DM 7,- Sfr 7.- Ös 56.-

April '89 Nr. 4

AMIGA

KICK START

COMPUTER MAGAZIN

&{%¿£@!-|§^#?

Im Chaos der Computersprachen

RESETFEST

Residente Programme

CPU-TIME

14MHz 68000 vs. 68020

LINKSVERKEHR

BBC-Micro-Emulation für AMIGA



Tower-AMIGA • Autoboot-Festplatten • NEWS
Grundlagen • Tips&Tricks • KICKS • Spiele

Die erste Festplatte mit Fast-File Autoboot!

Hi-SPEED PLATTEN
werden die 40Q und 80Q Hardcards genannt. Durch einen integrierten Cache-Speicher erreichen diese Platten Zugriffszeiten von 11 Millisekunden! Weitere Merkmale sind das patentierte Autopark System (Airlock) und die doppelt hohe Lebensdauer (MTBF 50.000 Std.) gegenüber herkömmlichen Platten.

SCSI BUS
überträgt die Daten per DMA mit max. 4 MBytes/sec. auf das Controllerboard.

SCSI CONTROLLER
Autokonfigurierend mit 0 Waitstates. ANSI X3T9.2 kompatibel.

AUTOBOOT AMIGA & PC/AT
direkt von der Fast-File Partition bietet bisher nur der GVP SCSI Controller. Das Einrichten einer langsamen Boot-Partition entfällt, auch die PC oder AT-Karte kann jetzt direkt von der Impact Hardcard gebootet werden.



EXTERNER SCSI-BUS
für den Anschluß von bis zu 7 weiteren SCSI-Einheiten (z.B. Optische Platte).

100 MEGABYTE
bietet GVP auf einer Hardcard durch Verwendung modernster Platten mit minimalen Abmessungen. IMPACT Hardcards sind in folgenden Größen lieferbar:

IMPACT™ A2000-HC/20
A2000-HC/30
A2000-HC/40Q
A2000-HC/45
A2000-HC/80Q
A2000-HC/100

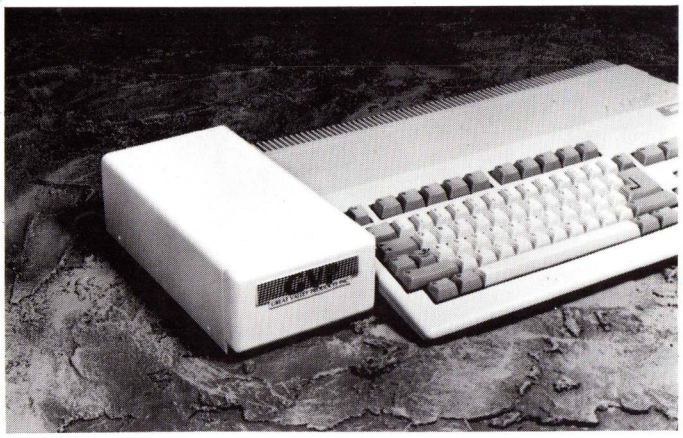
Q = Quantum Drive

16 BIT DMA ZUGRIFF
vom Cache-Speicher zum Amiga Board gewährleistet optimale Geschwindigkeiten ohne andere Amiga-eigene DMA Operationen (Blitter, Sound, etc.) zu beeinflussen.

CACHE-SPEICHER
puffert die Daten auf dem Controller und sorgt damit für einen reibungslosen Datentransfer.

EINFACHE INSTALLATION!
Die SCSI Autoboot Hardcard IMPACT ist eine professionelle Festplattenlösung für den Amiga 2000. Durch die direkte Montage der Platte auf der Steckkarte wurde Raum gespart und die Laufwerkseinschübe bleiben frei. Deutsche Installation und Workbench 1.3 im Lieferumfang.

IMPACT SCSI Platten für Amiga 500
Lieferbar in den gleichen Kapazitäten wie die A2000 Hardcards. Eigenes Netzteil mit Lüfter und internem Steckplatz für Speichererweiterung 2 Megabyte Fast-Ram und Autoboot Karte.



SCSI Controller mit 2 MB Fast-Ram
Gleiche Daten wie Hardcard jedoch ohne Festplatte. Stattdessen ist die Karte mit 2 MByte autokonfigurierendem Fast-Ram bestückt. Der Controller ist mit oder ohne RAM-Chips lieferbar.



GVP Produkte erhalten Sie im Fachhandel oder beim Distributor Microtron (Schweiz) und DTM

GVP
GREAT VALLEY PRODUCTS INC.

MICROTRON
COMPUTERPRODUKTE
Postfach 69 Bahnhofstr. 2
Tel. 032 872429 Fax 032 872482
CH-2542 PIETERLEN

Poststraße 25
6200 Wiesbaden-Bierstadt
(0 61 21) 56 00 84
fax (0 61 21) 56 36 43

DTM
Werbung und EDV GmbH

Die Messe der Messen

Es ist wieder soweit, die CeBIT '89 öffnet vom 8. bis 15. März ihre Pforten. Seit der Gründung vor vier Jahren ist die CeBIT immer populärer geworden. Jährlich werden es mehr Aussteller, die die aus der ganzen Welt kommen, um ihre neuen Soft- und Hardwareprodukte dem ebenfalls internationalen Publikum zu präsentieren.

Aber nicht nur die Ausstellerzahlen, sondern auch die Besucherzahlen sind von Jahr zu Jahr gestiegen. So kamen zur letzten CeBIT über 400000 Besucher. Immerhin fast 15% mehr als im Jahr davor. Diese Zahl verdeutlicht, daß nicht nur immer mehr Firmen sich des Werkzeugs Computer bedienen, sondern daß auch mehr und mehr Privatleute wissen wollen, was es Neues auf dem Computermarkt gibt. Aber nicht nur das Neue ist von Interesse. Auch die Vielfalt des Angebotenen lockt viele Leute an, die in der Absicht kommen, sich einen Computer oder neue Software zu kaufen.

Auf der letzten CeBIT ging der Trend aber in eine ganz andere Richtung. Man stellte nicht mehr neue, sondern zukünftige Produkte vor, was meiner Meinung nach nicht unbedingt Sinn einer Computer-Messe ist. Bei Commodore war das genauso. Als Neuheiten wurden die AT-Karte, ein Hires-Monitor und der AMIGA 2500 UX und verschiedenes andere vorgestellt. Was die AT-Karte betrifft, wurden bis zum heutigen Tag ausschließlich Testversionen angefertigt, die allerdings hoffen lassen, daß man sie spätestens nach der CeBIT '89 auch käuflich erwerben kann. Anders beim Hires-Monitor und der UNIX-Maschine. Von ihnen hörte man lediglich auf der CeBIT '88, und im Monat danach konnte man einige spekulative Zeilen in verschiedenen Computerzeitschriften darüber lesen. Bis zur letzten Entwicklerkonferenz wurde ich daher den Eindruck nicht los, daß nicht nur diese beiden Produkte auch auf der nächsten Messe nichts weiter als zukünftige Neuheiten sein werden, und ich dachte, daß es vielleicht bis zur übernächsten Messe dauern wird, bis man sie endlich kaufen kann. Aber wie sie selbst in diesem Heft lesen können, hat sich bei Commodore doch einiges getan. So sind die ersten AMIGA-UNIX-Entwicklergeräte noch für dieses Frühjahr angekündigt. Auch der Hires-Monitor war Bestandteil mancher Vorträge.

Auch wenn Termine nicht eingehalten werden, sorgt Commodore auf diese Weise dafür, daß die Produkte im Gerede bleiben. Ein kleiner Trost. Für Sie bedeutet das aber dennoch, abzuwarten, was die kommende CeBIT an Neuheiten zu bieten hat, und welche Termine für die einzelnen Produkte gemacht werden.

Auf jeden Fall sind wir wieder in Halle 7, Stand E42 zu finden, um das Messegeschehen für Sie in Wort und Bild festzuhalten.

Jürgen Haage

INHALT

Aktuelles

NEWS
AMIGA-Neuigkeiten 6

Biochips
Neue Erfolge in der Biotechnik 67

Spiele-NEWS
Das Neueste vom Spielemarkt 115

Software

Sherlock
Der Virendedektiv 12

Hardware

Prozessorwettstreit
14MHz 68000 vs. 68020 25

Tower-AMIGA
Der AMIGA im professionellem Design 31

Autoboot-Festplatten
Vortex vs. Quantum 41

Linksverkehr
BBC-Micro-Emulation 72

Listing

Im Dunstkreis des Neuntöters
Steuercodes zur Druckerzähmung 96

Grundlagen

Programmiersprachen
Ein Überblick 16

Kalte Wickel
Therapie für defekte RAM-Bausteine 48

WordPerfekt Praxis
Teil 2: Mailmerge 52

Resetfest
Die Theorie residenter Programme 55

Die Oberflächen des AMIGA
Grundlegendes über CLI und Workbench 62

Floppy im Griff
Teil 2: Die Devices 99



Der TOWER-AMIGA

Einen AMIGA in einem TOWER-Gehäuse bietet die Firma CSS an. Daß es sich dabei nicht um einen normalen AMIGA handelt, war zu erwarten. Was sich unter dem TOWER-Gehäuse an interessanter Hardware versteckt und was die Hardware zu leisten vermag, verrät unser ausführlicher Test. Oder trügt der TOWER-Schein? Mehr ab Seite

31

BBC-Micro-Emulation

Einer der verbreitetsten Rechner in England war und ist der BBC-Micro. Eine englische Softwarefirma hat eine Emulation auf dem AMIGA herausgebracht. Doch wie verhält es sich mit der Emulation, ist es wieder so ein Reinfall wie mit den diversen C64-Emulatoren? So viel sei hier verraten: Das BBC-Micro-BASIC schlägt das AmigaBasic in Sachen Geschwindigkeit um Längen. Mehr ab Seite

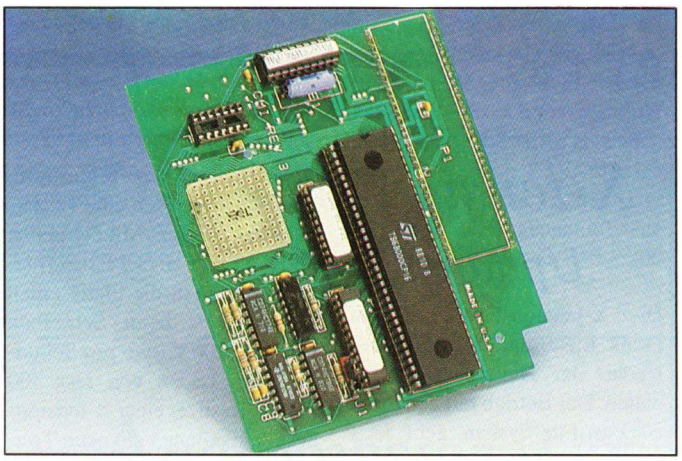
72

4 '89

Prozessorwettstreit 14 MHz 68000er vs. 68020

Ein mit doppelter Taktfrequenz betriebener 68000-Prozessor tritt zum Duell mit einem 68020-Prozessor an. Ob der beschleunigte 68000 mit dem 68020 mithalten kann, was er für Vor- bzw. Nachteile besitzt, wie es sich mit dem Mathe-Coprozessor 68881 verhält, erfahren Sie im ausführlichen Testbericht ab Seite

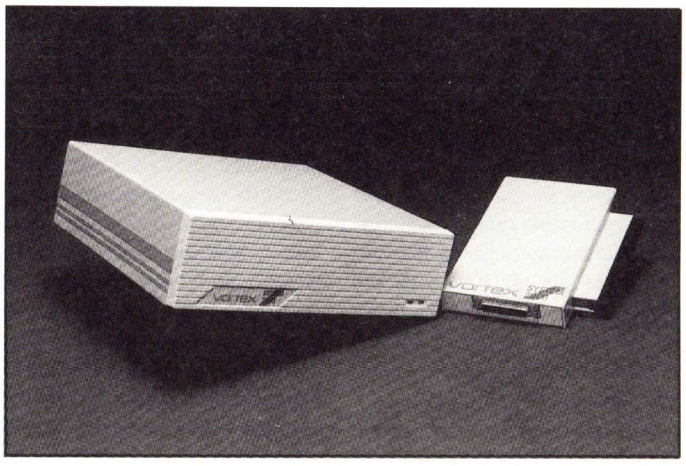
25



Autoboot-Festplatten

Bezüglich der Technik haben Commodore und die Entwickler etwas geschlafen. Erst ab der neuen Betriebssystemversion 1.3 ist das Booten von Festplatte möglich. Zwei Festplatten standen der Redaktion zum Test zur Verfügung. Die GVP SCSI Hardcard mit 80 MByte Speicherkapazität für den AMIGA 2000 und die VORTEX System 2000 für den A500 bzw. A1000. Letztere ist auch unter der Betriebssystemversion 1.2 autobootfähig. Was die neue Generation der AMIGA-Festplatten leistet, erfahren Sie ab Seite

42



KICKS FÜR INSIDER

Maus aus
Die Maus vom Bildschirm verbannt 76

Dem C-Compiler Beine gemacht
Schnelle Mathe-Routinen 83

Bis an's bittere Ende
Apfelmännchen in Assembler 86

MIDI IN / MIDI OUT
Casio Library Utility 90

Spiele

Spiele-News
Das Neueste vom Spielemarkt 115
TOP 12 117
Wall Street Wizard 119
Cosmo Piraten 119
Gauntlet II 120
Double Dragon 120
Tracker 121
Mickey Mouse 121
Zak Mc Kracken 122
Billard Simulator 122
Hollywood Poker Pro 123
International Karate Plus 123
Leben und sterben lassen 124
Advanced Ski-Simulator 124
Bozuma 125

Spielelösungen
FISH 111
Ultima III 113

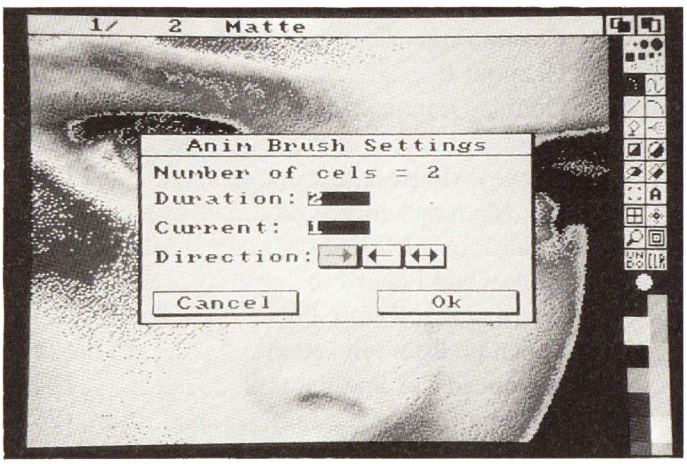
Rubriken

Bücher 40
Tips&Tricks 36
Einkaufsführer 5
Inserentenverzeichnis 125
Vorschau 130

NEWS

KICKSTART 4´89

DPAINT III



Das neue DPaint III besitzt jetzt auch Animationsfunktionen.

Kein Malprogramm ist so verbreitet auf dem AMIGA wie Deluxe Paint von Electronic Arts. Nach DPaint I und II bringt Electronic Arts jetzt die dritte Generation des Malprogramms auf den Markt - Deluxe Paint III. Hauptänderung zur Vorgängerversion sind die neuen Animationsfeatures und der Halfbright-Modus, der die Darstellung von 64 Farben erlaubt. Insgesamt stehen zwei Animationsmodi zur Verfügung, die an und für sich nichts weltbewegend Neues bieten. Es wird einfach ein sogenanntes Page-Flipping vorgenommen, d.h. ganze Bildschirme werden hintereinander aufgerufen. Die Animation wird durch eine Art Daumenkino erreicht. Dan Silva (Programmierer von DPaint) hat einige Funktionen dieser Art von Animation mitgegeben. Die zweite Animationsmöglichkeit wird durch ein Bilddifferenzverfahren erreicht. Die Unterschiede zwischen einzelnen Bildern werden gespeichert und nacheinander angezeigt. Animation kostet viel Speicher, aus diesem Grunde sind mindestens 1 MByte-RAM anzuraten. Auch sind einige weitere Features hinzugekommen.

X-tension Update

Die X-tension-Festplatte von Frank Elektronik hat ein neues Update der Treibersoftware erhalten. Der Betrieb mit dem neuen Fast-File-System geht jetzt problemlos. Die Bedienung der neuen Software ist wesentlich vereinfacht worden. Die Low-Level-Formatierung wird beispielsweise mit einem komfortablen Programm vorgenommen, dabei bekommen fehlerhafte Tracks automatisch eine Ersatztrack zugewiesen. Mit der neuen Software und einer FFS-Formatierung erreicht die Platte einen Datentransfer beim Schreiben von über 200 KByte pro Sekunde, beim Laden von fast 300 KByte. Besitzer einer X-tension-Festplatte können die neue Treibersoftware als Update erhalten.

Anbieter:
Frank Elektronik GmbH
Matthiasstr. 3
8500 Nürnberg 80
Tel. 0911-324380

Nachtrag zu:

"Tiefpaßfilter abschaltbar" (2/89, Seite 44) für

A500/B2000

Nicht immer besteht die Möglichkeit, mit dem Programm "Filter_aus" den Tiefpaßfilter auszuschalten. Nämlich dann nicht, wenn das Programm auf eine Diskette kopiert werden soll, auf der zu wenig Platz frei, oder sie nicht im AmigaDOS-Format beschrieben ist. Bei manchen Programmen, hauptsächlich Spielen, ist es nicht möglich, den Aufruf des Programms zum Ausschalten des Tiefpaßfilters in die Startup-Sequence einzubinden, weil die Programme durch einen Bootblock-Loader gestartet werden. Dadurch wird die Startup-Sequence nicht mehr ausgeführt.
Der Ausweg aus dieser Misere besteht in dem Einsatz eines Schalters, mit dem das Ausschalten des Filters zu jedem Zeitpunkt möglich wird. Mit einem 2 x EIN-Schalter kann der Tiefpaßfilter aus- und eingeschaltet werden, wenn die Verbindung zwischen Pin 8 und Pin 5 sowie Pin 14 und Pin 3 vom Verstärker-IC über je einen 520 Ohm Widerstand hergestellt bzw. unterbrochen wird. Der IC-Platz, auf dem Sie das Verstärker-IC finden, trägt beim A500 die Bezeichnung "U14" und beim B2000 "U204". Sie können auch auf den Schalter verzichten, weil heutzutage die Samplingrate von Musikstücken höher ist als bei früheren, die auf dem AMIGA mit wenig Speicher laufen mußten Durch eine höhere Samplingrate wird aber der Tiefpaßfilter, der nämlich ursprünglich eine Digitalisierung mit niedriger Samplingrate ohne Rauschen - aber mit der bekannten Nebenwirkung - ermöglichen sollte, überflüssig.

Nachtrag:

Zweite europäische AMIGA-Entwicklerkonferenz

Die auch unter der Bezeichnung 'DevCon' (Developers Conference) bekannte Entwicklerkonferenz fand diesmal in Frankfurt statt. Fast alle Commodore-Entwickler aus dem Bereich Soft- und Hardware waren zugegen. Die KICKSTART-Redaktion berichtete bereits ausführlich in der letzten Ausgabe. Leider haben sich einige Unklarheiten und Mißverständnisse ergeben, die auf diesem Wege ausgeräumt werden sollen. Die Betriebssystemversion 1.4 befindet sich zur Zeit noch in der Entwicklung. Entwickler erhalten lediglich eine Version, die einen kleinen Bruchteil der endgültigen Version 1.4 beinhaltet. Für den AMIGA-Anwender steht die Version nicht zur Verfügung, da sie für ihn keinen Nutzen bringt. Bis zur Auslieferung der Betriebssystemversion 1.4 wird aber noch einige Zeit vergehen. Die Version 1.4 wird das ECS (Enhanced Chip Set) unterstützen, die Chips befinden sich aber ebenfalls noch mitten in der Entwicklung. Sie sollen aber auch unter der Version 1.3 in Betrieb genommen werden können. Der im letzten Heft genannte neue Auflösungsmodus von 640 x 200 Pixel in Non-Interlace des ECS wurde fälschlicherweise als Super-Hires-Modus bezeichnet. Die richtige Bezeichnung lautet PRODUCTIVITY MODE. Bei diesem Mode sollen maximal 16 Farben aus einer Palette von 64 Farben ausgewählt werden können. Wann das ECS ausgeliefert werden soll, steht z.Z. noch nicht fest.

In der letzten Ausgabe waren unter der Überschrift *'Neue Commodore-Produkte'* einige Geräte aufgeführt, die zum Teil noch in der Entwicklung stehen. Lieferbar sind z.Z. folgende: A2286 Bridgeboard (AT-Karte), A2300 Genlock, A2620 68020-Card, A2058-Memory-Expansion-Card (aufrüstbar auf 4 und 8 MByte), A2090A-Harddisk-Controller (autobootfähig mit zwei EPROMS). Alle anderen Geräte, z.B. der A2024-HiRes-Monochrome-Monitor, die A2090B-Karte oder die serielle Karte A2232 stecken in der Entwicklung. Die amerikanische Firma ASDG hat ebenfalls eine serielle Karte fertiggestellt, sie unterscheidet sich im Grundkonzept erheblich von der in Planung befindlichen Commodore-Karte.

Audiomaster II



Bei Audiomaster II sind einige neue Features zu finden.

AEGIS bringt eine weitere Version von dem schon fast legendären Audiomaster. Bereits Audiomaster I zeichnete sich durch gute Bedienerfreundlichkeit aus. Bei Audiomaster II ist dies nicht anders. Gegenüber seinem Vorgänger besitzt es jetzt ein weiteres Edit und ein erweitertes 'User Options-Menu'. Auch bei den Lade- und Speicherroutinen ist etwas hinzugekommen. Die RAM-Scan-Ladefunktion kann auf drei verschiedene Adressen gelegt werden. Ansonsten hat sich nicht viel geändert. Der Verkaufspreis von Audiomaster II liegt bei 219.- DM.

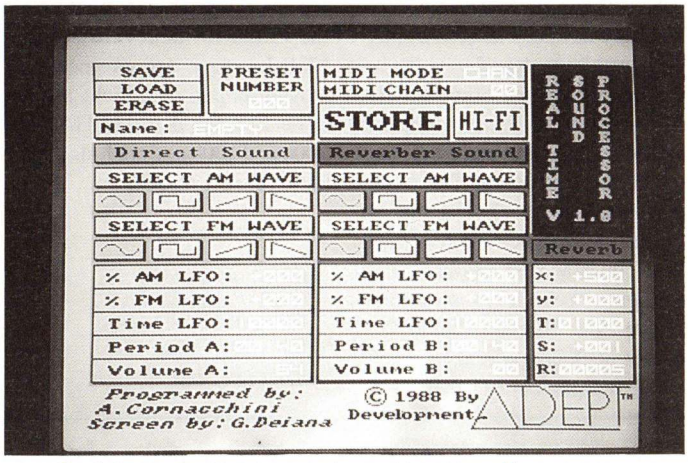
Anbieter:
IM
Borsigallee 18
6000 Frankfurt 60
Tel. 069-41072

Real Time Sound Processor

CASABLANCA vertreibt einen neuen Real Time Sound Sampler. Der Sampler wird an den parallelen Port des AMIGA 2000 bzw. 500 angeschlossen. Die mitgelieferte Software beinhaltet eine Vielzahl von Manipulationsmöglichkeiten. Allerdings ist die Software durch einen Dongle geschützt, was etwas hinderlich ist. Die mitgelieferte Dokumentation ist ausführlich und gut verständlich. Der Preis des REAL TIME SOUND PROCESSORS liegt bei 249.- DM

Anbieter: CASABLANCA
Nehringskamp 9
4630 Bochum 5
Tel. 0234-411994

Preis: 249.- DM.



Die Software des Samplers erlaubt eine Vielzahl von Manipulationen.

Low-Cost-Textverarbeitung

Aus Italien stammt eine Low-Cost-Textverarbeitung, die hierzulande von CASABLANCA vertrieben wird. Unter 100.- DM soll sie kosten. Wer meint, es handele sich lediglich um einen Editor, der etwas 'aufgemotzt' wurde, täuscht sich. Das Programm wartet mit vielen Features auf, wobei einige sogar einzigartig sind. Über 100 Funktionen sind implementiert. Beispielsweise kann die Bedieneroberfläche schnell und einfach geändert werden. Unfangreiche Blockoperationen sind ebenso vorhanden wie ein automatischer Viruswarner. Bilder können zwar nicht eingeladen werden, die Programmierer sind aber einen Kompromiß eingegangen, sie können separat verarbeitet und ausgedruckt werden. Den italienischen Programmierern ist mit dieser Textverarbeitung ein guter Wurf gelungen. Mit weniger als 100.- DM ist auch ein ausgezeichnetes Preis-/Leistungsverhältnis erreicht.

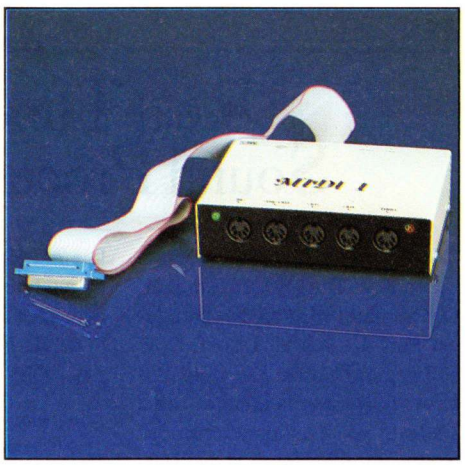
Anbieter:
CASABLANCA
Nehringskamp 9
4639 Bochum 5
Tel. 0234-411994



Die Low-Cost Textverarbeitung kann mit vielen Features aufwarten.

MIDI-1...

...nennt sich ein Midi-Interface von der amerikanischen Firma CMI. Mit diesem Interface ist es möglich, midikompatible Geräte mit dem AMIGA zu verbinden. Das Midi-Interface wird an den seriellen Port des AMIGA 500 oder 2000 angeschlossen, der durchgeschliffen ist, so daß noch Platz für andere Peripherie bleibt. Durch einen Umschalter kann zwischen dem Midi-Interface und der anderen Peripherie gewählt werden. Zwei Leuchtdioden zeigen die Aktivität des Midi-Busses an. Dem Anwender stehen ein Eingang und zwei Ausgänge zur Verfügung. Der Sync-Ausgang kann wahlweise mit 24, 48 und 96 Ticks betrieben werden. Dem Interface ist umfangreiche Software beigelegt, allerdings mußte die Dokumentation zunächst noch ausgedruckt werden, da sie lediglich auf Diskette vorlag. In Deutschland übernimmt die Firma IM den Vertrieb. Das Midi-Interface ist in mehreren Ausführungen erhältlich: als externe Box für den AMIGA 500, 1000 und 2000 und noch als Steckkarte für den A2000.



Das Midi-Interface kann entweder als externe Box oder als Steckkarte erworben werden.

Anbieter:
IM
Borsigallee 18
6000 Frankfurt 60
Tel. 069-410072

Preis: 169.- DM

Anti-Interlace-Filter

Es ist sicherlich allen AMIGA-Usern bestens bekannt, das Interlace-Flimmern. Die bislang entwickelten Lösungen bestanden in nicht unwesentlich teuren Hardwarezusätzen, die die einzelnen Bilder zwischenspeicherten und in doppelter Bildfrequenz zum Monitor schickten. Eine weitere Lösung befindet sich bei Commodore in der Entwicklung, die neuen Chips Denise und Fat Agnus, über die wir im letzten Heft berichteten.
Das Problem ist allerdings auch optisch zu lösen. Wie Sie wissen, werden im Interlace-Modus zwei Bilder leicht vertikal verschoben abgebildet. Die kürzlich in Japan entwickelte Anti-Interlace-Brille macht nichts anderes, als ein Halbbild durch vegetativen Polarisationseffekt auszufiltern. Das Flimmern wird somit stark begrenzt. Gerade bei empfindlichen Augen, die heutzutage besonders von der jungen Computergeneration getragen werden, konnte laut Hersteller eine deutliche Minderbelastung des Sehorgans registriert werden. Über den Zeitpunkt der Serienreife war bislang noch nichts in Erfahrung zu bringen.



Etiketten bedrucken

Mit dem Programm DETIKETT ist es möglich, Diskettenlabels mit nützlichen Texten zu versehen. Vorgesehen sind zum Beispiel Informationen über den Typ des auf der Diskette enthaltenen Programms, ob die Workbench benötigt wird, mit wieviel RAM es läuft usw. Der Benutzer kann aber auch selbst Texte erstellen.

Anbieter:
DETIKETT

Computer Ecke
Stresemannring 7
6070 Langen
Preis: 49.- DM

COMBITEC
DIE PARTNER

Combitec Software

Antivirus 4 **DM 29,90**
Komfortabler Viruskiller mit Boot-Block-Monitor, läuft auch im Hintergrund

PrintON 9-Dot **DM 29,90**
Brandneues Ausdruckprogramm für IFF-HiRes-s/w-Grafiken (z. B. DPaint 2), Ausdruck von 9-Nadlern in höchstmöglicher Auflösung (mit Vergrößerungsmöglichkeit)

Umfangreiche Software für Amiga lieferbar

PREISSENKUNG* · PREISSENKUNG*

Besuchen Sie uns auf der Hobby-tronic vom 12.-16. April 1989 in Dortmund. Sie finden uns in Halle 6, Stand-Nr. 6032.

