ellenbogen zwischen die Rippen! Bloß nicht aufgeben. Wenn einer - hollywoodreif und dramatisch - zu Boden geht, zählt der Ringrichter an. Animationen und technische Realisation sind über alle Kritik erhaben. Die gute Spielbarkeit läßt alle Skrupel vergessen. „Zong!" - und noch einen Tritt gegen die Rübe. Verwunderlich: man hat auch noch Spaß an diesem knochenbrecherischen Spiel und fühlt sich gefordert, den Gegner durch noch fiesere Kicks endlich zu Boden zu befördern. Ein humanistisches und ethisch wertvolles Spiel ist „Panza Kick Boxing" sicherlich nicht, und die Spielidee ist auch nicht gerade neu. Lediglich die geniale grafische und animatorische Umsetzung macht dieses Kampfspiel interessant. Außerdem wurde ein Videomodus eingebaut, so daß man ganze Sequenzen speichern und zusammenhängend abspielen kann. Übrigens: Skrupellosigkeit allein genügt nicht - Panza Kick Boxing ist nur was für wirklich geübte Bildschirmprügelknaben.



CBO





Scanface

Das UNIVERSALINTERFACE für PANASONIC-Scanner am ATARI ST, incl. Scansoftware V. 1.1

Unterstützt alle 3 Panasonic Scanner. Automatische Scannererkennung

Die HARDWARE

FX-RS505 (max 400 dpi, Pseudograustufen mit Ditherverfahren)
FX-RS506 (max 400 dpi, 16 echte Graustufen)
FX-RS307 (max 600 dpi, 256 echte Graustufen)

Scannen und Speichern aller gängigen Bildformate möglich:
IMG (GEM-Image Format, für z.B. Calamus usw.)
TIF (Standard TIF Format = ermöglicht Datenaustausch mit IBM kompatiblen PC's) / **IFF** (AMIGA Standard Format / **BLD** (Megapaint Format) / **PIC, PAC** (STAD Format) / **PI3** (Degas Format) / **IMC** (Signum und Creator Format)
DRUCKERTREIBER:
Treiber sind vorhanden für: HP-Laserjet kompatible / Canon Laserdrucker / ATARI Laserdrucker / NEC P6 und kompatible 24-Nadel Drucker / EPSON kompatible 9 Nadel Drucker

DIE SOFTWARE

arbeitet mit allen gängigen Festplatten zusammen (ATARI, Vortex) / Scannen von Schwarz-Weiß und Graubildern / Mehrere Bilder in versch. Fenstern gleichzeitig bearbeitbar / Schnelles Scrollen durch die Bilder / Maßangaben alternativ in Pixel oder Zoll / Wandlung von Schwarz-Weiß nach Graubild und umgekehrt / Auslagerung von zu großen Graubildern auf die Festplatte / Lupenfunktion (4 fache Vergrößerung, dadurch einfache Nachbearbeitung / Ausschnitte löschen und kopieren (aus mehreren einzelnen Bildern kann ein neues erstellt werden / Grauwertscala / Lineare Solarisation (Helligkeit und Kontrast können verändert werden / Anzahl der Graustufen kann verringert werden

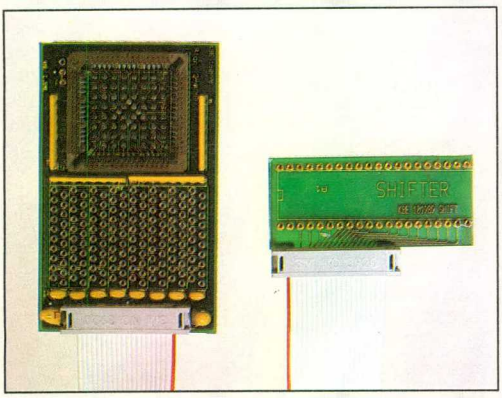
DAS BESONDERE

Scanface ermöglicht die Ansteuerung des ATARI-Laserdruckers schon mit Rechnern mit 512 KB / Software benötigt kaum Speicher / automatisches Erkennen des installierten Scanner - Modells / extrem schnelle Umrechnung von Schwarz-Weiß nach Grau / Unterschiedliche Rasterverfahren möglich ATARI Laserdrucker benötigt keinen eigenen Druckerpuffer

DAS NEUE

ohne Einschränkung auf dem TT lauffähig
Source-Code und Demo zur Programmierung der Schnittstelle liegt bei, dadurch ist die Anpassung aller Softwarepakete an Scanface möglich (für die Programmierer)

Neuer Preis 698,-- DM



Die sichere, saubere Lösung ihren ST auf 2,5 oder 4 MB zu erweitern

Die Leistungsdaten:

- geeignet für alle ATARI ST
- einfacher Einbau, da steckbar
- keine Kontaktschwierigkeiten, da die MMU durch einen Sockel ersetzt wird
- Spezialzange zum Ausbau der MMU ist im Lieferumfang enthalten
- extrem kleine Platine
- blockiert keine anderen Erweiterungen (AT-Speed, Mega Screen oder Hypercache
- durch die verwendete Technik ist die Erweiterung sehr betriebssicher
- mit 2 MB oder 4 MB erhältlich
- Software zur Speicherprüfung wird mitgeliefert (Anzeige des defekten Speicherchips)
- Einbauanleitung liegt bei

260, 520 ST,1040 STFM, STE, Mega 1 auf **2,5 MB**
260, 520 ST, 1040 STFM, STE, Mega 1 auf **4 MB**
Mega ST 2 auf **4 MB**

Erweiterung bestückt
mit 2 MB 598,– DM
mit 4 MB 748,– DM

Preise sind unverbindlich empfohlene Verkaufspreise

BESTELLCOUPON

Heim Verlag
Heidelberger-Land-Str 194
6100 Darmstadt-Eberstadt
Telefon: 06151/56057-58
Telefax: 06151/56059

Bitte senden Sie mir:

____ Scanface ❑ für 505 ❑ für 505U, 307 a 698,-- DM
____ Speichererweiterung auf 2,5 MB a 598,-- DM
____ Speichererweiterung auf 4 MB a 748,-- DM

zzgl. 6,-- DM Versandkosten (Ausland 10,-- DM)
unabhängig von der bestellten Stückzahl

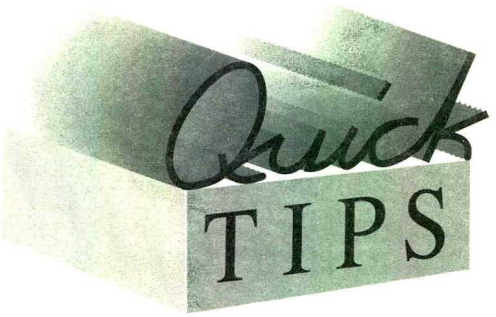
Preise sind unverbindlich empfohlene Verkaufspreise

Name: _____
Straße: _____
PLZ, Ort: _____

in Österreich
RRR EDV GmbH
Dr. Stumpfsr.118
A-6020 Innsbruck

in der Schweiz
Data Trade AG
Landstr.1
CH-5415 Rieden-Baden

Benutzen Sie die eingeheftete Bestellkarte



Andere Applikationen ans Ruder lassen

Das Pseudo-Multitasking unter GEM ist darauf angewiesen, daß alle gleichzeitig laufenden Applikationen (Hauptprogramm und Accessories) AES-Funktionen aufrufen. Dann nämlich tritt der sogenannte Dispatcher in Aktion und bestimmt, zu welchem Programm zurückgesprungen wird. Das muß nicht das sein, welches AES aufgerufen hat. Wenn man in einem Teil seines Programms keine AES-Funktionen benötigte, aber trotzdem anderen Prozessen eine Chance lassen wollte, blieb einem keine andere Möglichkeit, als einen Dummy-Aufruf zu machen, z.B. *evnt_timer()* ohne Wartezeit. Die GEM-Version 2.0 (bis jetzt nicht für Atari erhältlich) hat eigens dafür die Funktion *appl_yield()*. Doch auch beim Atari-GEM gibt es eine elegantere Version. Normalerweise erhält das Register D0 vor dem TRAP #2 den Wert 200 als Code für das AES. Nimmt man aber stattdessen 201, wird keine Funktion ausgeführt, sondern nur der Dispatcher aufgerufen. Der Zeiger auf den Parameterblock in D1 wird trotzdem benötigt, aber nur wegen des GLOBAL-Felds. Die Implementierung dieses Aufrufs in einer höheren Sprache sollte für jeden, der mit Assembler nicht gerade auf Kriegsfuß steht, keine Schwierigkeit sein.

Oliver Scheel

Haben auch Sie einen *Quick-Tip*?

Standen Sie auch einmal vor einem kleinen, aber schier unlösbarem Problem? Dann, durch Zufall bekamen Sie einen Tip und schon war es gelöst.

Ähnlich haben wir uns diese neue Rubrik in der ST Computer vorgestellt. Aufgerufen sind auch Sie, liebe Leser(innen)! Geben Sie Ihre Erfahrungen weiter, egal, ob es um Anwendungen, Programmieren o.ä. geht.

Wir sammeln Ihre (und unsere) Tips und stellen Sie ggf. in den Quick-Tips vor.

Einsendungen an: MAXON Computer
ST Computer Redaktion
Stichwort: Quick-Tip
Industriestr. 26
D-6236 Eschborn

Der Dreh vor der „ersten Hilfe"

Die Atari-Maus ist ein stark beanspruchtes Kleinod, weshalb nach einiger Zeit des Klickens gravierende Abnutzungserscheinungen vor allem der linken Maustaste zu schaffen machen. Möchte man den Zeigefinger vor einem ähnlichen Schicksal bewahren, muß man sich entweder nach einer neuen Maus umsehen oder kostengünstiger und entsprechend der Q-Tip-Anleitung in ST 3/90 die Tasten-Kontakte durch neue ersetzen. Zu dieser Einsicht kommt man in den meisten Fällen an einem Sonn- oder Feiertag, auf jeden Fall aber zu einem Zeitpunkt, wo es besonders schwierig ist, an neue Geräte oder Bauteile heranzukommen. Da nun beim ersten Mal nur der Kontakt der linken Maustaste stark lädiert ist, die rechte Taste aber noch einwandfrei funktioniert, bietet es sich als Sofortmaßnahme an, einfach die Funktionalität beider Tasten zu vertauschen (auch als Grundsatzlösung für Linkshänder empfehlenswert). Dazu öffnen wir, wie in ST 3/90 beschrieben und abgebildet, das Mausgehäuse und sichten in der oberen Hälfte am rechten Rand einen Steckverbinder. Mit einem Schraubenzieher hebeln wir den Buchsenstecker aus der Stiftleiste. Die beiden oberen Kontakte (weiß, gelb) sind die Schaltkontakte für die Maustasten (rechts, links). Diese Anordnung kann durch Messungen zwischen Steckverbinder und Anschlußbuchse der Atari-Maus ermittelt werden (Bild 1 und 2, siehe auch Profibuch). Der entscheidende Kunstgriff besteht nun darin, mit einer Nadel die beiden oberen Kontakte aus der Stekkerleiste herauszuziehen und vertauscht wieder hineinzustecken. Anschließend wird der Buchsenstecker wieder auf die Stiftleiste gesteckt und das Gehäuse zugeschraubt. Wer beim anschließenden Test das Gefühl hat, daß jetzt das Klicken überhaupt nicht mehr funktioniert, der traktiert wahrscheinlich nach wie vor die linke Maustaste. Nach einer Gewöhnungsphase jedoch funktioniert die Übergangslösung meist prächtig, und man hat Zeit gewonnen, nach dauerhafteren Lösungen Ausschau zu halten, bei denen dann ja wieder auf die „richtige" Maustastenbelegung umgeschaltet werden kann.

Editha Kuske, 6107 Reinheim 4



Bild 1: Der Buchsenstecker in der Maus

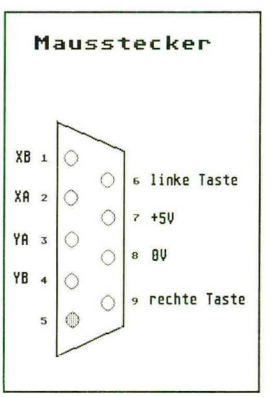


Bild 2: Der Stecker an der Maus

ST und SHARP-Pocket-Computer

Dieser Quicktip ist für alle Benutzer eines Sharp-Taschenrechners mit ST-Interface und des Programms Transfile gedacht. Mit Minimalaufwand kann man sich ein elektronisches Adreßbuch zimmern! Das folgende Mikroprogramm (als Anregung für ein komfortableres gedacht) ist in der Lage, die Telefonnummer zu einem bestimmten Namen (oder der ersten paar Anfangsbuchstaben desselben) auszugeben. Dabei wird einfach die DATA-Liste (ab Zeile 240) durchforstet und der erstpassende Eintrag angezeigt oder bei erfolgloser Suche eine Meldung ausgegeben (s. Listing 1).

Das Programm ist ja noch das wenigste, aber wie bekommt man seine Adressen in den Quellentext? Sein ganzes Adreßbuch in diese DATA-Zeilen eintippen ist nicht jedermanns Sache, umsomehr, weil bei einem Tippfehler eine falsche Person am anderen Ende der Leitung ist. Da ein echter ST-Benutzer ja sowieso seine ganze Bekanntschaft in irgendeiner Datenbank abgelegt hat, läßt sich das problemlos realisieren. Bei Adimens legt man sich eine Ausgabemaske in etwa dieser Form zurecht:

```
#COUNT# DATA
„#NAME#","#TELEFON#"
```

und läßt das ganze nicht beim Drucker hinaus, sondern legt es in eine Datei. Auch bei den Sharp-Modellen mit knickrigem Speicher lassen sich so an die hundert Telefonnummern speichern. Von Hand oder mit der Renumber-Option von Transfile werden noch die mit COUNT vorgezählten Zeilennummern zurechtgeschürbelt, mit der Zeile 'DATA „XXX"' (nicht vergessen!) ergänzt, die ganze Geschichte an obiges oder ähnliches Sharp-BASIC-Programm angehängt und in den Pocket-Computer geladen.

Marco Strehler, CH-8400 Winterthur

```
100 REM — ADRESSBUCH —110 „A":CLEAR
120 DIM S$(2)*20:REM DIMENSIONIEREN
130 DIM I$(0)*20
140 INPUT „NAME ? „;I$(0)
150 L=LEN I$(0):REM LAENGE ERMITTELN
160 RESTORE
170 READ S$(0):REM ZEILEN EINLESEN
180 IF S$(0)="XXX" THEN PRINT „NICHTS
    GEFUNDEN":GOTO „A"
190 READ S$(1)
200 S$(2)=LEFT$(S$(0),L)
210 IF I$(0)<>S$(2) THEN GOTO 170
220 PRINT S$(0):PRINT S$(1)
230 GOTO „A"
240 REM — DATEN —250 DATA „HAENSEL","11 11 11"
260 DATA „GRETEL","22 22 22"
270 DATA „BOESHEX","99 99 99"
... ........................
... ........................
900 DATA „XXX":REM ENDE
```

Listing 1: Eine Mini-Dateiverwaltung für Sharp-Taschenrechner

```
 1: 'Kopieren einer Datei (source, dest.)
 2: PROCEDURE copy(a$,b$)
 3:   LOCAL a%,b%,c%,i%
 4:   b%=1
 5:   IF EXIST(b$)
 6:     ALERT 3," | Datei existiert bereits!",
            1,"Weiter|Abbruch",b%
 7:   ENDIF
 8:   IF b%=1
 9:     OPEN „I",#1,a$          !Quelldatei
10:     OPEN „O",#2,b$          !Zieldatei
11:     a%=LOF(#1)              !Länge des Files
12:     b%=FRE(0)-3000          !Freier Speicher
13:     c%=a% MOD b%            !Rest
14:     DIM a|(b%)              !Speicher reser.
15:     FOR i%=1 TO (a% DIV b%) !Schleife
16:       BGET #1,V:a|(0),b% !Lesen
17:       BPUT #2,V:a|(0),b% !Schreiben
18:     NEXT i%
19:     BGET #1,V:a|(0),c%    !Rest lesen
20:     BPUT #2,V:a|(0),c%    !Rest schreiben
21:     CLOSE
22:   ENDIF
23: RETURN
```

Listing 2: Eine Kopierroutine in GFA-BASIC

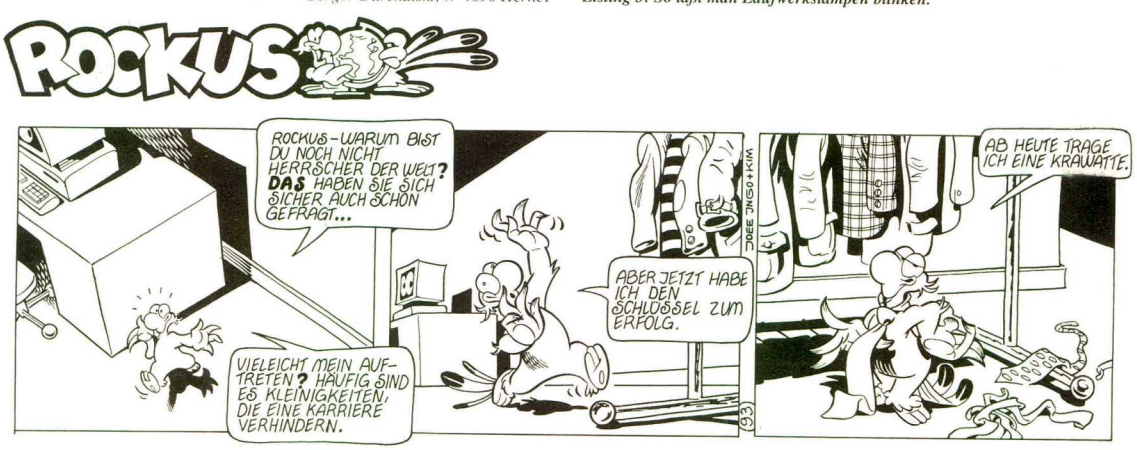
Kopieren und Blinken

Wer in GFA-BASIC Dateien kopieren möchte, muß sich eine eigene Routine dazu schreiben, da keine spezielle Routine für solche Zwecke eingebunden ist. Eine einfache und zudem schnelle Möglichkeit, Dateien zu kopieren, sehen Sie in Listing 2. Oder möchten Sie vielleicht einfach mal die Leuchtdioden Ihrer Laufwerke blinken lassen, um den Benutzer auf etwas aufmerksam zu machen? Nichts leichter als das, dazu müssen Sie lediglich Listing 3 abtippen!

Gergor Durchalski, W-4690 Herne1

```
 1: 'Läßt Laufwerkslampen blinken
 2: 'a&=Laufwerk 1 oder 2
 3: 'b&=wie oft blinken, c&=Blinklänge
 4: PROCEDURE drive_blink(a&,b&,c&)
 5:   FOR i&=1 TO b&
 6:     ~XBIOS(29,5-a&)
 7:     PAUSE c&
 8:     ~XBIOS(30,2+a&)
 9:     PAUSE c&
10:   NEXT i&
11: RETURN
```

Listing 3: So läßt man Laufwerkslampen blinken.



Friedrich Belzner
Sebastian Belzner

Kreative Grafik auf dem Atari ST

Neue Wege mit GFA-Basic 3.0

Haar bei München, 1990
Markt & Technik
339 Seiten, Diskette
16 Farbtafeln, DM 79,
-ISBN 3-89090-291-X

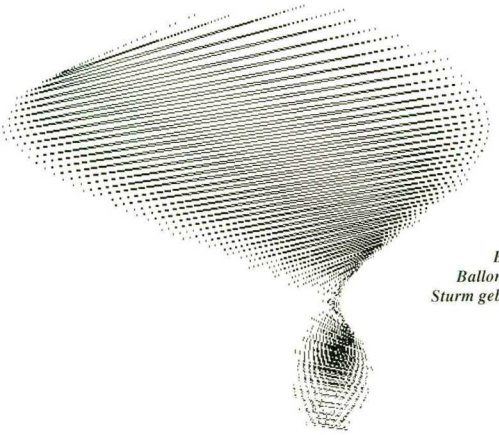


Bild 1: Ballon, vom Sturm gebeutelt



Bild 2: Kippschleife

Seit langem ist der Begriff „Art of Programming" nicht nur durch die Bücher von Donald E. Knuth bekannt, in „Kreative Grafik" geht es um „Programming of Art". Die Autoren wollen keinen Malkurs für die Verwendung von Zeichenprogrammen geben, sondern zeigen die man ansprechende Grafiken einfach programmiert und durch kleine Variationen interessante ästhetische Wirkungen erzielt.

Mit einer ungeheuren Menge kleiner Programme in GFA-Basic 3.0 entwickeln sie die verschiedensten Grafiken und nutzen dabei wenige geometrische Grundformen und einfache Farbmanipulationen. Die beiliegende Diskette enthält alle Programme, hinzu kommt ein GFA-Run-Only Interpreter. Als größere Anwendungen liegen die Programme SCREENY und COLORIX vor.

Nach einem kurzen Kapitel über Grundlagen wie Koordinatensysteme beginnen die Autoren ihre grafischen Experimente mit „Quadrat-Grafik". Sie verwenden dabei als Grundform ein einfaches Quadrat, das dann allerdings gedreht, in der Größe verändert, gefüllt und bewegt wird. Zusammen mit geeigneten Zeichenstilen ergeben sich ungewöhnliche Grafiken, wie Bild 1 zeigt.

Es geht weiter mit einem Kapitel über Farbeinsatz und einer Fülle von Variationen der Grundform „Bogenraute", einer auf Sinus und Cosinus basierenden eingedellten Raute. Ein Kapitel über weitere Formen der Liniengrafik beschreibt beispielweise Lissajous-Figuren, „Kippschleifen" (Bild 2) und die fraktale Drachenkurve.

Unter „Spurengrafik" verstehen die Autoren das Resultat der Bewegung eines Symbols bei Verwendung der XOR-Verknüpfung.

Zum Experimentieren dient das Programm SCREENY, das einige der in den vorhergehenden Kapiteln entwickelte Grafiken direkt anbietet. Der Clou sind allerdings die vielfältigen Möglichkeiten, diese und andere Grafiken zu manipulieren. Die Funktionen beinhalten beispielsweise Ausdünnen, Outlining und verschiedenste und einmalige Füllfunktionen. Die Blockoperationen beinhalten natürlich auch das Rotieren.

Man verliert sich schnell in SCREENY und kann sicherlich einige Zeit mit Experimenten verbringen. Es ist keineswegs zu verwechseln mit Bildbearbeitungsprogrammen - SCREENY soll spielerische Kreativität unterstützen.

COLORIX ist ein Programm zur Farbmanipulation, das verschiedene Effekte wie beispielsweise Farbverläufe und farbliche Kombinationen verschiedener Bilder bietet.

Das letzte Kapitel beschreibt eine interessante Möglichkeit, mit einem normalen Matrix-Drucker Farbgrafiken zu erzeugen. Dabei werden selbstgefertigte Farbpapiere und Farbseparation verwendet. Die auf den Farbtafeln in der Mitte des Buchs abgebildeten Resultate können überzeugen und wirken erheblich besser als Hardcopys aus einem Farbdrucker. Zu bemerken ist dennoch, daß das Verfahren aufwendig ist und keineswegs einen schnellen Ausdruck ersetzen kann. Die Autoren bezeichnen das Resultat als „mit Pastellkreiden gedruckte Originalgrafik".

Das Buch wird abgeschlossen mit einem immerhin 65-seitigen Listing der beiden großen Programme, einem Disketten- und Abbildungsverzeichnis und sowie einem kleinen Register.

Das reichlich mit Abbildungen versehene Buch liest sich sehr angenehm und fordert immer wieder zu eigenen Variationen auf. Die GFA-Programme sind überraschend kurz und auch für BASIC-Anfänger nachzuvollziehen. An geeigneten Stellen erläutern die Autoren die notwendigen Computer-Begriffe, wie beispielsweise die Farbpalette des ST.