PREISSENKUNG* · PREISSENKUNG*

GIB DEINEM AMIGA EINE CHANCE

Neu! Combitec TDS
(Track-Dis-Station), Adapter zum Anschluß von bis zu 4 Trackanzeigen, Typ Combitec Track-Dis, an allen gängigen Diskettenlaufwerken sowie des internen Laufwerkes DFO (bei A 500 und A 1000), Anschluß an den Diskettenport (durchgeschliffen)
VK-Preis: **DM 69,–**



Combitec D-RAM 2/4/8 M
Externe dynamische Speichererweiterung 2 MB, intern aufrüstbar auf 4 bzw. 8 MB, Einsatz moderner 1 MByte DRAM-Module, integrierter DRAM-Controller (0 Wait-states), formschönes, amigafarbenes Gehäuse, Busdurchführung
VK-Preis Version 2 MB: **1248,–** *
Version 4 MB: **2348,–** *
Version 8 MB: **4548,–** *

Combitec Disk 3,5
Diskettenlaufwerk 3 1/2", Busdurchführung für bis zu 3 ext. Laufwerke, Qualitätslaufwerk TEAC Typ FD 135 FN, PC 1-kompatibel, Anschluß für Track-Anzeige Typ: Combitec Track-Dis abschaltbarer Controller, amigafarbenes Gehäuse
VK-Preis: **DM 298,–** *



Combitec Track-Dis
Track-Anzeige zur Combitec Disk 3,5, 2 LED's zur Anzeige der bearbeiteten Diskettenseite, 2-stellige 7-Segment-anzeige, Steckanzeige zur Combitec Disk 3,5, amigafarbenes Gehäuse
VK-Preis: **DM 49,–**

Combitec Disk 5,25"
Diskettenlaufwerk 5,25", Busdurchführung für bis zu 3 ext. Laufwerke, Qualitätslaufwerk TEAC Typ: FD-55-GPR. Abschalter 40/80 Track-Umschalter, PC 1-kompatibel, Anschluß für Combitec Track-Dis, amigafarbenes Gehäuse
VK-Preis: **DM 368,–** *

Combitec HD 20 A
Adaptergehäuse für Harddisk zum Anschluß an den Amiga 500, formschönes, flaches, amigafarbenes Gehäuse, nachträglich intern aufrüstbar als Speichererweiterung mit statischen oder dynamischen RAM-Bausteinen.
VK-Preis: **DM 184,–**

Combitec HD 20
Harddisk mit Controller, superschnelle Datenübertragung, **Autobooting von der Harddisk ab Version 1.3** möglich, eigenes Netzteil. Komfortabele Treibersoftware, amigafarbenes Gehäuse
VK-Preis: 20 MB/65 ms **DM 1089,–** *
VK-Preis: 40 MB/28 ms **DM 1498,–** *
Hinweis: Die Harddisk Typ Combitec HD 20 kann nur in Verbindung mit dem Adaptergehäuse Typ Combitec HD 20 A Adapter, oder einer Combitec Speichererweiterung am Amiga 500 betrieben werden.



*Aufgrund von höheren Auflagen möglich

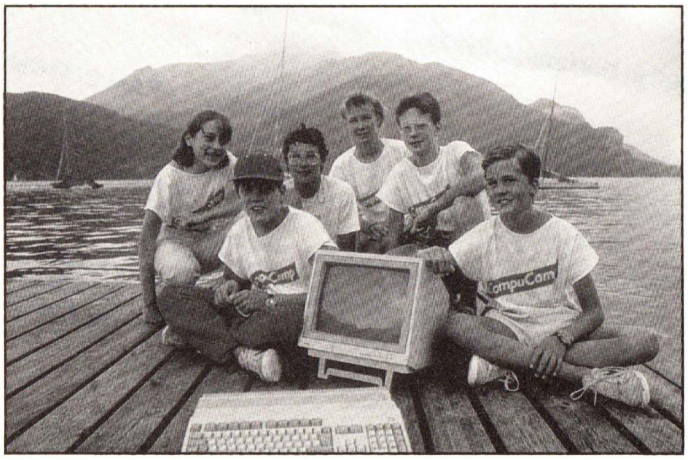
Weiteres Lieferprogramm:

Combitec S-RAM 500 (512 K)	DM 598,– *
Combitec S-RAM 1 M (1 MB)	DM 998,– *
Combitec Clock 77 S (P)	DM 359,–
Anbindungspaket (Kabel u. Software) zur Combitec Clock 77	
für Atari	DM 98,–
Amiga 500/2000	DM 98,–
Amiga 1000	DM 98,–
IBM kompatible	DM 98,–
Steckdosenmodul	DM 70,–
Combitec 5/3 ext. Netzteil	DM 115,–
Userport-Modul	DM 399,–

Rohlaufwerk (unmodifiziert, ohne Gehäuse und Kabel) TEAC, Typ FD 135	DM 198,–
Amiga 500	DM 940,–
Monitor 1084	DM 568,–
Toshiba P 321 24-Nadel-Drucker	DM 850,–
Mouse-Pad	DM 12,90 *
Joy-Stick m. Mikroschalter	DM 18,50
Commodore Disketten, 3,5", 2DD, 135 tpl, 10er Pack	DM 34,–
Disketten, neutral, 3,5", 2DD, 135 tpl, 10er Pack	DM 24,50
Commodore Disketten, 5,25", 96 tpl, HD, 10er Pack	DM 29,80
Diskettenbox für 80 Disketten, 3,5", abschließbar	DM 12,95

Besuchen Sie unseren Ausstellungsraum
Combitec Computer GmbH
Liegnitzer Straße 6-6a
5810 Witten
☎ Tel. 0 23 02/8 80 72

Computerferien



Viele verschiedene Computerkurse werden bei den Computercamps angeboten.

Wer sich auch in den Ferien nicht von seinem Computer trennen kann oder sich weiterbilden will, kann bei dem Reiseveranstalter CompuCamp GmbH das neue Urlaubskonzept CAMPS nutzen. Voraussetzung ist jedoch ein Alter zwischen 12 und 18 Jahren. Die Idee dieser Ferienform stammt aus den USA und hat dort schon eine längere Tradition. Jedoch bestehen die Ferien nicht nur aus Computerkursen, sondern es finden auch vielseitige sportliche Aktivitäten statt. Die Kurse sind in Könnensstufen eingeteilt, so daß wohl keiner über- bzw. unterfordert wird. CAMPS bietet unter anderen spezielle AMIGA-Kurse an, Themen sind beispielsweise AMIGABASIC oder C und Maschinensprachprogrammierung. Je nach Standort und individueller Kurswahl liegen die Kosten für einen Campaufenthalt für eine Woche bei mindestens 675.- DM. Im Preis ist alles enthalten: Unterbringung, Verpflegung, Kurs- und Freizeitbetreuung. Nähere Informationen erhalten Sie unter folgender Anschrift:

CAMPS
c/o CompuCamp GmbH
Wedeler Landstraße 93
2000 Hamburg 56
Tel. 040-811081

Link-Virus

Die Schreckensmeldungen reißen nicht ab, schon wieder ist ein neuer Virus aufgetaucht. Doch diesmal verbreitet er sich nicht wie bisher üblich über die Bootspur, sondern hängt sich an Programme. Diese Art von Viren bezeichnet man als Link-Viren. Die Gefährlichkeit des Virus' ist eindeutig. Bisherige Virusfinder beschränkten sich auf die Untersuchung der Bootspur und des Speichers, ließen aber Programme außer acht. Da sich der Virus aber an Programme hängt, ist es schwer, ihn zu finden und die Diskette zu 'säubern'. Das Eliminieren aus dem Speicher stellt kein großes Problem dar, übernehmen das doch fast alle gängigen Virusschutzprogramme. Das Aufstöbern hingegen setzt zur Zeit noch viel Handarbeit voraus. Die Funktionweise des Virus' ist recht einfach. Er hängt sich an bestehende Programme an, das Programm wird dabei zunächst in einen extra Puffer geladen, die Virusroutine hängt sich vor das eingeladene Programm und schreibt sich dann zurück auf das Speichermedium. Die Datei wird dadurch um ca 1000 Byte vergrößert. Beim Aufruf dieser infizierten Datei hängt er sich an den Systembefehl *'OldOpenLibrary'*. Von diesem Befehl aus setzt er sein zersrtörerisches Werk fort. Der Virus macht sich hin und wieder bemerkbar und gibt folgende kleine Meldung im DOS-Fenster aus:

AmigaDOS presents:a new virus by the IRQ-TeamV41.0

Ein kleiner Tip der Redaktion: Meistens werden die ersten Befehle der 'startup-sequence', z.B. der Echo-Befehl, befallen. Auf diese Befehle sollten Sie ein besonderes Augenmerk legen.
Neben der kleinen Meldung, die im DOS-Fenster ausgegeben wird, meldet sich der IRQ-Virus mit einem System-Requester, der das erneute Booten zur Folge hat. Vor diesem Virus muß ausdrücklich gewarnt werden, er kann enorm großen Schaden anrichten. Die bösen Absichten der Virusprogrammierer sind schon daran deutlich zu erkennen, daß selbst die Virusmeldung in der Datei zerhackt ist.



GFA-BASIC-UPDATE

Sicherlich ist es dem ein oder anderen Anwender von GFA-BASIC 3.0 aufgefallen, daß manche Befehle und Funktionen nicht einwandfrei arbeiten. Auch beim Hersteller GFA hat man dies festgestellt und arbeitet seitdem an den Bugs. Herausgekommen ist die offizielle Update-Version GFA-BASIC 3.02. Jeder, der seine Originaldiskette, einen frankierten und adressierten Rückumschlag und 5.- DM in Briefmarken einschickt, erhält diese Version.
Neu dazugekommen ist der Menüpunkt SAVE ICON, der im aktivierten Zustand bei jedem Speichervorgang auch ein entsprechendes Icon erzeugt. Die serielle Schnittstelle kann jetzt einfach mit

OPEN "I",#1,"COM: [Baudrate [,Parität [,Länge [,Stopbits]]]]"

angesprochen werden. Mit Hilfe einer Variablen (_IOExtSer) können auch die Parameter ermittelt werden.
Die Befehle GET und PUT benutzen nun das Clipping und arbeiten endlich fehlerfrei. Auch die Befehle, um einen gesamten Bildschirm in einer String-Variablen zu verwalten (SGET/SPUT), sind implementiert worden.
Auch ein Befehl namens SCROLL ist hinzugekommen, der einen rechteckigen Bereich innerhalb des Ausgabefensters rollt (siehe AmigaBASIC).
Wir haben die neue Version natürlich sofort getestet und festgestellt, daß fast alle Fehler, die zuvor bei den Programmen auftraten, behoben waren. Sollten Sie auch Erfahrungen mit GFABasic gemacht haben, dann teilen Sie sie uns doch bitte mit.

Auf den GFA-BASIC-COMPILER müssen Basic-Fans leider noch etwas warten. Angestrebt wird ein Termin Mitte diesen Jahres.

Eine weitere Ankündigung betrifft den GFA-Assembler, der bereits auf dem ATARI ST im Einsatz ist und nun auch für den AMIGA angepaßt werden soll. Schon auf der CEBIT 89 will man dieses neue Produkt vorstellen, zu dem auch ein Crossassembler geliefert werden soll. Ansonsten wird das Produkt der ATARI-Version sehr gleichen. Als Veröffentlichungstermin wird ebenfalls die Jahresmitte angegeben.

Infos und Update:
GFA Systemtechnik GmbH
Heerdter Sandberg 30-32
4000 Düsseldorf 11

COMBITEC D-RAM für A2000

Die Firma COMBITEC erweitert ihr Lieferprogramm für Speichererweiterungen. Neu aufgenommen wurden dynamische Erweiterungen für den internen Einbau in den AMIGA 2000. Die Steckerleiste ist vergoldet und gewährt damit eine hohe Kontaktsicherheit. Die Autokonfigurationsdatei befindet sich in einem werkseitig programmierten PAL-Baustein, der bei einer Aufrüstung nicht gewechselt werden muß. Somit ist es dem Käufer möglich, die Speichergröße selbständig in 2MB-Schitten zu erweitern. Die Sockel dafür sind bei der 2MB-Ausführung bereits vorhanden.

Preise:
COMBITEC D-RAM 2 A2000
DM 379.- (unbestückt)

COMBITEC D-RAM 2 A2000
DM 1298.-

COMBITEC D-RAM 4 A2000
DM 2449.-

COMBITEC D-RAM 8 A2000
DM 4499.-

COMBITEC Computer GmbH
Liegnitzer Straße 6-6a
D-5810 Witten

BC-FORTRAN77

Ein neues Fortran-Entwicklungssystem wird von BC-Softwaretechnik Andre Köstli vertrieben. Das System besteht aus integriertem Editor, Precompiler, Compiler, Linker, Versionsverwaltung, Debugger und Profiler.
Das Laufzeitsystem fängt 68000-Exceptions ab, und es sind Testhilfen für Parameter-, Feldgrenzen- und Stacküberwachung vorhanden. Der Linker arbeitet wahlweise mit einer Optimierungsoption, bei der Module in einem Objektspeicher gehalten werden, um sie nicht immer nachladen zu müssen.

Eine kleine Graphikbibliothek mit den Unterprogrammen PLOTS, PLOT, NEWPEN, FACTOR, WHERE und SYMBOL unterstützt die Grafikausgabe. Der Profiler ermittelt die Verteilung der Rechenzeit der einzelnen Unterprogramme.

BC-Softwaretechik Andre Köstli
Martin-Luther-Straße 63
7000 Stuttgart 50
Tel.: 0711- 56 47 36

NACHTRAG

In Heft 1/89 haben wir bei dem Bericht HAM-FAX /AMIGA die Adresse vergessen. Wegen wiederholter Nachfage nun die Anschrift:

Karl J. Ebensberger
Fällhofstr. 11
D-8068 Pfaffenhofen 1
Tel.: 08441 - 61 45
BTX: 08441 - 7 22 13

VON SVEN STILLICH

Ein Fall für SHERLOCK:

Virenjagd in finsteren Registern

Sicherlich ist jedem AMIGA-Besitzer dieses oder ähnliches schon einmal passiert: Man will ein Spiel starten, das aus dem Bootblock nachlädt, und hat auf einmal nur das CLI-Fenster vor sich. Oder während man dabei ist, seinem Programm den letzten Schliff zu geben, schaltet sich plötzlich der Bildschirm ab. Jetzt ist wohl der Zeitpunkt gekommen, sich (oder dem AMIGA) die Kugel zu geben oder die Initiative zu ergreifen. Dies hat die neue Firma SUNTRONIC getan. Die Idee, ein Programm zu entwickeln, das Disketten und Speicher auf Viren testet, ist zwar so alt wie die Viren selbst, SHERLOCK stellt jedoch einen wertvollen Beitrag zum Abwehrsystem unseres Rechners dar.

Doch nun der Reihe nach. SHERLOCK wird mit einer 10seitigen deutschen Anleitung ausgeliefert. In diesem Manual werden alle Funktionen des Programms detailliert erläutert und weitere Tips gegeben, wie man seinen Rechner am besten gegen Virenbefall schützen kann. Hat man zum Beispiel den Verdacht, auf seiner Diskette könnte sich einer dieser Krankheitserreger eingenistet haben, kann man den Bootblock untersuchen lassen [SHERLOCK erkennt ca. 30 verschiedene Bootsektoren, darunter auch Virusprotektoren (System-Z,...)]. Gibt der Checker "Vorsicht! Kein Standard Bootblock!" aus (das heißt, der Bootblock ist ihm nicht bekannt), gibt es noch die Möglichkeit, sich den verdächtigen Block in hexadezimaler Form oder im ASCII-Format anzeigen zu lassen. Ist es wirklich ein Virus, löscht man ihn einfach. Natürlich gibt es auch Funktionen zum Durchforsten des RAMs. Hierbei wird sowohl nach bekannten Viren als auch nach "virulenten Datenstrukturen" (O-Ton Anleitung) gesucht. Doch all dies gehört natürlich zu den obligatorischen Funktionen eines Virus-Searchers. SHERLOCK gibt dem geneigten Anwender jedoch auch die Möglichkeit, die Zerstörung seiner Daten durch Viren zu verhüten oder wenigstens das Restrisiko derselben möglichst gering zu halten. So steht es dem Benutzer zum Beispiel frei, seine Disketten vor dem unkontrollierten Zugriff anderer zu schützen. Eine hiermit behandelte Diskette wird nicht mehr als Systemstartdiskette anerkannt, kann also nicht mehr gebootet werden. Ein anderes Hilfsmittel ist der "Sherlock-Checker". Dies ist ein Programm, das sich - ganz im Stile anderer Virus-Protektoren - nach dem Booten im Speicher installiert und bei jeder neueingelegten Diskette auf einen nicht standardgemäßen Bootblock prüft. Die Optionen "Bootfile laden/speichern", mit denen man Bootblöcke auf eine Datendiskette retten und später (wenn die Diskette trotz aller Vorsicht infiziert ist) wieder installieren kann, runden das Programmpaket ab.

```
rom-wack PC: FC1B64  SR: 2318
USP: 228F28  SSP: 07FFB6  XCPT:
0003 TASK: 000000 DR: FFC7F3E1
0002B646 00000B84 0000000A
00000B84 0000002B 0000003B
0022F904  AR: FFC7F3E1 00027F4A
00200AA
```

Diese Informationen sind ein klarer Fall für Sherlock

Fazit

Fast alle der SHERLOCK-Funktionen sind in dieser Form schon als Programme erhältlich. Jedoch vereinigt SHERLOCK diese Programme in sich, was zweifellos einen Vorteil darstellt. Doch wo viel Licht ist, ist auch Schatten. Da SHERLOCK beim Laden mittels trackdisk.device direkt auf die Diskette zugreift, ist es bis jetzt nicht möglich, den Searcher auf einer Festplatte zu installieren. Weiterhin verträgt sich SHERLOCK nicht mit der KICKSTART-Version 1.3.

Das Programm ist direkt bei der Firma SUNTRONIC, die auch einen UPDATE-Service anbietet, erhältlich. Da ein Virus in der Lage ist, den Wert einer Diskette auf den einer Leerdiskette zu reduzieren, halte ich den Preis von 39.95 DM für angemessen.

SHERLOCK-Version 1.0

+ große Funktionsvielfalt
+ gute Benutzeroberfläche
+ komplett in Deutsch
+ UPDATE-Service

- läßt sich nicht auf Festplatte kopieren

Anbieter:
SUNTRONIC
Am Schäfergraben 6
6231 Sulzbach

Preis 39.95 DM

ENDE

KICK ME, AMIGA

R.C.T.

KICK

Der Weg in völlig neue Dimensionen

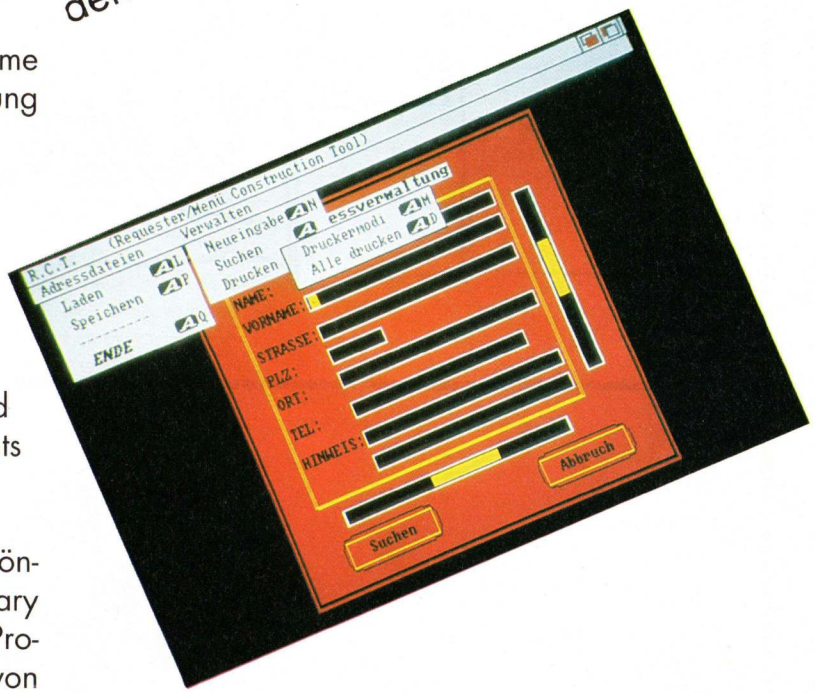
Requester und Menü Construction Tool für den *AMIGA*

Wollten Sie nicht schon immer ihre Programme mit einer professionellen Benutzerführung versehen?

Mit dem R.C.T. ist das kein Problem, egal welche Programmiersprache Sie wählen.

Das Erstellen einer grafikorientierten Oberfläche ist für viele ein Buch mit sieben Siegeln. Das R.C.T. hilft Ihnen, Requester mit Gadgets und Images oder Menüs in verschiedenen Schriftfonts zu konstruieren.

Die konstruierten Requester und Menüleisten können mit den Funktionen der mitgelieferten Library verwaltet werden. Die Library kann von allen Programmiersprachen genutzt werden, auch von Basic aus.
Bei Programmiersprachen wie C oder Assembler besteht die Möglichkeit, die Funktionen der Library direkt an das Programm zu linken.
Der Sourcecode-Generator ermöglicht die Erzeugung von C-, Assembler- und GFA-BASIC-Source von den konstruierten Requestern und Menüleisten.



R.C.T. DM 129.-

Bestellcoupon

Bitte senden Sie mir:

☐ R.C.T. DM 129.-

zuzüglich Versandkosten:
Inland DM 7.50
Ausland DM 10.-

Name

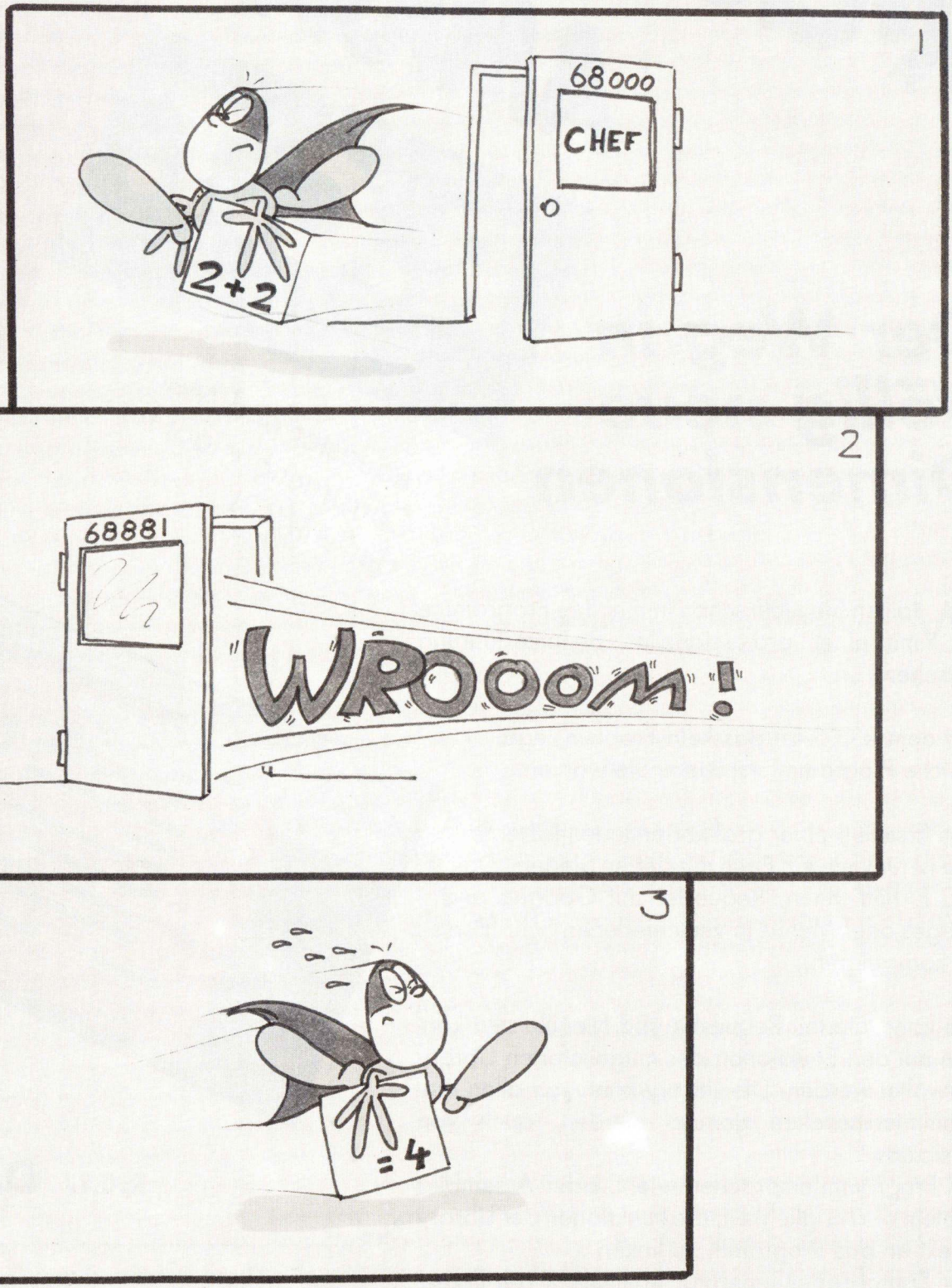
Straße

Ort

Datum Unterschrift

Auslandsbestellungen nur gegen Vorauskasse
Bei Nachnahme zuzüglich DM 3.50 Nachnahmegebühr

MAXON-Computer / Industriestr. 26 / 6236 Eschborn



10/89 © MAXON

CSS Consulting · System · Software

Auf der Warte 46 · 6367 Karben 1 · Tel. (0 60 39) 57 76 · Fax (0 60 39) 4 36 21

Der Rechner für Unis, Video-/Grafikstudios, prof. Anwender!

CSS - Amiga - Tower! Bekannt aus dem Fernsehen von Tele 5. Unser Spitzenmodell weist folgende Leistungsdaten auf:

- Kickstart 1.3 mit Virusfinder
- 2 x 3.5" Laufwerke
- 5 MB Arbeitsspeicher, davon 2 MB als 32 Bit RAM
- Unix Board A 2620, erweitert mit 68882, getaktet mit 30 MHz
- Programm zum Kopieren des Kickstart in das 32 Bit RAM
- Autoboot Controller A 2090A mit SCSI Festplatte
- AT-Karte incl. MS-DOS 3.3 und GW-Basic sowie 1.2 MB Laufwerk angeschlossen als Massenspeicher:
  das **erste beschreibbare CD Laufwerk** mit 552 MB!
  Der Preis für die Workstation ..... **DM 23.950,—**

Selbstverständlich sind die Komponenten auch einzeln erhältlich:

A 2090A/40 MB	2.290,—	A 2300 Genlock	450,—
AT-Karte, komplett	2.490,—	PC-Karte, komplett	950,—
A 2620 Unix Board	6.500,—	A 2058, 2 MB best.	1.500,—

CD Laufwerk incl. Controller und Software ..... 8.500,—
CSS-Turboboard 3, 68020/68882, 30 MHz, getaktet, bis 1 MB mit Static RAM bestückbar ..... 1.950,—
CSS-Piggy Pack für 68030, bestückt mit 030/16 ..... 1.500,—
CSS-Programm zum Kopieren des Kickstart ins 32 Bit RAM ..... 150,—

Kickstart-Platine mit 1.3/Virusfinder + WB + Extra + deutsches Handbuch nur ..... 150,— ohne Kickstart-Platine ..... 50,—

Fordern Sie unsere Listen an!

Vertriebspartner:

CERBERUS COMPUTER, Ansprechpartner Herr M. Stein, Spezialist für Unis Btx: 0271-57655

SCHWEIZ — Basel-Land: Meikel Hard & Software, Tel. 0041-61-8214761, von 8.00—17.00 Uhr

Empfehlung für Druckereibereich:
Fachgebiet FOTOSATZ, Konvertierung z.B. MS-DOS zu Linotype, FM Druck: Tel. 06039-41525, Fax 06039-6148

Elsa – die Harddisk für AMIGA
500/1000 mit Gehäuse
2000 als Einbausatz

– mit Netzteil und Controller in einem Gehäuse
– Superschnell! (400 KB pro Sek.)
– leicht zu installieren
– von 20 MB bis 120 MB

Preise: 20 MB 848,-
30 MB 998,-
40 MB 1169,-

Kickstartumschaltplatine mit Kick 1.3 Eproms 160,-
Expansionsboard = zusätzliche Steckplätze für A2000 PC-Teil 165,-

H & P Rodat
Der Computerladen
Comp. Mate

Versand per UPS
Nachnahme 10,-

Sudbrackstraße 31 • 4800 Bielefeld • Telefon 0521/133621

Aztec C Prof. V3.6	DM	279,–
Aztec C Dev. V3.6	DM	399,–
Source Level Debugger	DM	109,–
MCC-Shell + Toolk. + Assemb.	DM	149,–
Golem 2 MB-Rambox	DM	1298,–
Golem 20 MB-Harddisk	DM	899,–
Golem 3.5 Z-Laufwerk	DM	249,–
Flightsimulator II	DM	75,–
TDI-Modula V3.01 Dev.	DM	199,–
2-MB-Speicherweiterung	DM	1199,–

Kostenlose Prospekte auch für Atari ST und IBM von

CWTG

Computerversand CWTG Joachim Tiede
Bergstraße 13 · 7109 Roigheim
Tel/BTX 0 62 98/30 98 von 17 – 19 Uhr

IDEE-SOFT-Programme
– Exzellent in ihrer Struktur – alle Programme in Deutsch –

TYPIST AMIGA

Der AMIGA als elektronische Schreibmaschine mit zeilenweisem Ausdruck und 15-zeiligem Bildschirm-Display. Je nach Drucker bis zu 30 verschiedene Schriften File auf Disk, Kopie-Ausgabe DM 88.-

GELD AMIGA

30 Routinen für den Umgang mit Geld - Anlage - Vermögensbildung - Rentensparen Rendite - Lasten - Zinsen - Kredit - Hypothek - Laufzeit - Amortisation - Raten Gleitklausel - Nominal-/Effektiv-Zinsen Akonto bei Verzinsung - Diskontierung - Konvertierung - Komplette Tilgungspläne für alle Modi - Bild-/Druck DM 98.-

ASTROL. KOSMOGRAMM AMIGA

Nach Eingabe von Namen, Geburtsort (geographischer Lage) und -datum werden errechnet: Sternzeit, Aszendent, Medium Coeli, Zodiakradianten, Positionen der Objekte im Tierkreis, Häuser nach Koch/Schaeck, Aspekte. Ausgabe einer allgemeinen Persönlichkeitsanalyse mit Partnerschaftsskala. Bild-/Druckerausgabe auf 3 DINA4-Seiten, Horoskop-Diagramm - Erfassung aller Planeten mit Sonne, Mond und Mondknoten, Minutengenaue Ausgabe - Koordinaten-Tafel DM 78.-

BIOKURVEN AMIGA

Zur wissenschaftlichen Trendbestimmung der biologischen Rhythmik und des seelisch-geistig-körperlichen Gleichgewichtes - Bildschirmausgabe monatlich vor- und zurückschreitend, auf Drucker beliebig lang mit täglicher Analyse und Kennzeichnung kritischer Tage - Absolut- und Mittelwerte - Ideal für Partnervergleich Beschreibung wissenschaftl.Grundlagen - Texteditor für Zusatzdaten etc. DM 58.-

KALORIEN-POLIZEI AMIGA

Nach Eingabe von Körpergröße, Gewicht, Geschlecht und Arbeitsleistung erfolgen Bedarfsrechnung und Vergleich mit eingegebener tatsächlicher Ernährung nach Eiweiß, Fett und Kohlenhydraten - Ideal-, Über-/Untergewichtsbestimmung - Ausgabe der Vitalstoffe und der Gehalte - Liste der Aktivitäten und Verbräuche - Lebensmittel-Kalorientabelle - Bildschirm und/oder Drucker auf einigen DINA4 DM 58.-

In Computer Shops oder bei uns per NN + DM 5.- oder Vorkasse + DM 3.-
Preise unverbindlich. Liste gegen adressierten Freiumschlag DINA5

I. DINKLER
IDEE-SOFT
Am Schneiderhaus 7
Tel. 02932/32947 D-5760 ARNSBERG 1

GESCHÄFT AMIGA

Ein Editor erstellt Formular-, Adressen-Artikel- und Dienstleistungsdateien. Im Hauptmenü wählt die Maus unter Angebot/Kostenvoranschlag, Auftrag/Bestellung, Auftragsbest., Rechnung, Lieferschein, Mahnung. Eingabe-Optionen Hand/Datei - Erfassung von je 20 Positionen/DINA4, die über Menge, Preis, Aufschlag/Rabatt, Umsatzsteuer, Skonto durchgerechnet werden. Texteditor für Zusätze - Kein Verbund zu Lager und Buchhaltung - Schnell, übersichtlich, bedienfreundlich DM 198.-

Provisionsabrechnung AMIGA

Editor für Vertreter- und Kundenadressen sowie Formulardaten. Eingabe Hand/Datei, 25 Positionen/DINA4, Provision 0.01 - 99.99%, Storno und Spesengutschriften - Durchrechnung bis zum Endbetrag mit/ohne Umsatzsteuer, schnell und genau DM 118.-

Inventur, Fibu-gerecht AMIGA

Kontinuierliche Bestandsverwaltung mit laufender Auswertung. Neuerfassung durch Streichen, Ändern, Hinzufügen, gruppenweise Selektion nach Code, max.1000 Pos./Liste, Blätter-Addition DM 118.-

Dateiverwaltungen AMIGA

Datenfelder von 8 Zeilen a 33 Zeichen je Datei max.1000 - Suchcode von max.33 Zch selektiert ab 1 Zeichen, mit jedem mehr die Zielgruppe weiter einengend - Optionen: Code, Nummer, alle, Blatt vor/zurü. Ändern (zeilenw.), Streichen, Hinzufügen Druck: 80-Zeichen-/Block-Liste, Seitenvorschub, Etiketten, Datenfeldmaske -Gezielte Aufgaben, schnell am Ziel - Übersichtlich, schnell, bedienfreundlich -

ADRESSEN	68.-	GALERIE	118.-
BIBLIOTHEK	118.-	LAGER	118.-
BRIEFMARKEN	118.-	PERSONAL	118.-
DISKOTHEK	78.-	STAMMBAUM	118.-
EXPONATE	118.-	VIDEOTHEK	78.-

DEFINDATA zum Selbstdefinieren der Inhalte DM 148.-

VON MARCELO MERINO

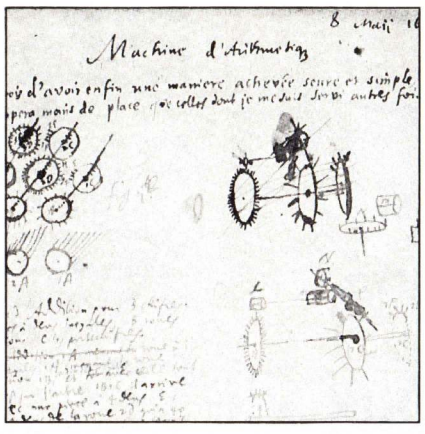
Programmiersprachen am AMIGA



WÄRE DAS LEBEN NICHT EINFACHER, WENN WIR ALLE DIE GLEICHE SPRACHE SPRECHEN WÜRDEN? MAN KÖNNTE SOGAR DAVON TRÄUMEN, DASS DAMIT ALLE SPANNUNGEN ZWISCHEN DEN MENSCHEN VERSCHWINDEN UND KRIEGSGEFAHREN ODER ÄHNLICHE NICHT NATÜRLICHE KATASTROPHEN GEBANNT WÄREN. LEIDER IST DAS NUR EIN TRAUM. VIELE HABEN IHN GETRÄUMT. DAS ESPERANTO SOLLTE EINEM BESSEREN VERSTÄNDNIS AUF DER BASIS EINER GEMEINSAMEN SPRACHE DIENEN. KURZ NACHDEM DER POLNISCHE ARZT L. ZAMENHOF DIESE UNIVERSELLE SPRACHE ENTWICKELT HATTE, KAMEN SOLCHE 'KULTURELLEN BRÜCKEN' IN MODE. KEINE DIESER KUNSTSPRACHEN HAT SICH INDES DURCHGESETZT; DIE MENSCHHEIT BLEIBT DURCH VERSCHIEDENE SPRACHEN GETRENNT.

Vielleicht weil Computer ein menschliches Produkt sind, spiegeln sich hier ähnliche Verhaltensweisen wider. Die Existenz verschiedener Computersprachen beruht jedoch nicht nur auf der Tatsache, daß diese von verschiedenen Menschen geschaffen werden. Sicher hat sie mehr mit der Vielfältigkeit der Aufgaben, die der Mensch dem Computer gestellt hat, zu tun. Für jedes Problem, das in der Geschichte des Menschen gelöst werden mußte, wurden die notwendigen Werkzeuge geschaffen. Deshalb ist die Entwicklung der Wissenschaften und der Technik eng mit den zu lösenden Problemen verbunden.

Bei der Entwicklung des Computers ist es nicht anders gewesen. Die Automatisierung von Rechenvorgängen war mit Sicherheit einer der ersten Vorfahren des jetzigen Computers. Im 17.Jahrhundert entwickelte Wilhelm Schicard, ein Astronom, eine Rechenmaschine, die die vier Grundrechenarten beherrschte. Blaise Pascal baute in der gleichen Zeit eine Additions-/Subtraktionsmaschine und machte sich dabei schon Gedanken darüber, wie ein Computer aussehen könnte. Leibniz entwickelte im 20.Jahrhundert eine Rechenmaschine, die nicht nur die vier Grundrechenarten beherrschte, sondern auch Wurzeln ziehen konnte. Das alles waren aber rein kalkulatorische Maschinen. Der Mensch war jedoch auf der Suche nach der Automatisierung produktiver Abläufe.


Handskizze der arithmetischen Maschine

Joseph Marie Jaquard entwickelte im Jahr 1805 einen automatischen Webstuhl, der verschiedene Vorgänge von selbst wiederholte. Jacquard hatte sich eine Art gelochte Karten ausgedacht, die den verschiedenen Teilen eines Ablaufs entsprachen. Man konnte sie fast beliebig ändern, so daß eine flexible Automatisierung gewährleistet war. Dies war die Geburt des Lochkartensystems als universelle Eingabe von Befehlen und/oder Daten. Die Abläufe waren aber nur für diese Maschine konzipiert, von einer 'Sprache', die diese steuerte, kann nicht die Rede sein.

Charles Babbage opferte sein ganzes Leben der Entwicklung universeller Rechenmaschinen. Seine erste Maschine, 'Differential Engine', ist zweifellos ein echter Vorläufer des Computers. Später entwarf er eine zweite Maschine, 'Analytical Engine', die aber nie gebaut wurde. Interessant ist aber auch Babbages Helferin, nämlich Lady Augusta Ada Lovelace, die Tochter von Lord Byron. Sie half Babbage bei der Entwicklung der analytischen Maschine und prägte Begriffe wie 'Schleife' und 'Subroutine'. Hundert Jahre später sind dieser Begriffe ein Muß in jeder hohen Programmiersprache.


Leibniz, der deutsche Denker und Erfinder

Das Verteidungsministerium der USA nannte eine Programmiersprache, die für die Steuerung von Vernichtungsmaschinen entwickelt wurde, ADA. Man sagt, zu Ehren Augusta Ada Lovelaces. Welch trauriges Schicksal. Augusta, wie können wir das wiedergutmachen?

Aber die Erfindungen von Jacquard und Babbage waren mechanische Versuche. Mit Beherrschung des elektrischen Stroms, der Erfindung von Rohren und später der Halbleitertechnik wurde das Konzept wieder erneuert. Elektrische Impulse, die von Reglern oder Schaltern kontrolliert wurden, veranlaßten diese analytischen Maschinen, bestimmte Vorgänge zu realisieren. Die einzelnen Befehle wurden mit verschiedenen Schaltereinstellungen den Maschinen übermittelt.

Herman Hollerith machte von der Jaquard-Karte zur Webstuhlsteuerung bei der Volkszählung 1884 in den USA Gebrauch. Jahre später entwickelte Hollerith seine 'Tabulating Machine'. Diese Maschine funktionierte, im Gegensatz zu Jaquards automatischem Webstuhl, elektromechanisch. Viele elektromechanische Rechner folgten dieser Maschine. Nach einfachen Schaltern wurden Relais' und andere Elektrovorrichtungen eingesetzt. Erst nach dem Zweiten Weltkrieg wurde mit ENIAC der erste elektronische Computer verwirklicht.

ENIAC rechnete trotz seiner Röhrentechnik noch dezimal, was immer noch nicht dem Stand unserer heutigen Computer entspricht.

John von Neumann ist verantwortlich für eine revolutionäre Entwicklung in der Geschichte der Computer-Mensch-Kommunikation. Er gab dem Rechner weder mechanische Instruktionen noch dezimale Eingaben, sondern eine schier endlose Folge von Nullen und Einsen. Mit dem EDVAC wurde die neue Generation digitaler Rechner geboren. Doch auch damit wurde immer noch nicht eine vernünftige Kommunikation mit dem Rechner realisiert. Erst mit dem nächsten Schritt wurde ein für den Menschen bequemes und verständliches Bindeglied zur Maschine geschaffen. Dieser Schritt war die Entwicklung von Assembler. Er hat den Programmierer von der Codierung der Befehle befreit und die Programme um einiges deutlicher und verständlicher gemacht, sie waren besonders schnell in ihrer Ausführung und an die Maschinen sehr gut angepaßt. Die Schnelligkeit ist zweifellos ein Vorteil, aber der maschinenabhängige Code macht diese Programme nicht bzw. nur sehr schwierig übertagbar.

Da jeder Prozessor seine eigenen Befehle und Adressierungsarten hat, wurde vom Hersteller ein eigener Assembler entwickelt. Der Assemblerprogrammierer muß deshalb, arbeitet er mit verschiedenen Systemen und unterschiedlichen Prozessoren, mehrere Assemblersprachen lernen. Trotz aller Nachteile bleibt Assembler die effektivste Art, einen Rechner zu programmieren. Da Assembler direkt den Maschinencode erzeugt, sind Assemblerprogramme die schnellsten und kürzesten. Assembler wird in vielen Anwendungen eingesetzt, die zeitkritisch oder von einer Hochsprache aus nicht erreichbar oder lösbar sind.

Problemorientierte Sprachen

In den 50er Jahren erkannte man, daß die Eingabe des Dualcodes oder das Verschieben von Schaltern in verschiedene Stellungen nicht nur mühsam und langwierig war, sondern auch eine Fehlerquelle darstellte. Man forschte nach besseren Kommunikationsmethoden mit dem Rechner und wollte eine Sprache schaffen, die so weit wie möglich die Technik der menschlichen Kommunikation widerspiegelte. Die Struktur der Aufgaben, die dem Computer gestellt wurden, sollte diesem angepaßt werden und auf verschiedenen Rechnern laufen können.

FORTRAN, der Grossvater

Zu Anfang der 50er Jahre, als der massive Einsatz der Computer im kommerziellen Bereich begann, stiegen die Kosten der Software-Entwicklung enorm. Der größte Teil der Zeit wurde für die Fehlerkorrektur aufgewendet.
Im Jahr 1954 gelang es dem IBM-Konzern, die erste höhere Programmiersprache zu schaffen. FORTRAN (Formula Translator) ist eine Programmiersprache, die von Anfang an für die Lösung von naturwissenschaftlichen Problemen gedacht war. Da zu diesem Zeitpunkt die meisten Aufgaben der Datenverarbeitung aus dem mathematischen Bereich kamen, hatte man mit FORTRAN eine Sprache gefunden, die die Entwicklung und Lösung komplizierter arithmetischer und algebraischer Operationen beherrschte. Da die Sprache von einem großen Konzern entwickelt wurde, ist sie in der Welt der Datenverarbeitung schnell zum Standard geworden. Mit der Zeit wurden viele Verbesserungen vorgenommen, so daß die neueste Version FORTRAN 77 um einiges leistungsstärker als frühere Versionen ist. FORTRAN ist trotz der Verbreitung von Personal-Computern und trotz des riesigen Angebots verschiedener Softwarehäuser eine der Sprachen großer Rechenanlagen geblieben.

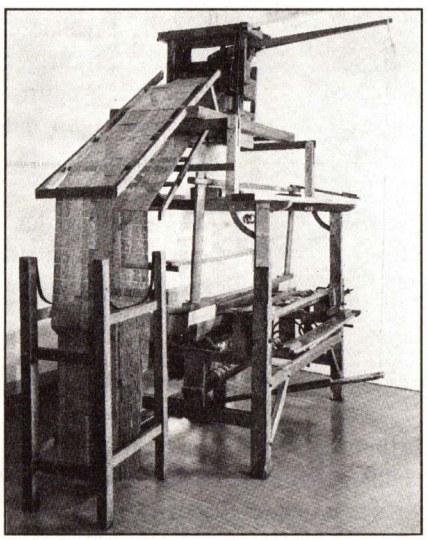
```
#include <intuition/intuition.h>
#include <functions.h>

struct IntuitionBase *IntuitionBase;
struct Window *Win;

struct NewWindow NewWindow =
{
  10,10,400,200,
  0,1,
  CLOSEWINDOW,WINDOWCLOSE|ACTIVATE,
  NULL,NULL,
  (UBYTE*)"EIN FENSTER",
  NULL,NULL,
  0,0,0,0,
  WBENCHSCREEN
};


main()
{
  if(!(IntuitionBase =(struct IntuitionBase *)
                        OpenLibrary("intuition.library",0L)))exit (10L);
  if(!(Win = (struct Window *)OpenWindow(&NewWindow)))
         { CloseLibrary (IntuitionBase); exit(10L);}
         Wait(1L<<Win->UserPort->mp_SigBit);
         CloseWindow(Win);
         CloseLibrary (IntuitionBase);
}
```

Dieses 'C'-Programm öffnet ein Window am AMIGA



Der automatische Webstuhl von Jaquard

COBOL, der Wirtschaftssaurier

Professor Dr. Edsgar Dijstra, einer der bedeutendsten Forscher der modernen Form des Programmierens, sagte über COBOL:
"Der Gebrauch von COBOL paralysiert den Verstand; das Lehren dieser Sprache sollte daher als Verbrechen angesehen werden."
Der gute Professor hat eine Reihe solcher polemischer Aussagen über Programmiersprachen und Programmieren überhaupt auf Lager. Die wollen wir aber nicht alle auf einmal bringen.

COBOL ist eine Programmiersprache, die heutzutage mit Sicherheit nicht die gleiche Bedeutung hat wie vor 30 Jahren. Zweifellos hat sie in der Welt der Personal-Computer nie den Rang von BASIC, PASCAL oder dem Neuling 'C' erreicht. Aber bis vor 10 Jahren waren ungefähr 80% der gesamten professionellen Software in COBOL geschrieben.

COBOL wurde in einer Gemeinschaftsarbeit im Auftrag des Amerikanischen Verteidungsministeriums entworfen. Sie wurde nach bestimmten Ansprüchen für dieses Amt erdacht und sollte dabei helfen, die gesamte Verwaltung, die von

DAS GFA-BASIC 3.0 BUCH

KLAUS SCHNEIDER, OLIVER STEINMEIER, PETER FRITZEN

Lieferbar!

Als optimale Ergänzung zum Handbuch des neuen GFA-BASIC 3.0-Interpreters bietet sich dieses Buch an. In zwei Hauptteilen wird zunächst eine systematische Einführung in die Programmierung von BASIC unter Berücksichtigung der besonderen Fähigkeiten von GFA-BASIC auf dem Amiga gegeben. Hier werden dem Neuling vom ersten Einzeiler bis zu abstrakten Datentypen alle Möglichkeiten der strukturierten Programmierung mit zahlreichen, durch Flußdiagramme transparenter gemachten Beispielen nahegebracht. Doch auch BASIC-erfahrene Programmierer lernen hier die neuen Strukturen kennen, die sich doch sehr von denen anderer Dialekte unterscheiden.

Der zweite Teil bauf auf dem ersten auf und vermittelt weitere Kenntnisse der Programmierung, anhand von Programmen, die wiederum ausführlich beschrieben und erklärt sind. Hier seien ein leistungsfähiges Grafikprogramm sowie zahlreiche Beispiele zur Betriebssystemprogrammierung genannt.



Die Benutzung der verschiedenen Libraries und Intuition-Funktionen wird detailliert erklärt, so daß die Verwendung dieser Möglichkeiten in eigenen Programmen keine Schwierigkeiten bereitet.

Durch zahlreiche Anhänge – neben vielen Tabellen finden Sie auch ein sehr ausführliches Stichwortverzeichnis – wird das Buch optimal ergänzt und kann problemlos zum Nachschlagen von Details benutzt werden.

AUS DEM INHALT:

Erklärung der Schleifen- und Programmstrukturen
- FOR-NEXT, WHILE-WEND, REPEAT-UNTIL, DO-LOOP
- Prozeduren, Funktionsunterprogramme und Verzweigungen
- Rekursion
- Beispielprogramme

Variablentypen und Arrays
- numerische und Zeichenkettenvariablen
- Arrays zur Aufnahme großer Datenmengen

Multitasking in GFA-BASIC
- Reagieren auf Ereignisse
- Zeitabhängige Prozeduraufrufe

Programmentwicklung
- Programmplanung und -entwurf
- strukturierte Programmierung
- TOP-DOWN-Prinzip
- Fehlersuche
- Debugging-Möglichkeiten

Dateiverwaltung
- sequentielle Dateien
- Random-Access-Dateien
- Funktionen und Befehle zur Diskettenverwaltung

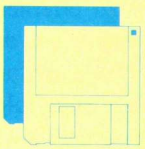
Grafikprogrammierung
- Die vielseitigen Möglichkeiten
- Turtlegrafik, Spriteprogrammierung
- Arbeiten mit mehreren Bildschirmen
- HAM-Modus
- Entwicklung eines vielseitigen Grafikprogramms

Sound- und Sprachprogrammierung
- Die Möglichkeiten der Befehle
- Verwendung der Anweisung in eigenen Programmen

Abstrakte Datentypen
- Verkettete Listen
- Bäume

Betriebssystemprogrammierung
- Pulldownmenü-Steuerung
- Requester-Einsatz in Programmen
- Intuition und Graphics-Library
- Aufruf von Systemroutinen

Für Einsteiger ★ Fortgeschrittene ★ und Profis



ÜBER 500 Seiten EINSCHLIESSLICH PROGRAMMDISKETTE

Dem Buch liegt eine Programm-Diskette bei mit über 150 Übungs- und Beispielprogrammen

BUCH & DISKETTE KOMPLETT HARDCOVER **DM 59,-**

Preise sind unverbindlich empfohlene Verkaufspreise

Heim Verlag

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

Schweiz: DataTrade AG
8021 Zürich
Langstr. 94

BESTELLCOUPON AMIGA GFA 3.0 BUCH

Bitte senden Sie mir _____ St. DAS GFA-BASIC **3.0 BUCH** einschließlich Programm-Diskette für DM **59,–** zzgl. DM 5,– Versandkosten (unabhängig von bestellter Stückzahl)

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname _____

Straße, Hausnr. _____ PLZ/Ort _____

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte.

Anfang an reichlich mit Computern ausgestattet war, zu reprogrammieren. Die Sprache sollte in ihrer Syntax ähnlich dem Englischen strukturiert sein und von jedem gelesen werden können. Außerdem sollte sie so leistungsfähig sein, alle möglichen Wirtschaftsprobleme oder Verwaltungsaufgaben lösen zu können. COBOL wurde später vom American National Standard Institut (ANSI) stark standardisiert, so daß diese Sprache in ihren verschiedenen Versionen fast immer die gleiche geblieben ist.

BASIC,
DER INTERPRETER PAR EXCELLENCE

BASIC wurde Ende der 60er Jahre entwickelt. Der direkte Vorfahr ist eindeutig FORTRAN, obwohl die Sprache auch Elemente von ALGOL besitzt. Kemeny und Kurtz, die BASIC am Darmouth College in New Hamphshire entwickelten, wußten damals wohl nicht, daß sie eine Sprache geschaffen hatten, die heute der Standard für fast alle Home-Computer und die meistverbreitete Sprache im Personal-Computer-Bereich ist. Ihre große Stärke liegt in erster Linie darin, daß sie eine Interpreter-Sprache ist. Obwohl ihr Sprachumfang relativ groß ist, bleibt sie dennoch sehr einfach zu erlernen. Der Vorteil eines Interpreters ist, daß der Dialog Mensch-Rechner sofort stattfindet. Nach der Anweisung folgt die sofortige Ausführung ohne Zwischenphasen. Deswegen ist diese Sprache vor allem bei Schülern sehr beliebt und für Unterrichtszwecke gut geeignet.

Ein großer Nachteil von BASIC war in der Anfangszeit, daß sehr viel Freiheit bei der Programmierung zugelassen wurde. Die Möglichkeit, in Unterroutinen hin- und herzuspringen, machte die Programme fast unlesbar. Solche Mankos wurden aber im Laufe der Zeit behoben, so daß die neuen Generationen wesentlich erweitert wurden und mittlerweile so aussehen, daß sie eher an PASCAL erinnern.

APL,
EINE EINFÜHRUNG IN DIE HIEROGLYPHEN

APL steht für "A Programming Language", was natürlich auf alle Interpreter und Compiler zutrifft. APL wurde ursprünglich als universelle Notation zur Vermittlung von logischen (mathematischen) Zusammenhängen entwickelt und erstmals am legendären IBM 360 implementiert. Seither hat sich die Sprache in dieser Größenordnung rapide verbreitet. Dennoch blieb APL großen Anwenderkreisen unbekannt. Das Erscheinen sogenannter "Supermicros" Ende der 70er Jahre bescherte APL eine Verbreitung in der dezentralen Datenverarbeitung.

Die Interpreter-Sprache APL ist ständig verbessert worden. Bei APL-Befehlen handelt es sich nicht um pseudo-englische Sprachschöpfungen wie "ON...GOSUB", sondern um Symbole: eine Bildsprache also, wie sie im alten Ägypten verwendet wurde. Die Programme in APL werden modular aufgebaut. Die einzelnen Routinen werden als Funktionen berechnet - und es sind auch welche, da sie im Programm mit exakt derselben Syntax verwendet werden wie die im Sprachumfang enthaltenen Funktionen. Dieser Umstand deutet schon an, daß (ähnlich wie im FORTH) APL vom Anwender erweitert und ausgebaut wird.



Charles Babbages opferte sein ganzes Leben der Entwicklung von Rechenmaschinen

APL-Programme sind meist sehr kurz: Üblicherweise nimmt ein vergleichbares Programm in BASIC oder PASCAL etwa zehnmal soviele Zeilen in Anspruch.

BASIC	APL	PASCAL

10 LET S=0
20 INPUT 1,M
30 FOR N=1 TO M
40 INPUT 1,V
50 LET S=S+V
60 NEXT N
70 PRINT S/M
	(+/B)÷pB	PROGRAM mean (input,output);
		VAR
		value,sum,mean,real;
		count:integer;
		BEGIN
		sum:=0;
		count:=0
		read (value);
		WHILE NOT eof DO
		BEGIN
		sum:=sum+value;
		count:=count+1
		read (value)
		END;
		mean:=sum/count
		writeln (mean)
		END

Hier kann man sehen, wie mächtig die Sprache APL ist

FORTH,
DIE VIERTE DIMENSION

Charles Moore schuf diese Sprache in den 60er Jahren als Konkurrenz zu den damals vorhandenen Sprachen, die für Steuerungszwecke ineffektiv waren. Der erste Einsatz dieser "vierten Generation" (daher der Name) war die Ansteuerung eines Teleskops in einem Observatorium.

Wie die anderen höheren Programmiersprachen besizt FORTH Sprachelemente bzw. feste Vereinbarungen, die der Anwender beachten muß. Ein wesentlicher Punkt, in dem sich die Sprache von anderen unterscheidet, ist, daß sich aus den definierten Sprachelementen neue Funktionen ableiten lassen. Dadurch hat der Programmierer unbegrenzte Möglichkeiten, verschiedene Funktionen selbst zu definieren. Ein anderer Vorzug

von FORTH liegt in der Maschinennähe, die es erlaubt, die jeweilige Prozessorarchitektur effektiv auszunutzen, und in der daraus resultierenden hohen Verarbeitungsgeschwindigkeit.

PASCAL, DER STRUKTURPHILOSOPH

Ein Nachteil von BASIC und FORTRAN war die Schwierigkeit, den Ablauf eines Programmes nachzuvollziehen. Die häufigen Sprünge in Unterroutinen machten das Programm sehr unlesbar. Heute gibt es BASIC-Versionen, die Prozeduren erlauben, und auch FORTRAN 77 kann besser strukturiert werden.
Nach dieser Erkenntnis wurde Mitte der 60er Jahre die Sprache ALGOL geschaffen, der ein völlig neues Konzept zugrunde lag. Man erkannte aber auch an dieser Sprache gewisse Schwachstellen.

Niklaus Wirth, der selbst an einer ALGOL-Version mitgearbeitet hatte, schuf 1970 an der Technischen Hochschule in Zürich PASCAL.

Diese Sprache, benannt nach dem bekannten Philosophen Blaise Pascal, wurde als Sprache für Informatiker, Mathematiker und Naturwissenschaftler konzipiert, ist aber ebenso für andere Zwecke geeignet (das Betriebssystem von MacIntosh z.B. wurde in PASCAL programmiert). Ihr Hauptmerkmal liegt im übersichtlichen Aufbau, der durch spezielle Befehle ermöglicht wird. Dazu gehören verschiedene Anweisungen zur Schleifensteuerung sowie komfortable Verzweigungsmöglichkeiten. Dies bildet die Grundlage für eine strukturierte Programmierweise, der PASCAL seine große Popularität verdankt. Ein anderer wesentlicher Pluspunkt von PASCAL ist die Menge an Datentypen sowie deren komplexe Zusammenfassung, die man Records nennt.

```
FUNCTION dateilaenge (   name     : string;
                     VAR laenge   : long_integer) : boolean;

 TYPE dta_type          = RECORD
                            reserviert : PACKED ARRAY [0..21] OF byte;
                            zeit       : integer;
                            datum      : integer;
                            groesse    : long_integer;
                            name       : PACKED ARRAY [1..14] OF char;
                          END;
   dta_ptr_type  =  ^dta_type;

   sys_str       =  PACKED ARRAY [1..80] OF char;

 VAR  dta_ptr                          :  dta_ptr_type;
      pfad                             :  sys_str;
```

Ein PASCAL-Programm, wo RECORDS angewandt werden

'C' IST NICHT NUR EIN BUCHSTABE

Die Sprache 'C' wurde in den 70er Jahren von Dennis Ritchie bei Bell Laboratories, einer Tochtergesellschaft von AT&T, entwickelt. Man wollte das bis zu diesem Zeitpunkt in Assembler PDP11 geschriebene Betriebssystem UNIX umschreiben, um es auch für andere Maschinen lauffähig zu machen.

Bei der Entwickung von 'C' wurde besonders darauf geachtet, eine Sprache zu schaffen, die so unabhängig wie möglich von jeder Hardware-Architektur blieb. So gehören zum tatsächlichen Sprachkern keine Anweisungen für Ein- und Ausgabe, Dateizugriff und Speicherverwaltung. Alle diese Funktionen werden in externe, sogenannte C-Libraries implementiert. Damit erreicht man eine fast absolute Portabilität von Programmen aus verschiedenen Rechnern.