Etwas erstaunlich für einen großen Verlag wie Markt & Technik ist, daß in unserem Rezensionsexemplar ein Druckbogen - also 32 Seiten - kopfstehend eingeheftet ist. Die schönen Farbtafeln entschädigen allerdings für diesen herstellungstechnischen Fehler.

„Kreative Grafik auf dem Atari ST" ist ein sicher ungewöhnliches Computerbuch. Es verleitet zum Spielen und Experimentieren und beweist nachhaltig, daß Computergrafik keineswegs nur aus der schillernden Nachbildung der Realität und Spiegelkugeln besteht. Wessen ästhetisches Empfinden über Einrücktiefen in C-Programmen hinausgeht wird an dem Titel lange Freude haben.

RT

SALIX-Prolog für den ST

Leistungsdaten:
- Compiler/Interpreter System mit einer Geschwindigkeit von 1200 Lips
- Edinburgh-Standard mit zusätzlich eingebauten Funktionen (ca.160)
- Integer- und Gleitkommaarithmetik
- Syntax Error-Corrector
- Lector. Ein in Prolog geschriebenes Hilfmittel um Tippfehler und falsche Verwendung zu prüfen
- Anschluß an vorhandenen Editor z.B. 1ST-Word möglich. Prolog muß nicht verlassen werden um eine Bibliothek zu editieren
- GEM-Top-Level Interpreter, dadurch komfortable Bedienung mit Drop-Down Menüs und Formularen
- Startup-File um das System nach den jeweiligen Anforderungen anzupassen.
- eigene Exceptionbehandlung, dadurch mehr Sicherheit
- Datenbankeditor
- Zyklische Strukturen werden verarbeitet
- Leistungsfähiges Testsystem
- Exception Handling
- Benutzerdefinierte Funktionen
- Globale Variable
- Clipboard Device
- sehr schnell durch clause indexing
- großer Komfort durch programmierbare Funktionstasten

Normalversion **DM 198,--**

Salix-Prolog Professionell
wie Salix-Prolog Normalversion plus:
- Modulkonzept (Schnellden fertiger Module, Definition abgeschlossener Module mit Namenräumen vermeidet die Gefahr von Namenskonflikten und erlaubt die Teamarbeit bei größeren Projekten
- C/ASSEMBLERschnittstelle
C und Assenbler Programme könne als benutzerdefinierte Primitives definiert und eingebunden werden. Beispielprogramme sind enthalten
- TOS und GEM Oberfläche
- kostenlose Runtime Lizensen

Prolog Prof. **DM 498,--**

Aufstieg von Normalversion auf Professionell DM 300,–

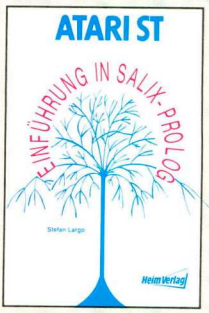
Heim-Verlag

Heidelberger-Land-Str.194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57-58
Telefax 0 61 51 / 5 60 59

Preise sind unverbindlich empfohlene Verkaufspreise

Benutzen Sie bitte die in der Zeitschrift eingefügte Bestellkarte oder rufen Sie uns an

Das Buch zu Salix-Prolog



Das Fachbuch für den Salix-Prolog Besitzer und jeden der sich für Künstliche Intelligenz interessiert. Ziel des Buches ist es den Umsteiger oder Neueinsteiger fundierte Grundkenntnisse in der Prolog Programmierung zu vermitteln. Teil 1 erklärt die Installation, Teil 2 führt in die Programiertechniken ein.

Hardcover, über 400 S.
Bestell Nr. B-448
ISBN 3-923250-

DM 49,--

MegaFakt

Sonst nichts.

"Eines der besten Fakturierungsprogramme für den Atari ST/TT".

novocom
hardware software

novocom • rathausstr. 1–3 • d–5650 solingen
telefon 0212–45888 • fax 0212–47399

ST PROFI-PARTNER

Mönkhofer Weg 126 2400 Lübeck Tel.: 0451 / 505 367 Fax: 0451 / 505 531

in Zusammenarbeit mit
DEIKE PRESS INTERNATIONAL

DPI vektor
Vektor-Grafikserie
Neu!

Die Neue Dimension in der Welt der
Grafik

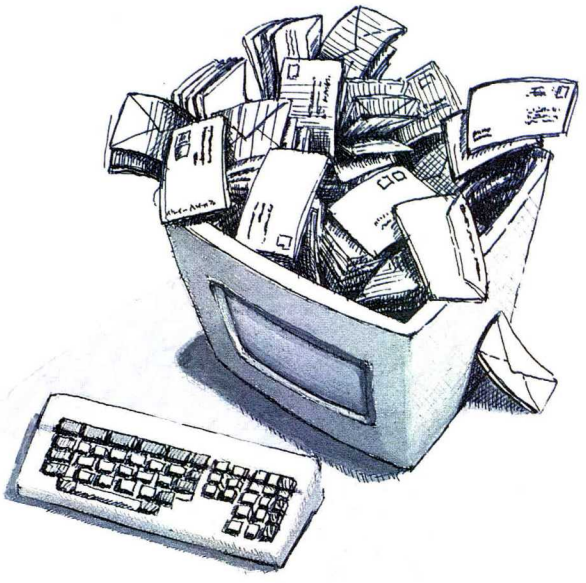
Infomaterial und Übersichten kostenlos !

- *jeden Monat neu*
- *Themen- und saisonorientiert*
- *6 Themenausgaben im Jahr zusätzlich*
- *Professionelle Qualität*



- *Top Illustrationen für DTP- und Grafikprogramme*
- *Alle Saison-Themen werden 2 Monate im Vorlauf geliefert*
- *Superpreis*

abgespeichert im Calamus .CVG-Format
oder im GEM/3-Metafile-Format



Ein Wort in eigener Sache

In den Jahren, die unsere Zeitschrift existiert, haben wir immer wieder versucht, durch die Beantwortung der bei uns eingehenden Briefe ein wenig Licht in das Dunkel zu bringen, das bei der Arbeit mit dem ATARI ST schon so manch einen aus der Fassung bringen konnte - eine Tatsache, die nicht nur Ihnen, verehrter Leser, sondern auch uns oft genug zu schaffen machte. Nichtsdestotrotz haben wir uns bemüht, die Probleme zu lösen und diverse Leserbriefe zu veröffentlichen, da wir der Meinung waren, daß die jeweilige Thematik auch einen größeren Leserkreis interessieren könnte. Trotzdem gibt es immer wieder Briefe, die wir nicht beantworten können oder dürfen. Damit Sie nicht allzusehr enttäuscht zu sein brauchen oder keine Antwort erhalten, möchten wir Sie bitten, sich an folgende Spielregeln zu halten, die sich aus unserer Erfahrung ergeben haben. Fällt Ihr Brief nicht unter die folgenden Kriterien, hat er gute Chancen, positiv beantwortet oder wenigstens als Hilferuf an unsere Leserschaft gedruckt zu werden.

1. Leider gehen immer wieder Briefe mit dem Wunsch ein, ein Produkt für diesen oder jenen Anwendungsfall vorzuschlagen, verschiedene Produkte bezüglich der Vor- und Nachteile gegeneinander abzuwägen und zu bewerten. Es ist uns aus Wettbewerbsgründen nicht erlaubt, ein bestimmtes Produkt zu favorisieren, selbst wenn wir das eine oder andere in der Redaktion überzeugt einsetzen. Wir können Sie in diesem Fall ausschließlich auf die von uns möglichst objektiven Tests und eventuell anstehende Fachmessen hinweisen. Bedenken Sie bitte, daß auch wir nicht jede Textverarbeitung, jedes Malprogramm und so weiter kennen und bestimmte Produkte dadurch in das Abseits drängen würden.

2. Oft erreichen uns Briefe, die sich positiv oder auch negativ über bestimmte Händler, Softwarehäuser oder deren Produkte auslassen. Sicherlich interessieren uns solche Bemerkungen. Bitte haben Sie aber Verständnis, daß wir weder Lob noch Tadel abdrucken dürfen, da diese Aussagen meist subjektiv sind. Anders sieht die Sache beispielsweise bei Gerichtsurteilen aus, die Sie, verehrte(r) Leser(in), erfochten haben.

3. Aufgrund der Vielzahl an Briefen, die uns täglich erreichen, sind wir leider nicht in der Lage, Programmfehler anhand von Listings oder ähnlichem zu korrigieren. Dennoch sollte ein Problem möglichst detailliert beschrieben sein, denn Ferndiagnosen sind prinzipiell sehr schwer, jedoch mit genauerer Angabe der Symptome eventuell durchführbar.

4. Von Zeit zu Zeit erreichen uns Briefe mit der Bitte, die Adresse des Lesers zwecks allgemeiner Kontaktaufnahme zu veröffentlichen. Würden wir dies in die Tat umsetzen, würde sich der Umfang des anderen redaktionellen Teils beträchtlich verkleinern. Ausnahmen stellen Leser in fernen Ländern dar, für die eine Kontaktaufnahme im eigenen Land recht schwierig ist.

Zum Schluß sollen ein paar Tips eventuell voreilig geschriebene Briefe verhindern.

1. Wenn Sie ein Problem bezüglich einer bestimmten Problematik haben oder an einem bestimmten Produkt interessiert sind, finden Sie interessante Artikel darüber eventuell in vorhergehenden Ausgaben unserer Zeitschrift. Zur Auswahl eignet sich das Jahresinhaltsverzeichnis besonders gut, das immer am Jahresende in der ST Computer abgedruckt wird.

2. Sollten die Probleme mit der Handhabung eines Produktes zu tun haben, wenden Sie sich zunächst an Ihren Händler und über diesen an den Distributor beziehungsweise an das Software-Haus. Die Wahrscheinlichkeit, daß Ihnen das Software-Haus weiterhelfen kann, ist um ein Vielfaches höher als die, daß wir Ihnen helfen können.

3. Lesen Sie aufmerksam die Leserbrief-Seite. Viele Fragen wiederholen sich immer wieder, obwohl wir bestimmte Probleme schon mehrfach angesprochen haben.

Booten stoppen

Bei TOS 1.0 konnte man bei angeschlossener Festplatte via Diskette booten, indem man die Alternate-Taste beim Boot-Vorgang festhielt. Wie kann ich bei TOS 1.4 vom 6.4.89 von Diskette booten, ohne die Festplatte auszuschalten? Die Version von AHDI ist 3.01 vom 16.5.89. Das TOS 1.4 ist mit den Patches aus ST-Computer 1/90 modifiziert.

Dietrich Carsten, W-8416 Hanau

Red.: Nichts einfacher als das. Während des Boot-Vorgangs halten Sie Alternate gedrückt und spielen ein wenig mit der Control-Taste herum. Das Booten von Diskette funktioniert nicht mehr, wenn Sie nur die Alternate-Taste festhalten, sondern Sie müssen zusätzlich noch einige Male auf Control drücken.

*

Kopf defekt

Ich habe ein Problem mit meinem 1040 ST: Der obere Schreib- und Lesekopf ist unwiederbringlich defekt. Meine Diskstation funktioniert wie ein SF354. Mit viel Mühe habe ich versucht, einen kompletten Kopf zu bekommen und ihn einzubauen, doch vergeblich. Ich selbst bin Techniker und habe damit keine Probleme. Für ein Austauschlaufwerk möchte ich einfach nicht investieren! Was ist zu tun?

Bernhard Knöfel, W-7516 Karsbad 1

Red.: Einen passenden Kopf erhalten Sie entweder bei jedem Atari-Vertragshändler oder direkt im gutsortierten Elektronikhandel. Allerdings ist die Justierung des Schreib-/Lesekopfes nur dann ordnungsgemäß durchzuführen, wenn Sie über entsprechende Gerätschaften verfügen. Der Kopf läßt sich zwar auch „einfach so" justieren, dann könnten Sie aber eventuell Probleme mit fremden Disketten bekommen. Hier stellt sich schon die Frage, ob der Einbau eines neuen Kopfes, der recht teuer ist, noch wirtschaftlich ist!

*

Formatfragen

Ich bin seit kurzer Zeit im Besitz der Megafile 60. Nachdem diese Platte in vier logische Laufwerke aufgeteilt war, wollte ich versuchen, eines der Laufwerke für MS-DOS-Anwendungen zu reservieren. In diesem Zusammenhang habe ich einige Fragen. Ist es grundsätzlich möglich, die Platte für beide Formate zu formatieren und beide Arten von Daten zu speichern? Wenn ja: Wie wird es bewerkstelligt bzw. wie öffne ich das Laufwerk; muß es gesondert formatiert werden?

Heinrich Velke, W-8504 Stein

Red.: Die Frage ist nicht mit einem klaren 'Ja' oder 'Nein' zu beantworten, sondern mit 'Jein'. Die Art der Formatierung ist von Emulator zu Emulator unterschiedlich. Manchmal können nur bestimmte Partitionen benutzt werden, andere Emulatoren geben sich mit dem Atari-Format zufrieden, wieder andere müssen eine komplette, speziell formatierte Partition zugewiesen bekommen. Am besten können Sie das in der Anleitung Ihres Emulators nachlesen.

*

Strickmaschine

Vor etwa einem Jahr kaufte ich mir eine vollelektronische Strickmaschine Brother KH-940. An der Rückseite des Geräts befindet sich eine Anschlußbuchse zur Datenübertragung, welche ursprünglicherweise zur Kommunikati-

on mit einem Floppylaufwerk von Brother bzw. mit einem von Brother entwickelten Minicomputer gedacht ist. Die Schnittstelle läßt sich aber auch über ein spezielles Interface-Kabel an einem RS-232C-Schnittstelle anschließen. Für den ST gibt es ein Programm zur bildschirmgestützten Strickmusterentwicklung. Leider leistet mir das Programm zu wenig. Um meinen Wunsch nach „gestrickten Computergrafiken" erfüllen zu können, fehlt mir also einzig und allein das Interface-Kabel zwischen dem ST und der KH-940. Weder Brother noch der Hersteller des Kabels konnten mir jedoch Informationen über die Schnittstelle geben.

Volker Abel, W-6101 Roßdorf

Red.: Der Anschluß dürfte sich als relativ unkompliziert herausstellen. Die benötigten Schaltpläne, die wir natürlich aus Urheberrechtsgründen hier nicht abdrucken können, sind beim technischen Brother-Service unter der Rufnummer (06151) 26775 erhältlich.

*

Emulatoren

Aladin, Spectre, PC-/AT-Speed, PC-Ditto, ATonce, Delta-Modul - ein Emulator jagt den anderen und verspricht uns die seligmachende Kompatibilität zum Äpfelchen oder zur Big Blue Mother. Und ganz begeistert über diesen ungemeinen Zuwachs an Professionalität stürzt der Atarianer mit der Atarianerin zum nächsten Händler und läßt sein neues Spielzeug und den Traum vom geeinten Computer-Paradies einbauen. Doch spätestens zwei Wochen später ist der Traum kaputt, der Wert des Spielzeugs fragwürdig. Eigentlich hatte man sich doch den ST gekauft, und nun kann man seine mühsam geschriebenen Texte und Grafiken nicht mal auf dem Firmencomputer weiterverwenden. Also alles nochmal eingeben und in Zukunft nur noch im IBM- oder MAC-Modus arbeiten? Warum hat sich der arme Tropf denn dann überhaupt einen ST gekauft (wobei der auch nicht mehr so billig ist, wie er - im Vergleich - mal war)? Das kann doch wohl nicht mit Kompatibilität gemeint sein? Die Lösung? Wo bleibt das intelligente Laufwerk, das TOS-, DOS-, Mac- und Amiga-Disketten selbständig erkennt und sowohl lesen als auch beschreiben kann? Wo bleibt die nicht minder intelligente Software, die Daten (Text und Grafik) aus einem Format in ein anderes und zurückübersetzt, wobei zumindest einfache Attribute wie fett, hochstellen, Sonderzeichen usw. erhalten bleiben (nötigenfalls natürlich auch ohne)?

Markus Winter, W-8700 Würzburg

Red.: Es ist sicherlich schade, daß der Austausch zwischen den Emulatoren und den Originalgeräten nicht ganz so einfach ist, doch problematisch ist es nun auch wieder nicht. DOS-Disketten sind beispielsweise dann vom ST und dem DOS-Rechner lesbar, wenn sie unter DOS formatiert wurden. Auf dem MAC können Sie eine ST-Diskette dann lesen, wenn Sie sich ein entsprechendes Laufwerk in den MAC einbauen und dort 720 kB DOS formatieren. Der Austausch von Grafiken stellt ebenfalls kein Problem dar, wenn Sie die entsprechenden Konvertierungsprogramme besitzen. Für den MAC können Sie beispielsweise Ihre PIC-Dateien umwandeln in TIFF, für den Austausch zwischen ST und PC gibt's etwa das GIF-Format (eines von vielen), das beide Rechner verstehen. Texte auf den MAC zu transportieren ist kein Problem mit dem Programm „Dateien konvertieren", mit dem Sie übrigens auch Grafiken transportiert bekommen. Auf dem PC ist es noch einfacher, wenn Sie 1st_Word PC benutzen, denn dann bleiben tatsächlich alle Formatinformationen erhalten! Mit anderen Textverarbeitungsprogrammen gibt es allerdings Schwierigkeiten beim Austausch der entsprechenden Steuerzeichen, da diese nicht genormt sind. Natürlich gibt es kein Standardprogramm für alle Anwendungen, und das ist auch gut so, denn für ein Programm, das alles kann, bräuchten Sie einen Arbeitsspeicher von mehreren MB...

*

Jahreszahlen

Man hört oft in der Werbung: „Was auch immer in den 90er Jahren geschieht...". Doch das ließ mich auf die Frage kommen, was danach passiert. Kommen die heute gebräuchlichen Software-Pakete mit einem vierstelligen Jahreswechsel zurecht, oder werden wir dann einen Versicherungsbescheid aus dem Jahr 1900 bekommen, weil die Software bzw. Hardware der Versicherung XY nicht auf ein solches Ereignis vorbereitet war? Wie sieht dieses Problem auf dem Atari TT und in der MS-DOS-Welt aus? Ein entsprechender Software-/Hardware-Überblick unter diesem Aspekt würde mich sehr interessieren.

Detlef Jockheck, W-4994 Pr.Oldendorf

Red.: Professionelle Software benutzt meistens die im ST eingebaute Uhr, um sich zu orientieren. Dadurch ist die Zeiteingabe bereits begrenzt auf den folgenden Zeitraum bis zum 31.12.2107 - weiter rechnet weder die Uhr des ST noch die des TT. Das ist zwar schade, doch es ist kaum anzunehmen, daß Sie im Jahre 2107 noch den ST/TT benutzen - zur Not wird auch ein Patch helfen, es gibt ja bereits genügend. Ein wesentlich schwerwiegenderes Problem dürfte jedoch mit der Umstellung der Postleitzahlen von 4 auf 5 Ziffern anstehen, die durch die Eingliederung der Ex-DDR notwendig geworden ist. Hier werden die Programme beweisen müssen, ob sie diesem Novum gewachsen sind.

*

Intensitätsverlust

Ist es möglich, daß ein SM124 nach drei Jahren an Intensität verliert (Nachlassen der Helligkeit), oder hat dies andere Gründe (z.B. Nachlassen eines Steuersignals)?

Hartmut Broich, W-8000 München

Red.: Der SM124 läßt, wie jedes andere (Daten-)Röhrengerät auch, an Kontraststärke nach. Das ist durch die Funktionsweise solcher Geräte bedingt und läßt sich nicht ändern. Der Zeitpunkt des Nachlassens hängt stark von der Beanspruchung des Geräts ab und kann deshalb schwer vorausgesagt werden. Sollte Ihnen der Kontrast jedoch zu schwach sein, können Sie durch Vorsetzen einer Kontrastscheibe (keine Folie!) noch recht gute Ergebnisse erzielen.

*

Editor

Ich suche zum Betrieb mit Aditalk einen ASCII-Texteditor unter TOS. Wordplus eignet sich zwar, ist aber mit zirka 160 kB sehr langsam zu starten, überdies ist der Betrieb nicht sehr komfortabel, da die Mausanzeige von Wordplus unter Aditalk unterdrückt wird.

Martin Schwaar, CH-4663 Aarburg

Red.: Auf der PD-Diskette 55 befindet sich ein passender Editor, mit dem Sie schnell und unkompliziert Texte bearbeiten können.

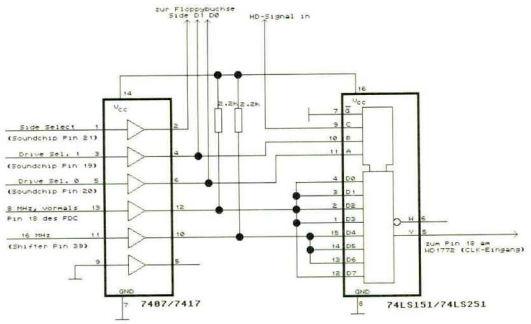


Bild 1: Die neue HD-Schaltung mit dem gewissen Etwas

KLI-KLA-KLEISTER

Sie haben Ihren Rechner beim Einbau der Modifikation für HD-Laufwerke [1] sauber abgekocht? Nein? Dann erhalten Sie diesen Monat weitere Gelegenheit, Ihre Zerstörungslust an Ihrem ST auszuleben: Joachim Bohs hat sich eine Variante der HD-Schaltung einfallen lassen, die einige Vorteile gegenüber der bisherigen Lösung hat.

Die Schaltung aus [1] tut zwar das Gewünschte, kann aber - wie alles Gute - noch verbessert werden. Zum Beispiel könnte man versuchen, mit noch weniger Bausteinen und Verdrahtungen auszukommen, oder aber man integriert weitere Funktionen, wenn man ohnehin schon mal seinem Floppycontroller mit dem Lötkolben zu Leibe rückt. Letzteres ist Ziel der hier vorgestellten Schaltung.

Zum Betrieb von HD-Laufwerken am ST ist nur wenig nötig:

- ein Floppycontroller, der 16 MHz Takt verträgt (wie wir inzwischen wissen, geht das zuverlässig nur mit dem Typ WD1772-02 und wahrscheinlich auch mit dessen Lizenznachbau von VLSI)
- 16 MHz vom Shifter oder von einem externen Quarzbaustein
- eine kleine Schaltung, die mittels des HD-Signals, das die meisten Diskettenlaufwerke hoher Schreibdichte ausgeben können, entscheidet, welcher Takt (8 oder 16 MHz) dem Floppycontroller zugeführt wird.

In Bild 1 finden Sie den neuen Schaltungsvorschlag. Er basiert auf den beiden Bausteinen 74LS151 (oder 74LS251) und 7407 (oder 7417). Der 7407-Baustein sorgt dafür, daß beide Taktsignale gepuffert werden; somit sollte es nun in den meisten Fällen möglich sein, den 16-MHz-Takt direkt vom Pin 39 des Shifters abzunehmen. Wer mag, kann natürlich trotzdem einen externen Takt zuführen, sollte daraus aber am besten auch gleich den 8-MHz-Takt durch einen Teiler gewinnen. Insbesondere sollten Sie dann auf einen externen Taktgeber umsteigen, wenn Sie beim Zugriff auf die Diskettenlaufwerke plötzlich Bildstörungen (leichtes Flimmern) bemerken.

Der 7407-Baustein puffert aber auch die Signale Side Select, Drive Select 0 und Drive Select 1, die vom Soundchip kommen. Das ist nämlich auf jeden Fall eine gute Idee, wenn Sie Fremdlaufwerke anschließen. Man weiß ja, was man dem bisweilen mimosenhaften Soundchip schuldig ist.