Die eigentlichen Anweisungen, Operanden und Operatoren beschränken sich bei 'C' auf ein paar Dutzend. Damit vereinfacht sich das Erlernen der Sprache, und der Compiler selbst kann sehr kompakt sein. Einerseits tendiert man dazu, zu jedem Compiler eine Standard-Library mitzuliefern, andererseits berücksichtigt man, daß die einzelnen Eigenschaften der eigenen Betriebssysteme optimal ausgenützt werden können. Gerade diese beliebige Anzahl von Routinen macht aus 'C' eine besonders leistungsfähige Sprache.



Alan Turin, einer der Großväter der modernen Computertechnologie

Man behauptet, daß in 'C' geschriebene Programme fast genauso schnell wie Assembler sind und zusätzlich allen Elementen strukturierte Programme zur Verfügung stehen. Zur Entstehung solcher Gerüchte trägt sogar der Schöpfer dieser Sprache bei. In seinem Buch ('The C Programming language', zusammen mit Kernigham) beschreibt er, daß diese Sprache die Möglichkeiten heutiger Computer widerspiegelt und daß es keinen Grund mehr zur Programmierung in Maschinensprache gäbe. Natürlich reicht diese Aussage nicht zu der Annahme aus, daß 'C' doch so schnell wie Assembler sein könnte. Es gibt aber Elemente, die diese Behauptung bekräftigen. Die Datentypen von 'C' sind sehr 'maschinenherkömmlich' (Integer, Longs, Char, etc.). Prozessoroperationen lassen sich durch 'C'-Operatoren sehr leicht nachbilden. Durch die Bildung von Zeigern lassen sich Adressen direkt ansprechen bzw. Inhalte manipulieren.

Eine andere Besonderheit dieser Sprache ist der sogenannte Präprozessor. Dieser erledigt eine Menge Aufgaben, bevor der Compiler mit der Übersetzung anfängt. Alle Header werden integriert, die Konstanten mit ihrem tatsächlichen Wert vorbelegt, Funktionen, die durch Makros definiert wurden, substituiert usw.

'C' hat in fast allen Bereichen Anwendung gefunden. Von maschinennaher Programmierung, bei der die schnelle Ausführung sehr wichtig ist, bis zum Einsatz in der Herstellung von CAD und

Textverarbeitungsprogrammen findet man Programme, die in 'C' geschrieben wurden. Viele Betriebssysteme wurden ebenfalls in dieser Sprache geschrieben.

MODULA-2

Modula ist eine Weiterentwicklung von Pascal. Wie bei den eben genannten Sprachen ist auch hier N. Wirth der geistige Vater. 1978 wurde Modula der Öffentlichkeit in Form einer Maschine gezeigt, die nur diese Sprache beherrschte. Modula ist stark nach Pascal aufgebaut, aber in der Leistung wesentlich verbessert. Die Programme werden in kleine Einheiten zerlegt, die man Module nennt. Jedes Modul ist völlig unabhängig und kann getrennt übersetzt werden. Die Module entsprechen etwa Bibliotheksstücken, die jederzeit in ein anderes Programm implementiert werden können. Die Prozeduren lassen sich bei Modula als Datentypen deklarieren, so daß eine sehr große Flexibilität angeboten wird. Durch die implementierten Coroutinen können in Modula nebenläufige Prozesse (Multitasking) realisiert werden.

LOGO, NICHT NUR FÜR KINDER

LOGO wurde Ende der 60er Jahre von Seymour Papert, Professor im Institute of Technology, Massachusetts, das sich mit künstlicher Intelligenz beschäftigt, entwickelt. LOGO gehört zu den Interpreter-Sprachen und ist daher sehr schnell zu beherrschen.
Papert, Mathematikprofessor und ehemaliger Student des schweizerischen Psychologen Jean Piaget (bekannt als Vater der modernen Psychologie), strebte nicht nach einer neuen Sprache, die das Optimum eines Computers bilden sollte, sondern eher nach einem pädagogischen Werkzeug, das die intellektuellen Fähigkeiten der Studierenden entwickeln sollte - genauso, wie der tatsächliche Lernprozeß komplizierte Vorgänge erleichtert.

So entstand eine Programmiersprache, die im Dialogbetrieb mit dem Anwender arbeitet und in der Lage ist, schwierige Prozesse anschaulich zu lösen. Dazu dienen kurze Befehle in Form von einfachen und bekannten Begriffen. Beim damaligen Stand der Technik war es unmöglich, solche Lösungen auf einem Sichtgerät darzustellen. Papert schuf eine mechanische "Schildkröte", an deren Unterseite ein Stift befestigt war. Beim Fahren auf einer Papierunterlage hinterließ sie Linien. Heute hat sich diese motorisierte "Schildkröte" in ein stilisiertes Dreieck verwandelt, das auf dem Bildschirm Figuren zeichnen kann. Zusätzlich zu den Befehlselementen, die zur Verfügung stehen, kann man die Sprache durch selbstdefinierte Funktionen erweitern, die in jedes Programm eingebaut werden können. LOGO hat heute im Bereich der künstlichen Intelligenz seinen Platz als Untermenge der Sprache Prolog.

LISP UND PROLOG, SPRACHEN DER KÜNSTLICHEN INTELLIGENZ

LISP und PROLOG sind ohne Zweifel die Muttersprachen der künstlichen Intelligenz. Am Anfang hatten sie nicht dieselbe Verbreitung gefunden wie andere Sprachen. Aber seit einigen Jahren hat sich der Computer-Fachmann immer mehr mit dem Thema Computer-Intelligenz beschäftigt, und so wurden die Sprachen neu entdeckt. Sie gehören zu sogenannten applikativen bzw. deklarativen Sprachen. Bei den imperativen Sprachen (FORTRAN, BASIC, PASCAL, FORTH, C ...) wird ein Problem nach einem bestimmten, vom Programmierer festgelegten Ablauf gelöst. Es hängt von der Fähigkeit des Programmierers ab, die verschiedenen Ausnahmesituationen, an die ein Programm während seines Ablaufs stoßen könnte, zu berücksichtigen. Hat der Programmierer darin wenig Erfahrung, wird sich das Programm entsprechend wenig zuverlässig verhalten.

Ein Programm, das in einer applikativen Programmiersprache geschrieben wurde, besitzt keinen festen Algorithmus, sondern eher eine Softwarewelt aus Regeln und Schlußfolgerungen. Die eigentliche Lösung von Fragestellungen besorgt dann ein (in den Compiler oder Interpreter bereits integrierter) Backtracking-Algorithmus. LISP ist die älteste KI-Sprache (und nach FORTRAN die zweitälteste Computersprache überhaupt) und existiert in vielen Versionen, in denen meist auch reichlich imperative Sprachelemente vorhanden sind. LISP entstand zunächst aus einer Kombination von FORTRAN und IPL. John McCarthy, der Vater von LISt Processor (LISP), vereinigte eine der wichtigsten Eigenschaften von IPL, die 'Listen', als Datenstrukturen mit den Grundfunktionen von FORTRAN. 1960 war ein komplettes LISP-System fertig, das aus einem Interpreter und einem Compiler bestand.

PROLOG steht für 'Programming in Logic' und bezieht sich auf eine Idee, die Anfang der 70er Jahre in den Kreisen der Programmierer ziemlich neu war: das Programmieren nach einer 'logischen' Denkweise. PROLOG wurde jedoch wiederentdeckt, als die Japaner diese Sprache als Sprache der 'fünften Generation' und als Bestandteil ihrer neuen Technologie gewählt hatten. PROLOG befreit den Programmierer vom imperativen Ballast seines Problems bzw. seines Problemlösungswegs. Man könnte sogar sagen, daß mit PROLOG der Traum erreicht worden ist, einen Computer programmieren zu können, ohne eine bestimmte Programmiersprache lernen zu müssen: ein völlig neues Konzept.

Auch eine PROLOG-Version ist von unserem PD-Service zu erhalten. TOY-Prolog ist eine Implementation des bekannten DEC-10-Dialekts. Diese Version ist nicht sehr schnell, doch trotzdem für den PROLOG-Neuling ein gutes Werkzeug zum Einsteigen in die Welt der fünften Generation.

Wie Sie sehen, wurde der Computer mit der Zeit ein Polyglotte. Zahlreiche Sprachen wurden für die unterschiedlichen Probleme geschaffen, so daß heute die Landschaft der Computersprachen unübersichtlich geworden ist.
Einige haben wir hier überhaupt nicht erwähnt. Entweder sind diese Sprachen für den AMIGA gar nicht vorhanden (das ist z.B der Fall bei PEARL, ADA etc.), oder sie sind wegen ihrer Struktur mehr oder weniger obsolet. Programmiersprachen wie ALGOL oder PLI finden in der Welt der großen Anlagen immer ihren Platz.

Trotzdem haben sich im Bereich der Personal-Computer die Programmier-

sprachen ebenso schnell vermehrt wie in der Welt der größeren Anlagen. Nur wenige völlig unbedeutsame Sprachen sind an Rechnern wie dem AMIGA nicht implementiert worden. Von einem Standard kann insofern nicht die Rede sein. Vor allem bringt jede Programmiersprache verschiedene Versionen hevor, die zueinander nicht vollständig kompatibel sind. Eine Sprache als 'die Sprache überhaupt' zu betrachten, ist nicht gerechtfertigt. Es ist fast wie bei den Geschmäckern. Sie sind einfach unterschiedlich.

APL:
APL 68000 Amiga
Philgerma, München
298,- DM

Assembler:
K-Seka Assembler
Philgerma, München
168,- DM

MCC-Macro-Assembler
Metacomco, Bristol, United Kingdom
168,- DM

DevPac-Assembler
Philgerma, München
148,- DM

Profimat Assembler
Data Becker, Düsseldorf
98,- DM

KICK-ASS
MAXON Computer, Eschborn
89,- DM

ASM68K
PD-Software
KICKSTART PD-Sammlung

BASIC:
AC-BASIC-COMPILER
ABSoft Corp., Royal Oak, USA
298,- DM

TrueBASIC Interpreter/Compiler
Pfotenhauer, Achern
398,- DM

GFA-BASIC 3.0 Interpreter
GFA, Düsseldorf
198,- DM

C:
Lattice C 5.0
Lattice Inc., Glen Ellyn, USA
698,- DM
Lattice C++
Lattice Inc., Glen Ellyn, USA
998,- DM

Aztec C68k 3.6
MAXON Computer, Eschborn
298 - 598,- DM

Forth:
J-Forth
Delta Research, Palo Alto, USA
348,- DM

Fortran:
AC/Fortran 77 Compiler
ABSoft Corp., Royal Oak, USA
598,- DM

Modula:
TDI Modula-2
TDI Software Inc., Dallas, USA
198 - 598,- DM

M2S Modula-2
A.+L. Meyer-Vogt, Bonstetten, Schweiz
338,- DM

Benchmark Modula-2
Philgerma, München
428,- DM

Pascal:
MCC Pascal-Compiler 2.0
Metacomco, Bristol, United Kingdom
298,- DM

Prolog:
Philgerma Prolog
Philgerma, München
198,- DM

Stony Brook-Prolog
PD-Software
KICKSTART PD-Sammlung

Lisp:
Cambridge Lisp
Metacomco, Bristol, United Kingdom
448,- DM

XLisp
PD-Software
KICKSTART PD-Sammlung

Alle Sprachen sind im gutsortierten Fachhandel zu erhalten.

ENDE

SOFTWARE AUS DEUTSCHLAND

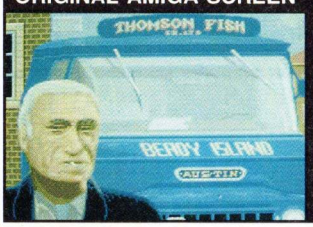
Schalten Sie die Glotze aus ...
Legen Sie das Buch weg ...
Erleben Sie:

HOLIDAY MAKER

Die neue Art von Computer-Unterhaltung
EIN PM-ARTVENTURE



ORIGINAL AMIGA SCREEN



SPANNEND WIE EIN AUFREGENDER FILM AUFREGEND WIE EIN SPANNENDES BUCH

Story und Grafik dieses deutschen Adventures von PM ENTERTAINMENT setzen neue Maßstäbe. Ein unterhaltendes Vergnügen für alle, die »shoot-them-up games« satt haben.

2 Disketten inkl. deutscher Anleitung DM 89,-
Empfohlen ab 16 Jahren.

Sprite Animator

Sprite-Editor der Luxusklasse
Sprites bis zu 16 Bewegungsphasen unterstützt
Assembler-, C-, Basic- und IFF-Format
sehr hohe Verarbeitungsgeschwindigkeit durch Assemblerprogrammierung
Preis DM 99,—

BUNDESLIGA Manager

Werden Sie zum Manager Ihres Fußballvereins.
Verhelfen Sie Ihrem Verein zu Siegen und Geld.
Sehr gute Wirtschafts-Simulation, hoher Spielspaß,
bis zu 4 Spieler
Preis DM 69,-

Unsere Programme laufen auf allen Amiga-Modellen mit mind. 512 K Speicher. Gegen 1,90 DM in Briefmarken erhalten Sie ausführliche Produkt-Info's. Versand gegen Nachnahme oder Vorkasse (V-Scheck), Ausland nur gegen Vorkasse, zuzügl. 5,- DM Versandspesen.

Preise sind unverbindliche Preisempfehlungen.

Distributoren:

BRD:
Casablanca GmbH
Nehringskamp 9
4630 BOCHUM 5
Tel. 02 34 / 41 19 94
Es betreut Sie Herr Hollax

ÖSTERREICH:
Intercomp A. Mayer
Heidendankstraße 24
A-6900 BREGENZ
Tel. 0 55 74 / 2 73 44

SOFTWARE 2000
Lübecker Straße 10
2320 Plön/Holstein
Telefon:
04522/1379

GENLOCK 8802 DM 998,-
POLAROID PALETTE
mit IMPRINT neu DM 6750,-
. **Vorführmodell** DM 4500,-

DIA BELICHTUNGEN:
je DIA DM 4,95
ab 36 Stück DM 3,31
Rank Xerox OVERHEADFOLIE
Tintenstrahausldruck DM 15,-

VIDEO EFFECTS 3D
mit dt. Handbuch DM 379,-
3 DEMON
mit dt. Handbuch DM 215,-
TURBO SILVER V 3.0a
mit dt. Handbuch DM 359,-
THE DIRECTOR
mit dt. Handbuch DM 159,-
MODELER 3D
mit dt. Handbuch DM 218,-

VIDEOSCAPE 3D V 2.0
mit dt. Handbuch DM 379,-
PRO VIDEO PLUS
mit dt. Handbuch DM 598,-
ANIMATE 3D
mit dt. Handbuch DM 259,-
SCULPT 3D
mit dt. Handbuch DM 198,-
PHOTON PAINT
mit dt. Handbuch DM 249,-
PIXMATE
mit dt. Handbuch DM 149,-
STUDIO MAGIC
mit dt. Handbuch DM 129,-
SONIX
mit dt. Handbuch DM 175,-
FANTAVISION
mit dt. Handbuch DM 159,-
FISH und RPD Disks je DM 5,-
weitere INFO siehe LOFTPOST

AUDIO MASTER
mit dt. Handbuch DM 134,-
PERFECT SOUND Stereo Digitizer *dt.Handb.* DM 198,-
DeLuxe Music Construction Set *mit dt. Handbuch* DM 179,-
AZTEC C V 3.6 Prof.
mit dt.Übersetzung DM 499,-
AZTEC C V 3.6 DEV.
mit dt.Übersetzung DM 655,-
dt. Übersetzung des Manual
AZTEC C V 3.6 DM 138,-

ZING KEYS. *dt. HB* DM 129,-
DIGA *m. dt. HB* DM 169,-
CREDITS deutsch DM 98,-
Schriftgenerator - deutsche Umlaute (ä, ö, ü, ß) - Overscan -Grafikeinbingung - einfachste Bedienung

Für Besitzer der Originalprogramme gibt es deutsche Handbücher einzeln für DM 39,95

weitere INFO: LOFT POST
video LOFT film HARD & SOFTware GmbH
Fiedlerstr. 22 - 32
BRD 3500 Kassel
tel.: 0561 - 87 33 99 / 87 79 28
fax: 0561 - 87 80 48

video LOFT film HARD&SOFT ware GmbH

GET IT

Einsteiger aufgepaßt! Einsteiger zugefaßt!

Amiga Public Domain-Disketten

4,- DM ABO-Preis pro Disk (Normalpreis 5,- DM)
Ab 10 Stück. Porto/Verpackung frei.
Totale Public Domain-Inform.: 6 INFO-Disk. = 15,- DM
Die zur Zeit umfassendste, ausführlichste Information!
Neu: die 13te Ausgabe vom PD-Magazin auf Diskette: ab 10.04. »GET IT«.
Haben Sie Ihren Amiga gerade neu?
Lassen Sie sich durch unseren superaktuellen PD-POOL (über 2300 Disks) nicht abschrecken. Wir helfen Ihnen mit **Einsteiger-Paketen 9 Themen: jew. 10 Disks = 40,- DM**
> **GRAFIK** MODULA-2 **SPIELE** CLIhelps **IHRE WAHL** < > > > **SLIDESHOWS** BASIC **MAGAZINE** TOOLS < < <

AIT-UG B. Rönn
Erlenkamp 13, 4650 Gelsenkirchen
Telefon 02 09 / 14 63 14

Vertrauen Sie auf unsere 2jährige PD-Erfahrung.
Qualität und Service seit zwei Jahren.
Achten Sie auch auf unsere anderen Mini-Anzeigen.

AIT UG

---

fischer Leddinweg 14
Hard & Software 3000 Hannover 61
24 Stunden Bestellannahme 05 11 / 57 23 58

PUBLIC DOMAIN

ab 10 Stck. auf **2.75** **incl. 2 DD 3,5" Disk**

No Name	3,5"	2DD	2.75
Fuji	3,5"	2DD	3,65

Dragons Lair	99,–
Dungeon Master	79,–
Emerald Mine II	37,–
Elite	67,–
Hostages	59,–
Katakis	48,–
Pacmania	58,–
Laufwerk intern A2000	**188,–**
extern alle Extras	**255,–**

!! Laufend Neuerscheinungen !!
Versandkosten NN + 7 DM, VK + 4 DM

RHEIN-MAIN-SOFT

Ihr Public Domain-Partner

mit über 2200 Disketten aus ca. 40 Serien wie Fish, RPD, Taifun, Chiron, Kickstart, Panorama, Auge usw.

Fish	-172	Taifun	-90	S.A.F.E.	-31
RPD	-160	ACS	-130	Franz	-20
Auge	-30	RHS	-94	GERMAN	-40 (5,- DM)
Kickstart	-140	Cactus	-24	usw.	

ab 0,80

Preise: 3.5"/5.25"-Diskette(n) Disketten von uns
von Ihnen **3,5" 2,60 DM**
0,80 DM 5,25" 1,20 DM

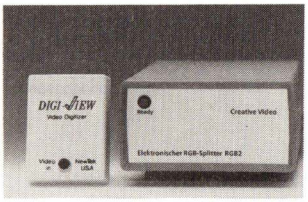
Katalogdisketten gegen 7,00 DM (V/Scheck/Briefmarken) anfordern

Preise zzgl. 4,00 DM Versandkosten b. Vorkasse (7,00 DM b. Nachnahme)

Auch Sonderserien, z. B. wie Amiga-PD-Buch, Buchhaltung, Haushaltsbuch, Etikettendruck, Perfect English usw., Abomöglichkeit

Rhein-Main-Soft · Postfach 39 · 6500 Mainz 32

Wollen Sie professionell in Farbe digitalisieren?



Wenn ja, dann führt kein Weg an unserem RGB-Splitter vorbei. Wir haben für jeden Digitizer den passenden RGB-Splitter. Wenn Sie einen DIGI VIEW besitzen erfolgt die Steuerung des RGB-Splitters sogar vollautomatisch über den Joystick-Port.

RGB2 für AMIGA 500/2000 (Gender Changer eingebaut)	**DM 498,–**
RGB3 für AMIGA 500/2000 zur Verwendung mit DIGI VIEW Gold	**DM 458,–**
RGB4 zur Verwendung mit DIAMOND Digitizer	**DM 358,–**

Händleranfragen erwünscht

Creative Video
Postfach 1501, 8520 Erlangen, Telefon 0 91 95 / 27 28

AMIGA PUBLIC DOMAIN

> 2500 Disketten!

auf 3,5" und 5,25"
– ab DM 1,20 – inkl. Markendisk

»A.P.S. -electronic-
Sonnenborstel 31, 3071 Steimbke
Tel. 0 50 26 / 17 00 (ab 18.00 Uhr)

VON ANDREAS KRÄMER

Prozessorwettstreit

14 MHz 68000er vs. 68020

Eine Familie tritt zum Wettstreit untereinander an. Gemeint ist die Motorola-Familie. In einem Vergleichstest soll herausgefunden werden, ob ein 68000-Prozessor mit doppelter Taktrate mit einem 68020-Prozessor mithalten kann. Die im Test verwendete 68000-Prozessorplatine ist im Vergleich zu einer 68020-Platine im Preis wesentlich günstiger. Bringt der Kompromiss etwas für den Anwender, oder ist auf jeden Fall der 68020-Prozessor, trotz des höheren Preises, vorzuziehen?

Der AMIGA wird serienmäßig mit einem 68000-Prozessor ausgeliefert. Die Taktrate, mit der der Prozessor sowie die Customchips (Paula, Denise, Agnus) betrieben werden, beträgt 7.16 MHz. Dieser Wert wird durch die Taktteilung eines 28 MHz-Quarzes erreicht. Die Taktrate kann mit der Geschwindigkeit, mit der der Computer rechnet, gleichgesetzt werden. Eine höhere Taktrate bringt also mehr Geschwindigkeit. Doch warum ist der 68020-Prozessor so wesentlich schneller, obwohl beide mit derselben Taktrate betrieben werden? Die erhöhte Geschwindigkeit des 68020-Prozessors beruht hauptsächlich auf der Tatsache, daß die interne Verarbeitung optimiert wurde. Auch der 256 Byte große Cachespeicher trägt dazu bei, beim 68000 gänzlich fehlt. Der Cachespeicher ist nichts anderes als ein schneller RAM-Speicher, in dem die nächsten Befehle bereits eingeladen werden können, während andere Teile des Prozessors noch mit der Bearbeitung des momentanen Befehls beschäftigt sind.

Im Gegensatz zum 68000er ist der 68020er ein reiner 32-Bitter, der in der Lage ist, einen 32-Bit breiten Bus zu verwalten, über den der Datentransfer erfolgen kann. Mit einem 32-Bit breiten RAM zeigt sich die wahre Geschwindigkeit des 68020-Prozessors. Unsere Testplatine des 68020 verfügte aber nicht über ein 32-Bit breiten Bus, so daß auch kein RAM angeschlossen werden konnte. Ein weiterer Vorteil des 68020-Prozessors ist, daß er direkt mit dem mathematischen Coprozessor 68881 bzw. 68882 harmoniert. Der 68000-Prozessor ist dazu nicht in der Lage.

Doch genug der Vorrede, kommen wir zu den Testkandidaten.

Die Testkandidaten

Das CSS Turbo Board mit 68020-Prozessor und Floating Point Unit 68881 stand dem CMI Prozessor Accelerator mit doppelter Prozessortaktgeschwindigkeit, demnach mit 14.32 MHz und 68881-FPU, gegenüber.

Das CSS Turbo Board besitzt keinen 32-Bit breiten Bus, weshalb auch kein 32-Bit-RAM angeschlossen werden kann. Die Abmessungen der Platine sind mit 9 x 8.5 cm sehr gering, dadurch kann die Platine leicht in alle AMIGA-Modelle eingebaut werden. Hierzu wird der 68000-Prozessor aus seinem Sockel entfernt und die 68020-Platine stattdessen eingesteckt. Der 68881 kann mit einem eigenen Takt betrieben werden, dazu steht ein Quarzsockel zur Verfügung. Auf einer mitgelieferten Diskette finden sich einige Demoprogramme und ein Programm, um den Cachespeicher des Prozessors ein- bzw. auszuschalten.

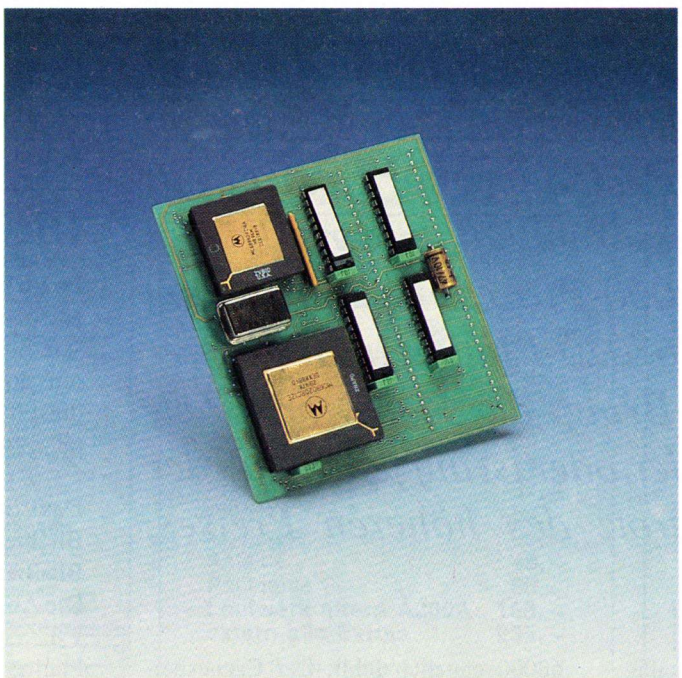
Der CMI Prozessor Accelerator besitzt einen normalen 68000-Prozessor, der jedoch mit 14.32 MHz, also mit doppelter Taktfrequenz betrieben wird. Nur eine spezielle Baureihe des 68000 verkraftet diese Geschwindigkeit, der serienmäßig im AMIGA eingebaute Prozessor würde bei dieser Taktrate nicht

funktionieren. Auf Wunsch kann die Karte auch mit der 68881-FPU ausgerüstet werden, dazu ist ein freier Sockel auf der Platine zu finden. Unser Testexemplar war mit FPU ausgerüstet, die mit 20 MHz getaktet wurde. Da der 68000 aber nicht in der Lage ist, die FPU direkt anzusprechen, ist etwas Logik und Software vonnöten. Bevor die FPU angesprochen werden kann, muß ein Programm von der mitgelieferten Diskette gestartet werden. Der Accelerator ist in verschiedenen Abmessungen erhältlich. Unsere Testplatine war für den AMIGA 2000 bestimmt.

Doch gehen wir auf die mitgelieferte Software ein wenig näher ein. Auf der Diskette befinden sich etliche Programme, die hauptsächlich für die Installation des mathematischen Coprozessors vorhanden sind. Aber auch die Taktfrequenz kann per Software bestimmt werden. Zusätzlich finden sich einige Demoprogramme, die die Geschwindigkeitssteigerung demonstrieren.

Am interessantesten sind jedoch die Programme zum Ansprechen der FPU. Hier stehen mehrere zur Verfügung. Warum? Nun, auf der mitgelieferten Diskette ist eine spezielle FPU-Library zu finden, die mit dem Programm 'FPUInstall' aktiviert werden kann. Die FPU ist jetzt multitaskingfähig. Wird die FPU mit einem anderen Programm angesprochen, steht nur ein Task zur Verfügung.

Das große Manko des 68881 besteht jedoch darin, daß er nicht über das Betriebssystem erreichbar ist, sondern über eigene Routinen eingebunden wird. Das hat zur Folge, daß Programme, wie beispielsweise die Floating Point-Version von 'TURBO SILVER', die FPU nicht unterstützen. Diese TURBO SILVER-Version funktioniert nur mit einem 68020-Prozessor, der mit einer FPU ausgerüstet ist. Der Programmierer ist auf sich gestellt. Eine neue Library macht dies erst möglich. Auf der neuen Workbench 1.3 findet sich die neue Library *'mathieeedoubtrans'*, die das Ansprechen des 688881 ermöglicht. Wird diese Library geöffnet, spricht der Prozessor Accelerator die FPU ohne Probleme an. Zu bemerken ist noch, daß nur der 68000-Prozessor schneller getaktet wird, die Taktrate der Customchips bleibt hingegen unverändert und beträgt nach wie vor 7.16 MHz. Programme, die nur den Prozessor nutzen, erleben eine deutliche Geschwindigkeitssteigerung. Bei Programmen, die die Customchips nutzen, ist die Steigerung der Geschwindigkeit natürlich geringer.



Das CSS Turbo Board ist mit 68020-Prozessor und der Floating Point Unit 68881 ausgerüstet und beschleunigt den AMIGA im Durchschnitt um über 50%.