Ein weiterer Vorteil dieser Schaltung: Sie kann relativ einfach umkonfiguriert werden, sobald man die Ausrüstung wechselt - beispielsweise, wenn man den Rechner an einen anderen Standort bewegt oder aber das externe Laufwerk seinen Geist aufgibt und dergleichen Unbill mehr geschieht. Dazu manipuliert man die Beschaltung der Signale D0 bis D7 am 74LS151:

- 2*DD: HD-Eingang (in der Zeichnung Eingang C des 74LS151, also Pin 9) auf Masse und D0 bis D7 am 74LS151 auf 8 MHz legen
- 2*HD: HD-Eingang mit Laufwerken verbinden, D0 bis D3 auf 8 MHz, D4 bis D7 auf 16 MHz legen
- 1*DD als Laufwerk A, 1*HD als Laufwerk B: Wie zuvor, nur zusätzlich D6 auf 8 Mhz legen
- 1*HD als Laufwerk A, 1*DD als Laufwerk B: Wie 2*HD, nur zusätzlich D5 auf 8 MHz legen

Mehr zum HD-Umbau finden Sie in [1] und [2].

CBHD vs. Rest der Welt

Im allgemeinen darf ich mir auf die Schulter klopfen: Mein Plattentreiber CBHD verträgt sich mit einer weiten Palette von Hard- und Software. Im besonderen allerdings schwitzt der Autor dieser Zeilen aber immer wieder über seinem Plattentreiber, um dessen Zusammenspiel mit dieser und jener Neuheit abzuklopfen. Um auch unseren Posteingang etwas zu entlasten, gibt es diesmal zwei Tips für den Umgang mit dem CBHD:

- CBHD 3.5 vs. TUNEUP und FCOPY III: TUNEUP meldet am Ende des Optimierungsvorganges, es könne den Bootsektor nicht schreiben. Außer der Fehlermeldung hat dies allerdings keine weiteren Folgen - wenn ich mich auch frage, was TUNEUP im Bootsektor einer Plattenpartition zu suchen hat. In FCOPY III funktioniert das Restaurieren einer Partition nicht, wenn CBHD 3.5 installiert ist. Ursache beider Verwirrungen ist, daß CBHD V3.5 sich hier an das Verhalten des Disketten-BIOS anlehnt: Das meldet nämlich vorsorglich einen Diskettenwechsel, wenn der Bootsektor einer Diskette beschrieben wird - was ja auch sinnvoll ist, weil sich beim Beschreiben alle Formatparameter im Bootsektor geändert haben könnten. Da das aber offensichtlich ein paar Programme nicht mögen, wird dieses Verhalten in der nächsten Version des CBHD abschaltbar sein. Wer mag, kann von mir auch eine gepatchte CBHD-Version haben. (Original-KLEISTER-SCHEIBE und frankierten Rückumschlag nicht vergessen!)

- CBHD vs. STAcy: Der Controller der in der STacy eingebauten Platte stürzt ab, wenn man ihn nach einem zweiten angeschlossenen Laufwerk fragt - genau das tut aber CBHD beim Booten. Um diesen Fehler des Controllers zu umgehen, gibt es, wie bereits im STacy-Test der ST-Computer [3] erwähnt, eine angepaßte STacy-Version des CBHD, der man untersagen kann, sich nach einem zweiten Laufwerk zu erkundigen. Auch andere Festplattentreiber haben übrigens so ihre Schwierigkeiten mit der STacy.

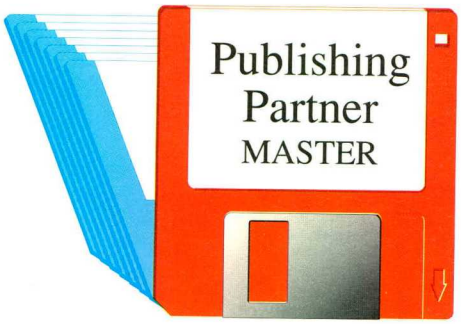
Die Firma FSE aus Kaiserslautern weist in der Gebrauchsanleitung, die zu deren Platten ausgeliefert wird, darauf hin, daß CBHD bis dato nicht sicher sei und auf ihren Platten zum Datenverlust führen könne. Daß mich das aufgeschreckt und geärgert hat, werden Sie verstehen. Nun hilft mein gekränkter Programmierer-stolz aber niemandem, und so habe ich selbst eine FSE-Platte zwei Monate im Dauereinsatz mit CBHD getestet - keine Fehler.

Bei einer kleinen Umfrage unter Bekannten und Lesern fand sich ebenfalls kein Hinweis auf lauernde Gefahren. Kein Leser hat mir bisher von Unverträglichkeiten des CBHD mit Platten dieser Marke berichtet, daher können wir im Moment wenig tun. Wir bitten Sie, verehrte Leser und Leserinnen, deshalb darum, uns von Schwierigkeiten, die Sie mit dem CBHD und irgendwelchen Platten (v.a. natürlich Platten der erwähnten Firma) haben oder hatten, zu berichten. Nur so können wir überhaupt herausfinden, ob an Behauptungen wie der zitierten etwas dran ist.

Zum Schluß wie immer unsere Kontaktadresse:

Claus Brod
Am Felsenkeller 2
8772 Marktheidenfeld

Literatur:
[1] Bohs/Brod/Stepper:
„Ganz schön dicht".
ST-Computer 1/90
[2] Bohs/Brod/Stepper:
„Ganz schön dicht - die Software".
ST-Computer 2/90
[3] Schormann/Brod:
„Man geht nicht ohne STacy".
ST-Computer 1/90



Publishing Partner Master (P.P.M.) stellt ein mächtiges und komplexes Werkzeug zur Gestaltung von Text und Grafik dar. Ein Programm mit einer Vielzahl von Hilfsmitteln zur Erstellung von Anzeigen oder Plakaten, aber auch von mehrseitigen Zeitungen oder Büchern. P.P.M. bietet neben seiner Funktionsvielfalt auch einige völlig neue Konzepte, wobei auf eine einfache und durchsichtige Bedienerführung Wert gelegt wurde.

Import
ASCII-Texte oder formatierte Texte werden direkt aus Textverarbeitungsprogrammen unter Beibehaltung ihrer Textattribute übernommen (z.B. aus CTX, That's Write, First Word oder Word Perfekt). Grafiken werden in den Formaten Neochrome, DEGAS, IFF, IBM, Metafile, Tiny, IMG, Arabesque, STAD, AegisDraw Plus, EPSF (Encapsulated PostScript Files) importiert.

Dokumente
Vollständige Implementierung des GEM-Window-Systems mit bis zu sechs gleichzeitig geöffneten Textfenstern einschließlich der Möglichkeiten der Kopier- und Einfügefunktion zwischen Texten, der Kombination von Texten mit verschiedenen Formaten, Makro-Instruktionen und Tastaturbenutzung für alle Funktionen.

Seiten-Format
Elf vordefinierte und ein vom Benutzer zu bestimmendes Format. Vertikale und horizontale Seitenausrichung. Gegenüberliegende Seiten. Seitennummerierung arabisch oder römisch. Links-rechts unabhängige Master-Seiten. Einfügen und Löschen von mehreren Seiten. „Gehe zu Seite..."-Anweisung. Gruppieren von Objekten mit Transfer zu einer beliebigen Seite.

Seiten-Visualisierung
Gleichzeitige Anzeige von zwei aufeinander folgenden oder nicht folgenden Seiten, sechs zusätzliche, vordefinierte Anzeigeformate und ein benutzerformuliertes Format (15 bis 1500 Prozent). Variables Zoom. Benutzerdefinierbares Gitter. Magnetische Ausrichtung an Führungs- und Gitterlinien. Verschiedene Maßeinheiten (Inch, Pica, US-Point, Cicero, Didot-Punkte, metrische Punkte, Zentimeter und Milimeter).

Seiten-Layout und Textbearbeitung
Grundfunktionen der Textverarbeitung sind kombiniert mit internationalen Befehlen zum Setzen von hoch- und tiefgestellten Zeichen, Linien und Buchstabenzwischenräumen, sowie zum Einrücken von Absätzen. Weitere Funktionen sind Suchen und Ersetzen, automatische und manuelle Zentrierung, Fließtext um ein Objekt oder eine Stilvorlage (Formatschablonen).

Textmanagement
Kontrolle über Textobjekte. Automatischer und manueller Spaltensatz (bis zu 512 Spalten pro Seite!). Frei definierbare Spaltentextanbindung über mehrere Seiten. Tabulation für jede Spalte frei wählbar. Deutsche, englische und französische Silbentrennung.

Grafik-Management
Kompletter Grafik-Editor mit den Funktionen: veränderbare Polygone, Ellipsen, Rechtecke, Kreise, Linien, Rahmen, Rahmen mit abgerundeten Ecken, Kreisbogen, elliptische Bogen. Freihandzeichnen. Wählbare Linienstärke (von 2 bis 1360 Punkt), verschiedene Linientypen und Füllmuster.

Farben
Definition der Farbpaletten (Primärfarben oder gemischte Farben). Wahl des Farbsystems (RGB, CMY, HLV, HLS, YIQ).

Manipulieren von Text und Graphik
Gruppieren und Trennen von Objekten, automatische Objektausrichtung. Einfügen und Löschen von Objekten von oder zu Seiten. Dreidimensionale Drehung von Texten und Objekten in 1-Grad Schritten.

Typographie
Zahlreiche Zeichensätze. Kombinierte Effekte (Schrift linksgeneigt, fett, aufgehellt, gespiegelt, kursiv, negativ, schattiert, unterstrichen, doppelt unterstrichen, durchgestrichen, Konturschrift). Frei wählbare Zeichengröße und -abstand. Genauigkeit von 2 bis 1360 Punkt. Automatische und manuelle Abstandsregelung mit Darstellung pro Buchstabenpaar.

Ausdruck
Vergrößerung oder Verkleinerung des Dokumentes (15 bis 999 Prozent). Überlappungsfunktion erlaubt Ausdruck im Posterformat über mehrere Normalseiten. Gesamtanzeige Negativ- und Spiegeldruck (Overhead-Transparente, Offset). Farbtrennung auf mehrere Seiten. Laden von nichtresidenten Zeichensätzen. P.P.M. unterstützt alle PostScript- und kompatible Systeme (S/W, Farb-Laserdrucker und hochauflösende Fotosatzmaschinen) sowie den Atari SLM 804 Laserdrucker, den HP LaserJet und kompatible, Epson kompatible Matrixdrucker NEC-Drucker und andere S/W- und Farbdrucker.

DM 798,-
Unverbindlich empfohlener Verkaufspreis

P.P.M.

Die genannten Firmen- und Produktnamen sind Warenzeichen der jeweiligen Inhaber

Hotline
Info + Betreuung
COMPO SOFTWARE
GmbH
Tel. 06551-6266

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

BESTELL-COUPON

Bitte senden Sie mir: ___ Publishing Partner Master à 798,- DM

Name, Vorname _____
Straße, Hausnr. _____
PLZ, Ort _____
(Oder benutzen Sie die eingeheftete Bestellkarte)

In Österreich:
RRR EDV GmbH
Dr. Stumpfstraße 118
A-6020 Innsbruck

In der Schweiz:
Data Trade AG
Landstraße 1
CH - 5415 Rieden-Baden

DUFFNER COMPUTER
Habsburgerstr. 43
7800 Freiburg
Tel: 0761/56433
FAX: 0761/551724
ATARI in Freiburg

ARTWORKS 398.–
Das professionelle DTP–Gestaltungspaket

FONTS
ARTWORKS Designer Fonts – bei uns zu haben

LogiMouse Pilot 89.–
Der Präzisions–Mäuserich

Wir suchen noch Autoren wie Sie.

Haben Sie eine gute Programmidee und wollen ein Buch schreiben und mitgestalten. Kennen Sie eine Menge Tips und Tricks. Möchten Sie Ihre Erfahrungen weitergeben.

Wir bieten Ihnen unsere Erfahrung und unterstützen Ihre Ideen. Als leistungsstarker Verlag freuen wir uns bald von Ihnen zu hören.

Heim Verlag Kennwort: Autor Heidelberger Landstr. 194 6100 Da.-Eberstadt Tel.: 06151/56057

ST-Fibu

Das Programmpaket
für Ihr Büro auf
ATARI ST/TT

FIBU

- ST-Fibu- komplette Finanzbuchhaltung mit Offener Postenverwaltung ab DM 348.00
- ST-Fibu-Mini-Version ab DM 148.00
- GMa-Text-Textverarbeitung mit Mahnwesen/Serienbrief * ab DM 149.00
- ST-Fakt-Fakturierung * ab DM 199.00
- ST-Inven-Inventarverwaltung ab DM 79.00

Neu: ST-Giro- Abwicklung des Zahlungsverkehrs * ab DM 99.00

* Programm mit Schnittstelle zur ST-Fibu

Demoversionen mit Handbuch DM 60.00
(wird beim Kauf verrechnet)
Kostenlose Info anfordern!

GMa / Soft
Gerd Matthäus
Betriebswirt

Bergstr. 18 - 6050 Offenbach
Tel. 069 / 89 83 45 - Fax 89 84 21

Public-Domain Software

Hier stellen wir Ihnen unsere neuesten ACTION - Pakete vor. Die Pakete werden alle drei Monate erneuert und wieder mit aktuellster PD-Soft bespielt!
Jedes ACTION - Paket besteht aus 3 doppelseitigen PD-Disketten.
Jedes Paket kostet nur DM 10,--

Ausgabe: 12/90 bis (einschl.) 2/91

GAS.-01/ *Monochromspiele - Action* (3 Disks, DM 10,--)
Darkness is for ever: ein Adventure, daß Sie viele Stunden in seinen Bann ziehen wird. **Rennhund:** sportliches Managerspiel für 1 bis 8 Teilnehmer. **Quizmaster 0.5:** nettes Ratespiel für 1 bis 2 Spieler. **Knack den Tresor:** sehr aufwendiges Spiel, bei dem Sie 40 alte Bilder erstehen müssen. **Chicago:** Glücksspiel. **Play Me:** sehr schöne Tetris-Variante. **Hinem:** Memoryspiel mit 198 Karten. **Cave Miner II:** super Boulder-Dash-Variante. **Fly A Jet:** Flugsimualtor einer Boing 727; mit vielen Funktionen. **Segelflug:** Segelflug-Simulator. **Memory:** "normale" Memory-Variante.

GAF.-02/ *Farbspiele - Action* (3 Disks, DM 10,--)
Persifal: spannendes Abenteuerspiel. **Megamix 1.2:** Memoryspiel. **TriDo 1.5:** super Geschicklichkeitsspiel. **Xenotron 2.3:** schnelles Weltraum-Ballerspiel. **Crisslefridge:** geniales Boulder-Dash Spiel mit super Grafik. **Two for Two:** Panzerspiel, das eine schnelle Reaktion und gute Nerven voraussetzt. **Double Bounce:** Tennisspiel für 2 Sportler. **Gilbert:** Memo-Variante. **Deathstar:** Kampf gegen Monster und vieles mehr. **Damonold:** abwechslungsreiches Ballerspiel mit digit. Stimmen. **Shymer:** Adventure.

ANW.-01/ *Anwender - Action* (3 Disks, DM 10,--)
Beropress 1.0: Druckt Poster, Grußkarten etc. Verarbeitet sogar Signum-Zeichensätze. **Formular 1.1:** Schreibprog. zum Ausfüllen von Formularen. **Stundenplan. Bildklau:** ladt Bilder ein und verarbeitet diese in Data-Zeilen für GfA-Basic. **Disk Manager:** archiviert Diskettenінhalte. **Houdini :** Tool für Signum. **Getfish - Tab:** verwaltet bel. Sportligen. **Adress 5.0:** Adreßverwaltung.

MUG.-01/ *Musik und Grafik - Action* (3 Disks, DM 10,--)
Van Gogh III: eines der besten Mal/Zeichenprogramme. **Click & Beep:** belegt die Tastatur mit digit. Sounds. **Butterfly Artist:** ausgefallenes Malprogramm. **Freedrum 2.5:** Sound/Trommelmaschine mit digit. Effekten. **ST-Design:** leistungsfähiges Zeichenprog.

CLP.-01/ *Clip Art - Action* (3 Disks, DM 10,--)
die besten Clips im "PAC"-Format (Stad).

SIG.-01/ *Signum Zeichensätze - Action* (3 Disks, DM 10,--)
ca. 30 verschiedene Signum (Script) Zeichensätze (für 9 und 24 Nadeldrucker), sowie verschiedene Tools.

Fordern Sie auch bitte unseren Gesamtkatalog (gedruckt) für DM 3,-- an.
Ach: Bei uns kostet jede PD-Disk nur DM 4,--

COMPUTER-SERVICE KOHLER
Don-Carlos-Str. 33B, D-7000 Stuttgart 80
Telefon: 07 11 / 6 78 73 92 (24h-Bestellservice)

Bei uns werben bringt

GEWINN



Sprechen Sie mit uns.

Heim Verlag
☎ 06151/56057

DISKETTEN-SERVICE

Absender bitte deutlich schreiben

Vorname / Name

Straße / Nr.

PLZ / Ort

Telefon

Postkarte

Bitte freimachen

MAXON Computer
Schwalbacher Straße 52

D-6236 Eschborn

ST COMPUTER

Kontaktkarte

ST COMPUTER

Kontaktkarte

Bitte Adresse der Firma, bei der Sie Informationen, oder etwas bestellen möchten, auf der rechten Seite eintragen →→→→→→→→→→

Absender
(Bitte deutlich schreiben)

Vorname/Name

Beruf

Straße/Nr.

PLZ/Ort

Telefon Vorwahl/Rufnummer

Postkarte

Bitte freimachen

Firma

Straße/Postfach

PLZ Ort

ST COMPUTER

ABO

ST COMPUTER

ABO

Absender = Rechnungsadresse
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Postkarte

Bitte freimachen

Heim Verlag

Heidelberger Landstr. 194

6100 Darmstadt 13

COMPUTER

ATARI

Public Domain Service

Bitte senden Sie mir folgende PD-Disketten / Sonderdisks

Datum / Unterschrift

Zahlung erfolgt:

___ Vorauskasse (per Scheck) ___ per Nachnahme

- Pro PD-Diskette entfällt ein Unkostenbeitrag von DM 10,-
- Preis der Sonderdisks je nach Angabe
- Bei Nachnahme zuzüglich DM 4,- Nachnahmegebühr
- Versandkosten (Porto und Verpackung) DM 5,- (Ausland DM 10,-)
- Auslandsbestellungen nur gegen Vorauskasse möglich

Ab 5 Disketten versandkostenfrei

SONDERDISK

Bitte beachten Sie, daß Sonderdisks ein Copyright besitzen und nicht weiterkopiert werden dürfen.

DISKETTEN-SERVICE

COMPUTER

☐ Ich bitte um weitere Informationen

☐ Ich gebe folgende Bestellung auf

in Bezug auf Ihre Anzeige in ST-Computer Heft _____ Seite ___

Menge	Produkt/Bestellnummer	DM	gesamt DM

Datum, Unterschrift (für Jugendliche unter 18 Jahren der Erziehungsberechtigte)

Abgesandt am:

COMPUTER

Firma:

Bemerkungen:

COMPUTER

Abonnement

Kd.-Nr.:

Ja, bitte senden Sie mir die ATARI-Computer Fachzeitschrift ab der nächsten erreichbaren Ausgabe für mindestens 1 Jahr (11 Hefte) zum ermäßigten Preis von jährlich DM 80,- frei Haus.
(Ausland: nur gegen Scheck-Voreinsendung DM 100,- Normalpost, DM 130,- Luftpost).
Der Bezugszeitraum verlängert sich nur dann um ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekündigt wird.

Geschenk- bzw. Lieferadresse:

Vorname

Name

Straße / Postfach

Land PLZ Ort

Gewünschte Zahlungsweise bitte ankreuzen

☐ Bequem und bargeldlos durch Bankeinzug

Konto-Nr. BLZ

Institut Ort

☐ Ein Verrechnungsscheck über DM ____ liegt bei.

☐ Vorauskasse per Zahlung auf unser Postscheck-Konto Ffm, BLZ 500 100 60, Kto.-Nr. 5537-602

Diese Vereinbarung kann ich innerhalb von 8 Tagen beim Heim-Verlag, Heidelberger Landstr. 194, 6100 Darmstadt-Eberstadt widerrufen. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs. Ich bestätige die Kenntnisnahme des Widerrufsrechts durch meine 2. Unterschrift.

Datum, 2. Unterschrift

COMPUTER

ABO

Diese Vereinbarung kann ich innerhalb von 8 Tagen beim He
Verlag, Heidelberger Landstr. 194, 6100 Darmstadt-Eberst
widerrufen. Zur Wahrung der Frist genügt die rechtzeit
Absendung des Widerrufs. Ich bestätige die Kenntnisnahme
Widerrufsrechts durch meine 2. Unterschrift.

PD NEWS

Public Domain ganz groß

Vor etwa vier Jahren begann die Redaktion der Zeitschrift ST Computer, Public Domain-Programme zu sammeln und in einer eigenen Serie der Öffentlichkeit allgemein zugänglich zu machen. Inzwischen ist die Sammlung auf fast 400 Disketten angewachsen und gehört zu den meistkopierten auf dem Atari ST. Mit Beginn des neuen Jahres wollen wir in der ST Computer nun auch mehr auf Public Domain-Programme eingehen. Da in der kurzen Übersicht am Heftende viele Programme nur wenig oder gar nicht berücksichtigt werden konnten, sollen an dieser Stelle künftig die interessantesten PD-Programme näher vorgestellt werden. Dabei liegt der Schwerpunkt sicherlich auf den Neuerscheinungen, aber auch ältere Programme sollen Berücksichtigung finden. Ferner wollen wir über Updates und Tricks informieren.

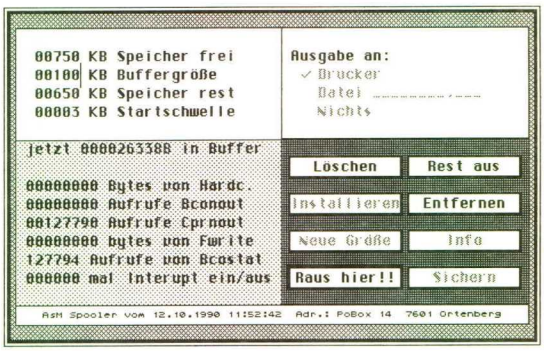
Drucken im Hintergrund

Was nützt einem ein mit 16, 20 oder gar 32 MHz laufender Superrechner, wenn beim Ausdrucken der Texte der angeschlossene Drucker mit 200 Zeichen/Sekunde im Schnelldruck oder 50 Zeichen/Sekunde bei Schönschrift das ganze System quasi zum Stillstand bringt? Will man gar eine hochauflösende Grafik ausdrucken, vervielfacht sich die Datenmenge, so daß man eine längere Kaffeepause - viel gesünder wäre natürlich eine Gymnastikpause - einlegen muß.

Wie kann man also Texte und Grafiken drucken und gleichzeitig einigermaßen unbehindert weiterarbeiten, lautet hier die große Frage. Die Lösung, die einem ein verkaufstüchtiger Händler aufdrängen würde, wäre eine Speichererweiterung für Ihren Drucker oder ein zwischengeschalteter RAM-Puffer. Beides kostet eine Menge Geld. Preiswert löst man das Problem dagegen, indem man auf einen sogenannten *Spooler* zurückgreift.

Was macht ein Spooler?

Als Spooler bezeichnet man im allgemeinen speicherresidente Programme, die im Hintergrund auf die Druckausgaben der Anwenderprogramme (Textverarbeitung, Malprogramm) lauern und diese in einem Puffer zwischenlagern. Ohne sich über den Spooler weiter Gedanken zu machen, schicken Sie wie gewohnt Ihren Text an den Drucker. Er wird wie bisher seine Arbeit sofort aufnehmen, aber schon nach kurzer Zeit wechselt der Mauszeiger von der fleißigen Biene wieder zum Pfeil, um zu signalisieren, daß Sie in der Hauptanwendung weitermachen können.



Da Sie - außer bei sehr rechenintensiven Programmen wie zum Beispiel einer Tabellenkalkulation - Ihren Rechner bei vielen Aufgaben nicht voll auslasten, hat der Mikroprozessor noch genügend Zeit, sich mit anderen Dingen zu befassen. Während Sie also in Ihrer Textverarbeitung über einer Formulierung brüten, langweilt sich der 68000er ganz fürchterlich. Der Spooler nutzt diese Zeit, um zwischendurch immer wieder einige „Häppchen" an den Drucker zu schicken. In der Regel klappt das so reibungslos, daß der Ausdruck verzögerungsfrei vonstatten geht.