Die Geschwindigkeit

Die grundlegende Frage, die schon weiter oben gestellt wurde und wohl jeden interessiert, ist natürlich die: Kann ein mit doppelter Taktrate betriebener 68000-Prozessor mit einem 68020-Prozessor mithalten? Die Frage muß klar und deutlich mit einem 'Nein' beantwortet werden. Der 14 MHz-68000er kann nicht mithalten. Das CSS-Board mit dem 68020-Prozessor ist noch einmal um ein Vielfaches schneller. Um dies zu verdeutlichen, habe ich natürlich etliche Tests unternommen. Es mußte wieder einmal ein Benchmark herhalten. Der erste Benchmark war eine Leerschleife, die 1000000mal durchlaufen wurde; der zweite Benchmark führte eine einfache Addition bzw. Subtraktion durch, diese Rechenoperation wurde ebenfalls 1000000mal durchgeführt. Der dritte Benchmark rechnete mit Float-Zahlen. Der Sinus, Cosinus bzw. Tangens mußte 10000mal berechnet werden. Hier mußte die 688881 zeigen, was in ihr steckt. Ein weiterer Geschwindigkeitstest wurde mit dem in diesem Heft abgedruckten Apfelmännchen-Programm durchgeführt. Hier zeigte sich mit 35% die größte Geschwindigkeitssteigerung gegenüber einem mit normaler Taktfrequenz betriebenen 68000-Prozessor. Der 68020 war aber noch einmal um über 40% schneller. Der 68881 kam bei diesem Test nicht zum Zuge, da das Apfelmännchenprogramm lediglich mit Integer-Werten rechnet. Schon dieser Test zeigt, daß der Prozessor Accelerator nicht mit dem 68020 mithalten kann.

Die erste Benchmark (Leerschleife) brachte eine Geschwindigkeitssteige-

rung von knapp 12%, die zweite knapp 17%, lediglich die dritte kam auf über 30% ohne FPU, mit FPU lag die Steigerung bei 90%. Die Prozentangaben beziehen sich auf einen normalen 68000 mit 7.16 MHz Taktrate.

Das CSS-Board mit 68020 legt gegenüber dem 68000-Prozessor mit 14.32 MHz und FPU bei den Zeiten noch einmal kräftig zu. Bei den Benchmarks 1 und 2 ist eine weitere Steigerung von über 50%, bei dem Floating Point-Benchmark jedoch nur 10% Steigerung festgestellt worden.

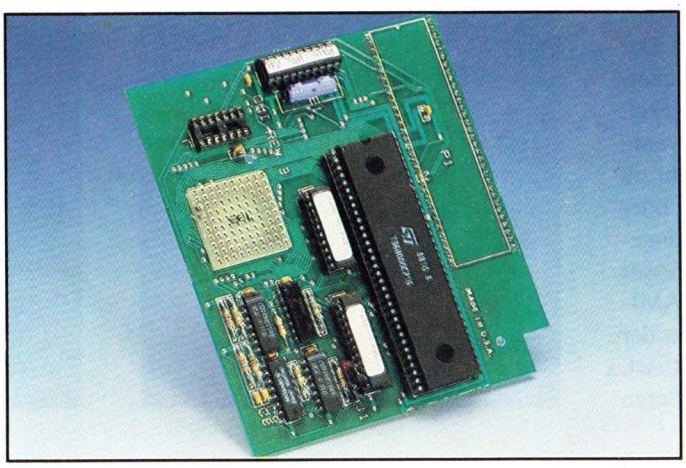
Die genauen Zeiten können Sie in der nebenstehenden Tabelle noch genauer studieren.

Fazit

Ein mit doppelter Taktrate betriebener 68000-Prozessor, wie er mit dem CMIProzessor Accelerator vorlag, kommt an die Geschwindigkeit eines 68020-Prozessors nicht heran. Der 1. Platz im Prozessorwettstreit geht somit eindeutig an den 68020. Das CSS Turbo Board ist mit dem 68020 schneller. Jedoch besitzen beide 'AMIGA-Beschleuniger' ihre Vor- bzw. Nachteile. Der Hauptvorteil des CSS Turbo Boards liegt bei der Geschwindigkeit, der Hauptnachteil ist jedoch der Anschaffungspreis. Ein weiterer Nachteil besteht darin, daß manche Software nicht mit dem 68020-Prozessor harmoniert, besonders bei Spielen kommt es des öfteren zu Systemabstürzen, aber auch bei verschiedenen Anwenderprodukten meldet sich der Guru.

Der Prozessor Accelerator kennt Kompatibilitätsprobleme nicht. Jede Software funktionierte ohne Probleme. Da die CMI-Platine über sogenannte Jumper verfügt, kann die Einstellung auf 7.16 oder 14.32 auch hardwaremäßig und nicht nur über die mitgelieferte Software erfolgen. Aus diesem Grunde müssen nicht auf jede Boot-Diskette die Programme kopiert werden, die das Umschalten der Taktrate ermöglichen.

Der Preis des CMI Prozessor Accelerators liegt bei etwa 500.- DM, ohne Floating Point Unit. Eine Steigerung von bis zu 35% sind mit ihm möglich, damit bietet er ein angemessenes Preis-/Leistungsverhältnis. In der Regel liegt die Steigerung aber zwischen 10 und 20 Prozent. Mit der 68881-FPU ist der Prozessor Accelerator um etliches teurer, weil die FPU in vergleichweise geringen Stückzahlen produziert wird. Die Geschwindigkeitssteigerung mit FPU ist aber enorm und hinkt der des 68020 mit 68881-Lösung kaum hinterher. Der Nachteil der 68881-FPU auf der Accelerator-Platine besteht jedoch darin, daß die meiste Software, die den 68881 unterstützt, den Matheprozessor über den 68020 anspricht. Bei solcher Software nutzt der 68881 des Accelerators recht wenig. Mit der neuen Workbench 1.3 ist aber Besserung im Anmarsch. Eine neue Mathe-Library unterstützt die FPU. Mit ihr können Sie die FPU ohne Probleme ansprechen und deren Geschwindigkeit nutzen. Kommerzielle Programme müssen freilich erst



Der CMI Prozessor Accelerator besitzt einen 16 MHz 68000-Prozessor, der mit doppeltem Systemtakt betrieben wird.

Die Geschwindigkeiten im Vergleich

	Benchmark 1	Benchmark 2	Benchmark 3
CSS Turbo Board (68020 mit 68881-FPU	2.10	2.84	9.84
CMI Prozessor Accelerator 14.32 MHZ Taktrate	4.46	6.52	93.72
CMI Prozessor Accelerator 14.32 MHz und 68881-FPU	4.46	6.52	11.02
Normaler AMIGA 7.16 MHz	5.02	7.86	136.54

Alle Zeitmessungen in Sekunden

an die neue Library angepaßt werden, was hoffentlich etliche Softwarefirmen beherzigen werden.

Die Dokumentation des Accelerators ist bisher nur in englischer Sprache, so daß deutsche Anwender evtl. ihre Schwierigkeiten bekommen können. Selbiges gilt für das 68020-CSS-Board. Hier müßte doch Abhilfe geschaffen werden. Der Einbau der beiden Platinen ist denkbar einfach, jedoch erlischt gegebenenfalls die Garantie, da der Rechner geöffnet werden muß. Aus diesem Grunde sollte der Einbau vom Fachmann vorgenommen werden. Lassen Sie mich noch kurz versuchen die verschiedenen Zielgruppen der Platinen zu definieren. Der CMI Prozessor Accelerator ist sicherlich für AMIGA-Anwender entwickelt worden, die ihren AMIGA ein wenig tunen möchten, dabei jedoch nicht zu tief in die Tasche greifen wollen oder können. Das CSS Turbo Board hingegen bietet einen echten 32 Bit-Prozessor mit 68881-FPU und ist damit für AMIGA-Anwender entworfen, für die der High-Tech-Prozessor interessant ist, und die auch nicht vor einer größeren Ausgabe zurückschrecken.

CSS Turbo Board 68020/ 68881

+ eigener Takt für die 68881-FPU möglich
+ leichter Einbau
+ saubere Verarbeitung
+ Geschwindigkeitsteigerung von über 50% (Durchschnittswert, ohne FPU)
+ kompatibel zu kommerzieller FPU-Software

- mangelhafte Dokumentation
- kein 32 Bit-Bus

Preis: 1490.- DM

Anbieter:
C.S.S. / Kàrben 1
Tel. 06039-5776

CMI Prozessor Accelerator

+ eigener Takt für die 68881-FPU möglich
+ leichter Einbau
+ saubere Verarbeitung
+ software- und hardwaremäßiges Einstellen des Prozessortaktes
+ kompatibel zu jeder AMIGA-Software

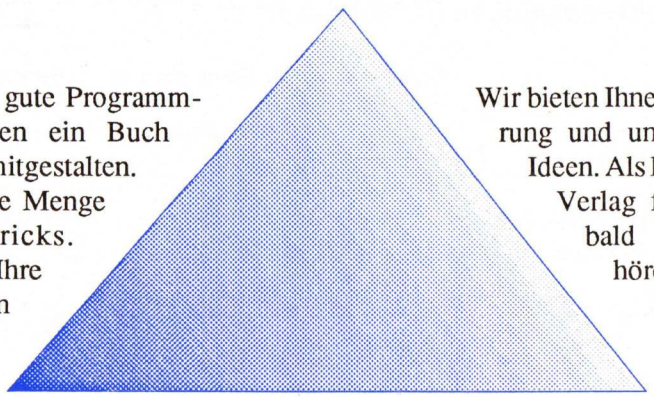
- englischsprachige Dokumentation
- FPU nur bedingt ansprechbar (mit neuer Mathe-Library von WB 1.3)

Preis: 498.- ohne FPU
998.- mit FPU

Anbieter:
IM / Frankfurt
Tel: 069/410072

ENDE

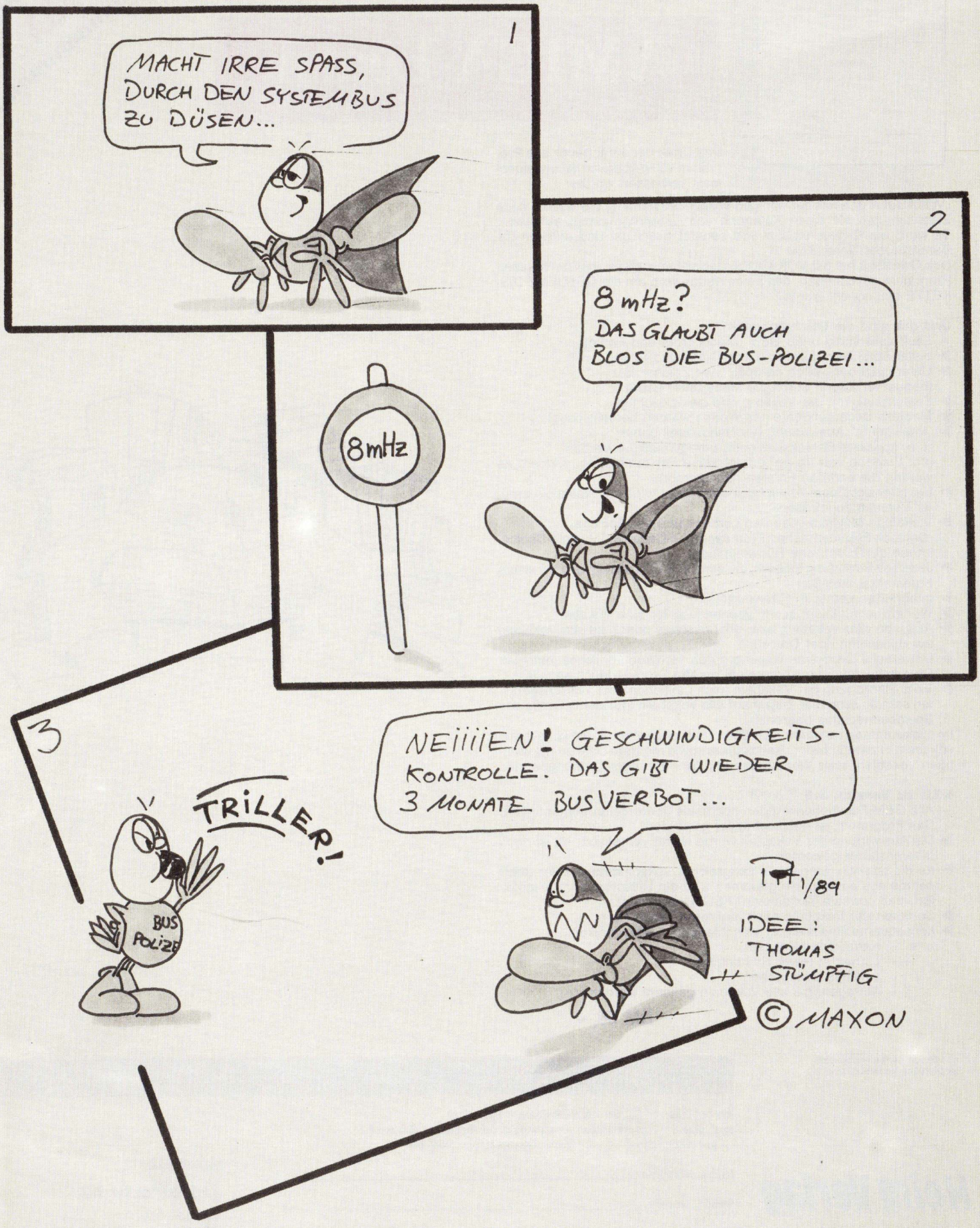
Haben Sie eine gute Programmidee und wollen ein Buch schreiben und mitgestalten. Kennen Sie eine Menge Tips und Tricks. Möchten Sie Ihre Erfahrungen weitergeben.



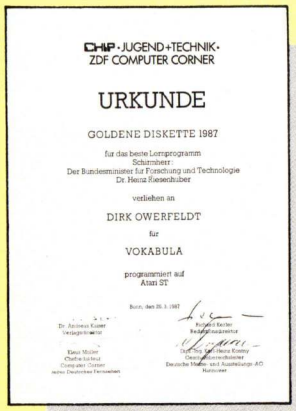
Wir bieten Ihnen unsere Erfahrung und unterstützen Ihre Ideen. Als leistungsstarker Verlag freuen wir uns bald von Ihnen zu hören.

Wir suchen noch Autoren wie Sie.

HeimVerlag Kennwort: Autor Heidelberger Landstr. 194 6100 Da.-Eberstadt Tel.: 06151/56057


1
MACHT IRRE SPASS, DURCH DEN SYSTEMBUS ZU DÜSEN...
2
8 mHz?
DAS GLAUBT AUCH BLOS DIE BUS-POLIZEI...
8mHz
3
NEIIIIEN! GESCHWINDIGKEITS-KONTROLLE. DAS GIBT WIEDER 3 MONATE BUSVERBOT...
TRILLER!
BUS POLIZEI
TT 1/89
IDEE: THOMAS STÜMPFIG
© MAXON

Jetzt für AMIGA



CHIP · JUGEND+TECHNIK · ZDF COMPUTER CORNER

URKUNDE

GOLDENE DISKETTE 1987

für das beste Lernprogramm
Schirmherr:
Der Bundesminister für Forschung und Technologie
Dr. Heinz Riesenhuber

verliehen an

DIRK OWERFELDT

für

VOKABULA

programmiert auf
Atari ST

Bonn, den 20. 3. 1987

Dr. Andreas Koster
Verlagsdirektor

Richard Becker
Redaktionsdirektor

Klaus Müller
Chefredakteur
Computer Corner
beim Deutschen Fernsehen

Dipl.-Ing. Ludolfine Koenig
Chefredakteurin
Deutsche Messe- und Ausstellungs-AG
Hannover

Dieses Siegerprogramm des Programmierwettbewerbes GOLDENE DISKETTE 87 ist die Grundlage für den neuen Vokabeltrainer aus dem Heim-Verlag:

AMIGA-Learn

Vers. 3.5

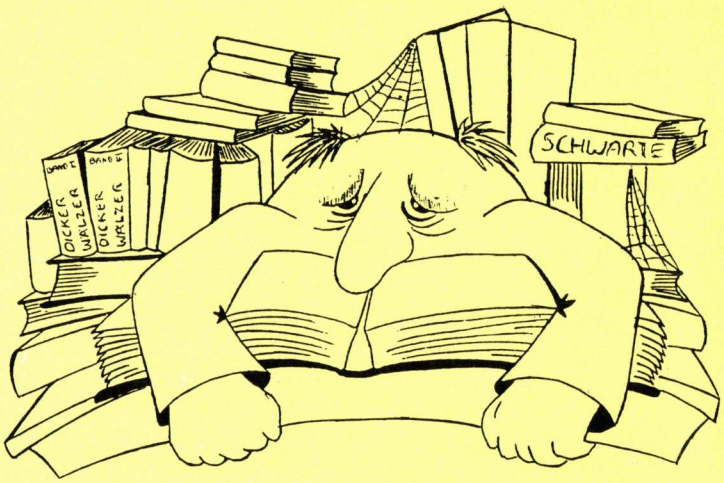
Das Urteil der Jury, **bevor** das Programm völlig überarbeitet und nochmals verbessert wurde:

„VOKABULA ist eines der wenigen Vokabelprogramme, das wirklich neue Eigenschaften für diese Kategorie von Lernprogrammen aufweist... Es wurd ein C geschrieben und benutzt geschickt und effizient die Benutzeroberfläche GEM...
Dirk Owerfeldt hat mit VOKABULA das bisher professionellste Vokabel-Programm geschrieben, das beim Wettbewerb um die GOLDENE DISKETTE eingereicht wurde."

Und das sind die Leistungsdaten:

- Läuft vollständig unter GEM (Monochrom und Farbe)
- ▶ bietet Spaß beim Lernen durch Grafik und Sound
- ▶ Unterstützt den vollen europäischen Zeichensatz (bequemer Zugriff durch die Maus unter GEM)
- ▶ Fehlerhäufigkeit der Vokabel wird berücksichtigt
- ▶ Mehrere Bedeutungen eines Wortes werden berücksichtigt
- ▶ „Intelligente" Auswertung der Benutzereingaben u. a. spezielle Berücksichtigung unregelmäßiger Verben (bei Eingabe von 'to go' erfolgt keine Fehlermeldung, sondern es werden die anderen Formen nachgefragt)
- ▶ Bei offensichtlicher Ähnlichkeit der Wörter wird wahlweise ein zweiter Versuch zugelassen
- ▶ Vielfältige Möglichkeiten des Lernens und der Abfrage (Deutsch-Fremdsprache, Fremdsprache-Deutsch, Multiple-Choice, lernen durch optische Rückkopplung)
- ▶ Jederzeit Bewertung möglich, die den Lernerfolg anzeigt und mit einem Kommentar motiviert
- ▶ integriertes Lernspiel HANGMAN
- ▶ Wörterbuchfunktion, sucht Übersetzung für eine Vokabel
- ▶ Ausgabe aller Vokabeln bzw. aller falschen oder richtigen Vokabeln auf Bildschirm oder Drucker
- ▶ Universelle Druckeranpassung durch 1st-Word-ähnliches, externes Druckerkonfigurationsprogramm an **alle** Drucker (auch serielle)
- ▶ Trotz Einordnung der Vokabeln nach Lektionen oder Wissensgebieten ständig schneller Zugriff auf alle Vokabeln (nur durch Größe des Speichermediums begrenzt)

Im Lieferumfang befinden sich über 1600 englische Grundwortschatzvokabeln in zwei Schwierigkeitsstufen, sowie mehrere Dateien mit wichtigen Vokablen, etwa Wendungen und Struktur- oder Ordnungswörter.



NEU ab Version 3.5 (1.3.89)

- Alle GEM-Funktionen wurden nochmals überarbeitet und optimiert. Das Programm ist noch schneller geworden.
- ▶ Die Auswertung der Vokablen wurde weiter verbessert, sie ist noch differenzierter geworden.
- ▶ für die schnelle Eingabe umfangreicher Vokabeldateien oder Übernahme aus anderen Programmen wird ein Hilfsprogramm mitgeliefert, daß normale Textdateien (ASCII) verarbeitet.
- ▶ Sortieren der Vokabeln nach mehreren Parametern
- ▶ konsequente Realisierung verschiedener Lerntechniken:
  u. a. – Karteikarten-Konzept
  – Lernen in fester Reihenfolge
  – zufällige Stichproben
  – Abfragen, bis alle Vokabeln gekonnt werden

DM 69,–

* alle Preise sind unverbindlich empfohlene Verkaufspreise

BESTELL-COUPON an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Ich bestelle: _____St. AMIGA-Learn á 69,– DM
zzgl. DM 5,- Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname _____

Straße, Hausnr.. _____

PLZ, Ort _____

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte.

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

SCHWEIZ
DataTrade AG
Langstr. 94
CH-8021 Zürich

Was IBM kann, können wir schon lange. So oder ähnlich dachte wohl ein AMIGA-Entwickler und konstruierte den TOWER-AMIGA. Daß es sich hier nicht um einen normalen AMIGA handelt, ist anzunehmen. Suggeriert doch ein TOWER-Gehäuse dem Betrachter, daß es sich um einen leistungsfähigen Rechner handelt, der seine Kontrahenten weit hinter sich läßt. Ob es sich bei dem TOWER-AMIGA ähnlich verhält, oder ob der Schein trügt, verrät der Bericht.



VON ANDREAS KRÄMER

Der TOWER-AMIGA

In der Computerbranche beschreibt das englische Wort TOWER (zu deutsch: Turm) ein senkrechtes Computergehäuse. Der Computer wird nicht wie sonst üblich auf den Schreibtisch gestellt, sondern neben ihn oder darunter. Das macht die Bauart des Computerturms möglich. Der erste Vorteil des TOWER-AMIGA ist also klar. Auf dem Schreibtisch müssen lediglich der Monitor und die Tastatur plaziert werden. Der TOWER-AMIGA

gehört auf den Boden. Das hilft, Platz zu sparen. Doch kommen wir zu den Leistungsdaten des ungewöhnlichen AMIGA. Wird der Begriff TOWER doch mit mehr Leistung gleichgesetzt.
Den TOWER-AMIGA erhält man in verschiedenen Ausführungen. Der kleinste AMIGA-Turm wird mit einem 68000-Prozessor, der mit 14.32 MHz Taktfrequenz betrieben wird (siehe Prozessorwettstreit), und dem mathematischen Coprozessor 68881, zwei $3^1/_2$" Laufwerken, drei MByte RAM, erweiterbar auf 9.5 MByte, KICKSTART-Version 1.3 mit integriertem Virusfinder und einer autobootenden 20 MByte Festplatte ausgeliefert. In dieser Version schlägt der TOWER-AMIGA mit 7950.- DM zu Buche. Der nächstgrößere TOWER ist mit einem 68020-Prozessor, der FPU 68881 und einer 30 MByte Festplatte versehen. Ansonsten sind die Daten mit dem Standard-TOWER identisch. Die größte TOWER-AMIGA-Version besitzt eine 80 MByte Festplatte, einen 68020-Prozessor und den schnelleren Coprozessor 68882. Der Preis liegt hier bei 9390.- DM. Nach Absprache mit der Firma CSS kann der TOWER-AMIGA auch in anderen Konfigurationen erworben werden, denkbar sind ein größerer RAM-Speicher oder eine größere Festplatte, eine andere 68020-Karte, z.B. die A2620-Karte von Commodore, eine AT- bzw. XT-Karte usw. Natürlich steht dem Käufer offen, den TOWER-AMIGA nach und nach selber aufzurüsten.
Zum Test stand uns ein TOWER-AMIGA zur Verfügung, der bereits einige Besonderheiten beinhaltete: eine 68020-Karte, die bis auf 1 MByte 'on board' aufgerüstet werden kann und natürlich über einen 32 Bit breiten Bus verfügt. Des weiteren war die AT-Karte eingebaut, eine 20 MByte-Festplatte verrichtete ihren Dienst, die mit dem Fast-File-System formatiert war. Der komplette Preis eines solchen AMIGA-TOWERs wächst natürlich in die Höhe, muß doch der Preis der AT-Karte mit einbezogen werden.

Der Aufbau

Grundeinheit des TOWER-AMIGA bleibt die AMIGA 2000B-Mutterplatine. Im TOWER-Gehäuse wird die Platine hochkant montiert. Die Festplatte wird an deren oberem Teil befestigt, darunter findet sich das $5^1/_4$"-Laufwerk der eingebauten AT-Karte, worunter sich die zwei $3^1/_2$"-Laufwerke befinden. Hier können noch weitere Laufwerke montiert werden, ob ein $5^1/_4$" oder $3^1/_2$", bleibt dem Anwender überlassen. Unter den Laufwerken findet sich die Bedienungseinheit des TOWER-AMIGA: der Ein-/Ausschalter, ein Kickstartumschalter, der es ermöglicht, zwischen zwei Betriebssystemversionen zu wählen, ein zusätzlicher Reset-Schalter und verschiedene Leuchtdioden, die den Betrieb der Laufwerke und der Festplatten anzeigen. Zusätzlich gibt es dort ein Computerschloß, das von AT-Rechnern her bekannt ist. Ist der Rechner 'abgeschlossen', kann kein Unbefugter den AMIGA in Betrieb nehmen.
Die Anschlüsse für die Tastatur, die Maus und den Joystick, die sich beim original AMIGA auf der Frontseite befinden, sind auf die Rückseite des TOWER-Gehäuses verlegt worden. Dadurch ergibt sich zwar ein aufgeräumtes Bild, bei ungünstiger Plazierung des TOWERs kann es jedoch zu Problemem kommen, da die Maus-, Joystick-, Monitor- oder Tastaturkabel zu kurz sind. Eine günstige Plazierung ist also anzuraten. Auf der Rückseite des Gehäuses sind des weiteren die parallele und serielle Schnittstelle, der RGB-Ausgang und die Laufwerksbuchse zu finden. Sie wurden ebenfalls verlegt und sind leicht zu erreichen.

Das Innere des TOWER-AMIGA wirkt nicht ganz so aufgeräumt wie ein normaler A2000, das liegt aber an den umfangreichen Umbauarbeiten und dem Anschluß der zusätzlichen Hardware. AMIGA 2000-Karten können ohne Probleme in die verschiedenen Slots der A2000B-Platine gesteckt werden, da der Slotteil frei von irgendwelcher Verdrahtung ist. Auffällig ist das Kickstart-ROM, das in Form einer Zusatzplatine eingesteckt wird. Wie schon erwähnt, handelt es sich hierbei um die Kickstart-Version 1.3 mit integriertem Virusfinder. Der Virusfinder ist ein alter Bekannter, der Guardian kommt zum Einsatz. Der Virusfinder untersucht die Bootspur der Bootdiskette und gibt bei einem 'Non-Standard-Bootblock' eine Fehlermeldung und das Bootspur-Hexdump-Listing aus. Jedoch kann die Meldung übergangen und wie gewohnt fortgefahren werden. Neben diesem nützlichen Virusschutz zeigt sich ein weiterer Vorteil des TOWER-AMIGA. Direkt nach dem Einschalten bootet der Rechner von Festplatte. Voraussetzung ist jedoch, daß sich keine Diskette im Boot-Laufwerk befindet, ansonsten wird von dieser gebootet. Unser Test-TOWER war mit einer 20 MByte-Platte ausgerüstet, die mit dem neuen Fast-File-System formatiert werden kann. Die Geschwindigkeit der Platte steigt dementsprechend.
Als Festplatten-Controller diente ein OMTI, der mittlerweile von vielen Festplattenproduzenten verwendet wird.
Apropos Geschwindigkeit: Der TOWER-AMIGA war mit einer 68020-Karte mit Floating Point Unit 68882 und 256 KByte 32 Bit-RAM ausgestattet. Der zusätzliche Speicher befindet sich auf der Karte und kann auf 1 MByte aufgerüstet werden. Mit diesem RAM wird der Rechner natürlich ausgesprochen schnell. Allerdings ist das RAM nur als Fast-MEM ansprechbar und nicht als Chip-MEM, auf das die Customchips zugreifen können. Das zusätzliche RAM ist leider nicht autokonfigurierend, so daß es mit den Programmen *'ADDMEM'* oder *'INSTALL'*, die sich auf der mit ausgelieferten Diskette befindet, eingebunden werden muß. Auf dieser Diskette sind außerdem einige Demoprogramme untergebracht, die hauptsächlich die erhöhte Geschwindigkeit des TOWER-AMIGA demonstrieren. Noch ein Wort zum 68882: Der größere Bruder ist kompatibel zum 68881, ist aber in der Rechengeschwindigkeit dem 68881 noch einmal überlegen. Die FPU wurde mit einer Taktgeschwindigkeit von 30 MHz getaktet, Programme wie TURBO SILVER 3.0 oder ANIMATE 4D, die die FPU unterstützen, erleben in ihrer Rechengeschwindigkeit einen wahren Geschwindigkeitsboom.
Unser Test-TOWER-AMIGA war mit einer AT-Karte ausgerüstet, Probleme mit dem 68020-Prozessor traten ebensowenig auf wie mit der KICKSTART 1.3 oder der Autoboot-Festplatte.

Verarbeitung?

Da beim TOWER-AMIGA doch recht umfangreiche Umbauarbeiten stattfinden, stellt sich die Frage nach der Verarbeitung fast von selbst. Sie ist fachmännisch durchgeführt. Die vielen Kabel

sind sauber verlegt, die Befestigungen der Laufwerke, des Netzteils, der Festplatte und deren Controller sind sauber und wirken aufgeräumt. Möchte man eine weitere Festplatte oder ein Laufwerk anschließen, muß eine Abdeckblende auf der Frontseite herausgedrückt und das zusätzliche Laufwerk eingesetzt werden. Zur Befestigung stehen bereits Halterungen zur Verfügung.

Fazit

Der TOWER-AMIGA entpuppt sich als aufgemotzter AMIGA 2000. Die grundlegende Änderung zu einem herkömmlichen besteht im Gehäuse. Natürlich kann auch der normale AMIGA 2000 in der Form des TOWER-AMIGA aufgerüstet werden, aber die Eleganz eines TOWERS besitzt dieser nicht. Auch wirkt er nicht so professionell.
Die Frage nach der Käuferschicht des TOWER-AMIGA läßt sich nicht so einfach beantworten. Hauptkäuferschicht wird mit Sicherheit der betuchtere Computeranwender sein, der seinen Rechner gleich mit mehr Leistung erwerben möchte. Einige 1000 Mark müssen auch schon für die Grundversion hingeblättert werden. Durch Absprache mit der Herstellerfirma kann der TOWER-AMIGA den jeweiligen Wünschen angepaßt werden. Der Preis berechnet sich dann natürlich individuell. Das Zusammenspiel der einzelnen Hardwarekomponenten funktioniert ohne Probleme, lediglich die bekannten Probleme mit mancher Software, die nicht mit dem 68020-Prozessor harmoniert, ist zu beklagen. Für die Zukunft hat sich die Firma CSS noch einiges ausgedacht, beispielsweise sollen in die AMIGA 2000-Tastatur verschiedene Leuchtdioden eingesetzt werden, die den Betrieb des Rechners, der Laufwerke und der Festplatte signalisieren. Der Anwender hat dann alle Funktionen im Blickfeld. Auch für die AT- bzw. XT-Karte ist einiges geplant. Ein optisches Speichermedium ist in Planung. Dessen Zugriffszeiten sind in der Regel denen von Festplatten unterlegen, dafür besitzen sie aber eine gigantische Kapazität. Das in Planung befindliche Gerät stammt von der Firma RICOH und besitzt eine SCSI-Schnittstelle. Der optische Massenspeicher soll auch auf die AMIGA-Seite angepaßt werden.
In naher Zukunft soll der TOWER-AMIGA auch mit einem 32-Bit breiten Kickstart-Bus ausgerüstet werden. Dadurch erlebt der Rechner eine deutliche Geschwindigkeitssteigerung.

TOWER-AMIGA

+ solide verarbeitet
+ autobootende Festplatte
+ keine Probleme im Zusammenspiel der einzelnen Hardwarekomponenten

- Kabel für Maus, Joystick, Tastatur und Monitor unter Umständen zu kurz

Anbieter:
CSS
Auf der Warte 46
6367 Karben 1
Tel. 06039-5776

Preise:
TOWER-AMIGA Grundversion:
2x3 1/2" Laufwerke, 3 MByte RAM, Kickstart 1.3 mit Virusfinder, autobootende 20 MByte-Festplatte, 68000-Prozessor mit 14 Mhz und 68881-FPU
7950.- DM

TOWER-AMIGA:
wie oben, jedoch 68020-Prozessor u. 68881-FPU und 30 MByte-Festplatte
8850.-DM

TOWER-AMIGA I:
mit 40 MByte-Festplatte
8950.- DM

TOWER-AMIGA II:
68882-FPU und 60 MByte-Festplatte
9390.- DM

Andere Konfigurationen auf Anfrage

ENDE



Grafik-Connection

für
Commodore AMIGA · ATARI ST · IBM-PC · Gepard
und andere Computer



WICHTIGE MERKMALE:
- ein Grafiksystem für den C-Programmierer
- professionelle Standardisierung
- Implementierung auf beliebigem Rechner
- vollständige Portabilität aller Programme
- Testprogramme, Druckeranpassung
- höchste Genauigkeit und Verständlichkeit
- mathematische Grundlagen
- ausführliche Erläuterung der Algorithmen

AUS DEM INHALT:
- Rosetten und Spiralen
- rekursive Tänze und Kurven
- fraktale Spielereien in der komplexen Ebene
- Einführung in die Vektorrechnung
- objektorientierter Ray-Tracer
- Hyperwürfel und Artverwandte
- Landschaften aus dem Rechner
- Programmdiskette im Buch enthalten

Hardcover
Bestell-Nr. B-
ISBN 3-923250-80-0

59,–
incl. Programmdisk.

„Grafik-Connection" wurde von 10 Finalisten des Bundeswettbewerbs Informatik geschrieben – alles Experten auf ihrem Grafikgebiet und in der C-Programmierung. Es ist für C-Programmierer gedacht, die in die Programmierung von mathematisch orientierter Computergrafik einsteigen möchten.

Das Grafiksystem „Gratia" genügt professionellen Ansprüchen in der Grafik-Programmierung. Angefangen bei grundlegenden Grafikelementen für beliebige Ausgabegeräte (Bildschirm, Drucker etc.) werden Module für punktorientierte Grafik, Kreisbögen und Ellipsen, anwendungsorientierte Grafik mit beliebigen Maßeinheiten, verzerrungsfreie Grafik, Turtle-Grafik und Vektortext vorgestellt.

Das gesamte System „Gratia" ist ohne Änderung auf jeden Computer mit vollständigem Kernighan/Ritchie-C-Compiler übertragbar. Voraussetzung ist die gerätespezifische Implementierung einer minimalen Systemgrafik; sie wird für den Atari ST und einige andere Rechner im Buch und auf Diskette mitgeliefert. Alle Buch-Programm laufen unverändert auf allen Rechnern, für die eine Gratia-Portierung existiert. Gratia ist Public-Domain und kann frei verwendet und weiterkopiert werden.

Auf der Grundlage von „Gratia" werden anwendungsorientierte Grafikprogramme vorgestellt. Jeder Autor hat sich auf sein Thema konzentriert; die Kapitel sind fachlich fundiert, anschaulich geschrieben und mit Grafik-Beispielen reichlich illustriert worden. Für die Gestaltung der Programme ist ein einheitlicher Standard entwickelt worden.

Besonderer Wert wurde auf Anschaulichkeit und Genauigkeit in der Sprache, der Mathematik und der Beschreibung der Programme gelegt. Alle verwendeten Algorithmen werden erklärt – ob es um das Ziehen von Linien oder das Verfolgen von Lichtstrahlen entlang von Brechungen und Spiegelungen geht. Für das Rechnen mit Vektoren und komplexen Zahlen werden die mathematischen Grundlagen dargelegt. Das zeitaufwendige Durcharbeiten von Sekundärliteratur entfällt deshalb.

BESTELL-COUPON an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

* alle Preise sind unverbindlich empfohlene Verkaufspreise

Ich bestelle: _____St. Grafik-Connection (incl. Programmdiskette) á 59,– DM
bitte ankreuzen: ☐ 3 ½'' Diskette oder ☐ 5 ¼'' Diskette
zzgl. DM 5,- Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname _____

Straße, Hausnr.. _____

PLZ, Ort _____

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte.

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

SCHWEIZ
DataTrade AG
Langstr. 94
CH-8021 Zürich

FROHE BOTSCHAFT VON AMIGAOBERLAND:
DIESE PREISE HAT UNSER HERR KOPPISCH ABGESEGNET!

ANIMATION: NEU

Multiplane					
Effects					
Apprentice	198.00	Stand			
Data: Drachen	98.00	Flipper			
Data: Christmas	298.00		98.00	Rotoscope	
Data: Dinosaurier	45.00		98.00	Editor	
	45.00				169.00
	45.00				128.00

Unsere Hits des Monats:

CMI Processor Accelerator				**Photon Cel Animator** ✎ ✺	**249.00**
Digi View Gold PAL ✎ ✺	**498.00**	**Dragons Lair** ✎ ✺	**115.00**	**Photon Paint PAL 2.0** ✎ ✺	**169.00**
	349.00	**Elite** ✎ ✺	**75.00**	**SoundOasis** ✎ ✺	**169.00**

Unser absoluter SUPER-HIT: **Evil Garden** ✎ ✺ **59.00** **Falcon** ✎ ✺ ✠ **85.00** **Lattice C V5.0** ✎ ✺ **530.00**

Unsere Topangebote:

ACTIONWARE

Capone ✎	
P.O.W. ✎ ✺	
Lichtpistole ✎ ✺	69.00
	69.00
	115.00

ANIMATION

3-Demon ✎✎	
Animation TV-Show ✎	169.00
Caligari ✎ ✺	129.00
Deluxe Productions	3995.00
Deluxe Video II ✎ ✺	325.00
Fantavision ✎ ✺ ✠	225.00
Lights, Camera, Action ✎✎ ✺	89.00
Sculpt Animate 4D ✎ ✺	128.00
Silver V.1.1 ✎✎ ✠	995.00
Turbo Silver V.3.0 ✺ ✠	149.00
TV-Show ✎ ✺	**385.00**
TV-Text 3D ✎	128.00
Video Effects 3D ✎ ✺	165.00
Video Effects 3D ✎✎ ✺ ✠	345.00
VideoScape 3D PAL 2.0 ✎✎ ✠ ✺	448.00
VideoScape 3D PAL 2.0 ✎✎ ✺	285.00
Videotitler V.1.1 ✎	279.00
Movie Setter ✠	179.00
Comic Setter ✎ ✺	198.00
	189.00

DATENBANK

Superbase II ✠ ✎ ✺	
Superbase Professional	195.00
	448.00

DIVERSES

Aegis Diga	
C-64 Emulator II ✺	115.00
	109.00

FESTPLATTEN

20 MB A500	
20 MB Filecard A2000	1028.00
	898.00

GRAFIK

Aegis Draw 2000 ✎ ✺	
Calligrapher ✎	449.00
Calligrapher ✎ ✠ ✺	175.00
Calligrapher II PAL+	189.00
Deluxe Print ✎✎ ✺ ✠	
Deluxe Paint III ✎ ✺	185.00
Deluxe Print II (4/89) ✎ ✺ ✠	a.A.
Printmaster Plus ✎ ✺	185.00
Deluxe Photo Lab ✎✎ ✺ ✠	70.00
Deluxe Photo Lab ✎✎	195.00
Digi Paint PAL ✎ ✠	179.00
Forms in Flight II ✎✎ ✺	90.00
Intro CAD ✎	198.00
Modeler 3D ✎✎ ✺	109.00
Photon Paint PAL ✎	148.00
Photon Paint PAL V2.0 ✎ ✺	159.00
PixMate ✎ ✺ ✠	169.00
Proffesional Draw ✎ ✺	98.00
X-CAD	a.A.

HURRICANE:

A-2000 Board	839.00
incl. Proz. (68020+68881)	2295.00
ohne Proz.	1395.00

2 MByte 32 Bit 100Ns (best.)	2795.00
0 MByte	1395.00
68030 Board incl. Proz.	2195.00
ohne Proz.	1749.00
A-1000 Board	1998.00
ohne Proz.	1395.00
CMI Prozessor Accelerator	498.00

KALKULATION

Logistix ✠	
Math-A-Mation ✎ ✺	339.00
Maxiplan Plus ✎	159.00
Maxiplan 500 ✠	295.00
Maxiplan Plus ✎ ✠	329.00
Maxiplan Plus Update ✎ ✠	678.00

MUSIK

Aegis Audiomaster II ✎ ✺	389.00
CMI MIDI Interface ✎ ✺	168.00
Deluxe Musik Constr. Set ✠ ✎✎	169.00
ECE MIDI Interface ✎ ✺	188.00
Future Sound II	128.00
Perfect Sound ✎	339.00
Pro Midi Studio V.1.4	165.00
Sonix ✎	268.00
Soundscape Sampler Mimetics	115.00
Sound Oasis ✎✎	198.00

SIMULATION

Flugsimulator II ✎✎ ✠	169.00
Flugsimulator III ✎✎	
F-16 Falcon ✎ ✺ ✠	79.00
Galileo Planetarium V.2.0 ✎ ✺	a.A.
Jet ✎	85.00
Original Jet Anleitung ✠	98.00
Scenery Disk #7	79.00
Scenery Disk #11	16.00
Scenery Japan	42.00
Scenery Europa	42.00
Surgeon Operations Sim.	42.00
Universal Military Simulator	69.00
UMS Data Disk Vietnam ✎ ✺	85.00
UMS Data Disk Civil War ✎ ✺	45.00
	45.00

SPEICHER

8 MByte Unbestückt ✺	
512 KByte A-500 ✎ ✺	579.00
8 MByte, 2 MByte Bestckt	359.00
	1498.00

SPIELE

Andromeda Mission ✎✎ ✺	
Balance of Power ✎	45.00
Barbarian (Psygnosis) ✎	65.00
Bards Tale ✎✎ ✠	55.00
Bards Tale II ✎✎ ✠	69.00
Battle Chess ✺ ✎ ✠	65.00
Blitzkrieg at the Ardennes ✎	69.00
California Games ✎✎ ✺	95.00
Carrier Command ✎✎ ✠	65.00
Chessmaster 2000 ✠ ✎	69.00
Corruption (Rainbird) ✎ ✺	65.00
Crono Quest (Psygnosis) ✎✎ ✺	79.00
Defender of the Crown ✎	79.00
Dragons Lair ✎✎	65.00
	115.00

Dungeon Master (1MB) ✎ ✺	79.00
Elite ✎ ✺ ✠	75.00
Empire ✎✎	81.00
Empire Strikes Back ✎ ✺	59.00
Evil Garden ✎ ✺	59.00
Faery Tale Adventure	67.00
Ferrari Formula One ✎ ✠	72.00
Gauntlet II ✎ ✺	58.00
Gettysburgh ✎ ✺	89.00
Hollywood Poker Pro ✎ ✺	55.00
Hostages ✎ ✺	75.00
Impossible Mission II ✎	69.00
Interseptor ✎✎ ✠	65.00
Jeanne d'Arc ✎ ✺	55.00
Kampfgruppe	69.00
Katakis ✎✎ ✺	50.00
Kings Quest I+II+III ✎	95.00
Leaderboard Golf+Tournam. ✎ ✺	72.00
Leasure Suite Larry ✎ ✺	65.00
LED Storm ✎ ✺	69.00
Legend of the Sword ✺	69.00
Marble Madness ✎ ✠	60.00
Menace (Psygnosis) ✎✎ ✺	54.00
Obliterator ✎	55.00
Off Shore Warrior ✎✎	69.00
Out Run ✎ ✺ ✠	59.00
PacMania ✎ ✺	62.00
Pioneer Plaque ✎ ✺	65.00
Ports of Call ✎ ✺ ✠	79.00
Powerdrome ✎ ✺	79.00
Return to Atlantis ✠	69.00
Rocket Ranger ✎ ✺	65.00
Roger Rabbit ✎ ✺	85.00
Sargon III ✎	85.00
Sentinel ✎	55.00
Sex Vixens from Space ✎ ✺	75.00
Shadowgate ✎	67.00
Shanghai	60.00
Sherlock	69.00
Sorcerer Lord ✎ ✺	89.00
Speedball ✎ ✺	79.00
Summer Olympiad ✎ ✺	55.00
Starglider II ✎✎ ✠	73.00
Starray Defender ✎ ✺	69.00
Starwars ✎✎	45.00
Strike Force Harrier ✎ ✺	60.00
Sub Battle Simulator ✎	79.00
Sword of Sodan ✎ ✺	73.00
Telewars II ✎ ✺	78.00
TV Football (Cinemaware) ✎ ✺	75.00
Terrorpods	55.00
Ultima III	79.00
Ultima IV ✺	75.00
Uninvited	67.00
Winter Challenge ✎✎	49.00
Winter Games	59.00
Winter Olympics 88	49.00
World Games	59.00
Zak McKracken ✎ ✺	79.00
Zoom	39.00
Zork Triology	115.00
4x4 Off Road Racing ✎ ✺	89.00

SPRACHEN

AC Basic Compiler ✎✎	265.00
AC Fortran ✎ ✺	498.00
Aztec Source Level Debugger ✎	129.00
Aztec C Developer V. 3.6 ✎	439.00
Aztec C Personal V. 3.6 ✎	319.00
Benchmark Modula II ✎	339.00
CygnusEd Professional ✎ ✺	189.00
GFA Basic 3.0 ✎ ✺	198.00
Lattice C V.5.0 ✎ ✺	530.00
Magellan-Ki (künstl.Intell.) ✎ ✺	398.00
Metacomco Pascal V.2.0 ✎ ✺	249.00
Update Aztec C V3.4 auf V3.6	59.00

TEXT

Excellence	
Go-Amiga Text ✎✎ ✺	389.00
Kind Words ✎ ✺	199.00
Professional Page ✎ ✠ ✺	159.00
Shakespeare ✺ ✠	589.00
Vizawrite 2.0 ✎ ✺ ✠	339.00
Word Perfect ✎ ✺ ✠	225.00
Zuma Fonts 1,2,3,4, je	658.00
Fancy Fonts (Scul. Ani. 3D) ✎ ✺	57.00
	145.00

TOOLS

CLI Mate	
Discovery Disk Editor ✎✎	60.00
Dos to Dos	185.00
Floppy Accelerator II ✎	90.00
Funktion ✎ ✺	59.00
Grabbit	79.00
Marauder II (Brain 10)	50.00
Project D	55.00
Quarterback HD Backup ✠ ✎	75.00
SYS Virus Checker ✎ ✺	125.00
Turbo Print II	49.00
TxED Plus	95.00
Zing V.1.2	129.00
Zing Keys ✠	97.00
	79.00

VIDEO

Diamond Digitiser +	
Zeichen Progr. ✠ ✎ ✺	298.00
Diamond m. RGB Spl. (3.89) ✺	598.00
Digi View Gold PAL ✎ ✺	**349.00**
Digi Droid ✎	129.00
Flicker Fixer PAL ✎ ✺	1098.00
Pro Video Plus PAL ✺	598.00

ZUBEHÖR

AT Erweiterung A-2000	2498.00
Disketten 3 1/2 Zoll 2DD ✎	2.25
Promigos 3 1/2 Zoll Extern	259.00
Trackball ✎ ✺	89.00
TV Modulator	59.00

UND:

Das Große Amiga Spielebuch	45.00
Das Grosse Amiga 2000 Buch	49.00
Erdkunde	49.00
Englisch	49.00

Vergleicht die Preise, Freunde, und freut Euch mit uns!

Wir setzen Zeichen!

✎ = im Preis gesenkt ✠ = in deutsch

✎✎ = SUPERBILLIG ✺ = völlig neu

Preisänderungen und Irrtümer vorbehalten.

AMIGAOBERLAND liefert

- **innerhalb von 10 Tagen (garantiert!) ((meistens))**
- **bei einem Mindestbestellwert von DM 50,–**
- **plus DM 6,– Versandkosten (sorry!)**
- **gegen Vorkasse oder per Nachnahme**
- **ins Ausland bitte nur Vorkasse (Scheck o.ä.)**

Unsere

Bestellservice-Hotline:

0 61 71 / 7 18 46 (day & night)

Fax: 0 61 71 / 7 48 05

Unsere Hochburg:

AMIGAOBERLAND
A. Koppisch
Hohenwaldstr. 26
D6374 Steinbach

AMIGAOBERLAND.
Soft- und Hardware vom Feinsten. Preise vom Kleinsten.



Tips & Tricks

ZEILE LÖSCHEN OHNE MÄUSE

Wer kennt es nicht, das Löschen einer Basic-Zeile oder deren rechten Rest. Früher mußte man mit der Maus in gedrückter Haltung bei langen Zeilen mehrere Bildschirmseiten nach rechts scrollen, bis man am Ende der Zeile dem Rest den Rest geben konnte. Heute ist das aber anders. Man positioniert einfach den Cursor an der Stelle, von der aus alles, was rechts von ihr steht, gelöscht werden soll, und drückt dann die Tastenkombination CTRL & SPACE. Nun braucht man nur die Zeile zu verlassen, und -schwupp- ist alles, was rechts stand, verschwunden.

PC UND DRUCKER

Wer es leid ist, bei der PC-Karte auf die langsame Druckerport-Emulation zu warten und nur Listings ausdrucken will, kann dies auch mit dem Befehl tun:

```
awrite pc_dateiname prt:
```

Das ist nicht nur schneller, man spart sich außerdem das Laden der Emulation und somit Arbeit und Speicherplatz.

(Stefan Helbig)

SYSBASE BEI S2F

In der Ausgabe 12/88 fragt sich der Autor von S2F in Zeile 27, warum SYSBASE nicht funktioniert. Die Lösung ist die Deklaration von "struct ExecBase *SysBase" *im C-Programm am Anfang und der Import mit* "XREF _Sysbase" *im Assemblerteil.*

Als Verbesserungsvorschlag für die C-Programme bietet sich bei Verwendung des Aztec C folgendes an:

```
long _stack    = 0;
long _priority = 0;
long _BackGroundIO = 0;
char *_procname = "Programmname";
```

Nach Definition dieser Variablen und Linken mit Detach.o (Warnung des Linkers kann ignoriert werden) wird das Programm als eigener Task gestartet. So kann zum Beispiel bei KICKSHELL V1.1 der run-Befehl beim Aufruf gespart werden. Weitere Informationen hierzu findet man im AZTEC C 3.6-Handbuch, Ausgabe Jan. 88 auf Seite 11ff.
Dieses Verfahren kann mit allen Programmen des Aztec C 3.6 verwendet werden, um einen eigenen Programmtask zu erzeugen. Der CLI-Task, aus dem das Programm gestartet wird, kann, falls das Programm ein Window oder einen Screen öffnet, geschlossen werden.

(Norbert Bazin)

HILFE FÜR SPIELER

SWORD OF SODAN

Wer bei diesem Spiel nicht mehr mit den 5 Leben auskommt, sollte den Cheatmodus benutzen. Um dorthin zu gelangen, muß man allerdings erst einmal alle Leben verwirken, denn erst dann kommt man zur HALL OF FAME. Dort tippt man NANCY ein und befindet sich im folgenden Spiel im Cheatmodus.

KATAKIS

Wenn man vor dem Laden der Datadisk aufgefordert wird, eine beliebige Taste zu drücken, wählt man "Y". Danach wird die Anzahl der Spieler eingegeben, die Maus in Port 2 gesteckt und die rechte Maustaste so lange gedrückt, bis das Spiel lädt.

(Daniel Laffien)

STARTUP mit BASIC

Für die normale Workbenchdiskette ist das Verfahren sicherlich nicht geeignet, aber für eine Demodiskette, bei der es auf Ladezeiten nicht so ankommt. So ist es auch für diejenigen, die "nur" in BASIC programmieren können und keinen Compiler oder Assembler besitzen möglich, eigene Programme mit einem Autostart zu versehen. Um z.B. ein Programm namens *"Demo"* ausführen zu lassen, fügt man in der Startupsequence einfach den Befehl *"amigabasic Demo"* ein (falls der Programmname Leerzeichen enthält, muß man ihn in Anführungszeichen setzen). So wird bei jedem Booten das Basicprogramm ausgeführt. Setzt man ans Ende des Programms den Basic-Befehl *SYSTEM*, wird nach dem Ende des Basicprogramms die Startup-sequence weiter ausgeführt.

(Claus Brunzema)

SWAP IN C

Das folgende C-Programm (in Assembler geht es genauso) tauscht zwei Integervariablen (oder andere), ohne eine dritte zu benötigen. Möglich wird dies durch das EXOR (Exklusiv-Oder).

(Klaus Weber, Detting)

```
enint a=4,b=7;                  /* Integerbeispiel */
int i,j;                        /* Schleifenzähler */
char wort[2][80]   { "An alle C - Freaks !\0",
                     "Tauschen, schnell und ohne 3. Variable !\0" };

main()
{
  printf("vorher : \n");
  printf("wort[0] = %s\n",wort[0]);
  printf("wort[1] = %s\n",wort[1]);
  a = a^b;          /* EXOR 1 */
  b = a^b;          /* EXOR 2 */
  a = a^b;          /* EXOR 3 */
  for(i=0;i<79;i++)
  {
     for(j=0;j<3;j++) wort[j%2][i] = wort[0][i]^wort[i][i];
  }
  printf("nachher : \n");
  printf("a = %d b = %d\n",a,b);      /* Ergebnis anzeigen */
  printf("wort[0]  %s\n",wort[0]);
  printf("wort[1]  %s\n",wort[1]);
}
)
```

REFRESHZEIT DES PC

Ein Prgrämmchen für die Sidecar- und PC-Kartenbesitzer, die für ein bißchen mehr Geschwindigkeit ihres PCs sicher dankbar sind.

Dies GWBasic-Programm erzeugt ein 19 Byte langes Maschinenprogramm auf Diskette, welches den REFRESH-Timer des PCs beeinflußt. Trägt man in Zeile 110 des Programms eine große Zahl -z.B. 100- ein , wird der REFRESH-Timer herabgesetzt und umgekehrt.

Background: Die Speicherbausteine verlieren ihren Inhalt nach kurzer Zeit, wenn sie nicht wiederaufgefrischt werden. Dies besorgt ein DMA-Baustein im PC. Alle soundsoviele Millisekunden werden die RAMs wiederaufgefrischt. Während dieses Vorgangs kann die CPU nicht auf die RAMs zugreifen und muß warten. Dieser Wert ist für die heutigen, qualitativ hochwertigen RAMs recht großzügig gewählt. Beim Sidecar gehen so etwa 8% der Rechenleistung verloren. Bei einem eingesetzten Wert von 100 wird dies auf 1.3% gesenkt. Ein Wert von 255 würde auf 0% Verlust, also beste Performance, senken. Dies ist aber nicht anzuraten, da dann Rechnerabstürze nicht auszuschließen sind. Ich selbst lasse mein Sidecar seit mehreren Wochen mit einer auf 1.3% gesenkten Rate arbeiten (also ca 6% sind nicht die Welt, aber für 0 Kosten?!

(Michael Holin)

```
 10 REM Erzeugtes Programm ändert den REFRESH-Timer des PCs.
 20 REM In Zeile 110 Werte zwischen 3 und 150 eintragen.
 30 REM Standardwert ist 18. Kleine Wert bremsen den
 31 REM Rechner, große beschleunigen ihn.
 32 REM
 33 REM Dies Programm ist für Sidecar/PC-Karte!
 34 REM Es läuft =nicht= auf dem AMIGA.
 35 REM
 40 OPEN "refresh.com" FOR OUTPUT AS #1
 50 FOR I=1 TO 19 60 READ A:PRINT #1,CHR$(A);
 70 NEXT
 80 CLOSE #1:END
100 DATA 250,176,116,230,67,184
110 DATA 100
120 DATA 0,230,65,136,224,230,65,251,180
130 DATA 76,205,33
```

Programm für GWBASIC auf PC-Seite

Eine kurze Zeichenroutine

Es gibt verschiedene Möglichkeiten, eine Zeichenroutine zu realisieren, doch besonders kurz und elegant ist der Weg über die Graphics-Library.

Das folgende Programm öffnet die Bibliothek und wartet erst einmal auf eine Reaktion des Benutzers. Drückt dieser die linke Maustaste, werden der Routine *WRITEPIXEL* die entsprechenden Mauskoordinaten übergeben und diese zeichnet den Punkt and diese Stelle - einfacher geht es kaum:

```
LIBRARY "graphics.library"
WHILE INKEY$=""
  WHILE (MOUSE(0)=0)
    CALL WritePixel (WINDOW(8),
                MOUSE(1), MOUSE(2))
  WEND
WEND
LIBRARY CLOSE
END
```

(Yann Cebron)

CP80X -

Und er druckt doch!

Der alte CP80X ist unter den Aufsteigern vom C64 weit verbreitet. Es wäre doch praktisch, wenn man ihn auch am AMIGA weiterbenutzen könnte. Das ist natürlich mit einer IEC-Schnittstelle, wie sie z.B. in der KICKSTART 7/88 beschrieben wird, möglich, aber es geht auch sehr viel einfacher.

Denn wer sich einmal genauer mit dem CP80X-Handbuch beschäftigt, sieht, daß der Drucker insgesamt vier (!!) eingebaute Schnittstellen hat, die man mit DIP-Schaltern im Innern des Druckers einstellen kann. Die am besten passende Schnittstelle für den AMIGA ist die CENTRONICS-Schnittstelle mit CBM-Zeichensatz. Man stellt sie wie folgt ein: Zuerst öffnet man das Druckergehäuse (siehe Handbuch S. 5-7). Danach sucht man den DIP-Schalterbaustein Nr.1, der sich unter einer schwarzen Plastikfolie dicht unter der Druckwalze befindet. Beim CP80X, der für den C64 vorbereitet ist, sind die Schalter Nr.3 und Nr.4 in OFF-Stellung (Zählrichtung und ON-Stellung sind auf den DIP-Schaltern markiert). Für den Betrieb mit dem AMIGA muß nun Schalter Nr.3 eingeschaltet werden. (Wer will, kann sich auch einen Kippschalter parallel zum DIP-Schalter legen und so von außen jeweils zwischen AMIGA- und C64-Betrieb umschalten.) Nach der Umstellung baut man den Drucker wieder zusammen und besorgt sich ein CENTRONICS-Kabel. Bei diesem Kabel muß UNBEDINGT die Leitung am Pin 14 unterbrochen werden!!!!!!!!!!! Der Drucker könnte durch die (eigentlich hier unnötige) Spannung an diesem Pin schwer beschädigt werden! Am besten bricht man hierzu den Pin im Stecker zum Parallel-Port des AMIGAs mit einer Spitzzange heraus. (Pin 14 beim A500 und beim A2000, beim A1000 Pin 23 entfernen!!). Mit einem so eingestellten Drucker (in Preferences den Epson-Treiber und "Custom" als Paper size einstellen) kann man nun schon Grafiken, z.B. mit DPaint oder PrintScreen von der ExtrasD-Diskette, ausdrucken, was mit der IEC-Schnittstelle nicht so einfach ist. Allerdings kann man leider (wegen einiger Unterschiede in den Zeichensätzen) Texte nicht ohne weiteres ausdrucken, sondern braucht dazu ein kleines Treiberprogramm, das die Texte nicht über PRT:, sondern über PAR: ausdruckt und evtl. eigene Codewandlungen vornimmt. Dieses ASCII-Printer genannte Programm unterstützt den gesamten deutschen Zeichensatz (also mit Umlauten) und versucht im Bereich von CHR$(32) bis CHR$(125) Zeichen auszugeben, die jeweils den im BASIC-Handbuch beschriebenen möglichst ähnlich sehen. Damit kann man also Texte und Listings (bei AmigaBASIC ein Programm mit ,a speichern) ausdrucken.