Eine Lösung

Voraussetzung für so ein Programm ist natürlich ein kleineres oder größeres Stück vom Speicherkuchen des ATARIs. Das können nur 5, 50 oder sogar 500 KByte sein. Bei den meisten Programmen dieser Art können Sie zwischen verschiedenen Größen wählen, bei dem Spooler von Andreas Mandel (ST-PD 382) läßt sie sich in dieser Version leider noch nicht frei einstellen (auch wenn es so aussieht). Sie ist auf 100 KByte festgelegt. Dies ist bei genügend Speicherplatz ein vernünftiger Wert, obwohl 35 kByte in vielen Fällen auch schon ausreichen. Wer auf kleinen Systemen Speicherplatzprobleme bekommt, hat allerdings die Möglichkeit, den Spooler nach Belieben an- und auszuschalten.

Über die etwas ungewöhnliche Funktion *Startschwelle* kann das Programm angewiesen werden, erst eine gewisse Datenmenge anzusammeln, bevor es mit dem Ausdrucken beginnt. Will dann doch vorher ausdrucken, braucht man nur *Rest aus* anzuklicken. Hat man aus Versehen einen (langen) Text an den Drucker geschickt, läßt sich der Puffer auch einfach löschen. Neben der Veränderbarkeit der Puffergröße und der Startschwelle ist in einem Update auch ein Umlenken der Druckausgabe in eine Datei geplant sowie ein Abspeichern eigener Voreinstellungen.

Alternativen

Als Alternativen zu dem oben beschriebenen Accessory bieten sich zwei weitere Programme an: Entweder das Mehrfachaccsseory *Werkzeugkiste* von Ingo Brückl (ST-PD 356) oder auch *Mega-Matic* von Lloyd E. Pulley (ebenfalls ST-PD 356). Beide Programme bieten neben anderen Funktionen auch einen variabel konfigurierbaren Druckpuffer. Während man bei Werkzeugkiste allerdings ein Dutzend weitere - im allgemeinen sehr nützliche, aber speicherplatzfressende - Funktionen mitnehmen muß, bietet Mega-Matic die Möglichkeit, nach Belieben Druckpuffer, RAM-Disk, Bildschirmschoner etc. als Autoordnerprogramm zu installieren.

Fazit

Ein Spooler ist eine feine Sache, wenn man nicht „stundenlang" auf den langsamen Drucker warten will. Es ist aber etwas gewöhnungsbedürftig, während des Lärms, den so ein Matrixdrucker im allgemeinen macht, konzentriert zu arbeiten. Vielleicht sollte man währenddessen doch weniger anspruchsvolle Aufgaben erledigen, wie das Formatieren und Kopieren von Disketten.

TL

Spooler
ST-PD 382

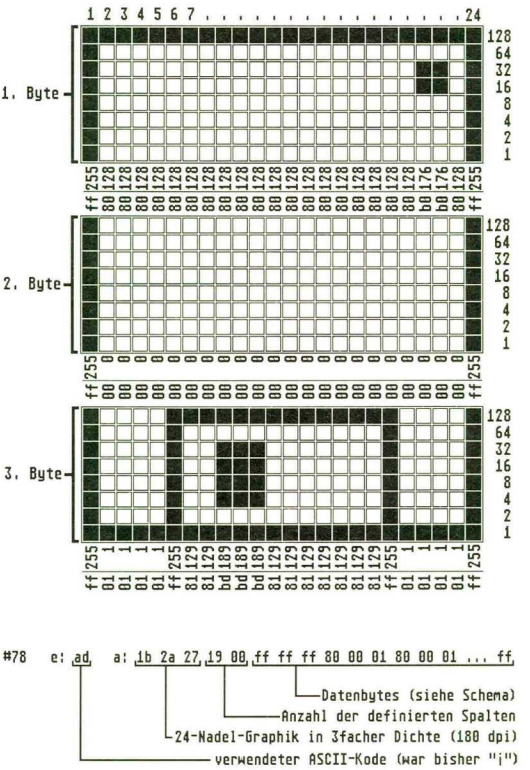
GRAFIK-ZAUBER

Eine professionelle Textverarbeitung ohne Grafikeinbindung dürfte sich heutzutage als unverkäuflich erweisen. Andererseits haben diese Programme auch ihren Preis. Was liegt da näher, als sich in der PD-Sammlung nach einer brauchbaren Lösung umzusehen?

MS-DOS läßt grüßen

In vielen Fällen reichen bei Grafikeinbindung in Text schon recht einfache Symbole. Wenn man einmal auf die MS-DOS-Welt schielt, findet man dort im Bereich der ASCII-Codes 128 bis 255 eine ganze Menge Sonder- und Grafikzeichen. Vor allem mit den Linien und Rastern lassen sich mit etwas Übung schon ganz passable Illustrationen erzeugen. Es muß dabei nur der ATARI-Zeichensatz umgestellt werden und der Drucker über den IBM-Zeichensatz #2 verfügen.

Von diesem Umstand ausgehend, habe ich mir überlegt, ob man nicht noch einen Schritt weiter gehen könnte und wirklich *beliebige* Zeichen auf Bildschirm wie Drucker zaubern kann. Da es für die Bildschirmdarstellung schon viele gute Lösungen gibt (z.B. FONTEDIT, ST-PD 217), soll dieser Aspekt an dieser Stelle weitgehend unberücksichtigt bleiben. - Die Programme, mit denen ich am meisten arbeite und mit denen ich mich am besten auskenne, sind Harlekin (MAXON) und Minitext (ST-PD 230). Diese beiden Programmen benutzen eine ähnlich aufgebaute Druckeranpassungsdatei, die unter anderem auch eine Tauschtabelle für einzelne Zeichen beinhaltet. Ursprünglich ist sie dazu gedacht, aus dem vom Textprogramm gesendeten ASCII-Code einen anderen zu machen, um so beispielsweise das ß (ASCII 158) behelfsweise in ein ß (ASCII 225) zu tauschen. Nachdem ich erfreut festgestellt hatte, daß dort nicht nur Byte gegen Byte, sondern gegen beliebig viele ausgetauscht werden können, war das Problem schon zur Hälfte gelöst!



Ein guter Tausch

Meine einfache, aber äußerst effektive Idee war, in die Tauschtabelle eine Steuersequenz für eine Minigrafik einzufügen. Da benutzerdefinierbare Zeichen in der Breite auf traurige 30 bis 40 Punkte eingeschränkt waren, habe ich mich auf die Einzelnadelansteuerung (Bit Image Mode) gestürzt. Im Klartext heißt das, daß das nicht benötigte Zeichen „¡" (auf dem Kopf stehendes Ausrufezeichen, ASCII 173, Sad) gegen die Sequenz *Einzelnadelansteuerung einschalten* (ESC *, $1b 2a), *24 Nadeln mit 180 dpi* (39 bzw. $27), die Spaltenzahl und abschließend die Daten-Bytes getauscht werden muß (gilt für EPSON-kompatible Geräte). Diese Minigrafiken dürfen 24 Punkte hoch (24-Nadel-Drucker, sonst nur 8) und bis zu 2880 Punkte breit sein. Gedruckt werden kann zwischen 60 Punkte/Zoll (dpi) und 360 dpi, wobei sich mit 180 dpi m. E. die besten Ergebnisse erzielen lassen.

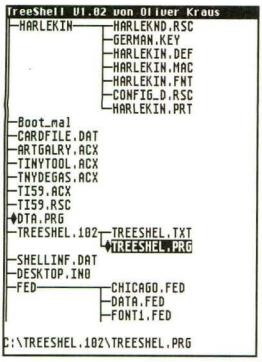
Käsekästchen

Zur Vorbereitung wird Kästchen- oder Millimeterpapier benötigt, auf dem man seine Sonderzeichen entwirft. Es gehört schon etwas Übung und Geduld dazu, das Zeichen im Raster zu entwerfen und korrekt in Bytes zu übertragen. Dabei sollte man die Auflösung nicht überschätzen, denn ein Pixel Abstand kann man hinterher kaum erkennen. Probieren geht hier über Studieren! - Die hübsche Grafik verdeutlicht an einem Beispiel, wie man vorgehen muß: Bei 24 Nadeln wird jede Spalte durch drei Bytes beschrieben, bei einem 8-Nadeldrucker entsprechend nur durch eines. Die Daten müssen spaltenweise (Byte 1 bis 3) hinter der Umschaltsequenz und der Spaltenanzahl angegeben werden (alle Zahlen für Minitext *hexadezimal*). Der Wert ergibt sich durch Addieren der einzelnen Zeilenwerte (1 bis 128). Um die kleine Diskette aufs Papier zu hämmern, muß man also 72 Zahlen ausrechnen und korrekt eingeben: 1b2a27 (Grafikmodus)1900 (Spalten, Schreibweise siehe Druckerhandbuch) ffffff 800001 800001 800001 800001 8000ff 800081 800081 8000bd 8000bd 8000bd 800081 800081 800081 800081 800081 800081 800081 800081 8000ff 800001 b00001 b00001 800001 ffffff. - Abschreckend, aber lohnend!

Fazit

Ich hoffe, daß ich Sie auf eine interessante, kreative Möglichkeit aufmerksam machen konnte. Nicht immer muß ein teures Programm her, sondern mit ein bißchen Phantasie läßt sich manches Problem lösen. Diese Art der Druckersteuerung ist allerdings schon eher etwas für Fortgeschrittene, aber das Funktionsprinzip dürfte nun soweit klar sein, daß sich jeder in die Materie einarbeiten kann.

TL



TreeShell 1.02

Eigentlich soll die vielgerühmte Festplatte ja eine Arbeitserleichterung sein (was sie in den meisten Fällen auch ist), aber aufgrund der hohen Speicherkapazität sieht man sich unversehens mit einem neuen Problem ganz anderer Art konfrontiert: man verliert so langsam den Überblick, wo welches Programm zu finden ist, oder man hat keine Lust, sich durch tausend Unterverzeichnisse zu klicken. Das Programm TreeShell von Oliver Kraus stellt eine interessante Lösung dar, des Problems Herr zu werden.

Nach dem Programmstart wird zunächst die gesamte Ordnerstruktur einer Partition eingelesen und in einer Hilfsdatei abgelegt. Anschließend wird das Inhaltsverzeichnis ähnlich wie bei MS-DOS als „Baum" dargestellt, auf dessen Ästen man beliebig wandern kann. Ausführbare Dateien (*.PRG; *.TOS; *.TTP) sind dabei durch eine Raute markiert. Hat man sie mit dem Cursor angewählt, reicht ein einfaches Return, um das Programm zu starten, egal in welchem Verzeichnis es sich befindet. Alternativ kann man sich eine Datei auch anhand der Anfangsbuchstaben suchen lassen (was äußerst schnell geht) oder häufig benötigte Programme über frei programmierbare Funktionstasten abrufen.

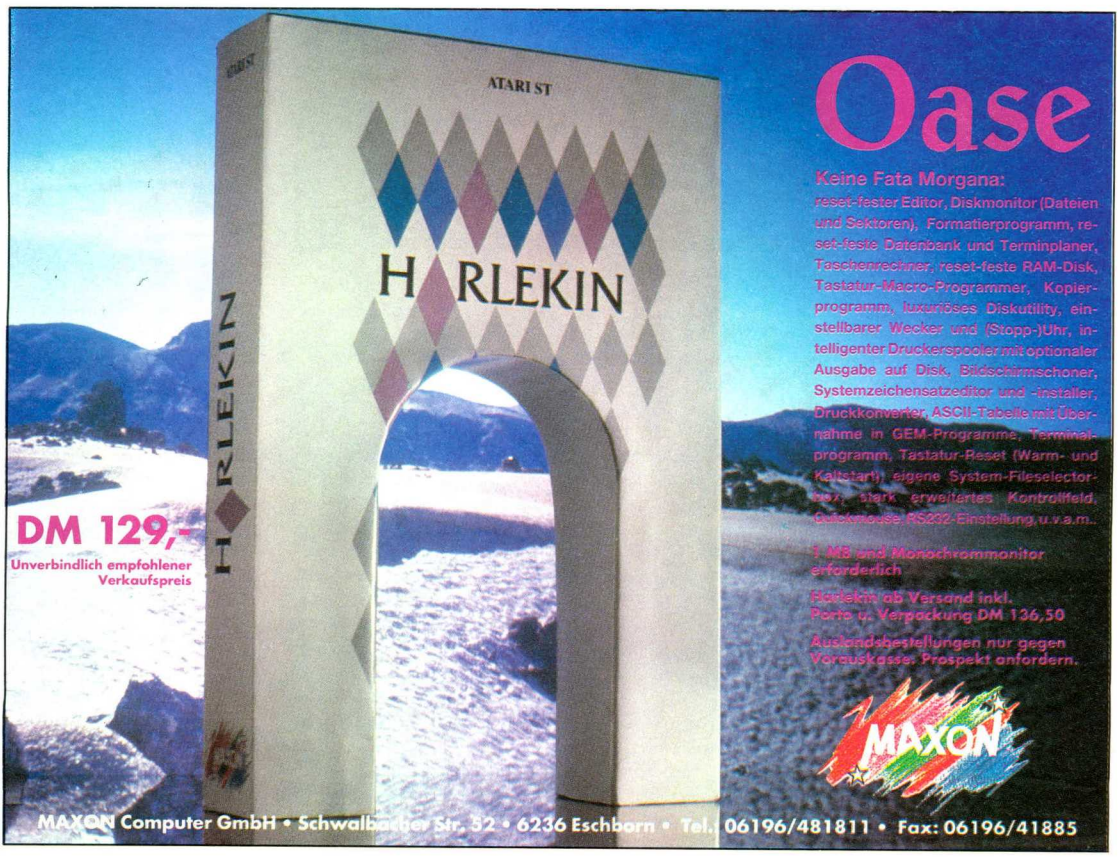
Ferner läßt sich der Baum über die Shift-Taste auf die Unterverzeichniseinträge reduzieren oder per Undo in der mittelgroßen Schrift darstellen, so daß die Übersicht auch in großen verschachtelten Verzeichnissen nicht so schnell verlorengeht. Hat man mit Return eine Datei ausgewählt, die kein Programm ist, wird sie wie gewohnt auf den Bildschirm gelistet, was natürlich nur bei ASCII-Texten sinnvoll ist.

Andere Partitionen oder Laufwerke werden einfach mit Control-Laufwerkskennung angewählt. Sofern die oben erwähnte Hilfsdatei existiert, wird auf die darin gespeicherte Information zurückgegriffen. Falls jedoch inzwischen Dateien hinzugefügt oder gelöscht wurden, empfiehlt es sich, per Clr-Home das Inhaltsverzeichnis neu einlesen zu lassen. Wird dabei das Diskettenlaufwerk A angesprochen, darf es nicht schreibgeschützt sein, da sonst leider ein hoffnungsloser Absturz des Programms folgt.

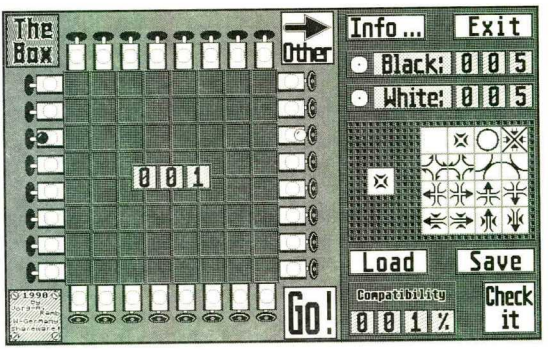
Wer nicht so sehr auf grafische Finessen (wie z. B. bei Gemini oder Opaque) Wert legt, sondern nur einen bequemen Vermittler zwischen seinen Anwendungsprogrammen sucht, sollte sich das Programm von der Diskette 339 ruhig einmal näher ansehen.

TL

TreeShell 1.02
ST-PD 339



Die Kiste!



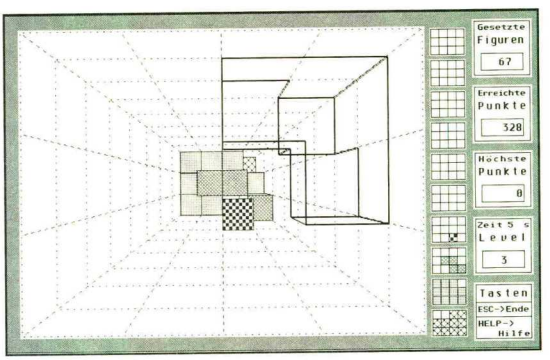
Eine harte Herausforderung für Knobler und Tüftler verbirgt sich hinter dem recht unscheinbaren Namen THE_BOX. Wenn man das Programm neugierig startet, ohne vorher einen Blick in die Anleitung geworfen zu haben, kann man mit dem merkwürdigen Flipperautomaten kaum etwas anfangen. Aber auch hier liegt der Reiz wieder in einem ganz einfachen Spielprinzip: Der Computer verteilt auf dem verdeckten Spielbrett verschiedenartige Hindernisse, die die beiden Kugeln ablenken oder reflektieren (THE BOX). Die Aufgabe des Spielers besteht „einfach" darin, das verdeckte Spielfeld auf einem zweiten Brett nachzubauen (TEST BOX). Dazu kann er die beiden Kugeln sooft wie nötig durch das erste Labyrinth schikken. Netterweise erhält man dann auch eine Laufzeitangabe, über die man zusätzliche Rückschlüsse über das verdeckte Innenleben erzielen kann.

Über einen Test-Knopf kann man anschließend die Übereinstimmung prüfen lassen, die dann in Prozent angegeben wird. Ziemlich gemein vom Programmautor Jörg-Andreas Ramb ist, daß es außer selbst nachzudenken keine Möglichkeit gibt, die Lösung zu finden, denn das Programm nimmt keine Kapitulation an! Bei 16^{64} verschiedenen Möglichkeiten hat man viel zu tun. Als kleinen Trost gibt es dafür die Möglichkeit, Partien abzuspeichern, um sie somit an einem besseren Tag fortsetzen zu können.

TL

The Box
ST-PD 346

AUSGE-BLOCKT



Eine neue TETRIS-Variante in 3D Niemand wird bestreiten, daß TETRIS eines der besten Computerspiele überhaupt ist. Wundern wird sich mancher, wenn ich nun behaupte, daß sich dieses genialeinfache Spielprinzip noch ein bißchen verbessern läßt: Sei es bei STETRIS (ST-PD 273), bei dem es komplexere Figuren gibt (jeweils aus fünf Segmenten bestehend) oder COLUMNS (ST-PD 273), bei dem verschiedene Farbkombinationen zusammengesetzt werden müssen, oder schließlich beim sechseckigen HEXTRIS (ST-PD 374).

Neidisch wurde man als Atarianer, als vor kurzem in der MS-DOS-Welt eine weitere Variante unter dem Namen BLOCK OUT auftauchte: Das gewohnte Prinzip wurde auf die dritte Dimension erweitert. Das heißt, man sieht den Schacht, in dem die Figuren kombiniert werden müssen, nicht mehr von der Seite, sondern perspektivisch von oben. Getarnt unter dem Namen SETZ_AB (ST-PD 346) hat der Neid dank Werner Lang aus Stuttgart nun ein Ende.

Die wie beim normalen TETRIS ebenfalls aus vier Segmenten bestehenden Figuren erscheinen am Bildschirm als durchsichtiges Drahtmodell. Über den Cursor-Block und in Kombination mit der Shift-Taste lassen sie sich wie gewohnt verschieben, aber zusätzlich auch um sämtliche räumlichen Achsen drehen. Mit der Leertaste kann man die Steine in den Schacht fallen lassen. Auch wenn die Steuerung zunächst sehr kompliziert aussieht, sollte man sich davon nicht abschrecken lassen. Schon nach wenigen Runden hat sich die durchaus sinnvolle Tastenbelegung soweit eingeprägt, daß man sich voll auf das Spiel konzentrieren kann. Wie beim Vorbild besteht der Sinn des Spieles darin, die einzelnen Figuren in dem Schacht so anzuordnen, daß sie eine geschlossene Fläche bilden, die dann verschwindet. Schafft man dies nicht, wächst der Turm immer weiter, und mit Erreichen der obersten Ebene endet das Spiel.

Als kleine Hilfe ist der zehn Ebenen umfassende Schacht mit einer Menge Hilfslinien versehen. Am rechten Rand wird im Miniformat zusätzlich nochmals für jede einzelne Ebene ein Schnitt angezeigt, an dem man sich ebenfalls orientieren kann. Wenn eine Etage vollständig gefüllt ist, verschwindet sie, und der darüberliegende Turm rutscht nach. Anders als beim normalen TETRIS sind die Figuren nicht nach ihrer Form, sondern nach ihrer Position im Stapel eingefärbt, so daß sie sich dabei in ihrer Farbgebung entsprechend verändern. Wenn man sich daran gewöhnt hat, kann man sich die räumliche Anordnung sogar ohne die Hilfsdarstellung am Rand gut vorstellen.

Vor dem Start kann man ein Level wählen, das einem mehr oder weniger Bedenkzeit läßt (zwischen einer und zehn Sekunden), bevor das Drahtmodell eine Etage tiefer fällt. Je weniger Zeit man benötigt, desto mehr Punkte kann man dabei natürlich einheimsen! Nach jeweils 25 gesetzten Steinen erhöht sich das Level um einen Schwierigkeitsgrad. Am Bildschirmrand wird ferner eine kleine Statistik geführt, die über die Anzahl der gesetzten Figuren, die erreichten Punkte, den High-Score sowie das aktuelle Level und die damit verbundene maximale Bedenkzeit Auskunft gibt. Nach Spielende besteht die Möglichkeit, sich in eine High-Score-Liste einzutragen.

Diese BLOCK OUT-Umsetzung ist für alle TETRIS-Fans ein Muß, für alle anderen Atarianer eine dringende Empfehlung! Das Spiel selbst ist spannend und eine echte Herausforderung, die programmtechnische Umsetzung (Quellcode in GFA-BASIC 3.0 liegt bei) und Bedienung einwandfrei! SUCHTGEFAHR!!

TL

Setz ab
ST-PD 346

PD-TeX 3.1 auf der Überholspur

Es hat sich mittlerweile herumgesprochen, daß TeX *das* System zum hochwertigen Satz von Texten ist, und die in der ST Computer 10/89 vorgestellte PD-TeX-Version hat schon eine beachtliche Verbreitung gefunden. Mit der neuen Version geht der Spaß jetzt aber erst richtig los. Diese Version ersetzt die alte auf den PD-Disks 250 bis 254 und 276 bis 278 komplett.



Bild 1: Eingabe der Sonderzeichen in TEMPUS

TeX ist anders als andere Kinder. TeX (sprich Tech, für griechisch Tau, Epsilon, Chi) ist keine gewöhnliche Textverarbeitung. Das von Donald E. Knuth geschriebene TeX wurde konsequent auf ein optimales Ausgabeergebnis hin entwickelt. Wenn Sie einmal eine mit TeX gesetzte Diplomarbeit mit einer von einer gewöhnlichen Textverarbeitung gesetzten verglichen haben, wissen Sie, was ich meine.

Die Kernstücke des TeX-Systems sind weitestgehend rechner- und druckerunabhängig. TeX-Versionen gibt es inzwischen auf praktisch allen Rechnern vom PC an aufwärts. Dies bedeutet, daß TeX-Benutzer bei einem eventuellen Rechnerwechsel ihre Texte gleich weiterbenutzen können. Auch auf zukünftigen Rechnergenerationen wird es mit Sicherheit TeX-Versionen geben.