ASCII-Printer ist in AmigaBASIC geschrieben, um möglichst einfach Erweiterungen einbauen zu können; denkbar wäre z.B. die Emulation der restlichen ASCII- Zeichen mit Grafikdruck.

(Claus Brunzema)

```
REM ASCII-Printer fuer CP80X
REM Autor Claus Brunzema

INPUT "Filename ";f$
OPEN f$ FOR INPUT AS #1
OPEN "par:" FOR OUTPUT AS #2

PRINT#2,CHR$(27);CHR$(68);
FOR x=8 TO 80 STEP 8
  PRINT#2,CHR$(x);
NEXT
PRINT#2,CHR$(0);

PRINT#2,CHR$(27);CHR$(82);CHR$(2); :REM deutscher Zeichens.
PRINT#2,CHR$(17);                  :REM TEXT-Modus
PRINT#2,CHR$(27);CHR$(85);CHR$(1); :REM unidirektional

FOR count=1 TO LOF(1)
  t$=INPUT$(1,#1)
  PRINT t$;

  IF t$=CHR$(10) THEN t$=CHR$(13)   :GOTO Printout  :REM LF
  IF t$=CHR$(34)  THEN t$=t$+CHR$(2):GOTO Printout  :REM QUOTE-Mode aussch.
  IF t$=CHR$(196) THEN t$=CHR$(219):GOTO Printout  :REM AE
  IF t$=CHR$(220) THEN t$=CHR$(221):GOTO Printout  :REM UE
  IF t$=CHR$(214) THEN t$=CHR$(220):GOTO Printout  :REM OE
  IF t$=CHR$(228) THEN t$=CHR$(91) :GOTO Printout  :REM ae
  IF t$=CHR$(246) THEN t$=CHR$(92) :GOTO Printout  :REM oe
  IF t$=CHR$(252) THEN t$=CHR$(93) :GOTO Printout  :REM ue
  IF t$=CHR$(223) THEN t$=CHR$(94) :GOTO Printout  :REM sz
  IF t$=CHR$(95) THEN t$=CHR$(164) :GOTO Printout  :REM _ (Unterstrich)
  IF t$=CHR$(94) THENt$=CHR$(143)+CHR$(145)+CHR$(94)+CHR$(143)+CHR$(17)
                     :GOTO Printout :REM ^ (mit grafik-charset)
  IF t$=CHR$(93) THEN t$=CHR$(143)+CHR$(145)+CHR$(93)+CHR$(143)+CHR$(17)
                     :GOTO Printout :REM ] (mit grafik-charset)
  IF t$=CHR$(92) THEN t$=CHR$(143)+CHR$(145)+CHR$(92)+CHR$(143)+CHR$(17)
                     :GOTO Printout :REM \ (mit grafik-charset)
  IF t$=CHR$(91) THEN t$=CHR$(143)+CHR$(145)+CHR$(91)+CHR$(143)+CHR$(17)
                     :GOTO Printout :REM [ (mit grafik-charset)
  IF t$=CHR$(96) THEN t$=CHR$(39)   :GOTO Printout   :REM Apostroph
  IF t$=CHR$(123) THEN t$=CHR$(40)  :GOTO Printout   :REM { wird zu (
  IF t$=CHR$(124) THEN t$=CHR$(165):GOTO Printout   :REM |
  IF t$=CHR$(125) THEN t$=CHR$(41)  :GOTO Printout   :REM } wird zu )
  IF t$=CHR$(167) THEN t$=CHR$(192):GOTO Printout   :REM §

  IF ASC(t$+CHR$(0))<91 AND ASC(t$+CHR$(0))>64 THEN
    t$=CHR$(ASC(t$+CHR$(0))+128)
  ELSEIF ASC(t$+CHR$(0))<123 AND ASC(t$+CHR$(0))>96 THEN
    t$=CHR$(ASC(t$+CHR$(0))-32)
  END IF

Printout:
  PRINT#2,t$;
NEXT

PRINT#2,CHR$(13)

CLOSE #1
CLOSE #2
```

SOUND-AUSTAUSCH ATARI - AMIGA

Das Digitalisieren von Musik mittels Computer ist immer noch nicht außer Mode gekommen, auch auf unserem "Erzrivalen" Atari ST gibt es viele Fans davon. Warum sollte man nicht unter seinen Bekannten, die einen ST und Digitizer besitzen, Musikstücke austauschen?

Digitalisierte Musik auf AMIGA und ST ist grundsätzlich kompatibel. Doch wer schon einmal versucht hat, ein AMIGA-Stück auf einem ST abzuspielen, war wohl enttäuscht. Da kommt so ohne weiteres nichts Vernüftiges raus.
Der Trick ist, daß jedes Byte des Stückes mit 128 geXORt werden muß. Das 7. Bit wird einfach umgedreht. Das ist alles. So lassen sich AMIGA-Stücke auf dem ST abspielen und umgekehrt. Man muß eventuell nur noch die Abspielgeschwindigkeit anpassen, da auf dem ST meist schneller digitalisiert wird.
Das Programm DigiSoundConvert übernimmt diese Arbeit für einen. Nachdem ein ST-Stück beispielsweise mit DOS-2-DOS (Test in Kickstart 12/88) auf eine AMIGA-Diskette übertragen wurde, kann man hiermit die nötige Umwandlung vollführen. Geben Sie Source- und Destinationsfile an (inkl. Pfad) und achten Sie auf genügend Platz auf der Zieldiskette. Sollte das Stück länger als 600 KB sein, hat man die Möglichkeit, es in 2 Teile zu spalten, damit es etwas handlicher wird. Das Stück wird anschließend in Happen wieder abgespeichert. Die Umwandlung dauert wegen des langsamen Basic-Interpreters recht lange. Ich empfehle daher dringend, das Programm zu compilieren.

(Michael Holin)

```
' DigiSoundConvert V1.1
' written 11.Dec.88 by Michael Holin, CLZ
'
' Dies Programm wandelt digitalisierte Musikstuecke vom AMIGA- in
' ST-Format um und umgekehrt. Bei Filelaengen ueber 600k kann das
' Musikstueck in zwei Files gesplittet werden, damit lange AMIGA-
' Stuecke (>720k) auf 2 ST-Disketten aufteilen werden können.
'

ON ERROR GOTO e:
IF FRE(0)<26000& THEN CLEAR ,40000&  '40 KB Speicher
DEFINT a-y
DEFLNG z
maxread=25000
PRINT "DigiSoundConvert V1.1  Amiga <=> Atari ST
PRINT
INPUT "Sourcefile      :";so$
INPUT "Destinationfile:";de$
PRINT
OPEN so$ FOR INPUT AS #1
z=LOF(1) IF z>600000& THEN INPUT"Sourcefile über 600k lang, in 2 Teile
                                                    splitten";j$
IF j$="j" THEN
   OPEN de$+".1" FOR OUTPUT AS #2
ELSE
   OPEN de$ FOR OUTPUT AS #2
END IF
WHILE z>0
   IF z<maxread+1 THEN
      laenge=z:z=0
   ELSE
   PRINT z1;"Byte":laenge=maxread:z=z-maxread:z1=z1+maxread
   END IF
   alt$=INPUT$(laenge,#1)            'maxread Bytes lesen
   z2=SADD(alt$)-1                   'Adresse von alt$ im Speicher
   FOR i=1 TO laenge
      POKE z2+i,PEEK(z2+i) XOR 128   'jedes Byte mit 128 XORen
   NEXT
   PRINT#2,alt$;
   IF j$="j" AND z1=500000& THEN     'evt. 2tes File erzeugen
      CLOSE #2:OPEN de$+".2" FOR OUTPUT AS #2
   END IF
WEND
e:
CLOSE
```


LAUFWERKE
3 1/2" Amiga Extern Formsch. Metallgehäuse helle Front, 880 KB, durchgef. Port, mit Schraubverr. abschaltbar 269,--
3 1/2" Amiga Intern Komplett mit Einbausatz und Anleitung 199,--
5 1/4" Amiga Extern Formsch. Metallgehäuse helle Front, 40/ 80 Spur, durchgef. Port mit Schraubverr. abschaltbar 329,--
3 1/2" Atari ST Extern wie oben, 720 KB, 2 x 80 Spur, eig. Netzteil 289,--
5 1/4 Atari ST Extern wie oben, 720 KB, 40/ 80 Spur, eig. Netzteil 349,--

SPEICHERERWEITERUNGEN
512 KB Ram f. Amiga 500 a. A.
1.8 KB Ram f. Amiga 500 a. A.
2 MB Box Extern z. Zt. auch teilbestückt mit 512 K und 1 MB für Amiga 500 u. 1000 a. A.

Rainbow Data
NEU »Wir finanzieren Ihre Anschaffung« NEU
Profitieren auch Sie von unseren Finanzierungsmodellen. Nähere Angaben auf Anfrage.

COMPUTER
Amiga 500 998,--
Amiga 2000 1998,--
Amiga 2000 und 1084 S 2579,--
20 MB FESTPLATTE f. Amiga u. Atari 949,--
Amiga-Filecard 25 ms, 31 MB Übertragungsgeschw. ca. 460 K
DRUCKER
Mannesmann Tally MT81 399,--
Star LC 10 598,--
Star LC 10 C 725,--
EPSON LX 800 549,--
EPSON LQ 500 879,--
NEC P 6 PLUS 1679,--

COMPUTERLEITUNGEN
Druckerkabel Amiga 500/ 1000/ 2000/ 23,00
Monitorkabel Amiga/ Scart 25,00
Emulatorkabel C 64 - Amiga 19,90
Bootselector DF 0/ DF 1 oder 2 - 3 19,00
Mouse-Pad antistatisch, rutschfest 12,50

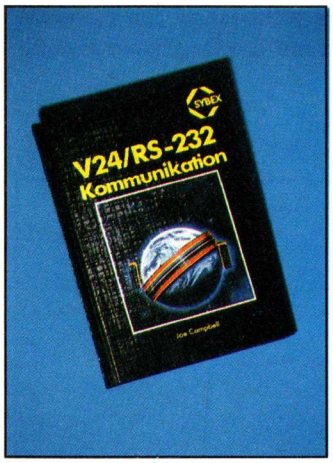
Weitere Angebote auf Anfrage
Preisänderungen vorbehalten

Erfragen Sie unsere aktuellen Tages- und Staffelpreise. Versand per Nachnahme.
Rainbow Data, Am Kalkofen 32, 5603 Wülfrath, Tel.: 02058/1366

MONITORE
Commodore 1084 S 629,--
Atari St SM 124 398,--
Neu im Angebot Kickstarterumschaltung 159,--
Highscorekiller f. alle 68000 3 Stufenschalter und stufenlosem Geschwindigkeitsregler 59,--
Akkustischer **Viruswarner** anzustecken an einen Laufwerksport 49,--

DISKETTEN
3 1/2" No Name 2 DD ab 21,00
3 1/2" Seika 2001 2 DD 24,90
3 1/2" TDK 2 DD 28,50
5 1/4" No Name 48 TPI 7,50
5 1/4" No Name 96 TPI 12,50
5 1/4" TDK 48 TPI 16,50
Public Domain
Wir führen ca. 800 PD f. Amiga auch für Atari und IBM komp. Wir kopieren auf 2 DD Disk.

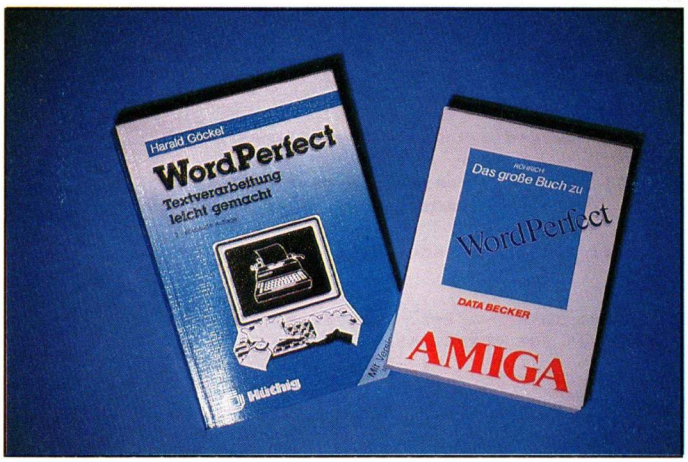
5 1/ 4" **ab 4,--** 10 **ab 3,50**
3 1/ 2" **ab 5,--** 10 **4,50**



V24/RS-232-Kommunikation

Joe Campbell
V24/RS-232-Kommunikation
Sybex-Verlag
217 Seiten
ISBN 3-88745-075-2

Für all diejenigen Zeitgenossen, die aufgrund ihres Bedürfnisses, ihren AMIGA (oder einen anderen Rechner) über die serielle Schnittstelle mit einem anderen Gerät zu verbinden, dem Wahnsinn nahe sind, ist dieses Buch gedacht.
Wer sich schon einmal mit der Verbindung zweier Rechner oder eines Rechners mit einem Peripheriegerät beschäftigt hat, wird um die diversen Probleme, die sich ergeben können, wissen. Der Autor des Buches hat nun seine Erfahrungen, die er gesammelt hat, in diesem Werk zusammengefaßt und will damit dem Leser bei dessen Problemstellungen behilflich sein.
Das Buch will hier aber keine fertigen Lösungen in der Art "Wenn Sie einen AMIGA mit einem XYZ verbinden wollen, dann..." anbieten. Es vermittelt vielmehr das Wissen, das benötigt wird, um die funktionierende Verknüpfung beider Geräte selbst herauszufinden. An fünf Fallstudien wird dabei die Vorgehensweise in der "Praxis" gezeigt. Zuvor wird aber ausführlich auf die Grundlagen der seriellen Schnittstelle eingegangen. Dabei werden auch die Unterschiede zwischen DTE (Datenendgerät) und DCE (Datenübertragungsgerät) genauestens erläutert. Ebenso wird der Schnittstellenstandard RS-232 und dessen Anwendung erklärt. Ein Kapitel zeigt dann einige nützliche, billig herzustellende Hilfsmittel, die das Austesten einer Schnittstelle erleichtern sollen.
Ein gesondertes Kapitel ist dem Anschluß von Modems an Computer und Terminals gewidmet. Auch hier wird wieder besonderes Augenmerk auf die Verwendung der Steuerleitungen gelegt. Im letzten Teil des Buches werden einige käufliche Testgeräte und Kabelzubehör vorgestellt. Hier macht sich negativ bemerkbar, daß dieses Buch eine Übersetzung aus dem Amerikanischen ist. Die vorgestellten Geräte wird man hierzulande wohl kaum erhalten können. Die an sich sehr nützlichen Informationen über den Leistungsumfang der Gerätschafften und deren Preiswürdigkeit sind für uns hierzulande nicht zu gebrauchen.
Die Zielgruppe dieses Buches ist eindeutig: Alle die, die eine serielle Verbindung erstellen wollen, die über die reinen Datentransferleitungen hinausgeht, werden in dem Werk Informationen finden, die es ermöglichen, die fast immer auftretenden Schwierigkeiten mit den Steuerleitungen zu bewältigen. Mit 32 DM ist der Preis erfreulich gering, so daß man dieses Buch getrost empfehlen kann.



WordPerfect-Textverarbeitung leicht gemacht

Göckel, Harald
WordPerfect-Textverarbeitung leicht gemacht
Hüthig-Verlag, Heidelberg
447 Seiten,
ISBN 3-7785-1592-6

Das Werk von H. Göckel ist auf die deutschen WordPerfect-Versionen 4.1 und 4.2 zugeschnitten. Es braucht Sie nicht zu verwundern, daß von der Version 4.2 die Rede ist, weil es diese für den AMIGA ja noch nicht gibt. Das Buch ist für die PC-Versionen des Textverarbeitungsprogrammes geschrieben worden. Das ist aber kein wesentlicher Nachteil, da die PC- und die AMIGA-Version zu 99 % funktions- und bedienungsgleich sind. Daher kann der AMIGA-Eigner getrost auch die Schriftwerke zu PC-WordPerfect als Ratgeber heranziehen.
Doch zum Inhalt des Buches: Nach der Einleitung, die in Kurzform auf die Neuerung der PC-Version 4.2 eingeht, folgt ein Kapitel, das sich Quickstart mit Wordperfect nennt. Hier werden die grundlegendsten Funktionen von WP beschrieben.
In den darauf folgenden Kapiteln werden dann die einzelnen Funktionen und Fähigkeiten von WP ausführlich erläutert. Die Erklärungen der einzelnen Befehle ist gut aufgegliedert. Neben der Angabe der aufrufenden Tastenkombination wird eine ausführliche Erläuterung der Funktionen und ihrer Anwendung gegeben. Danach folgt eine Auflistung der einzelnen Bedienungsschritte. Zum Schluß werden noch besondere Hinweise zu den Befehlen gegeben.
Insgesamt werden alle Befehle gut erklärt, so daß man sagen kann, daß dieses Buch seinen Zweck gut erfüllt. Was noch anzumerken wäre ist die Tatsache, daß man am Buch selbst die Fähigkeiten von WP gut erkennen kann. Es wurde nämlich komplett mit WordPerfect erstellt. Die Gestaltungsmöglchkeiten des Programmes hat man so immer vor Augen. Leider hat man alle Vorlagen mit einem eher mäßigen Nadeldrucker erstellt. Das Schriftbild ist daher dementsprechend dürftig. Mit einem Laserdrucker hätte man hier die Qualitäten der Textverarbeitung in ein besseres Licht rükken können.

Das große Buch zu WordPerfect

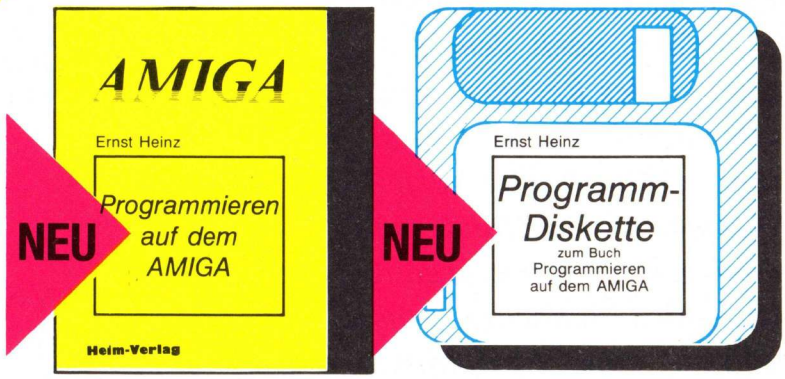
Röhrich, Peter
Das große Buch zu WordPerfect
Data Becker, , Düsseldorf
316 Seiten, 39,- DM
ISBN 3-89011-305-2

Dieses Buch ist für die AMIGA-Version von Wordperfect speziell geschrieben worden. Daher geht dieses Buch auch auf die Besonderheiten der AMIGA-Implementation ein (Mausbedienung). In der Erläuterung der WP-Funktionen bestehen keine wesentlichen "Qualitätsunterschiede" zu dem vorhergehend besprochenen Werk. Für meinen Geschmack sind die Erläuterungen der einzelnen Befehle etwas unübersichtlicher und weniger ausführlich.
Was das große Buch zu WordPerfect dem Produkt von H.Göckel voraushat, ist der Bezug zum AMIGA. Die Installation auf dem AMIGA ist zu Anfang des Buches ausführlich erläutert. Zusätzlich sind alle Abbildungen von der AMIGA-Version von WP übernommen.

Eine Empfehlung für das eine oder andere Buch kann nicht eindeutig gegeben werden. Jedes Werk hat seine speziellen Vorteile. Ein Anwender, der sich in der Bedienung des AMIGA und dessen DOS bereits gut auskennt und nur an einer möglichst ausführlichen Erläuterung von WordPerfect interessiert ist, wird von dem Buch H. Göckels mehr Informationen erhalten. Der AMIGA-Neuling hingegen (oder auch der Nur-Anwender ohne großes Interesse an der Funktionsweise des Rechners) wird sich mit dem Buch von Peter Röhrich eher anfreunden, da dieses auch auf die grundlegende Bedienung des Rechners im Zusammenhang mit WordPerfect eingeht.

PROGRAMMIEREN AUF DEM AMIGA

Das richtige Buch für Auf- und Umsteiger



VON
ERNST HEINZ

PROGRAMMIEREN AUF DEM AMIGA

WICHTIGE MERKMALE

▶ Dieses Buch weist Programmierern den Weg, wie Sie die fantastischen Fähigkeiten Ihres AMIGA auch von **AmigaBASIC** aus nutzen können. Es schließt die in den Handbüchern entstandenen Lücken, indem es die offengebliebenen Fragen in anschaulicher Weise beantwortet.

▶ Damit ist dieses Buch insbesondere für lernwillige Aufsteiger, d. h. mehr oder weniger erfahrene BASIC-Programmierer, sowie für Umsteiger, die bisher mit anderen Computersystemen gearbeitet haben, konzipiert.

▶ Besonderer Wert wurde auf guten Programmierstil gelegt. Anhand von über 40 sehr ausführlich dokumentierten Übungs- u. Beispielprogrammen kann der Leser trainieren, fremde Programme zu lesen und zu verstehen und richtige und gute Programmierung erlernen.

▶ Alle offenen Fragen und Probleme in Bezug auf AMIGA-spezifische Programmiertechniken und -befehle wird Ihnen dieses Buch zu beantworten versuchen.

▶ Besonders werden folgende Hauptthemen im Buch behandelt:
Fenstertechnik · Menue-Technik · Unterbrechung-Technik · Animation in Amiga-Basic · Grafik-Befehle · Normale Maussteuerung

▶ Zum Buch gibt es eine Programm-Diskette mit allen aufgeführten Übungs- und Beispielprogrammen

AUS DEM INHALT

▶ **CLI** und Möglichkeiten der **Execute**-Programmierung
▶ Erstellen einer eigenen **Startup**-Diskette
▶ Spezielle AmigaBASIC-Arbeitsdiskette erstellen
▶ Grafikprogrammierung (ohne Animation)
▶ Sound- und Sprachprogrammierung (Erzeugung menschlicher Sprache/Erzeugung von Tönen und mehrstimmigen Melodien/Tonerzeugung gemäß musikalischer Notennotierung)
▶ **Unterbrechungsfähigkeiten** von AmigaBASIC
▶ **Maussteuerung**
▶ **Menueprogrammierung** von Pull-Down-Menüs
▶ **Fenster-** und **Bildschirmtechnik**
▶ **Grafische Animation** mit **Sprites** und **Bobs**
▶ Ein **Potpurri** von AmigaBASIC-Programmen
▶ Kleine Einführung zur **Spezialhardware** des AMIGA

über 300 Seiten **DM 49,-***

PROGRAMMDISKETTE

AUS DEM INHALT

▶ Zur Unterstützung der praktischen Übung und Arbeit am Computer gibt es die Programmdiskette.
▶ An über 40 Übungs- und Beispielprogrammen können Sie guten Programmierstil nachvollziehen und trainieren.
▶ Damit hat die lästige Tipparbeit ein Ende; Diskette laden und los geht's...
▶ Die Diskette ist beim Verlag erhältlich
und kostet nur **DM 29,-***

* Preise sind unverbindlich empfohlene Verkaufspreise

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Ich bestelle: _____ St. *Programmieren auf dem AMIGA* á DM 49,—
_____ St. *Programmdiskette* zum Buch á DM 29,-

zzgl. DM 5,- Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname _____

Straße, Hausnr. _____

PLZ, Ort _____

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte.

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

VON ANDREAS KRÄMER

Autoboot-Festplatten

Eine neue Generation von Festplatten kündigt sich an - die Autoboot-Platten. Die Betriebssystemversion 1.3 macht es möglich. Das Einlegen der Workbenchdiskette entfällt, nach dem Einschalten des Rechners wird automatisch von der Festplatte gebootet. Der Vorteil liegt klar auf der Hand, zum einen geht der Bootvorgang wesentlich schneller vonstatten, sind Festplatten doch in ihrer Geschwindigkeit normalen Floppystationen weit überlegen, und zum anderen entfällt das lästige Suchen nach der Workbenchdiskette, die meistens nicht zu finden ist, wenn man sie braucht.

Des weiteren sind alle Systemordner auf die Festplatte zugewiesen. Ein Diskettenzugriff ist nicht mehr erforderlich, alle Bibliotheken, CLI-Befehle usw. werden von der schnelleren Festplatte bei Bedarf geladen. Das umständliche und langwierige Zuweisen der Systemverzeichnisse entfällt gänzlich.
Zum Test standen uns zwei Autoboot-Festplatten zur Verfügung. Die GVP SCSI Quantum-Festplatte mit einer Speicherkapazität von 80 MByte ist für den Einbau in den AMIGA 2000 geeignet, die Vortex System 2000 mit 20 MByte Kapazität arbeitet mit dem AMIGA 1000 bzw. A500 zusammen. Sie ist auch unter der Betriebssystemversion 1.2 autobootfähig, die GVP-Platte hingegen nur unter der Version 1.3.

Vortex System 2000

Das Vortex System 2000 hat nichts mit dem AMIGA 2000 gemein, wie vielleicht einige Anwender dachten, als sie im Namenszug '2000' lasen, vielmehr ist das Vortex System 2000 ein System, das für mehrere Rechner angeboten wird. Der Unterschied zwischen den verschiedenen Rechnersystemen besteht darin, daß lediglich die Grundeinheit bzw. das PERSONALITY MODUL gewechselt werden muß. Die eigentliche Festplatte ist bei allen Systemen gleich. Die Festplatte kann somit leicht für andere Rechnersysteme verwendet werden, und man muß keine weitere, teuer Festplatte kaufen.

Im Lieferumfang der Vortex-Festplatte befindet sich die Grundeinheit für den A500 bzw. dem A1000. In beiden Fällen wird die Einheit an den Expansionport des Rechners gesteckt. Zusätzlich findet man ein 20 Seiten starkes Handbuch, das die ersten Schritte erläutert, und zwei Disketten. Auf der einen ist Workbench 1.3D zu lesen, auf der anderen Installationsdiskette System 2000. Letztere dient zum Installieren der Platte, die Workbench 1.3 sollte sich der Anwender auf die Festplatte kopieren. Die Inbetriebnahme der Vortex ist denkbar einfach. Verschiedene Utilityprogramme vereinfachen das Low-Level- und AMIGADOS-Formatieren, das Partitionieren und das Installieren. Auch der Anfänger wird kaum Schwierigkeiten haben, lediglich wenn 3 oder 4 Partitionen installiert werden sollen, muß der Anwender ein wenig mitdenken.

Wie schon erwähnt, besticht die Vortex-Festplatte dadurch, daß sie sowohl unter der Betriebssystemversion 1.2 als auch unter der Version 1.3 autobootfähig ist. Der Vorteil liegt klar auf der Hand: Besitzt man das neue Betriebssystem nicht, fällt das nicht sonderlich schwer ins Gewicht, auch braucht man keine Kompatibilitätsprobleme zwischen V1.2 und V1.3 zu fürchten.

Die komplette Bootfähigkeit ist freilich nur beim A500 möglich, da sich das Betriebssystem bereits im ROM befindet. Beim A1000 muß zunächst die Kickstart-Diskette ins Laufwerk geschoben werden, daraufhin bootet die Festplatte wie gewohnt. Soll nicht von Platte gebootet werden, muß nach einem Reset einfach eine Diskette ins Bootlaufwerk eingelegt werden. Der AMIGA bootet dann von Diskette.

Die Verarbeitung der Vortex-Platte ist ausgesprochen gut, die Abmessungen des AMIGAfarbenen Plattengehäuses belaufen sich auf ca. 24x23x7 cm. Die Platte ist damit recht klein. Das Personality Modul ist flach und etwa 10 cm breit. Das mitgelieferte Handbuch ist in deutscher Sprache verfaßt, aber es läßt manche wichtige Information aus. Beispielsweise findet man keinen Hinweis über das Fast-File-System oder den Interleave-Faktor.

Doch kommen wir zur Geschwindigkeit der Platte. Um 300 KByte zu speichern, werden knapp 3,5s benötigt. Beim Laden der 300 KByte liegt sie knapp darunter. Mit diesen Werten liegt die Platte im Mittelfeld und kann weder zu den schnellsten noch zu den langsamsten Festplatten gerechnet werden. Die Zeiten wurden natürlich unter der Fast-File-System-Formatierung gestoppt.
Etwas störend wirkt sich das laute Lüftergeräusch aus, AMIGA 2000-Besitzer können ja ein Lied von Lüftergeräuschen singen. Mit der Zeit gewöhnt man sich aber daran.

Fazit

Das Vortex System 2000 hat seine Vorteile, können doch Festplatten von anderen Rechnern, die ebenfalls unter der Bezeichnung System 2000 arbeiten, leicht an den AMIGA angeschlossen werden. Daß die Platte unter Betriebssystemversion 1.2 bzw. 1.3 einwandfrei bootet, ist der große Pluspunkt, die verschiedenen Partitionen müssen weder 'gemountet' noch anderweitig angesprochen werden, sie stehen sofort zur Verfügung.