Die Rechnerunabhängigkeit legt dem System aber auch gewisse Beschränkungen auf. TeX ist kein WYSIWYG-System (what you see is what you get), sondern arbeitet mit einer Kommandosprache, die, ähnlich wie eine Programmiersprache, erst erlernt werden will. Dadurch kann TeX auch von einem normalen Text-Terminal ohne Grafikfähigkeiten gesteuert werden. Auf dem Atari ST nutzt das PD-TeX die vorhandenen Grafikmöglichkeiten aber natürlich aus.

TeX selber ist auch von den Ausgabegeräten unabhängig. Für Drucker oder Bildschirm braucht man daher jeweils einen passenden Treiber, der das Ergebnis auf dem entsprechenden Gerät ausgibt. Die Zeichensätze (Fonts), die TeX benutzt, werden alle mit dem zugehörigen MetaFont erzeugt; falls der Drucker selber intern Zeichensätze hat (soll's ja geben), so werden diese einfach ignoriert.

Hat man einen Zeichensatz also einmal im MetaFont-Format vorliegen, so hat man diesen mittels MetaFont auch gleich für alle nur denkbaren Ausgabegeräte. Infolgedessen kommt das Schriftstück auch auf jedem Ausgabegerät im wesentlichen (d. h. soweit die Druckqualität des Druckers das zuläßt) gleich heraus. Lediglich die Qualität des Druckers begrenzt dann noch die Schönheit des Ausdrucks.

Doch nun genug der Vorrede; weitere Informationen zu TeX finden Sie in der angegebenen Literatur. Jetzt will ich sie Ihnen nicht länger verheimlichen, die

Neuerungen

Als TeX vor etwa 10 Jahren entwickelt wurde, wurde das zunächst vom amerikanischen Standpunkt aus getan. Ab der Version 3.0 hat Donald E. Knuth einige Neuerungen eingebaut, die uns Europäern das Leben doch sehr erleichtern. So ist jetzt die Direkteingabe von 8-Bit-Zeichen (Umlaute, usw.) möglich, bei mehrsprachigen Texten kann TeX zwischen den Trennregeln umschalten. Es können also in einem Satz z. B. deutsche Wörter nach deutschen, und englische Wörter nach englischen Regeln getrennt werden.

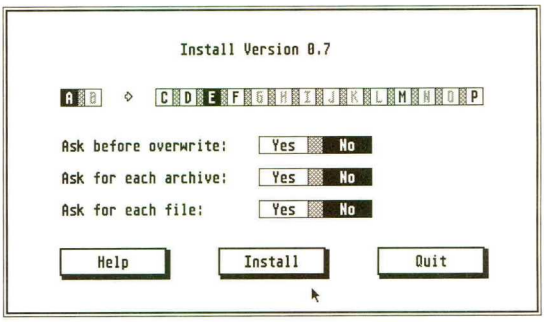


Bild 2: Automatische Installation per Knopfdruck

Die TeX Users Group (TUG) hat außerdem eine TeX-interne Codierung der Umlaute und Sonderzeichen festgelegt. In Bild 1 sehen Sie, welche Zeichen Sie damit direkt eingeben können (hier im Beispiel via TEMPUS). Die Tabelle zeigt nicht den Cork-Code, sondern den Eingabecode auf dem Atari für das PD-TeX 3.1. Man sieht insbesondere, daß alle Standardumlaute mit ihrem normalen Atari-Code eingegeben werden können. Weitere Sonderzeichen, wie Integrale, Wurzeln usw. werden dann über TeX-Kommandos angesprochen.

Die Version 3.1//2.1cs (sprich TeX 3.1 in der Atari-Version 2.1cs) läuft außerdem etwa doppelt (in Worten: zweimal) so schnell wie die Version 2.98//1.4cs. Dies wurde im wesentlichen durch einen Compiler-Wechsel von CCD Pascal nach GNU C erreicht. Auch MetaFont ist durch den Wechsel etwa doppelt so schnell wie bisher. Die Zeit zum Erzeugen des CMR10-Fonts auf einem normalen ST liegt unter 10 Minuten. Mit Beschleunigerkarten oder TT geht's entsprechend schneller. Das PD-TeX ist jetzt eine der schnellsten TeX-Versionen für den ST und braucht keinen Vergleich zu scheuen. Das komplette neue TeX-System läuft selbstverständlich auch mit TTs, 68030-Karten, 16-MHz-Karten, Großbildschirmen usw.

Die Druckertreiber, jetzt mit dem neuen Lattice C 5.06 entwickelt, wurden durch interne Verbesserungen erheblich schneller. In den Treibern wurden auch einige kleine Fehler entdeckt und behoben. Viele neue Funktionen und etliche ganz neue Treiber kamen hinzu. Neu sind z.B. die Unterstützung der Lindner-TeX-Grafik sowie das Einbinden von PostScript-Bildern beim PostScript-Treiber.

Die mitgelieferten Styles entsprechen nun dem internationalen Standard. Sagt man in LaTeX *\documentstyle[german]{article}*, so kommen die Überschriften auch tatsächlich in deutsch heraus.

Weil die Makro-Pakete für TeX immer mehr wachsen, stößt man nun leichter an die Grenzen des Systems (TeX capacity exceeded, sorry). Um dem zu begegnen, sind nun gleich zwei vergrößerte Versionen dabei: Large TeX kann von BenutzerInnen mit mindestens 2 MB Speicherausbau, Huge TeX

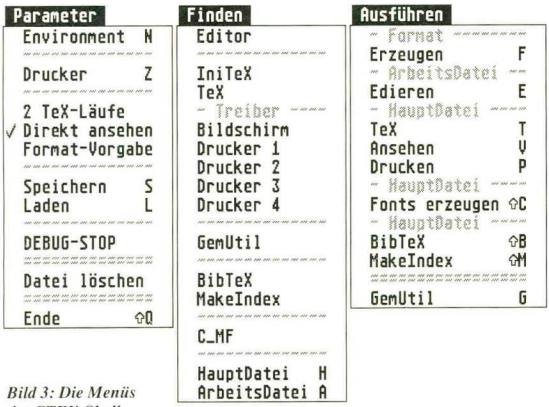


Bild 3: Die Menüs der CTEX-Shell

ab 4 MB Speicherausbau benutzt werden. Das dürfte dann für alle Anwendungen ausreichend sein.

Beständigkeit

Einer der Gründe für die Beliebtheit von TeX ist hohe Standardisierung des Systems. Texte lassen sich problemlos von einem Rechner auf den anderen portieren. Entsprechend ist es auch mit der Aufwärtskompatibilität: Wer schon längere Texte (Diplomarbeit usw.) mit einer alten Version entwickelt hat, kann diese einfach mit der neuen weiterbearbeiten. Es wurde darauf geachtet, daß die neue Version in diesem Fall das gleiche Ergebnis wie die alte produziert.

Die TeX-Version 3.1 ist gleichzeitig - von D. E. Knuths Standpunkt aus gesehen - eine der letzten Versionen von TeX. Es wird in Zukunft wohl nur noch kleinere Fehlerkorrekturen geben. Eine Version 3.2 ist nicht vorgesehen. Zukünftige Versionen werden 3.14, 3.141, 3.1415, usw. heißen. Ähnliches gilt für MetaFont: Nach der aktuellen Version 2.7 kommen 2.71, 2.718, 2.7182 usw.

Man kann sich trefflich darüber streiten, ob ein sich schnell entwickelndes, aber immer wieder mit neuen Fehlern behaftetes, oder ein stabiles, aber weitgehend fehlerfreies System besser ist. Bei TeX wurde diese Entscheidung zugunsten der Stabilität getroffen.

Installation

Um es gleich vorweg zu sagen: Für TeX brauchen Sie eine Festplatte. Die alte Version enthielt noch Hinweise für die Benutzung mit zwei Diskettenlaufwerken; es hat sich jedoch herausgestellt, daß damit kein vernünftiges Arbeiten möglich ist. Am besten legt man für TeX eine eigene Partition von 10-15 MB an.

Die komplizierte Installationsprozedur war einer der am häufigsten bemängelten Punkte am alten TeX-System. In der neuen Version wird ein komfortables Installationsprogramm mitgeliefert. Man drückt für jede der Disketten nur noch einmal auf *Install*, und schon ist das System installiert (siehe Bild 2). Das System wird normalerweise komplett auf einer einzigen Partition eingerichtet. Bei Platzmangel kann MetaFont aber auch auf einer anderen Partition als der Rest installiert werden.

Das TeX-System ist komplett und in sich abgeschlossen. Sogar ein kleiner Editor wird mitgeliefert. Die insgesamt 8 Disketten sind gut gepackt; einige sind randvoll. Die Anzahl der mitgelieferten Fonts ist allerdings nicht allzu groß. Die Beispieltexte können Sie damit wohl ansehen und teilweise auch ausdrucken. Für das Arbeiten mit TeX müssen Sie jedoch einmal für jeden Drucker die nötigen Fonts komplett mit MetaFont erzeugen. Aber keine Panik: Las-

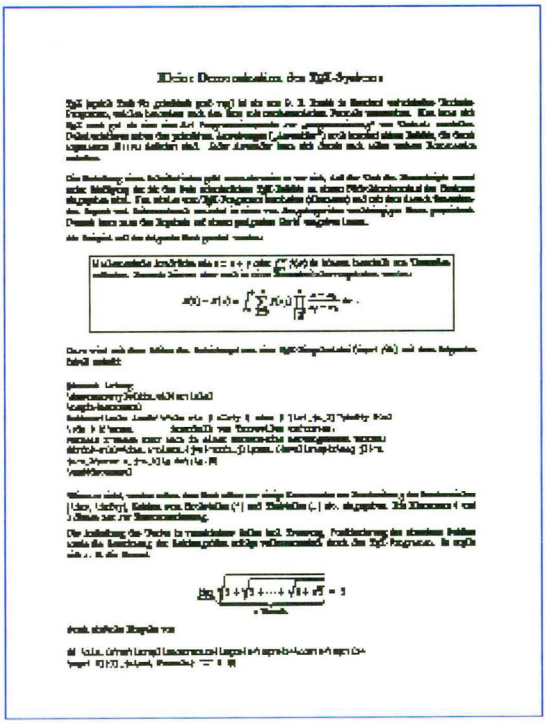


Bild 4: Überblick über den Aufbau der Seite mittels Zoom…



Kleine Demonstration des TEX-Systems

TEX (sprich Tech für griechisch groß τεχ) ist ein von D. E. Knuth in Stanford entwickeltes Textsatz-Programm, welches besonders auch den Satz mit mathematischen Formeln unterstützt. Man kann sich TEX auch gut als eine eine Art Programmiersprache zur „programmierung" von Textsatz vorstellen. Dabei existieren neben den primitiven Anweisungen („Assembler") noch komfortablere Befehle, die durch sogenannte *Macros* definiert sind. Jeder Anwender kann sich damit auch selbst weitere Kommandos erstellen.

Die Erstellung eines Schriftstückes geht normalerweise so vor sich, daß der Text des Manuskripts zuerst unter Einfügung der für den Satz erforderlichen TEX-Befehle an einem Bildschirmterminal des Rechners eingegeben wird. Nun wird er vom TEX-Programm bearbeitet (übersetzt) und mit dem danach feststehenden Layout und Seitenumbruch zunächst in einer von Ausgabegeräten unabhängigen Form gespeichert. Danach kann man das Ergebnis auf einem geeigneten Gerät ausgeben lassen.

Als Beispiel soll der folgende Text gesetzt werden:

Mathematische Ausdrücke wie $z = x + y$ oder $\int_{x_1}^{\infty} f(x)\,dx$ können innerhalb von Textzeilen auftreten. Formeln können aber auch in einer Formelzeile hervorgehoben werden:

$$F(b) - F(a) = \int_a^b \sum_{j=0}^{n} f(x_j) \prod_{\substack{k=0 \\ k \neq j}}^{n} \frac{x - x_k}{x_j - x_k}\,dx \quad .$$

Dazu wird mit dem Editor des Betriebssystems eine TEX-Eingabedatei (*input file*) mit dem folgenden Inhalt erstellt:

```
%format latexg
\documentstyle[din_a4]{article}
\begin{document}
Mathematische Ausdr"u"cke wie $ z=x+y $ oder $ \int_{x_1}^\infty f(x)
\;dx $ k"onnen          innerhalb von Textzeilen auftreten.
Formeln k"onnen aber auch in einer Formelzeile hervorgehoben werden:
$$F(b)-F(a)=\int_a^b\sum_{j=0}^n f(x_j)\prod_{k=0}\atop{k\neq j}}^n
{x-x_k\over x_j-x_k}\; dx\;\;.$$
\end{document}
```

Wie man sieht, werden neben dem Text selber nur einige Kommandos zur Beschreibung der Sonderzeichen (\int, \infty), Zeichen zum Hochstellen (^) und Tiefstellen (_) etc. eingegeben. Die Klammern { und } dienen nur zur Zusammenfassung.

Die Aufteilung des Textes in verschiedene Zeilen incl. Trennung, Positionierung der einzelnen Zeichen sowie die Berechnung der Zeichengrößen erfolgt vollautomatisch durch das TEX-Programm. So ergibt sich z. B. die Formel

$$\lim_{n\to\infty} \underbrace{\sqrt{2+\sqrt{2+\cdots+\sqrt{2+\sqrt{2}}}}}_{n\ \text{Wurzeln}} = 2$$

durch einfache Eingabe von

```
$$ \lim_{n\to\infty}\underbrace{\sqrt{2+\sqrt{2+\cdots+\sqrt{2+
\sqrt 2}}}}_{n\rm\ Wurzeln} "=" 2 $$
```

Bild 4a: …und als Postscriptausgabe auf Laserdrucker.

Die mit *Xpaint* erstellten Zeichnungen können direkt in TEX eingebunden werden:



Bild 5: Grafik in TeX

sen Sie Ihren ST einmal über Nacht rechnen, und schon haben Sie die Fonts beisammen. Die neue C_MF-Shell für MetaFont macht Ihnen das alles sehr bequem.

Wenn die alte TeX-Version nicht zum Laufen gebracht werden konnte, lag das meist an einer (versehentlich oder mutwillig vorgenommenen) falschen Installation. Durch die automatische Installation werden Fehler jetzt direkt an der Quelle vermieden.

TeX wird komplett über die GEM-Shell *CTEX* gesteuert (siehe Bild 3). In der neuen Version sind neben der Unterstützung für neue Systemteile (BibTeX und MakeIndex werden jetzt mitgeliefert) auch Tastaturkürzel für alle wichtigen Funktionen hinzugekommen. Benutzt man einen NEC P6, sind außerdem die Parameter der GEM-Shell richtig voreingestellt. Bei Benutzung anderer Drucker muß man noch unter *Finden* und *Parameter* den Drukker einstellen.

Jetzt kann man sofort das LATEXG-Format erzeugen und das Beispiel *DEMO.TEX* ausprobieren.

Vielfalt

Der neue Bildschirmtreiber bietet einige Leckerbissen. Er funktioniert nun auf alle Bildschirmen, die GEM-Unterstützung bieten, also auch auf OVERSCAN, Großbildschirmen, MGE, TT usw. sowohl in Farbe als auch in schwarzweiß. Er funktioniert also praktisch immer und überall.

Gelegentlich möchte man sich vielleicht einen Überblick über den Aufbau der gesamten Seite verschaffen. Dafür gibt es nun die Verkleinerung mittels Zoom-Funktion, die dies schnell (d. h. in drei Sekunden) erledigt (siehe Bild 4). Anders herum ist es auch möglich, den Bildschirm mit Drucker-Fonts zu benutzen und so effektiv eine Vergrößerung des Bildes zu erreichen. Dazu muß man lediglich eine kleine Zusatzoption (wie -v=180) angeben.

Bei den Druckertreibern ist einiges hinzugekommen. So existieren jetzt Druckertreiber für 9-Nadler, alle gängigen 24-Nadler (der Firmware-Fehler in den Star-Druckern stört nicht mehr), Tintenstrahldrucker Canon BJ-130 und HP DeskJet sowie für Laserdrucker HP LaserJet und Atari Laser. Last not least ist ein PostScript-Treiber vorhanden.

Besonders der HP DeskJet erfreut sich - als DeskJet plus oder DeskJet 500 -in jüngster Zeit zunehmender Beliebtheit. Er schont das Ohr und bietet fast Laser-Qualität.

Der LaserJet-Treiber ist sehr schnell geworden. Solange man keine großen Grafiken einbindet, ist er fast so schnell wie der Treiber für den Atari Laser. Die Ausgabedateien beim Drucken auf Diskette bleiben recht klein.

Für das ultimative Ergebnis kann man mittels PostScript bei Bedarf eine Satzanlage ansteuern. Die LinoType-Anlagen bieten beispielsweise schon bei 635 dpi ein hervorragendes Bild. (Die Belichtung auf einer solchen Anlage ist allerdings nicht ganz billig; rechnen Sie mit einer Größenordnung von 10 DM pro Seite.)

Wem etwas weniger genügt, und wer z. B. an der Arbeitsstelle oder Uni Zugang zu einem Laser- oder anderen Drucker hat, kann auch diesen nutzen. Zu diesem Zweck können die Druckertreiber alle auf Disk drucken. Das Ergebnis kann man dann auf eine 3.5-Zoll PC-Diskette kopieren und an die entsprechende Stelle tragen.

Features

Das Atari PD-TeX hat viele Eigenschaften, auf die andere TeX-Versionen neidisch sein können. Wie Experten schon lange erkannt haben, stecken in den Grafikfähigkeiten enorme Möglichkeiten. Im Prinzip kann das TeX alles, was Sie auch mit einem Plotter machen können - und noch mehr.

Einen kleinen Eindruck der Leistungsfähigkeit vermittelt Bild 5. Dieses Bild wurde mit dem Public Domain-ZPCAD (ST-PD 298,299) erstellt, in TeX eingebunden und mit den normalen TeX-Treibern ausgedruckt.

Zu erwähnen wäre vielleicht auch noch, daß die Druckertreiber bei fehlenden Fonts eine Liste der benötigten Fonts erstellen können, die dann mit C_MF gleich zu deren Erzeugung weiterverarbeitet werden kann. Dadurch gestaltet sich das Verfahren bei fehlenden Fonts sehr einfach. Wer mit TeX vielleicht nur ein paar Formeln setzen möchte, um diese nachher in ein Grafikprogramm einzubinden, findet einen Treiber, der nicht auf einen Drucker, sondern in ein GEM-Image-Bild 'druckt'. Diese Bilder lassen sich dann ganz normal mit anderen Programmen weiterverarbeiten.

Wie Sie sehen, wurde an vieles gedacht, was Ihnen das Leben angenehm macht. Natürlich gibt es kein völlig perfektes System, aber das Atari PD-TeX ist, wenigstens meiner Meinung nach, schon gar nicht übel.

Ausblick

Das Atari-PD-TeX ist in seiner neuen Version eine runde Sache. Das heißt nicht, daß es nichts mehr zu verbessern gäbe, aber zunächst sind Sie jetzt einmal an der Reihe. Sammeln oder entwickeln Sie Makro-Pakete, schreiben Sie Grafikprogramme, die mit TeX arbeiten, oder erfreuen Sie sich einfach an den Möglichkeiten, die TeX Ihnen bietet.

Der Einstieg in TeX ist zwar bestimmt nicht so einfach wie der in eine normale Textverarbeitung, aber seien Sie versichert: Wer TeX einmal richtig verstanden hat, will nichts anderes mehr anfassen.

Literatur:

ST Computer 5/89, Seite 148-154: TeX, der stille Superstar.
ST Computer 12/89, Seite 167-173: Public Domain TeX. Satzsystem zum Nulltarif.
Donald E. Knuth: The TeXbook. Computers & Typesetting Volume A. Addison Wesley, 1990.
Helmut Kopka: LaTeX - Eine Einführung. Addison Wesley, 1988.
Helmut Kopka: LaTeX - Erweiterungsmöglichkeiten. Addison Wesley, 1990.

TeX 3.1
ST-PD 290-297

ST-COMPUTER PUBLIC DOMAIN

ATARI ST COMPUTER Public Domain Service

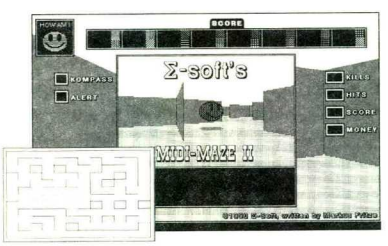
Liebe Leser,

kurz vor der 400sten Disketten freuen wir uns, Ihnen wieder einmal einen ganz besonderen Leckerbissen zu präsentieren; und zwar die ST-Umsetzung des aktuellen (und auch entgültigen) TeX-Systems 3.1. Daneben zwei Programme, die TeX-Vektorgrafik unterstützen. Insgesamt ein sehr leistungsstarkes Paket, das alle TeX-Anhänger sicher begeistert.
Wie schon erwähnt, ist im nächsten Monat Diskette 400 fällig. Allen PD-Autoren sei gesagt, daß die 400 noch nicht vergeben ist. Einsendungen die bis Anfang Februar eingehen, können noch berücksichtigt werden.

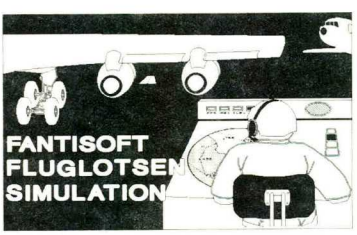
Bis dann

Ihre ST-Computer-Redaktion

385 SPIELE

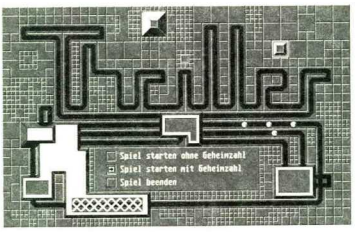


MIDIMAZE II: Stark erweiterte und verbesserte Version gegenüber MIDIMAZE I. Das Spiel funktioniert mit 2 bis 16 STs, kann also nur über MIDI-Netz (zwei einfache Kabel) gespielt werden. MIDIMAZE ist ein wahres Suchtspiel. Jeder Spieler steuert einen kugelförmigen Smiley durch ein Labyrinth. Trifft man einen Freund, hilft man ihm, kommt ein Feind, schießt man ihn ab (oder umgekehrt). Das Interessante ist, daß man den Mitspieler am anderen Rechner auf seinem Bildschirm trifft. Das Spiel ist Shareware. Gegen eine Registrierungsgebühr bei den Autoren erhält man neben Zusatzprogrammen auch ein original MidiMaze-T-Shirt, mit dem man sich auf jeder MidiMaze-Party sehen lassen kann.



FFS: Fluglotsensimulator. Simuliert die Aufgaben eines Fluglotsen. Auftretende Flugwünsche müssen durch Variation der Flughöhen- und Richtungen geregelt werden. Verschiedene Schwierigkeitsstufen (Häufigkeit der Flüge und Geschwindigkeiten von Zivil- und Militärflügen), Eingabe verschiedener Flugzeugdaten und diverser Flugräume, die wiederum mit dem beigefügten Karteneditor erzeugt werden können. Darstellung am ständig aktiven Radarschirm. Bei Fehlern wird dem Fluglotsen das Gehalt gekürzt. Sourcecode in GFA-BASIC und Assembler beiliegend. (s/w)

386 SPIELE



THRILLER: Action-Strategiespiel mit 10 gefahrvollen Level. Auf vorgegebenen Bahnen müssen Bonussteine eingesammelt werden. Doch ständig greifen Killerkugeln an, denen man geschickt ausweichen muß. Tolle Grafik und witziger Sound. Spiel per Joystick. (ab TOS 1.2) (s/w)

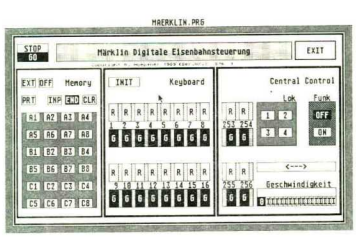
387 TECHNIK



AUTOKOST: Umfangreiches Programm zur Verwaltung aller anfallenden Autokosten. Neben der obligatorischen Benzinkostenauswertung, die in Autokost schon nach verschiedenen Gesichtspunkten und noch dazu grafisch ausgewertet wird, beinhaltet das Programm alle Fahrzeugdaten, Werkstattkosten, Öl-Verbrauch und Finanzierungskosten. Als Krönung sozusagen informiert es bei TÜV-Terminen, Wartungsintervallen und sagt, wenn man wieder einmal die ASU zu erledigen hat. (s/w)

GRAHAM: dient zur Umrechnung von Einheiten. Dabei sind die am meisten benötigten Einheiten für den physikalischen Chemiebereich verfügbar, die da wären:

m/s, cm/s, km/h, yd/s, mi/h, m/s^2, cm/s^2, in/s^2, yd/s^2, kg/m^3, g/cm^3, lb/ft^3, lb/gal(Br) , lb/gal(US), lb/in^3, kg, g, t, lb, cm, m, km, inch, foot, yard, rod, chain, furlong, mile, m^2, cm^2, sq. inch, sq. foot, sq. yard, acre, m^3, cm^3, cub inch, cub foot, cub yard, Pa, dyn/cm^2, bar, kp/m^2, kp/cm^2, torr, atm, mm H_2O, lb/in^2, J, erg, 1 at, 1 atm, kpm, kWh, cal, s, min, h, d
(s/w)



MAERKLIN: Steuerung einer MÄRKLIN-Digitaleisenbahn. Das Programm emuliert ein Central Control mit allen Funktionen, ein Keyboard mit allen Funktionen und ein Memory-Modul.

SCHEIBE: Spezielle Belastungsberechnungen der Technischen Mechanik. Nach Angabe des Materials, E-Modul, Querdehnzahl, Scheibendicke, Eckenzahl, Eckkoordinaten usw. berechnet Scheibe die wirkenden Kräfte und stellt es grafisch dar, samt Hauptsspannungstrajektoren, Belastung und Dreiecksnetz. (s/w)

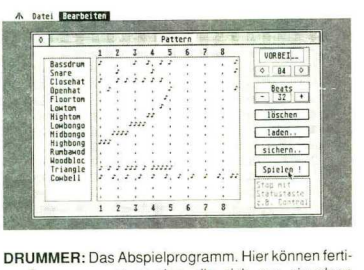
WINKEL: Berechnet typisierte Winkelanschlüsse anhand des Weißdruckes der DIN 18800. Nach Anwahl des entsprechenden Winkels werden die Grenzspannungen für verschiedene Stahlprofile ausgegeben. (s/w)

XRECHNER: Einfacher Taschenrechner mit 2 Nachkommastellen, speziell für Finanzrechnungen à la DM. Besonderheit: Das ACC speichert auf Wunsch den Rechenvorgang als ASCII-Datei ab, so daß er in ein Textprogramm eingeladen werden kann. (s/w)

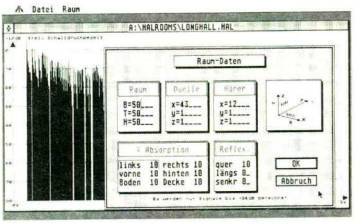
388 MUSIK

Drum- und Rhythmusgenerator auf der Basis synthetisch erzeugter Klänge. Im Gegensatz zu gesampelten Sounds, werden die Sounds hier mathematisch erzeugt, also berechnet. Das hat enorme Klangvorteile. Neben der Theorie kommt auch die Praxis bzw. die Anwendung nicht zu kurz. Leicht lassen sich tolle Rhythmen aus den bereits enthaltenen Samples (z.B. Schlagzeug, Karibik) zusammenstellen. Doch auch Melodien lassen sich beispielsweise zum Schlagzeug kreieren.

SAMPLER: Programm zum Erzeugen der Sounds. Das geschieht mittels mathematischer Formeln, die u.a. die Wellenform, die Frequenz, den Anstieg und Abfall der Lautstärke beschreiben. Ergänzt werden diese durch Schleifensteuerungen (IF THEN ELSE) und Variablenzuweisungen (A=3.14). Durch Variation der Frequenz kann ein Sample auch in verschiedene Tonhöhen umgerechnet, also eine komplette Tonleiter erzeugt werden.



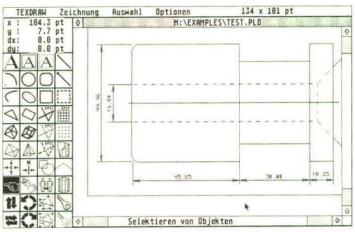
DRUMMER: Das Abspielprogramm. Hier können fertige Songs erzeugt werden, die sich aus einzelnen Pattern zusammensetzten, d.h. eine Ablaufliste in der bestimmte Passagen wiederholt werden können, ohne diese mehrmals zu schreiben. Ein Pattern kann sehr komfortabel mit der Maus erzeugt, Passagen verschoben oder ausgetauscht werden.



HALL: Der Hallgenerator. Dieser kann einen Sound in bestimmten räumlichen Gegebenheiten umrechnen, z.B. wie klingt die Bassdrum in einem langen schmalen Raum ohne gedämpfte Wände. Klangexperten werden feststellen, daß hiermit sehr komplexe Simulationen durchzuführen sind.

ST-COMPUTER PUBLIC DOMAIN

PD 389 TeX



TeX-Draw: Leistungsfähiges Zeichenprogramm und ideale Ergänzung zu TeX. TeX-Draw arbeitet vektororientiert, ähnlich GEM-Draw, produziert auf einem Ausgabegerät immer die maximale Auflösung. Sämtliche Grafikgrundfunktionen sind präsent. Hinzu kommen Bezierkurven und räumliche geometrische Funktionen. Objekte können, wie in objektorientierten Programmen üblich, gruppiert und als Komplettobjekt behandelt werden. Die Bilder können nun in TeX (z.B. Atari-TeX) importiert und dort auf allen Geräten (Drukker, Laser, Satzbelichter) ausgegeben werden.
TeXDraw kann neben TeX-Dateien auch HPGL importieren (ein Format, das u.a. von CAD-Programmen produziert wird), BGI-Vektorfonts verarbeiten und Rastergrafik (z.B. STAD) einlesen und konvertieren.
Ausgabe neben LaTeX- und PiCTeX- auch im Metafont-Format. (s/w)

PD 390 bis 397 TeX-System

AtariTeX 2.0: Komplettes TeX-System basierend auf der TeX-Endversion 3.1. Läuft auf ST,TT und unterstützt Grafikkarten und -erweiterungen.
Einbindung von Grafik, z.B. das GEM-IMG-Format, GEM-Metafile und TeX-Grafik-Befehle. Somit wird sowohl Pixel- als auch Vektorgrafik unterstützt.
Zoom-Funktion für Seitenübersicht oder Detailansicht.
Das System ist in einer komfortablen Shell mit Editor eingebunden und steht nach Durchlauf des automatischen Installationsprogramms betriebsbereit auf der Festplatte.
Festplatte (10MB frei) erforderlich.

390, 391, 392, 393

AtariTeX: TeX, LaTeX, Druckertreiber für alle 9- und 24-Nadeldrucker, HP Deskjet, HP LaserJet, Atari Laser bis hin zur PostScript-Ausgabe, die schließlich die Ausgabe auf Fotobelichtern (2540 dpi) ermöglicht. Damit kann AtariTeX auch zur Herstellung professioneller Druckvorlagen eingesetzt werden. Variable RAM-Disk.

394, 395

Metafont: Programm zum Erzeugen von Fonts in allen Größen für alle Ausgabegeräte samt komfortabler Shell.

396, 397

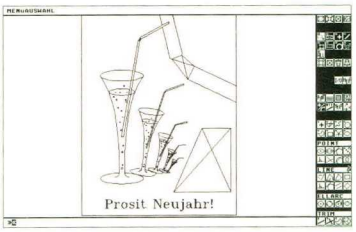
Fonts: Hochauflösende Zeichensätze für 9- und 24-Nadeldrucker.

MACHEN SIE MIT!

Möchten Sie ein selbstgeschriebenes Programm in unsere PD-Sammlung geben, um es auch anderen Usern zugänglich zu machen? Kein Problem. Schicken Sie es uns auf einer Diskette zu, samt einer Bestätigung, daß es von Ihnen geschrieben wurde und frei von Rechten Dritter ist. Bei Fragen steht Ihnen die Redaktion gerne zur Verfügung.

MAXON Computer • ST-Computer PD
Industriestr. 26 • W-6236 Eschborn

PD 398 und 399



ZPCAD: CAD-Programm mit Ausgabemöglichkeit für AtariTeX. Menü- und Kommandozeilen orientiert.

- Elementarobjekte: Punkt, Linie, Ellipse (Ausschnitt), Text, Schraffur, Detail (Teilzeichnung, rekursiv)
- vielfältige Konstruktionsmöglichkeiten
- 20 Ebenen
- Schraffur beliebiger Gebilde aus Linien und Ellipsen (-ausschnitten)
- Affine-Transformation in der Ebene
- bis zu 5 Vektor-Zeichensätze gleichzeitig
- Hilfswerkzeuge zur Erstellung eigener Zeichensätze
- beliebig umfangreiche nachladbare Befehlssätze (bis zu 5 gleichzeitig)
- Standardbefehlssätze mit komplexen Funktionen für Konstruktion, Text, Schraffur, Bemaßung, ...
- Plotterausgabe
- Standardbefehlssätze bzw. Plottertreiber zur Ein-/Ausgabe von Standardformaten (HPGL, CS-Graphics, AutoCAD-DXF, GEM-MetaFile)

(s/w)

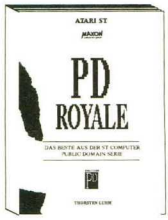
PD 389 bis 399 Paket TeX

Achtung! Komplettes TeX-Paket, bestehend aus dem kompletten TeX 2.0, Metafont, Fonts, TeXDraw, ZPCAD.
alle 11 Disketten für DM 89.-

UPDATE

PD 283 TurboASS 1.7: unterstützt Overscan, läuft auf STE und Grafikkarten, benutzt alternativ kleine Fonts. Tool-Programme wurden überarbeitet.

Schon gesehen?



"PD ROYALE - Das Beste aus der ST Computer Public Domain Serie" entweder direkt bei MAXON Computer für DM 34,- inkl. Versandkosten oder über den Buchhandel (unverbindlicher Verkaufspreis DM 29,-, ISBN 3-927065-07-2).

Die in PD ROYALE beschriebenen Programme sind auf über 60 Disketten verteilt. Wir haben uns daher dazu entschlossen, diese Programme neu gesammelt zusammenzustellen. Herausgekommen sind drei *'Pakete Royale'*, bestehend aus je 5 randvollen Disketten, gefüllt mit Programmen aus diesem Buch. Damit Sie je nach Ihren Interessensgebieten auswählen können, haben wir themenorientierte Pakete zusammengestellt.

PAKET ROYALE A - Anwendungen
Minitext, Deluxe Fontmaster, Fontedit, Tiny Editor, Keyhelp, Brief, Last Word, Umlaut, Prn-Send, Speedwriter, ASCII-Edit, Printing Press, Little Painter, Showtime, Sticker, LQ 800, Hardcopy, Public Painter, WDR-Bild, SBASE, GEM-Calc.

5 Disketten DM 39,-

PAKET ROYALE B - Utilities+Unterhaltung
ARC, Sagrotan, Bitte ein Bit, Cruncher, FCOPY III, Hyperformat, Packer, ZOO, Calendar, Chooseboot, Deskedit, Dump, FSelect, Give Up, Goodies, James, JClock, Lock, Megamatic, Simple, ST-Klick, Termin, Uhr, Werkzeugkiste, X-Utility, CHR-Tree, Diskkatalog, Goodview, Hide, SuperFile-Copy, XDir, XDirList, Bar, Biorythmus, ST-Kalender, Kalender, Shapes, Erdkugel.

5 Disketten DM 39,-

PAKET ROYALE C - Spiele
Tetrix, Stones, Columns, Diamond Miner, Solitaire, U-Boot, Laserschach, Superbreakout, Invaders, Mac Pan, Memory, Quiz, Wizzy, Go-Up, DGDB I+II

5 Disketten DM 39,-

ABKÜRZUNGEN
1MB = mind. 1MB Speicher notwendig
s/w = nur Monochrom; f = nur Farbe

ST COMPUTER Public Domain Service

DIREKT-VERSAND

Alle PD-Disketten unserer Sammlung gibt es nur direkt bei MAXON-Computer.

1. Schriftliche Bestellung

- Der Unkostenbeitrag für eine Diskette beträgt DM 10,-
- Hinzu kommen Versandkosten von DM 5,- (Ausland DM 10.-)
- Bezahlung per Scheck oder Nachnahme
- (Im Ausland nur Vorauskasse möglich)
- Bei Nachnahme zuzüglich DM 4,00 Nachnahmegebühr
- Ab 5 Disketten entfallen die Versandkosten (DM 5.- bzw. DM 10.-)
- Der Versand kann aus technischen Gründen **ausschließlich** gegen Nachnahme oder Vorauskasse erfolgen (auch für Händler!).

2. Telefonische Bestellung

MAXON-Computer GmbH
'PD-Versand'
Tel.: 0 61 96 / 48 18 11
Fax: 0 61 96 / 4 18 85
Mo-Fr 9^{00} - 13^{00} und 14^{00} - 17^{00} Uhr

- Lieferung erfolgt per Nachnahme

Adresse:

MAXON-Computer GmbH
'PD ST-Computer'
Schwalbacher Straße 52
W-6236 Eschborn

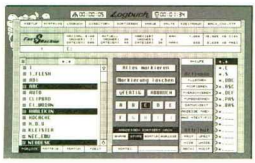
Nutzen Sie die PD-Karte in diesem Heft

Immer up to date

Programmname	Version	Daten		Programmname	Version	Daten	
Adimens ST	3.1	N HM		Musix32	1.01	J H	
Adiprog SPC Modula	1.1	N HM		NeoDesk	3.0	N HML	
Aditalk ST	3.0	N HM		Notator	3.0		
Adress ST / Check ST	1.0	N H		☛ NVDI	1.0	N HML	
Afusoft Morse-Tutor	2.0	N HML		Omikron Assembler	1.86	N HML	
Afusoft Radio-Writer	1.0	N HML		Omikron BASIC-Compiler	3.06	N HML	
Afusoft Radiofax plus	1.0	N HML	1M	Omikron BASIC 68881-Compiler	3.06	N HML	
AIDA	1.1	N HM		Omikron BASIC Interpreter	3.03	N HML	
AnsiTerm	1.4	N		Omikron DRAW! 3.0	3.01	N HML	
Assembler Tutorial	1.06	N HM		Omikron EasyGEM-Lib	1.0	N HML	
Banktransfer	1.0	N H		Omikron Maskeneditor	1.0	N HML	
1st BASIC Tool	1.1	N HML		Omikron Midi-Lib	2.1	N HML	
BTX/VTX-Manager	3.0	N H	1M	Omikron Numerik-Lib	1.2	N HML	
Calamus	1.09	N H	1M	Omikron Statistik-Lib	1.5	N HML	
Cashflow	1.0	N H	1M	PAM's TERM/4014	3.012e	N H	
Chips At Work	1.0	N HM		PAM's TurboDisk	1.7	N HML	
CIS-L&G	2.1	N H	2M	PAM's NET	1.1	N HML	
CiSystem	2.1		2M	PCB-layout	1.19	N H	
Creator	1.1	N H		PC ditto Euroversion	3.96	N HML	
Cubase	2.0			PegaDress	1.0	N H	
CW-Chart	8.0	N H	1M	PegaFakt	2.0	N H	
Daily Mail	1.2	N H		PegaStic	1.1	N H	
dBMAN	6.0	N HM		phs-BTX-Box	6.1	N HML	1M
Diskus	2.0	N HM		phs-ST-Box	1.2	N HM	
dBMAN	5.10	N HML		phs-Boxtalk	1.0	N HM	1M
Easybase	1.1	N HM		phs-Boxedi	1.0	N HML	1M
Easytizer	1.0	N HM		Platon	1.45	N H	
Easy Rider Assembler	2.04	N HM		1st Proportional	3.13	N HM	
Easy Rider Reassembler	2.31	N HM		Prospero Pascal	2.153	N HML	
Edison	1.00	N HM		Prospero Fortran	2.153	N HML	
fibuMAN	4.0	N H		Prospero C-Compiler	1.144	N HML	
fibuSTAT	2.3	N H		Prospero Developers Toolkit	1.111	N HML	
Flexdisk	1.4	N HML		Protos	1.1	N H	1M
FM-Meßtechnik	1.0.b	N HM		Querdruck2	2.05	N HM	
FTL Modula-2	1.18	N HM		Quick_Dialog	1.0	N HM	
Gadget	1.2.5b	N H		ReProk	1.10	N H	1M
GEMinterface ST	1.1.	N HML		Revolver	1.1	N HML	1M
GFA-Artist	1.0	N L		Rufus	1.04	N HML	1M
GFA-Assembler	1.5	N HML		Scarabus	2.0	N H	
GFA-BASIC 68881	1.3	N HML		Scigraph	2.0	J HM	
GFA-BASIC-Compiler	3.5	N HML		Script	2.0	N HM	
GFA-BASIC-Interpreter	3.5	N HML		Search!	2.0	N HM	
GFA-Draft plus	3.01	N		Signum! zwei	2.01	N H	
GFA-Farb-Konverter	1.2	N H		Simula	2.1	N HML	1M
GFA-Monochrom-Konverter	1.2	N ML		Skylink	1.5	N H	1M
GFA-Objekt	1.2	N HM		Skyplot+	4.3	N H	1M
GFA-Starter	2.0	N HML		Soundmachine II	1.0	N HM	
GFA-Vektor	1.0	N		SoundMerlin	1.01	N HM	
G+Plus	1.4	N HML		SPC-Modula-2	2.0	N HML	
GrafStar	1.0	N H		Spectre 128	1.9	J HM	
Hänisch Modula-2	3.1	N HML		1st_Speeder 2	1.0	N HML	1M
Hard Disk Accelerator	1.0	N HML		SPS ST	1.5	N H	1M
Hard Disk Sentry	1.10			STAD	1.3+	N H	
Hard Disk Toolkit	2.0	N HM		Steuer-Tax 2.9	1.10	N HM	
Harddisk Utility	3.0	N HM		Steuer-Tax 3.9	1.10	N HM	
Harlekin	1.0	N H	1M	STop	1.1	N HM	
Imagic	1.1	N HML		ST Pascal plus	2.08	N HM	
Intelligent Spooler	1.10	N HML		Supercharger	1.4	J H	
Interlink ST	1.89	N HM		Technobox Drafter/2	2.0	J H	1M
ISI-Interpreter	1.20	N HM		Technobox CAD/2-ST/TT	1.4	J H	2M
Junior Prommer	2.33	N HM		Tempus Editor	2.10	N HM	
K-Resource	2.0	N HM		☛ Tempus Word	1.0	N H	1M
Kleisterscheibe	2.2	N HM		That's Write	1.4	N HM	
Label ST	1.0	N HML		Theca Librarian	1.0	N HM	
Laser C (Megamax)	2.1	N HML		TIM	1.2	N H	
1st_Lektor	1.2	N HM		TIM II	1.0	N H	1M
Lern ST	1.22	N HML		Transfile ST 1600	1.1	N HM	
Link_it GFA	1.1	N HML		Transfile ST 850	1.2	N HM	
Link_it Omikron	2.0	N HML		Transfile ST plus	3.1	N HM	
MagicBox ST	7.78	N HM	1M	Transfile ST E500	2.0	N HM	
Mathlib	3.0	N HM		Transfile ST SF	2.0	N HM	
Mega Paint II	2.30	N H	1M	Transfile ST IQ	1.4D	N HM	
Mega Paint II Professional	2.31	N H	1M	Turbo C	2.0	N HM	
Megamax Modula 2	3.5	N HM		Turbo ST	1.8	N HML	
MGE Grafikkarte	1.27	N		UIS II + Hermes	2.5		
MGP GAL-Prommer	2.0	N H		V_Manager	3.1	N H	
Micro C-Shell	2.70	N HM		VSH Manager	1.0	N HML	1M
MPe II plus	1.02	N H	1M	WERCS Resource-Editor	1.0	N HM	
Mr Print	3.0	N H		Wordperfect	4.1	N H	
MT C-Shell	1.2	N HM	1M	Writer ST	1.4	N HM	
Multidesk	1.82	N HML		Wordplus	3.15	N HML	

Irrtum vorbehalten! Daten-Legende : N = kein Kopierschutz, J = Kopierschutz, H = hohe Auflösung, M = mittlere Auflösung, L = niedrige Auflösung, 1M = mindestens 1 Megabyte, ☛ = Änderung gegenüber letzter Ausgabe

SONDERDISK



FastSectorBackup 4.0

FastSectorBackup ist das ideale Tool für Ihre Datensicherung. Zum einen bietet es ein Image-Backup, welches komplette Partitionen sichert, und zum anderen ein sehr flexibles FileBackup. Damit lassen sich einzelne Dateien, welche nach Wildcards, Datum, Archiv-Bit oder einfach per Mausklick markiert werden, sichern. Weiterhin bietet FastSectorBackup die Möglichkeit, mehrere Backup-Vorgänge mit verschiedenen Markierungsarten in Batch-Dateien festzuhalten. Diese können dann automatisch ablaufen.

FastSectorBackup [1]
SD 35 *DM 25,-*

NEU

ORDNE HDB

Nach häufigem Schreiben und Löschen auf Festplatte sind die zusammengehörenden Teile einer Datei (Cluster) oft weit verstreut, was zu erheblichen Zeitverlusten führt. Das Programm ordnet die Struktur völlig neu, so daß alle Cluster einer Datei unmittelbar beieinander liegen. Der Plattenzugriff wird dadurch schneller.
Weitere Funktionen: Retten bzw. Regenerieren gelöschter Dateien, Umstrukturierung der Directory-Einträge, FAT-Analyse, Belegen defekter Sektoren, Ordner-Struktur zeigen, Namen (Platte/Ordner) ändern und anderes.
ORDNE HDB unterstützt die Treiber AHDI, CBHD, ICD, Eickmann und Vortex.

ORDNE HBD
SD 51 *DM 20.-*

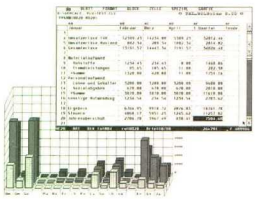


Dialog Construction Set

für GFA-BASIC 3.x

Mit dem Dialog Construction Set (DCS) lassen sich auf einfache Art und Weise LST-Dateien erstellen, die den Programmcode zur Behandlung von Dialogboxen unter GFA-BASIC 3.0 enthalten. So ist es möglich, diese schnell und bequem in eigene Programme einzubauen. Als Voraussetzung wird natürlich weiterhin das Resource Construction Set (wird bei GFA-BASIC mitgeliefert) benötigt. Einfach mit dem RCS erstellen und dann mittels DCS den Programmcode generieren. Grundkenntnisse über Dialogboxen und GFA-BASIC-Programmierung sind aber weiterhin erforderlich.

DCS [1]
SD 48 *DM 15,-*



GEM-CALCplus 3.0

Tabellenkalkulation

Überall dort, wo mit Zahlen hantiert wird, sei es zur betriebswirtschaftlichen Kostenrechnung, statistischen Auswertung von Meßreihen oder zur Erfassung der eigenen Finanzen, findet ein Kalkulationsprogramm seinen Einsatz. GEM-CALCplus ist ein flexibler und sehr leistungsfähiger Vertreter dieser Kategorie. Neben zahlreichen mathematischen und statistischen Funktionen bietet es eine exzellente Grafikausgabe der Daten als Kuchen-, Linien-, Balken-, Stapel-, Säulen-, Block- und Flächengrafik.

Funktionen und Operatoren:
+, -, *, /, PI, DAT, ABS(), INT(), RND(), LOG(), EXP(), CLG(), SQR(), SIN(), COS(), TAN(), ASN(), ACS(), ATN(), FAK(), NUN(N:n), SUM(), AVE(), STA(), STD(), MUL(), MIN(), MAX(), QMW(), QMN()

GEM_CALCplus ist eine Weiterentwicklung des weitverbreiteten GEM_CALC (PD)

Die Erweiterungen:
• Arcussinus und Arcuscosinus • Blatt und Block schützbar • Fehlermeldungen mit Erläuterung • Suchfunktion • verbesserte Grafikdarstellung, Grafikausdruck und Grafik-Datenauswahl (Block) • flexible Speicherverwaltung • fixierbare Spalte • erhöhter Eingabekomfort • schnelleres Scrolling • u.v.a.m.

Alte Datenblätter können übernommen werden. (1MB sinnvoll)

GEM-CALCplus 3.0 [1]
SD 44 *DM 25.-*

DAME

Computerumsetzung des alten Brettspiels, wobei der ST einen spielstarken Gegner darstellt. Die Figuren werden per Maus angewählt, die Züge protokolliert und analysiert. Verschiedene Spielstärken, Zugvorschläge, Laden und Speichern einer Partie, sowie verschiedene Spielvarianten dürfen nicht fehlen.

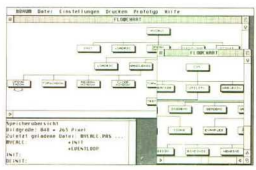
DAME
SD 29 *DM 15.-*

PANDA

Der Farbemulator

Der Farbemulator simuliert die Farbauflösungen des ST auf einem monochromen Monitor (SM,124,...). Dadurch kann man auch Farbprogramme laufen lassen, die sonst einen zweiten Monitor erfordern.

PANDA [1]
SD 18 *DM 15,-*



BBAUM

BBAUM ist ein äußerst leistungsstarkes Tool für die Programmdokumentation von C-, PASCAL- und GFA-BASIC-Programmen. Vor allem die Einarbeitung in fremde Quelltexte wird vereinfacht, indem grafisch in Form eines Baumes die Funktions- bzw. Prozedurabhängigkeiten dargestellt werden.

BBAUM untersucht:
- C-Quelltexte
- PASCAL-Quelltexte
- GFA-BASIC-Quelltexte (2.0, 3.0 und 3.5)
- DMP-Dateien (interne Baumstruktur)
- Verzeichnisse (Struktur Ihrer Festplatte/Diskette)

BBaum verwaltet Includes bzw. ausgelagerte Programmteile und fügt sie automatisch an die entsprechenden Stellen im Hauptprogramm an. Wahlweise werden auch die Routinen dargestellt, die in der System-Library definiert sind (z.B. *printf* oder *getchar*).
BBAUM ermöglicht weiterhin das Suchen eines bestimmten Namens und springt augenblicklich an diese Stelle in der Grafik. Gerade beim Erzeugen eines Directory-Baumes ist das sehr praktisch, da man auf diese Weise schnell eine bestimmte Datei findet.
BBAUM unterstützt alle 8-, 9- und 24-Nadeldrucker. Je nach Größe der Grafik werden auch mehrere aneinanderpassende Seiten bedruckt. Zur Druckzeitoptimierung ist zusätzlich eine direkte Ansteuerung der Centronics-Schnittstelle implementiert.
BBAUM unterstützt ebenfalls die Generierung von Funktionsprototypen, die den Umstieg auf den neuen ANSI-C-Standard erleichern.

BBAUM
SD 50 *DM 25.-*

KOALA

Der Monochromemulator

KOALA, der Monochromemulator ermöglicht es, Software, die für Monochrommonitor geschrieben wurde (z.B. SIGNUM!), auch auf einem Farbbildschirm laufen zu lassen.

Freie Einstellung der Bildwiederholfrequenz. So kann man zwischen hoher Bildrate oder hoher Rechenleistung wählen • Bildaufbau während Diskettenzugriff abschaltbar • Bildschirm-Hardcopy auf Disk (Farb- und s/w-Bild).

KOALA ist kompakt, schnell und für alle ST-TOS-Versionen (1.0-1.6).

KOALA [2]
SD 43 *DM 15.-*



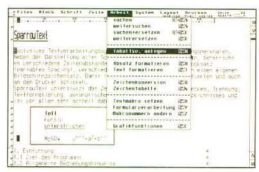
1stTrenn

vollautomatische Silbentrennung für 1stWordPlus

Darauf haben viele schon lange gewartet. Eine schnelle, automatische und präzise Silbentrennung für 1stWordPlus. 1stTrenn ersetzt die eingebaute Trennhilfe völlig, d.h. wird automatisch anstelle der eingebauten manuellen Trennung aktiviert (F10).

- arbeitet im Hintergrund (Accessory), 1stWordPlus muß nicht verlassen werden
- schnelle Trennung
- wahlweise mit Bestätigung oder vollautomatisch.
- hohe Trefferquote von über 98%, d.h. etwa eine falsche Trennung bei 8 Seiten Text.
- zusätzliche Autosave-Funktion des aktiven Textes
- läuft auf den deutschsprachigen 1stWordPlus Versionen 1.89, 2.02 und 3.15

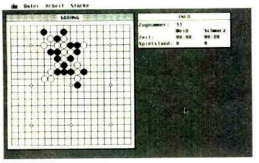
1stTrenn
SD 42 *DM 25.-*



SparrowText

Exklusives Textverarbeitungssystem mit besonderen Leistungsmerkmalen. Neben der Darstellung aller Schriftarten auf dem Bildschirm beherrscht es verschiedene Zeilenabstände, Proportionalschrift im Blocksatz (variables Spacing), verschiedene Font-Größen und vor allem einen eigenen Bildschirmzeichensatz. Damit lassen sich Sonderzeichen entwerfen und auch an den Drucker schicken.
SparrowText unterstützt das Zeichnen von Linien und Rechtecken, Trennung, Textformatierung, automatische Erzeugung eines Inhaltsverzeichnisses und ist vor allem sehr schnell dabei.
Als besonderen Leckerbissen ermöglicht es Formularverarbeitung, die sich hervorragend zum Ausfüllen von Briefbögen, Adreßfeldern oder allgemeinen Formularen eignet. Die Eingabefelder lassen nach Wunsch auch Eingabebeschränkungen (z.B. nur Zahlen) zu und bieten daher die Möglichkeit, gewisse Felder miteinander aufzuaddieren. Weiterhin kann man diese Felder automatisch ausfüllen lassen, da SparrowText Daten von einer Datenbank importieren kann und diese in die Felder einträgt. Dadurch läßt sich das Programm für Serienbriefe, Zeugnisse oder gar Rechnungen/Mahnungen einsetzen.

SparrowText [1]
SD 37 *DM 25,-*

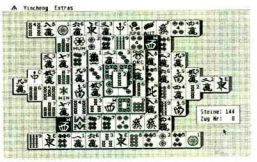


GOBANG

Ein Strategiespiel

GOBANG ist ein klassisches Brettspiel, bei dem abwechselnd Steine auf das Spielfeld gesetzt werden, wobei es gilt, 5 Steine in einer Reihe (senkrecht, waagrecht oder diagonal) zu plazieren. Der Computer bietet hier einen spielstarken Gegner, der nicht so leicht zu besiegen ist.
Neben dem Laden und Speichern einer Partie verfügt Gobang über verschiedene Spielstärken; vom Anfänger bis zum Profi. Auch die Blitzpartie, bei der jeder Spieler nur 30 Sekunden Bedenkzeit pro Spiel hat, bietet ihren speziellen Reiz. Ist man in einer schwierigen Lage, hilft der Rechner gerne mit einen Zugvorschlag aus.

GOBANG [1]
SD 49 DM 15.-



YINCHENG

Dieses Spiel beruht auf dem alten chinesischen Patience-Spiel Mah-Jongg. Es geht darum, das mit 144 Spielsteinen gefüllte Spielfeld zu entleeren, wobei immer nur zwei zueinander passende und nach bestimmten Regeln positionierte Steine entfernt werden dürfen. YINCHENG beinhaltet eine zwei- und eine dreidimensionale Spielvariante, die sich zwar in den Regeln, doch kaum in der Spielqualität unterscheiden.

YINCHENG [1]
SD 45 DM 20.-



ODYSSEUS

Schachprogramm

Hinter Odysseus steckt ein spielstarkes und komfortables Programm. Die Züge lassen sich leicht per Maus eingeben. Es verfügt über eine Zeit- und eine Tiefensteuerung (bis zu 12 Halbzüge) und beherrscht den Turniermodus. Die beigefügte, jederzeit erweiterbare Bibliothek erlaubt dem Programm den Zugriff auf wichtige Züge. Mit ihm kann man Partien speichern, nachspielen und analysieren lassen.

Odysseus [1]
SD 41 DM 25.-



ASSOZIATIX

Assoziative Datenbank

Assoziatix ist eine assoziativ-Muster orientierte Datenverwaltung, die es ermöglicht aus einer großen Datenmenge bestimmte Gruppen auszufiltern und daraus dank schneller assoziativer Suche nach bestimmten Konstellationen, Zusammenhänge zu finden (z.B. Rasterfandung).
Mit Hilfe des Formulareditors können die Eingabemasken leicht am Bildschirm gestaltet werden, sogar mit Grafikeinbindung.

Einige Besonderheiten:
- Paßwortschutz, Export- und Importfunktion, Serienbriefe, Reportdokumentation Statistische Berechnung numerischer Werte Expertfunktion, Volltextsuche Grafikeditor: Spiegeln, Drehen, Zoomen, Balken- Linien und Kuchengrafik.

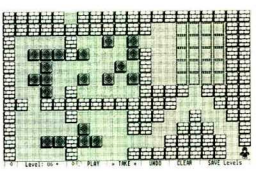
ASSOZIATIX (2 Disketten)
SD 27 a/b DM 30.-

COMPLEX

Quiz

Quiz mit über 3.500 Fragen aus den Wissensgebieten Geschichte, Geographie, Sport, Allgemeinbildung, Tierwelt, Kunst, Naturwissenschaft und Theater. Das Programm kann mit eigenen Fragen erweitert werden, somit steht die Möglichkeit zur Schaffung eines spezialisierten Quiz' (z.B. Motorwelt, Jura, Computerkunde oder gar Fremdsprache) offen. (1MB, 1-6 Spieler)

COMPLEX [1]
SD 47 DM 20.-

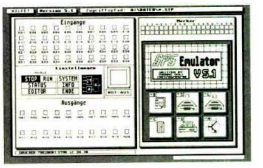


TAKE_1

Denkspiel

Die Regeln von Take_1 sind einfach - schieben Sie alle Diamanten auf die vorgesehenen Leerfelder. Leider gibt es ein paar Probleme... Der Vogel, den Sie mit den Cursortasten bewegen, kann immer nur einen Diamanten schieben und außerdem kann er sie nur schieben, nicht jedoch ziehen - das bedeutet, daß Diamanten an der Wand oder in einer Ecke nicht mehr verschoben werden können. Hier ist Taktik und Weitblick gefragt, denn so einfach, wie es im ersten Augenblick ausschaut ist es wirklich nicht. Durch den integrierten Level-Editor, lassen sich eigene Screens erzeugen.

TAKE_1 [1]
SD 46 DM 15.-



SPS-Emulator V 5.1

für programmierbare Steuerungen

Unser SPS-Emulator baut auf einem SIEMENS PG 605-Programmiergerät in STEP 5 auf. Mit ihm lassen sich SPS-Programme schreiben, auf Simulationsbasis austesten, laden, speichern, ändern, ausdrucken und als FUP (Funktionsplan mit logischen Gattern) ausgeben. Enthalten sind ein Editor, ein Interpreter und FUP-Generator. Alle Befehle wurden voll im Siemens S5 Standart umgesetzt •20 Timer als SE-, SA, SI-, SS-, SV-Timer zu verwenden • 20 Zähler (vorwärts/rückwärts), erhöhte Werte • Mehrfachzuweisungen nach einer Verknüpfung • wahlweise 20/40 Eingänge bzw. Merker • Schnellere Interpreterroutine (20-25%) • Startmerker für Autostart • Not-Aus-Merker/-Schalter • Blinkmerker: Vier Merker werden als astabiler Multivibrator angesteuert. •Sprungmarken (A-Z) •Komfortables Drucken der Awl •Die Merker-, Eingangs- und Ausgangsbezeichnungen können nun dezimal, hexadezimal oder byteweise bezeichnet werden. •Klammerbefehl —> U(,) • Oder vor Und —> O • Neuer Texteditor • Erhöhter Bedienungskomfort •Programmierung von Netzwerken

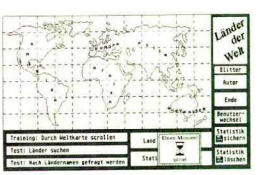
SPS Emulator V5.1 [1]
SD 14+ DM 25.-

OPAQUE

Das Desktop mit neuem Gesicht

Wie wäre es mit einem zweckmäßigen und originellen Desktop? Opaque bietet die Möglichkeit, jedem Programm ein eigenes, sinnbezogenes Icon zuzuordnen. Auch die Laufwerke lassen sich ändern. Weiterhin kann man die Icons mit Wildcards definieren. Samt Icon-Editor und über 100 Icons.

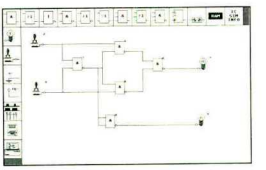
OPAQUE
SD 22 DM 15.-



LÄNDER DER WELT

Geographie-Lernprogramm mit leicht verständlicher Bedienung. 'Länder der Welt' vermittelt die Lage der einzelnen Länder auf der Weltkarte. Wo liegt z.B. Togo? 'Länder der Welt' hilft weiter und sorgt mit seinen Trainings- und Prüfungsfunktionen dafür, daß der Anwender diese Frage nicht ein zweites Mal stellen muß.

Länder der Welt [1]
SD 39 DM 15.-



ICSIM

Logik-Simulator

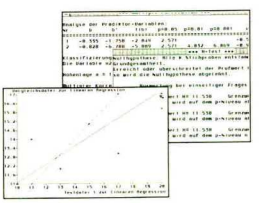
Das Programm simuliert das Verhalten von logischen Schaltungen. Bausteine und Verbindungen werden frei per Maus positioniert bzw. verbunden. Eine Schaltung läßt sich somit leicht austüfteln, testen und erst dann in die Praxis umsetzen.
Es sind die Logikbausteine nach DIN 40900 enthalten: AND, OR, NOT, NAND, NOR, XOR, RS-FF, KLEMME, LAMPE, SCHALTER, 0V und +5V.
Die Simulation wird als Impulsdiagramm oder Logiktabelle ausgegeben. Weiterhin liefert das Programm den Schaltplan und eine Liste der benötigten Bauteile.

ICSIM [1]
SD 25 DM 20.-

DATEI LOGIK

Datenbank, die einfache Handhabung und große Flexibilität miteinander vereint. So ist es für jedermann möglich, sich ohne große Anstrengung eine Datenbank nach seinen Vorstellungen aufzubauen. Mit Hilfe des integrierten Formulareditors kann eine individuelle Abfragemaske erstellt, mit dem Etiketteneditor das Layout von Aufklebern oder Karteikarten für jeden Aufgabenbereich festgelegt und mit der Mailmerge-Funktion mit den Daten auch Serienbriefe erstellt werden.

Datei Logik [1]
SD 36 DM 20.-



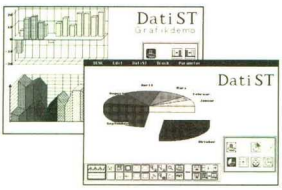
STatiST

modulares Statistik-Programmpaket

STatiST ist ein umfangreiches Paket zur Auswertung statistischer Daten. Zu jedem Prüfverfahren werden sämtliche Ergebnisse mit dem entsprechenden Wertungen und Kommentaren ausgegeben und, falls möglich, grafisch angezeigt. STatiST eignet sich für sämtliche, z.B. im Studium erforderlichen statistischen Auswertungen und macht das zeitaufwendige Rechnen per Hand und das Arbeiten mit Tabellen überflüssig.

STATIST [1] *(2 Disketten)*
SD 32a/b DM 30.-

SONDERDISK



DATIST

Präsentationsgrafik

Grafiken sagen oft mehr als 1000 Zahlen, daher sollte man sich bei der Auswertung von Daten auf DatiST verlassen. DatiST stellt Ihre Daten als, Kuchen-, Reihen-, Balken-, Säulen- und Liniengrafiken dar, entweder in 2D oder 3D, gefüllt oder als Rahmen. Lage, Größe, Dehnung und der Nullpunkt einer Grafik lassen sich frei mit der Maus einstellen; dafür sorgen die iconisierten Pop-Up-Menüs. Im 3D-Modus kann gar die räumliche Perspektive frei variiert werden. Die so erzeugten Grafiken, lassen sich beschriften (z.B. mit SI-GNUM!-Fonts) oder mit dem integrierten Zeichenprogramm bearbeiten, das vom Linienziehen über Blockoperationen bis hin zur Lupe alles bietet was man braucht. Um die Grafik zu Papier zu bringen bietet DatiST eine variable Druckeranpassung, die folgende Drucker unterstützt: Epson 9N/24N, NEC 24N, IBM PPR 24N, IBM AGM 24N, HP Laser, Atari-Laser!!.

DatiST [1]
SD 40 DM 25.-
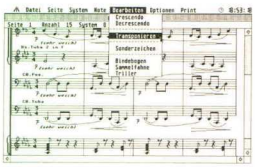


Special Paint 2

Grafik de Luxe

Grafikprogramm der Extraklasse. Neben den vielen nützlichen Funktionen zeichnet sich Special Paint vor allem durch seine Geschwindigkeit, seine bequeme Bedienung und seine Kompatibilität zu bekannten Malprogrammen aus. Special Paint bietet umfangreiche Blockfunktionen, Lasso, superschnelle Lupe, Maskierungen, Clippen, schnelle Bieg-, Zerr- und Drehoptionen, Animation und vieles mehr. Clipboardunterstützung, umfangreiche Textfunktionen (ladbare Fonts, Blocksatz, Zeilenumbruch).

Special Paint [1]
SD 21 DM 20.-



TRISTAN

Notensatzsystem

Für alle Musikfreunde, die nicht nur vom Blatt spielen, sondern auch aufs Blatt schreiben, bietet das Notensatzsystem TRISTAN die ideale Möglichkeit, ihre Noten professionell zu Papier zu bringen. Es lassen sich Partituren mit bis zu 100 Seiten mit max. 32 Notensystemen je Seite bearbeiten. Alle im klassischen Notensatz gebräuchlichen Zeichen lassen sich bequem mit der Maus edieren. Ebenfalls stehen mehrere Notenschlüssel, Sammelfahnen, Triller und Bindebögen zur Verfügung. Automatische Transponierfunktion. Ausdruck auf 9- und 24-Nadeldruckern, im 24-Nadelmodus in *maximaler* Druckerauflösung.

TRISTAN
SD 24 DM 25.-

FORMULA

2D-/ 3D-Plotter

Für mathematisch-wissenschaftliche Anwendung. Der eingebaute Formel-Interpreter beherrscht neben allen gängigen Operationen auch die Definition verschiedener Formeln in bestimmten Teilbereichen, logische Operationen und IF...THEN...ELSE. 3D-Grafiken lassen sich aus verschiedenen Blickrichtungen anzeigen und mit Schattierungen versehen.

FORMULA
SD 23 DM 20.-

Ultra-Disk

RAM-Disk-Tool

Ultradisk ist eine ultraschnelle, größenveränderbare, resetfeste und reset-residente RAM-Disk. Die Größe und die Laufwerkskennung kann frei bestimmt werden, und das alles ohne Inhaltsverlust und ohne den Rechner neu zu booten. Weiterhin enthalten ist ein ultraschneller Drucker-Spooler, der dafür sorgt, daß Sie weiterarbeiten können, während der Rechner noch Daten an den Drucker schickt. Auch darf der Maus-Speeder, die Zeitanzeige und der Bildschirmschoner nicht fehlen.

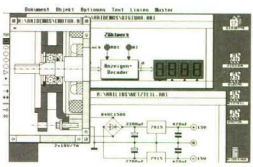
ULTRA-DISK
SD 33 DM 15.-

DER MOTOR

Der Motor erklärt mit zahlreichen Grafiken die Funktionsweise eines Verbrennungsmotors. Sehr anschaulich sind die bewegten Grafiken. Das gezeigte Wissen wird zusätzlich in einem Quiz abgefragt. Mit geregeltem Katalysator!!

DER MOTOR [1]
SD 20 DM 15.-

[1] nur für monochromen Monitor (SM 124)
[2] nur für Farbmonitor



ARIADNE

ARIADNE ist ein objektorientiertes Zeichenprogramm, d.h. Objekte können auch im Nachhinein ohne Auslösungsverlust verändert werden. Es bietet die Möglichkeit, jedes beliebige Grafikobjekt (mit Doppelklick) zu öffnen, worauf eine neue Zeichenebene bereitgestellt wird. Die Objekte auf dieser Ebene können dann wiederum geöffnet werden usw. Diese hierarchische Struktur eignet sich besonders zur Darstellung komplizierterer Dinge, z.B. Blockschaltbilder, Schaltungen etc.

ARIADNE [1]
SD 8 DM 15,-



ST-HIMMEL

Mit dem Programm kann der Anblick des Sternenhimmels für verschiedene Orte und Zeitpunkte berechnet werden. Ein ideales Programm für den Hobby-Astronomen.

Es zeigt:
- alle mit bloßem Auge (bei gutem Wetter) sichtbaren Sterne (~3000) mit Bezeichnungen, Helligkeiten und Entfernungen
- die mit bloßem Auge sichtb. Planeten
- den Mond mit seiner Phase
- die hellsten Sternhaufen und Nebel
- einen Kometen
- die Höhe der Sonne über oder unter dem Horizont
- die Namen der sichtbaren Planeten
- die verschiedenen Sternbilder
- den Tierkreis
- die Eigennamen von 190 Sternen (z.B. Großer Bär statt Ursa Maior)
- die Tag- und die Nachtseite der Erde auf einer Weltkarte.

ST-Himmel ist besonders anwenderfreundlich, so kann beispielsweise der Standort auf einer zoombaren Welt- bzw. BRD/DDR-Karte angeklickt werden.

ST-HIMMEL [1]
SD 38 DM 20.-

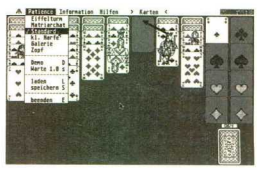
HARDCOPY II

Die erste Farb-Hardcopy für den ST

Universelles Hardcopy-Tool. S/W- und **Farb**-Hardcopy auf *allen* Druckern in *allen* Größen, Screendump auf Disk, Formatkonvertierung, läuft als Accessory, einfachste Bedienung, optimale Druckqualität.

HARDCOPY II
SD 15 DM 15.-

Sonderdisks unterliegen trotz des niedrigen Preises einem Copyright.



PATIENCE

Das Patiencespiel (patience = franz.: Geduld) stammt aus Frankreich. Es ist ein Kartengeduldsspiel, das hohe Aufmerksamkeit erfordert. Es schult das Denkvermögen, fördert die Kombinationsfähigkeit, entspannt und beruhigt zugleich. Im Programm sind folgende Patience-Varianten enthalten: Standard, Eiffelturm, Zopf, Kleine Harfe, Matriarchat und Bildergalerie. Patiencen verfolgen das Ziel, Karten nach bestimmten Regeln sortiert abzulegen. Sind alle Karten abgelegt, gilt die Patience als gelöst. Das Programm gibt auf Wunsch Lösungsvorschläge. Eine ausführliche Anleitung zu den Patiencen fehlt ebenfalls nicht.

Patience [1]
SD 11 DM 15,-

Programmierer aufgepaßt!!

Haben Sie nicht auch ein Programm geschrieben, das in diese Serie paßt? Sonderdisketten enthalten leistungsstarke Programme aus allen Bereichen zu günstigen Preisen. Als Autor erhalten Sie eine attraktive Umsatzbeteiligung. Lassen Sie doch mal was von sich hören.

MAXON Computer
Idee Sonderdisk
Industriestr. 26
6236 Eschborn

Weitere Sonderdisks

01	TOS 1.0	nicht mehr lieferbar
02	RCS 1.4	15,-
03	Extended VT52 [1]	15,-
04	Lovely Helper	15,-
05	Accessories	15,-
06	NIKI [1]	15,-
07	VirusEx	15,-
09	Legende [2]	15,-
10	Quinemac [1]	15,-
12	MagicBox ST	15,-
13	Robotwar [1]	15,-
16	Easy Adress [1]	15,-
17	IconDesign	15,-
19	MAKI [1]	15,-
26	Hauskasse [1]	15,-
28	Master Etikett [1]	15,-
30	Würfelpoker	15,-
31	EasySTat [1]	25,-
34	Fußball [1]	15,-

Sonderdisk-Bestellung

Sonderdisks können Sie telefonisch oder schriftlich bestellen, oder nutzen Sie einfach die Bestellkarte im Heft.

Bei Nachnahme zzgl. DM 4.- Gebühr, Versandkosten DM 5.- (Ausland DM 10.-)

MAXON Computer
Schwalbacher Str. 52
6236 Eschborn
Tel: 06196/481811

SONDERDISK

Sonderdisks beinhalten Programme aus den verschiedensten Bereichen (z.B. Utilities, Grafik, Schulung, Spiele). Sonderdisks ermöglichen den Usern, qualitativ hochwertige Software zu einem kostengünstigen Preis zu erhalten. Im Preis ist eine Beteiligung der Autoren enthalten.

INSERENTENVERZEICHNIS

AB-COMPUTER ..159
AKZENTE 47
ALTEX 149
APPLICATION.....2
AS-DATENTECH. 155
ASTON 127
BEGEMANN &
NIEMEYER 107
BELA COMPUT.....55
BOSSART SOFT ...81
CALTEC.....33
CATCH COMP.....88
CCD 13
CEW COMP 77
CIECHOWSKY 159
CREATIV-VIDEO .149
COMPEDO.....149
CSH 124
CSR 154
CTN 153
CWTG.....95
DATA 2000 77
DIGITAL DATA 129
DREWS EDV.....95
DUFFNER COMP.170
EICKMANN 189
ELZNER 81
FISCHER.....121
FSE.....107
GALACTIC 107
GEERDES SOFT.153
GENG TEC.....95
GMA-SOFT 170
GÜNTHERBERG ...25
HAASE 124
HANISCH MOD. ..159
HCS 25
HEBER-KNOBL. ..155
HEIER.....155
HEIM 26,27,39,49
.....85,92,95,99,139
.....143,161,185,196
HERBERG 110,111
HERGES 154
HL-COMPUTER...155
HÖFER 154
HÖGER.....77
HORN-EDV 152
HESSE+HERWIG.127
ICP-VERLAG 37
IDEE GMBH 152
IDEE SOFT 153
IDL 40,41
IKS 114
KAHLERT 155
K+L 152
KOHLER 127,155
KUHLMANN 149
LACOM 114
LIGHTHOUSE 9
MAASSEN 152
MARKERT 114
MAXON 16,17,30
...89,103,135,146,173
MCS 153
MIELKE 25
NOVOPLAN 71
NÖTHE 81
OLUFS.....155
OMIKRON 190
PD-EXPRESS 127
PORADA 152
PRINT TECHNIK....11
RICHTER.....47
RTS-ELEKTR.....159
RUPP.....153
RÜCKEMANN 154
SCHLICHT 152
SCHLICHTING.....81
SCHNEIDER 123
SCHÖN.....152
SEBASS 77
SCILAB 77,124
SENDER.....51
SHIFT 29
SOFTHANSA 154
SSD SOFTWARE ..77
SW SOFTWARE ..152
TAS 153
TETRA 71
THOBE 152
TK COMPUTER...123
TKR.....149
TMS 23
TORNADO 81
TRADE IT 127
UNICOMP 23
VORTEX GMBH.....45
WACKER 113
WAVE 154
WBW SERVICE ...153
WEESKE 117
WITTICH 131
WOHLFAHRT-
STÄTTER.....60,61



Neuheiten

I.D.A.
Integrierte Datenbankanwendungen für den ATARI ST/TT

Schnelles, ausbaufähiges Multitasking-Datenbanksystem
I.D.A. läuft in leicht bedienbarer Fensterumgebung mit Umschaltmöglichkeit von einer Aufgabe zur nächsten (Multitasking). Beispielsweise muß eine Maske nicht erst vollständig ausgefüllt und abgespeichert werden, wenn man schnell nach einer Telefonnummer sucht. Auch Spezial-Anwendungen lassen sich einfach in das I.D.A.-System integrieren und sind jederzeit verfügbar.

Bei I.D.A. bekommen Sie alles aus einer Hand:
• Variables Datenbanksystem • Schneller Zugriff auch auf große Datenmengen • Parallelbetrieb von Datenbanken, Masken, Listen und Zusatzprogrammen (Multitasking) • Multiuserbetrieb in Netzen (z.B. Bionet, ATARI Net) • „Persönliche" Klemmbretter für jede Datei • Mehrfachsortierung auf 4 Ebenen • Leistungsfähige Wahl- und Rechendefinitionen • Drucken im WordPlus-Format • Spezialanwendungen benutzen I.D.A.-Funktionen und -Oberfläche komplett mit

Programmierbar in zwei Stufen: • durch Algorithmen, eine Programmierweise, die BASIC ähnelt. Als Rechenoperationen stehen auch sin, cos, zur Verfügung, gerechnet wird in doppelter Genauigkeit • durch SPC-Modula-2, um auch komplexe Anwendungen zu programmieren, dazu gibt es eine eigene Version von DBProg.

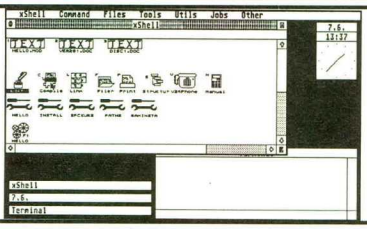
Frei definierbare Masken mit folgenden Elementen: • Titel für Überschriften • Items vom Type String, Integer, Langzahl, Datum, Zeit, Geld, Extern • Graphikelemente • Makrobutton, um einen kompletten String auf einen Button zu legen • Frei belegbare Funktionsbuttons, bedienbar mit Maus- und Funktionstasten. Sie lösen auch evtl. definierte Algorithmen aus • Alle Fonts können in verschiedenen Größen (7 bis 20 Punkt / wenn vorhanden auch GEM-Fonts) sowie in verschiedenen Farben dargestellt werden • Verbundmasken, die aus den Elementen der Dateimasken zusammengesetzt sind • Zu jeder Maske gehört ein Listenfenster, das die Maskendaten in Listenform darstellt. Die Liste ist editierbar (Browse-Modus) • Jedes Maskenfenster hat ein eigenes Clipboard, um Datensätze zu sammeln

Dateien, Verbünde, Algorithmen, persönliche Einstellungen werden in einem „Worksheet" zusammengefaßt – auch mit unterschiedlichen Zugriffsrechten (für Sekretärin, Mitarbeiter, Chef), regelbar in Paßwortstufen. Bei besonderen Sicherheitsanforderungen wird die Datei automatisch verschlüsselt. Mirrordatei sorgt auf Wunsch nach jeder Datenbankänderung für ein Spiegelbild auf einer anderen Partition: wichtige Sicherungsmaßnahme für Multiuser. Exportmöglichkeiten zwischen Dateien. Adreß- oder Lagerdaten können automatisch in Rechnungsdatei übertragen werden – mit Algorithmen programmierbar.

DM 398,- Unverbindlich empfohlener Verkaufspreis

SPC MODULA-2

Der SPC Modula-2 Compiler ist eine Portierung des original ETH Zürich Single-Pass-Compilers. Der Compiler hat einen typischen Durchsatz von 5000 Zeilen pro Minute und erzeugt 68000-Maschinencode. Der Code kann entweder mit einem Linker zu Stand-Alone Programmen gelinkt werden oder in das Load-Time-Linking System der Entwicklungsumgebung einbezogen werden. Dadurch enfällt die langwierige Link Phase und das Programm kann sofort nach dem Compilieren gestartet werden.



Das SPC-Modula-2 System hat eine komfortable und multitaskende grafische Shell, die sowohl mit der Maus als auch mit der Tastatur bedient werden kann. Der spezielle Modula-Editor ergänzt auf Wunsch nach der Eingabe von wenigen Buchstaben eines Schlüsselwortes dieses selbständig. Das Online-Handbuch können Sie immer anschauen. Nach dem Editieren eines Quelltextes sorgt die bedingte Compilierung mit automatischer Make-Funktion für die Übersetzung.

Der symbolische Debugger zur Fehlersuche in Programmen beim Testen arbeitet mit 5 Fenstern (Quellcode, Variablen, Prozeduren, usw.). Ausführliche Bibliotheksmodule, Beispiele und alle nur denkbaren Utilities sind auf ingesamt fünf doppelseitigen Disketten gespeichert. Basic-Library für einfache Umsetzung von Gfa-3 Programmen in Modula. Einfache Fensterprogrammierung mit der portierbaren Windowschnittstelle SSWiS. SPC Modula-2 unterstützt den 68881 Coprozessor und PAK68. Das Resource Construction Set (Digital Research) RCS 2.1 und ein Modula-Kurs wird mitgeliefert.

Das SPC Modula-2 System arbeitet auf allen ATARI ST-Konfigurationen mit Monochrom-, Farb- oder Großbildschirm. Ausführliches englisches oder deutsches Handbuch.

Hotline und Kundenbetreuung: Compo Software GmbH, Tel. 06551-6266

DM 398,- Unverbindlich empfohlener Verkaufspreis

BESTELL-COUPON

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

Bitte senden Sie mir: ___I.D.A.-Datenbanksystem à 398,- DM
___SPC Modula-2 à 398,- DM
zzgl. Versandkosten DM 6,- (Ausl. DM 10,-) unabhängig von der bestellten Stückzahl

Name, Vorname _____
Straße, Hausnr. _____
PLZ, Ort _____
(Oder benutzen Sie die eingeheftete Bestellkarte)

In Österreich:
RRR EDV GmbH
Dr. Stumpfstraße 118
A-6020 Innsbruck

In der Schweiz:
Data Trade AG
Landstraße 1
CH - 5415 Rieden-Baden

In der nächsten ST-Computer lesen Sie unter anderem

Arabesque Professional

Eines der besten Grafikprogramme des Jahres 1990, Arabesque, hat würdigen Nachwuchs bekommen. „Arabesque Professional" nennt sich der Nachfolger. Welche Unterschiede sich zum normalen Arabesque ergeben, und ob es sich lohnt, das Programm zu kaufen, erfahren Sie in der nächsten Ausgabe der ST-Computer.

Maxidat

Daß ein Datenbankprogramm nicht unbedingt mehrere hundert Mark kosten muß und dennoch eine enorme Leistungsfähigkeit an den Tag legen kann, beweist Maxidat. Nach dem relationalen Verwaltungsprinzip aufgebaut, können aber auch assoziative Suchkriterien eingebaut werden. Maxidat erlaubt das Einbinden von Bilddateien, hat einen Statistikteil und errechnet automatisch Diagramme. Lesen Sie unseren Testbericht in der März-Ausgabe der ST-Computer.

Überraschung

Wenn wir schon von Grafikprogrammen berichten, darf auch das neueste nicht fehlen. Der Name? Der bleibt eine Überraschung. Was es kann? Vollflächenfenstertechnik, Autoscroll, Online-Lupe, beliebig viele Fenster, TT-Tauglichkeit, Schnittstelle zu Signum!, dynamische Speicherverwaltung, nur 100 kB lang, alle Funktionen über Tastatur abrufbar. Jetzt sind Sie sicher neugierig geworden.

TT-Programmierung

Der TT ist in vielen Bereichen noch unerforscht. Wir durchbrechen diese Schallmauer und bringen Ihnen knallharte Fakten zur TT-Programmierung. Mehr verraten wir allerdings auch hier noch nicht - immer einen Schritt voraus. Natürlich mit der neuen ST-Computer im nächsten Monat.

Die nächste ST-Computer erscheint am Fr., dem 22.02.91

Fragen an die Redaktion

Ein Magazin wie die ST-Computer zu erstellen, kostet sehr viel Zeit und Mühe. Da wir weiterhin vorhaben, die Qualität zu steigern, haben wir Redakteure eine große Bitte an Sie, liebe Leserinnen und Leser: Bitte haben Sie Verständnis dafür, daß Fragen an die Redaktion nur **donnerstags von 14^{00}-17^{00} Uhr** unter der Rufnummer 06196/481814 telefonisch beantwortet werden können.
Natürlich können wir Ihnen **keine** speziellen Einkaufstips geben. Wenden Sie sich in diesem Fall bitte an einen Fachhändler. Wir können nur Fragen zur ST-Computer beantworten.

Vielen Dank für Ihr Verständnis!

Impressum ST Computer

Chefredakteur: Harald Egel (HE)

Redaktion:
Harald Egel (HE)
Joachim Merz (JM)
Dieter Kühner (DK)
Martin Pittelkow (MP)

Redaktionelle Mitarbeiter:
C.Borgmeier (CBO)
Claus Brod (CB)
Ingo Brümmer (IB)
Derek dela Fuente (ddF)
Stefan Höhn (SH)
Raymund Hofmann (RH)
Claus P. Lippert (CPL)
Thorsten Luhm (TL)
Chr. Schormann (CS)
U.Seimet (US)
R.Tolksdorf (RT)
Thomas Werner (TW)

Autoren dieser Ausgabe:
D. Brockhaus
M. Chakravarty
H.Günterberg
K.Elsbernd
A.Hollmann
W.Keller
S. Krüppel
D. Rabich
H.J.Richstein
W.Waßmus

Auslandskorrespondenz:
C.P.Lippert (Leitung), D.Dela Fuente (UK)

Redaktion: MAXON Computer GmbH
Postfach 59 69
Industriestr. 26
6236 Eschborn
Tel.: 0 61 96/48 18 14, FAX : 0 61 96/4 11 37

Verlag: Heim Fachverlag
Heidelberger Landstr. 194
6100 Darmstadt 13
Tel.: 0 61 51/5 60 57, FAX : 0 61 51/59 10 47 + 5 60 59

Verlagsleitung:
H.J.Heim

Anzeigenverkaufsleitung:
U.Heim

Anzeigenverkauf:
K.Margaritis

Anzeigenpreise:
nach Preisliste Nr.6, gültig ab 2.1.91
ISSN 0932-0385

Layout:
Manfred Zimmermann (vtl.), Angela Heischmann

Titelgestaltung:
Axel Weigend

Fotografie:
Martin Pittelkow

Illustration:
Manfred Zimmermann

Produktion:
B.Kissner

Druck:
Frotscher Druck GmbH

Lektorat:
V.Pfeiffer

Bezugsmöglichkeiten:
ATARI-Fachhandel, Zeitschriftenhandel, Kauf- und Warenhäuser oder direkt beim Verlag

ST Computer erscheint 11 x im Jahr
Einzelpreis: DM 8,-, ÖS 64,-, SFr 8,-
Jahresabonnement: DM 80,-
Europ. Ausland : DM 100,- Luftpost : DM 130,-
In den Preisen sind die gesetzliche MWSt. und die Zustellgebühren enthalten.

Manuskripteinsendungen:
Programmlistings, Bauanleitungen und Manuskripte werden von der Redaktion gerne angenommen. Sie müssen frei von Rechten Dritter sein. Mit seiner Einsendung gibt der Verfasser die Zustimmung zum Abdruck und der Vervielfältigung auf Datenträgern der MAXON Computer GmbH. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.

Urheberrecht:
Alle in der ST-Computer erschienenen Beiträge sind urheberrechtlich geschützt. Reproduktionen gleich welcher Art, ob Übersetzung, Nachdruck, Vervielfältigung oder Erfassung in Datenverarbeitungsanlagen sind nur mit schriftlicher Genehmigung der MAXON Computer GmbH oder des Heim Verlags erlaubt.

Veröffentlichungen:
Sämtliche Veröffentlichungen in der ST-Computer erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes, auch werden Warennamen ohne Gewährleistung einer freien Verwendung benutzt.

Haftungsausschluß:
Für Fehler in Text, in Schaltbildern, Aufbauskizzen, Stücklisten usw., die zum Nichtfunktionieren oder evtl. zum Schadhaftwerden von Bauelementen führen, wird keine Haftung übernommen.

© Copyright 1991 by Heim Verlag

HDPlus 5.02 **198 DM**

HDPlus ist die speziell für unsere Festplatten entwickelte Treibersoftware mit allem, was zur komfortablen Arbeit mit Massenspeichern und deren Wartung nur vorstellbar ist. Die wichtigsten Funktionen sind einfach zu bedienen, aber auch für den Experten ist HDPlus das universelle Werkzeug. Auf Datensicherheit wurde besonderer Wert gelegt, so können Sie den Rootsektor sichern, Partitionen schreibschützen, oder den Zugriff per Paßwort schützen. Booten verschiedener Accessories von beliebigen Partitions, beliebig viele Partitionen u.V.m.

eickmann Harddisks EX 30·60·120 MB

Alle eickmann Festplatten werden mit dem neuen HDPlus 5.02 und HDPlus-UTILITIE ausgeliefert. (Fast Filemover von First GbR, Optimizer von Projekt.FPS, Hard Disk Utilitie von Application Systems) Und selbstverständlich anschlußfertig, formatiert, partitioniert, autobootfähig.

z.B.: **EX 60/L** **1598 DM**
24 ms Zugriffzeit, extrem leise, Autopark

EX 120/L **2498 DM**
24/24 ms Doppellaufwerk, extrem leise, Autopark

Minidrive Festplatten 40·60·75·80·100·200 MB

Die schnellen SCSI-Platten im Mini-Gehäuse mit der starken Leistung. Hardwaremäßiger Schreibschutz. Die eickmann Mini Drives wurden gezielt auf Platzeinsparung und freie Plazierungsmöglichkeiten hin konzipiert. Das Gehäuse ist im Design der Mega-Serie gehalten, aber kaum halb so groß!

z.B.: **Minidrive 60** **1498 DM**
24 ms Zugriffzeit, Single-Laufwerk, SCSI, Autopark

Minidrive 200 F **3498 DM**
15 ms Zugriffzeit, Single-Laufwerk, 48K-Cache, SCSI, AP.

Megadrive Einbauplatten für Mega ST

z.B.: **Megadrive 60** **1398 DM**
24 ms Zugriffzeit, SCSI, Autopark

Megadrive 100 F **2098 DM**
18 ms Zugriffzeit, 16K-Cache, SCSI, Autopark

Wechselplatte EX 40 W
EX 40 W + 44MB Medium **1998 DM**
25 ms Zugriffzeit, Wechselplatte

Wechselplatte + Festplatte in einem Gehäuse

z.B.: **EX 40 W/75 F + Medium** **3398 DM**
25/18 ms Wechselplatte + eingeb. 75 MB Platte, Autopark

EX 40 W/80 + Medium **3198 DM**
25/24 ms Wechselplatte + eingeb. 80 MB Platte, Autopark

EX 40 W/200 F + Medium **5098 DM**
25/15 ms Wechselplatte + eingeb. 200 MB Platte, Autopark

eickmann EM 124 Multi **498 DM**
640x400, 640x200, 320x200 Graustufenmultisync

eickmann FolioTalk **98 DM**

Interfaceprogramm und Verbindungskabel zwischen Atari ST und Portfolio. Die Übertragungssoftware läuft als Accessory oder GEM-Anwendung und ermöglicht den einfachen und sicheren Datenaustausch zwischen ST und Portfolio. Parallele Schnittstelle erforderlich.

Portfolio Komplettpaket **598 DM**

Der kleinste PC der Welt. Inkl. Parallel-Interface und FolioTalk (Schnittstelle zum ST)

weitere Angebote und Preise auf Anfrage

ET-der eickmann Tower Preise auf Anfrage

Der Tower macht Platz auf dem Schreibtisch!
Computer (ST/TT), Festplatte, Wechselplatte, Diskettenlaufwerke, Grafikkarte, Beschleuniger, alternative Betriebssysteme (z.B. Spectre GCR, MS DOS-Emulatoren), Laserinterface, DMA-Buffer, DMA-T-Switch, MS DOS Tastaturmodul, Modem, u.V.m. finden im neuen Gehäuse Platz – unter dem Schreibtisch.
Einfach einschalten und mit der Arbeit beginnen. Auf Ihre zig-fach Steckdosen werden Sie verzichten müssen, denn die Grundkonfiguration, Tower, Bildschirm und Drucker, kommt mit einem Dreifachstecker aus.

mouseWare
MOUSE CONTROL





mouseWare DESIGNER MAUS 98 DM

Das optimale Arbeitstier für höchste Ansprüche in den Bereichen DTP, Bildverarbeitung, Grafik und CAD.
Die ergonomisch richtige Form macht die Maus zur sensiblen Fortsetzung der Hand.

►ergonomisch◄ ►schnell◄ ►langlebig◄





mouseWare PAD 19,50 DM

Die Spezialbeschichtung ist genau auf die Gleitflächen der Maus abgestimmt. Mit diesem Pad gleitet die Maus wie auf einem Luftkissen und stoppt exakt dort, wo Sie es wünschen.

►abwaschbar◄ ►flächenoptimiert◄
►nahezu unverwüstlich◄

ET- der eickmann Tower

NICHT in den Tower gehören:
Tastatur, Monitor,
Scanner und Drucker.
Ihre gesamte restliche Hardware zieht gern in diesen Tower ein.

- ► vollklimatisiert
- ► ruhige Lage
- ► zentrale Energieversorgung
- ► repräsentative Architektur





Bei der individuellen Ausstattung Ihres eickmann Towers berät Sie kompetent:

Das Planungsteam
von eickmann computer

eickmann computer

eickmann computer • In der Römerstadt 249 / 253
6000 Frankfurt / Main • Telefon 069 / 76 34 09 • Fax: 069 / 7 68 19 71

EIN GUTER FREUND

»Mortimer ist ein wirklich gelungenes Programm, das man jedem ans Herz legen kann.«
PD-Journal 8/90, S. 26

»Die Firma OMIKRON hat sich offenbar ganz am Endbenutzer orientiert, und das hat zu einem wirklich guten Ergebnis geführt.«
XEST (österreichisches ATARI-Magazin) 2/90, S. 18

»... ein Butler, von dem man sich wirklich gern verwöhnen läßt.«
ST-Magazin 5/90, S. 21



MORTIMER PLUS

Für viele unserer Kunden ist Mortimer ein guter Freund geworden. Er war stets da, wenn er gebraucht wurde; verstand sich gut mit allen anderen Programmen – und packte immer kräftig mit an. In diesem Jahr hat er nochmals kräftig dazugelernt. Und ist so – wie wir meinen – ein noch besserer Freund geworden. Näheres erfahren Sie im Prospekt oder telefonisch.

Mortimer Plus DM 129,– Upgrade DM 60,–

Mortimer DM 79,– (Upgrade bis zum 31.12.90 DM 50,–)

NEUHEITEN

+ Texteditor mit automatischem Zeilenumbruch, Blocksatz und Menüzeile
+ Speichermonitor: Daten retten nach Absturz beliebiger Programme
+ Dateiauswahlbox ins Betriebssystem eingebunden
+ erweiterter Tastaturmakro-Treiber
+ lauffähig auf ATARI TT
+ Uhrzeit einstellen & über Kaltstart retten
+ trotzdem weniger als 80 Kbyte – kein Problem selbst für einen 520 ST

Mortimer Plus kann natürlich alles, was Mortimer kann – und das ist eine ganze Menge.

OMIKRON.Soft + Hardware
Sponheimstr. 12a · D-7530 Pforzheim
Telefon 0 72 31 / 35 60 33

OMIKRON.

XEST, Webgasse 21, A-1060 Wien
OMIKRON. France, 11, rue dérodé, F-51100 Reims
Elecomp, 11, avenue de la gare, L-4131 Esch/Alzette
Jotka Computing, Postbus 8183, NL-6710 AD Ede