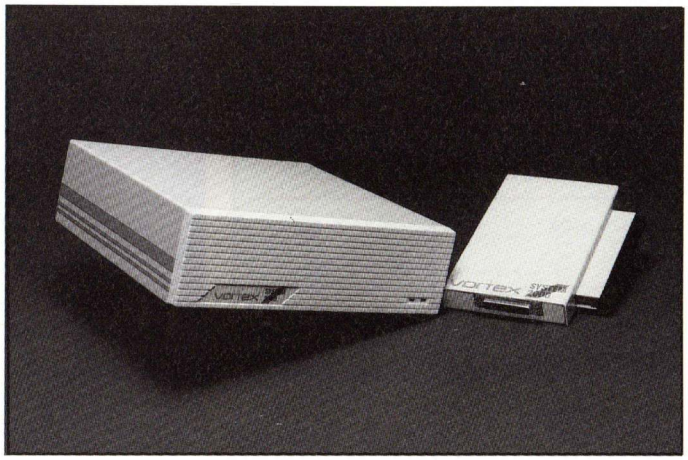
Vortex System 2000

+ einfache Installation
+ Autoboot unter Version 1.2 und 1.3
+ solide verarbeitet
+ andere System 2000-Platten können problemlos angeschlossen werden

- Dokumentation in Deutsch, könnte aber ausführlicher sein
- keine weiteren Festplatten anschließbar

Anbieter:
Vortex Computersysteme GmbH
Falterstr. 51-53
7101 Flein
Tel.: 07131-50880
Preis: 1398.- (20 MByte)

Andere Plattengrößen auf Anfrage



Die Vortex-Platte mit der Grundeinheit hat einen schicken 'Look' und ist unter Betriebssystem-V1.2 und V1.3 autobootfähig.

In Sachen Datensicherheit kann der Festplatte nichts angekreidet werden, während der gesamten Testphase traten weder READ/WRITE - noch CHECKSUM ERRORS auf. Der Expansionport des Personality Moduls ist durchgeschleift, weitere Peripherie kann somit noch angeschlossen werden. Tests mit verschiedenen Speichererweiterungen ergaben ein einwandfreies Funktionieren (obwohl bei der einen Erweiterung das Gehäuse etwas bearbeitet werden mußte, um den Anschluß möglich zu machen).

Auf der beim Kauf fertig formatierten Festplatte findet der Käufer des weiteren ein Programm, mit dem die Festplatte vor dem Transport geparkt werden kann, und ein Festplatten-Backup-Programm, das aus dem Public Domain-Pool stammt.

GVP SCSI/Autoboot-Festplatte

Die GVP-Festplatte ist für den AMIGA 2000 bestimmt, eine für den AMIGA 500 ist aber ebenfalls erhältlich. Zum Test stand der Redaktion eine 80 MByte-Platte zur Verfügung, die in einen freien AMIGA-Slot gesteckt wird. Die 3 $^1/_2$"-Quantum-Platte ist hierbei direkt auf der Steckkarte befestigt. Ihre 80 MByte sind ihr nicht anzusehen, 20 MByte-Platten besitzen die gleichen Abmessungen. Die Autobootfähigkeit wird mit Hilfe zweier EPROMs erlangt, allerdings nur unter der Betriebssystemversion 1.3. Die Autoboot-EPROMs sind für ca. 50.- DM erhältlich.
Im Lieferumfang befindet sich die Steckkarte, eine englischsprachige Dokumentation und eine Installationsdiskette. Wenn die Autoboot-EPROMs mit erworben werden, erhält man eine zusätzliche Dokumentation.
Die Installation der Quantum-Festplatte ist denkbar einfach. Die Steckkarte wird in einen freien AMIGA-Slot gesteckt, und die mitgelieferte Diskette anstatt der Workbench eingeschoben. Wenn die Autoboot-EPROMs mit erworben wurden, müssen sie noch in die beiden leeren Sockel auf der Platine gesteckt werden. Nach dem Ladevorgang kann über die Workbench die eigentliche Installation beginnen. Dazu stehen verschiedene Programme zur Verfügung. 'Install' formatiert die Festplatte, wobei der Anwender zuvor bestimmen kann, in wieviele Partitionen die Platte zerlegt werden soll und wie groß die einzelnen sein sollen. Das File-Format muß ebenfalls zuvor festgelegt werden. Sind alle Angaben gemacht, werden die Partitionen formatiert. Das Programm 'createbootfd' erstellt eine Bootdiskette, mit der die Festplatte gebootet werden kann. Beide Programme sind sehr einfach gehalten, Anfänger werden mit deren Hilfe keine Probleme mit der Installation der Festplatte bekommen. Obwohl die Platte eine reine AMIGA-Platte ist, ist es möglich, eine PC-Partition zu erstellen. Das Programm 'PCBridgeboard' macht dies möglich. Mit diesem Programm wird eine PC-Datei erzeugt, die eine PC-Festplatte simuliert, von der sogar das MSDOS-Betriebssystem gebootet wer-

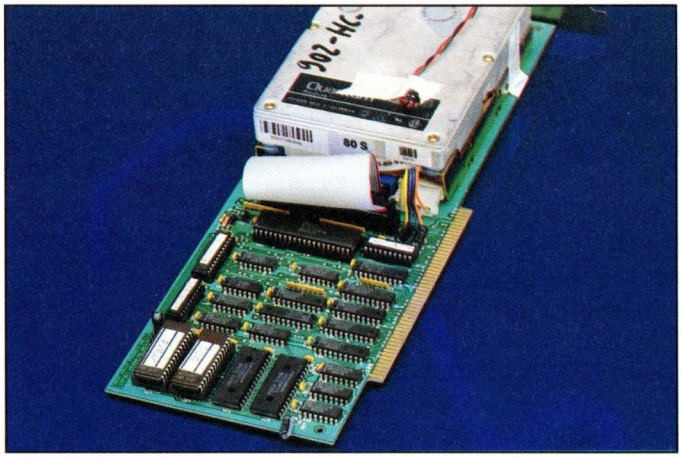
Geschwindigkeitstest:

Es wurden 300 KByte geladen und gespeichert.

	Speichern	Laden
Vortex 20 MByte mit FFS	3.56	2.54
GVP SCSI 80 MByte mit FFS	2.11	1.10
Diskette	64.52	24.56
RAM-Disk	1.16	0.41

FFS=Fast-File-System
Alle Zeitmessungen in Sekunden

Die Geschwindigkeiten im Vergleich. Zusätzlich haben wir noch die Zeiten einer RAM-Disk und einer normalen Floppystation gemessen.



Die 80 MByte-Quantum-Festplatte ist unter der Betriebssystemversion 1.3 autobootfähig.

den kann. Das Arbeiten mit dieser Partition erfolgt genauso wie mit einer herkömmlichen PC-Festplatte. Besitzer einer PC-Karte (AT-Karte) werden dies zu schätzen wissen.

Das automatische Booten

Die GVP SCSI-Festplatte ist nur unter Betriebssystem 1.3 autobootfähig. Aus diesem Grunde müssen Sie bereits im Besitz der Version 1.3 sein, ansonsten verweigert die Platte jedes automatische Booten, und die Autoboot-EPROMs sind unnütz bzw. erweisen sich sogar als äußerst störend. Sind die EPROMs eingesetzt, und schaltet man zum Betriebssystem 1.2 (vorausgesetzt, man ist im Besitz einer Kickstart-Umschaltplatine, die es ermöglicht, zwischen verschiedenen Betriebssystemen hin- und herzuschalten), verweigern die Platte und der Rechner jeglichen Dienst. Der Betrieb unter der Betriebssystemversion 1.2 ist nicht möglich. Die Programmierer hätten gut daran getan, dieses Manko zu beseitigen. Weiß man doch, daß einige Programme mit der Version 1.3 nicht harmonieren. Mit ein wenig Geschick kann aber ein kleiner Schalter angebracht werden, der die Autoboot-EPROMs abschaltet.
Wenn sich eine Diskette in Laufwerk df0: befindet, versucht das System, zunächst von dieser zu booten, gelingt das nicht, wird von Platte gebootet.

Die GVP-Festplatte besitzt einen SCSI-Controller, der mit einer 16 Bit-DMA ausgerüstet ist. Der DMA greift direkt auf einen eigenen Datenpuffer zu. Die Zugriffszeit erhöht sich dadurch auf 11 msec. An den Controller können bis zu 7 weitere SCSI-Geräte angeschlossen werden, der SCSI-Bus ist von der Rückseite des A2000 aus erreichbar.

Die Geschwindigkeit der SCSI-Festplatte ist als sehr schnell zu bezeichnen. Um 300 KByte zu speichern bzw. zu laden, werden lediglich knapp über 2 bzw. 1 Sekunde benötigt. Die RAM-Disk ist da auch nicht viel schneller. Auch die simulierte PC-Partition erwies sich als recht schnell. Diese Geschwindigkeit wird freilich mit dem Fast-File-System erreicht.

Fazit

Die GVP-Festplatte ist ein durchdachtes System. Die Möglichkeit, auf einer AMIGA-Festplatte eine PC-Partition einzurichten, sei hier nur als Beispiel genannt. Die Installationssoftware ist kinderleicht zu bedienen. Anfänger werden keinerlei Schwierigkeiten bekommen. Die Geschwindigkeit der SCSI-Platte ist ebenfalls sehr erfreulich, waren bisher andere SCSI-Platten doch eher mit Floppystationen zu vergleichen. Daß das Betriebssystem 1.2 nicht mit den Autoboot-EPROMs genutzt werden kann ohne den oben erwähnten kleinen Umbau, muß jedoch als Manko betrachtet werden, ebenso wie die in Englisch gehaltene Dokumentation. Lediglich eine DIN A4 große Seite ist in deutscher Sprache beigelegt und erklärt das Nötigste. Ebenfalls erwähnenswert erscheint mir, daß die Platte nur einen AMIGA-Slot belegt. Probleme irgendwelcher Art, seien sie seitens der Platte oder im Zusammenspiel mit anderweitiger Peripherie, traten nicht auf.

Gute Technik kostet ihren Preis. Die SCSI-Platte ist gegenüber anderen Festplatten etwas teurer. Die 80 MByte-Version mit einer Quantum-Platte schlägt mit 3298.- DM zu Buche, die 20 MByte Seagate-Platte mit 1498.- DM.

GVP SCSI Festplatte

+ einfache Installation
+ schnelle Datenübertragung
+ PC-Partition möglich
+ Autoboot unter Kickstart 1.3
+ belegt nur einen AMIGA-Slot
+ bis zu sieben weitere SCSI-Geräte anschließbar

- V 1.2 nicht nutzbar mit Autoboot-EPROMs
- englische Dokumentation

Anbieter:
DTM
Poststr. 25
6200 Wiesbaden-Bierstadt
Tel. 06121-560084
Preis: 3298.- (80 MByte)

Andere Plattengrößen auf Anfrage

AMIGA BASIC
Profibuch



NEU!

Für Theoretiker

Dieses Buch gibt Ihnen einen hervorragenden Überblick über die wichtigsten Möglichkeiten und Methoden der Systemprogrammierung unter AMIGA-BASIC. Neben einer detaillierten Beschreibung und Analyse der wichtigsten Funktionen finden Sie in diesem Buch eine ausführliche Erläuterung der verschiedensten System-Unterroutinen sowie den Aufbau des Gesamtsystems.

Für Praktiker

Im Hinblick auf die praktische Anwendung in eigenen Programmen dürfte die Sammlung von über 60(!) neuen AMIGA-BASIC-Unterroutinen zur Systemprogrammierung ein absoluter Lekkerbissen für jeden BASIC-Programmierer sein. Mit diesen neuen Unterroutinen werden die fantastischen Möglichkeiten des AMIGA nun endlich auch all denjenigen in einfacher Weise zugänglich gemacht, die selbst nicht tiefer ins Sytem eindringen möchten.

Für jeden

Blitzschnelle und variationsreiche Textausgabe, die Benutzung beliebiger Textfonts, die Erzeugung von Pull-Down-Menüs, die Gestaltung eigener Requester sowie die komplette Fenstersteuerung werden ab sofort für jeden BASIC-Programmierer leicht realisierbar.

Aber noch viel mehr Geheimnisse erwarten Sie während der Lektüre der einzelnen Kapitel.

MAXON

Bestellcoupon MAXON Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Name: _____
Vorname: _____
Straße: _____
Ort: _____
Unterschrift: _____

Hiermit bestelle ich:

☐ Exemplar(e) von "AMIGA BASIC Profibuch".
Mit Diskette für DM 59,00
☐ Vorauskasse
☐ Nachnahme

Versandkosten: DM 7,50
Nachnahme zuzgl. DM 3,50 Nachnahmegebühr.

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
Vertrtieb in Holland: JOTKA COMPUTING Postbus 8183 NL-6710 AD Ede Tel.: 08380/38731 Fax.: 08380/21675

LESERFORUM

Briefe an KICKSTART

Die Qual der Wahl

Ich habe eine Frage: In welcher Sprache lassen sich Spiele am besten programmieren? Und wie lerne ich das dann auch gut? Bitte helfen Sie mir!

M. Heckmair (12 Jahre)

Die Sprache, in der Spiele wohl am besten zu programmieren sind, ist Assembler. Da diese Sprache aber recht schwer zu erlernen ist, würden wir Dir empfehlen (wir nehmen einmal an, daß Du noch keine andere Programmiersprache erlernt hast), mit einer sogenannten Hochsprache (Basic, C) erst einmal Erfahrungen zu sammeln. Programmierfehler sind in Hochsprachen leichter zu finden, und die Programme sind auch besser zu überblicken. Als Einstieg würden wir AMIGA-BASIC empfehlen, da diese Sprache im Lieferumfang des Rechners enthalten ist. So bleibt noch Geld übrig für Bücher, die man als Anfänger sicherlich gut gebrauchen kann. Zudem findet man in der Public Domain viele Programme in AMIGA-BASIC, aus denen man jede Menge Anregungen und Programmiertricks entnehmen kann.

Die Red.

TTL-Monitor am AMIGA

Seit Tagen suche ich nach einer Möglichkeit, den A500 an einen TTL-Monitor für einen IBM-PC anzuschließen, da diese Monitore relativ billig und zudem von der Bildqualität her sehr gut sind. Vielleicht können Sie mir bei der Lösung meines Problemes helfen?

U.Siebraße

Mit einem normalen IBM-TTL-Monitor wird Ihr Unterfangen nicht von Erfolg gekrönt sein. Die Video-"Formate" des AMIGA und des IBM-MonoChromAdapters sind leider zu unterschiedlich. Der AMIGA verwendet eine Zeilenfrequenz von 15750 Hertz, der IBM-PC hingegen schreibt seine Videodaten mit einer Zeilenfrequenz von 18432 Hertz. Da die Monitore sich nur im Bereich von etwa plus/minus 1 kHz synchronisieren, ist es dem Monitor nicht möglich, das Bild zu "fangen". Eine Möglichkeit, einen monochromen Monitor anzuschließen, besteht aber trotzdem. Monitore mit "Composite-Video"-Eingang können das Videosignal des AMIGA verarbeiten. Die Bildqualität liegt zwar etwas unter der eines TTL-Monitors, da bei dieser Signalform Helligkeits- und Synchroninformationen gemischt übertragen werden, ist aber immer noch als gut zu bezeichnen.

Die Red

Nachbestellung

Wie wäre es, wenn man bei Ihnen die Kurse als Kopie nachbestellen könnte? So müßte man nicht unbedingt das ganze Heft nachbestellen. Kann man die Unkosten auf den Preis abwälzen oder sind sie zu groß? Die Kurse könnte man so wegordnen, ohne die Zeitschrift zu zerstören. Lassen Sie sich die Idee einmal durch den Kopf gehen.

E. Zühlke

Den Kernpunkt der Angelegenheit haben Sie schon selbst angerissen: die Kosten! Das Herauskopieren und der Versandaufwand sind dermaßen kostenintensiv, daß eine solche Vorgehensweise nicht in Erwägung gezogen werden kann, solange ein Nachbestellen des Heftes möglich ist. Die Kosten für einen kopierten Artikel kommen in die Größenordnung eines Einzelheftes, was es nicht als sinnvoll erscheinen läßt, einen solchen Service anzubieten. Um Ihre geliebten KICKSTARTs nicht verhackstückeln zu müssen, bleibt es Ihnen natürlich unbenommen, für den Eigenbedarf Kopien eines Artikels zu erstellen.

Die Red.

BASIC

Bitte veröffentlichen Sie mehr Sachen in BASIC, und zwar aus folgenden Gründen:

- Viele Einsteiger haben bereits Erfahrung in dieser Sprache.
- BASIC wird beim Kauf von Commodore mitgeliefert.
- C-Compiler sind zu teuer für Leute, die nur mal so programmieren wollen.

H.-W. Hennes

HD

Bitte testen Sie doch einmal die "Add-on-Box" von Tandon für die PC/AT-Seite des A2000. Mir scheint dies die optimale HD-Lösung zu sein.

A.Theiler

Prinzipiell wäre die von Ihnen angeführte Wechselplatte eine interessante Lösung. Gegen die Verwendung auf dem PC-Teil des AMIGA sprechen aber technische Gründe. Diese Wechselplatte arbeitet nach dem RLL-Prinzip. Hiermit treten nun Probleme bei AMIGA-Partitionen auf. Da diese Fehlfunktionen von dem verwendeten Rechner abhängig sind (vom jeweili-

gen Rechner, nicht dem Rechnermodell!), erscheint es für uns nicht ratsam, Platten zu testen, die nach diesem Aufzeichnungsprinzip arbeiten. Ein Funktionieren des Testgerätes mit unseren Rechnern würde nicht eine allgemeine Funktionstüchtigkeit der Festplatte mit allen AMIGAs garantieren. Der Informationswert und die Hilfe beim Kaufentscheid wären dadurch hinfällig.

Die Red.



Multitasking

Der AMIGA ist bekanntermaßen ein Multitasking-Computer, was selbst einige MS-DOS-Zeitschriften zugeben. [Und die geben garnix zu, was dazu führt, daß den AMIGA-Zeitschriften "Identitätsprobleme" angedichtet werden, wobei es eine Tatsache ist, daß es die PC-Zeitschriften sind, die übereifrig Erweiterungskarten testen (= anpreisen), welche den PC auch grafik-, midi- und soundfähig machen usw. Für viel Geld, versteht sich... Ich habe einen AMIGA 2000 mit PC-Karte, weiß also, wovon ich rede.]

Ein Anliegen hätte ich aber noch:
Es müßte ein wichtiges Testkriterium sein, ob ein Programm, egal aus welcher Sparte, multitaskingfähig ist! Wenn mein PC im AMIGA nebenbei einen Serienbrief druckt, kann ich mit DPaintII meiner Malfreude Ausdruck geben. ...Aber schon, wenn ich mit Interceptor fliegen will, ist Sense mit der Arbeit am PC! Vielleicht überdenken Sie meine Idee einmal.

H.J. Röcknagel

Eines expliziten Vermerks über die Multitaskingfähigkeit eines Programmes bedarf es unserer Meinung nach nicht, da man grundsätzlich sagen kann:
Multitasking und Anwendungen - ja;
Multitasking und Spiele - nein.
Das Dilemma bei den Spielen ist hauptsächlich auf zwei Punkte zurückzuführen: Die Erfordernisse der Spiele an die Rechenleistung (hauptsächlich aufwendige Grafiken) ist so groß, daß der Programmierer alle Ressourcen des AMIGA für sich beansprucht.
Der andere Hauptpunkt ist der leidige Kopierschutz: Die meisten Programme arbeiten mit Kopiersicherungen, die das Booten notwendig machen, und machen damit dem Multitasking den Garaus. Beim Verzicht auf solchen Kopierschutz wären vielleicht auch einige Spiele multitaskingfähig.

Die Red.

Rätselprogramm

Ich suche für einen AMIGA 1000 ein Rätselprogramm. Da die Suche bisher vergeblich verlief, wende ich mich heute an Sie. Vielleicht können Sie mir weiterhelfen.
Das Programm sollte verschiedene Funktionen beinhalten, so etwa das Edieren, Löschen, Eingeben und Drucken von einzelnen Datensätzen. Das Programm sollte die bekannten Schwedenrätsel lösen helfen. Die dazu benötigten Begriffe und Lösungen sollten zum größten Teil vom Anwender eingegeben werden können.
Bitte teilen Sie mir mit, ob Sie so ein Programm kennen. Falls nein, könnten Sie mir vielleicht Vorschläge machen, wo ich es sonst noch bekommen könnte.

M. Weisgerber

Leider kennen auch wir kein solches Programm. Aber vielleicht hat ein Leser ein derartiges Programm oder arbeitet an einer Lösung.

Die Red.

Externe Tastatur

Da Sie sich seit längerem in Ihrer Zeitschrift regelmäßig mit dem Bridgeboard für den AMIGA 2000 beschäftigen, möchte ich mich mit zwei Fragen an Sie wenden. Zum einen möchte ich gerne wissen, ob man auch als Besitzer der A2088-Karte die neue Software für die AT-Karte von Commodore erwerben kann. Zum anderen habe ich festgestellt, daß es sowohl auf der PC- als auch auf der AT-Karte eine Steckleiste (J7) gibt, an die man angeblich eine externe Tastatur anschließen kann. Ich würde gerne erfahren, ob und wie z.B. eine IBM-Multifunktionstastatur angeschlossen werden kann.

Jürgen Dexel

Die neue Software wird als käufliches Update erhältlich sein. Sobald sie verfügbar ist, werden wir unsere Leser darüber unterrichten.

Vom Anschluß einer externen Tastatur an die von Ihnen erspähte Steckleiste raten wir Ihnen ab, da diese Tastatur zwangsweise mit der eingebauten kollidieren würde. Zum einen träte ein Hardwareproblem auf (beide Tastaturen senden auf denselben Port), und zum anderen würde sich die Zusatztastatur aller Voraussicht nach nicht mit dem Programm PCWindow der Janus-Software vertragen. Theoretisch ließen sich beide Probleme beheben, uns ist jedoch noch keine entsprechende Lösung bekannt.

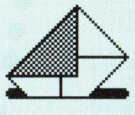
Die Red.



AMIGA DOS 1.3 & Janus-Harddisk

Sie haben schon viel über die PC-Karte und die damit verwendbare (und auch für den AMIGA nutzbare) Janus-Harddisk berichtet. ebenso haben Sie die neue Workbench 1.3 vorgestellt, die gegenüber WB1.2 doch manche Verbesserungen mit sich bringt.
Das Betreiben der Janus-Harddisk unter WB1.2 ist relativ problemlos: Installieren mit 'DJMOUNT' und die Sache läuft. Doch wo bleibt dieser Befehl bei WB1.3?
Im Manual zu WB1.3 habe ich keinen Befehl dieser Art gefunden. Auch der 'DJMOUNT'-Befehl der WB1.2 brachte mit WB1.3 kein Resultat.

Das Problem hat seinen Grund darin, daß verschiedene DOS-Versionen nicht von vornherein aufeinander abgestimmt sind. Dies ist nicht nur beim AMIGA so, sondern auch bei IBM. So funktionieren die 3.2-Dienstprogramme auch nicht unter DOS 3.3.
Bei dem von Ihnen angesprochenen Phänomen liegt die Ursache aber eher in der benutzten Vorversion der WB1.3. Darin funktionieren einige Befehle noch nicht wie in der Endversion geplant. Versuche zeigten, daß DJMOUNT (V1.2) sich bei uns mit WB1.3 vertrug.



Leserbriefe geben nicht unbedingt die Meinung der Red. wieder. Die Red. behält sich vor, Zuschriften sinngemäß zu kürzen.

STEFAN OSSOWSKI'S SCHATZTRUHE

①	**Haushaltsbuch**	bis zu 25 Konten, flexibel, leicht bedienbar, mit umfangreicher **deutschsprachiger** Dokumentation
②	**ASDG-RamDisk**	resetfeste RamDisk mit deutscher Anleitung
③	**Mountain Cad**	professionelles CAD-Programm, deutsche Anleitung
④	**Spiele I, II, III**	10 erstklassige PD-Spiele aus allen Bereichen wie Action, Geschicklichkeit, Strategie...
⑤	**Anti Virus**	5 Programme gegen alle Viren
⑥	**M.S.-Text**	hochwertige **deutsche** Textverarbeitung
⑦	**Utility-Disk**	25 nützliche Utilities aus allen Bereichen
⑨	**Sonix-Paket**	Original Sonix-Player + 4 weitere Disketten mit phantastischer Sonix-Musik. **Top-Hit!**
⑩	**Business**	3 Disketten: Tabellenkalkulation, relationale Datenbank, sehr gute Textverarbeitung
⑪	**Grafik**	3 phantastische Slideshows: Politiker, Tiere und Ray-Tracing-Bilder
⑫	**Bundesliga**	Bundesligaverwaltung mit grafischer Darstellung in **deutsch**
⑬	**Paranoid**	sensationelles Breakout-Spiel
⑭	**Buchhaltung**	erstes **deutsches** PD-Buchhaltungsprogramm
⑮	**Perfect Englisch**	deutscher Englisch-Vokabeltrainer
⑯	**AMIGA-Paint**	sehr gutes deutsches Malprogramm
⑰	**Videodatei**	bringt Ordnung in Ihre Videodatei, **deutsch**
⑱	**Fußballmanager**	bei diesem Spiel können Sie Ihre Fähigkeiten als Manager eines Fußballclubs testen, **deutsch**
⑲	**Platten/Cassetten/CD-Liste, deutsch**	
⑳	**Giroman**	Komfortables **deutsches** Programm, mit dem Sie Ihr Girokonto einfach verwalten können.
㉑	**Spiele-Tips & Tricks-Lexikon**	zu ca. 100 populären Spielen erhalten Sie zahlreiche Tips **deutsch**
㉒	**Kampf um Eriador, V2.0**	taktisches Strategiespiel für 2 Personen mit sehr guter Grafik u. Sound, **deutsch**
㉓	**Mandelbrot**	verschiedene ausgezeichnete Programme zur Berechnung u. Darstellung von **Apfelmännchen**
㉔	**Funckey**	hilfreiches Programm zur F-Tastenbelegung
㉕	**Label 2.0**	Etikettendruckprogramm, **deutsch**
㉖	**Risk**	die Amiga-Umsetzung des bekannten Brettspiels, **deutsch**
㉗	**Ray-Tracing-Construction-Set, V 2.0**	phantastisches Programm zur Berechnung von Licht u. Schatten – siehe Test AMIGA 1/88 – komplett auf **3** Disketten mit **deutscher Anleitung**

Für unsere **Schatztruhe** benutzen wir Markendisketten.

Preise:
Porto/Verpackung: DM 3,– V-Scheck
DM 6,– Nachnahme

Paket Nr. 9 = DM 50,–
Pakete Nr. 4, 10, 11, 27 = je DM 30,–
alle sonstigen Nr. = je DM 10,–

Wir führen alle bekannten PD-Serien wie Fish, Panorama, Faug Amicus, Auge, Taifun, Chiron Conception. RPD, Kickstart, Slideshows, Slipped, TBAG, PD, Franz...
Viele Programme deutschsprachig

Taifun ★ Taifun
= PD-Serie der Extraklasse

Sonderpreise: incl. Porto/Vp. geg. Vorausk./ V-Scheck

1-35	DM 175,-
36-70	DM 175,-
1-70	DM 315,-
71-80	DM 65,-

5,-DM

kosten unsere aktuellen **2** Katalogdisketten. Mit deutschem Inhaltsverzeichnis unseres gesamten PD-Angebotes von weit über 1000 Disketten. Ab DM **4,50**. Gegen Verrechnungsscheck oder in Briefmarken anfordern.

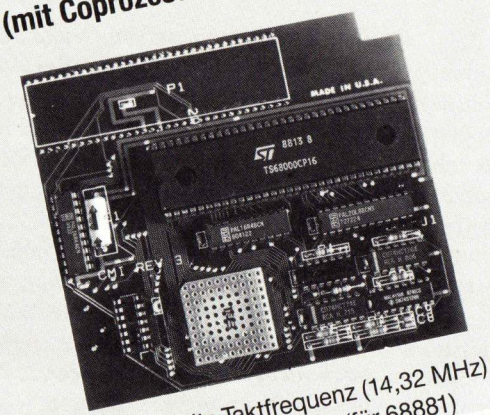
Stefan Ossowski – Ihr PD-Spezialist –
Veronikastr. 33, 4300 Essen 1, Tel. 02 01 / 78 87 78

CMI Creative Microsystems Inc

Processor Accelerator

DM 498,–

(mit Coprozessor!!!) DM 998,–



- Doppelte Taktfrequenz (14,32 MHz)
- Coprozessor Sockel (für 68881)
- Macht aus 50% unbedingt nur 40%!!!
- Üb. 500% höhere Rechenleistung m. Coprozessor
- Vollkompatibel, blitzschneller Bildschirmaufbau, schnellere Berechnung (z.B. Raytracing)
- Für alle Amiga Modelle lieferbar!

RS 232 MIDI Interface

mit Tool Software DM 169,–



- Selektierbare Ausgänge (max. 2) und Durchführungen
- MIDI In und Sync out (24, 48 oder 96 Ticks)
- LED-Anzeige für MIDI Aktivität
- Auch als A2000 Steckkarte lieferbar
- Alle Produkte ab Lager lieferbar!!!

Intelligent Memory
Borsiallee 18, 6000 Frankfurt/M. 60
Tel. 069-41 00 71/72
Fax 069-41 40 68
Händler erfragen bitte unsere Konditionen!

MICROTRON-Computerprodukte
Postf. 69, Bahnhofstr. 2
CH-2542 Pieterlen
Tel. 032-87 24 29
Fax 032-87 24 82

Erhältlich in allen Filialen

SCHAULANDT

L + S Schaulandt	Tel. 040/47 70 07
Schaulandt Harburg	Tel. 040/77 24 64
Schaulandt Wandtsbeck	
Schaulandt Kiel	Tel. 0431/67 6051
Schaulandt Flensburg	Tel. 04 61/1 30 91
Schaulandt Ahrensburg	Tel. 0 41 02/5 04 03
Schaulandt Braunschweig	Tel. 05 31/40 637
Schaulandt Wiesbaden	Tel. 0 61 21/30 50 18
Schaulandt Frankfurt	Tel. 069/28 08 93

Kalte Wickel

Therapie für defekte RAM-Bausteine

Wenn Sie in der Ausgabe 2/89 das Programm RamTest abgetippt oder Sie es sich sonstwie beschafft haben, dann haben Sie vielleicht erschrocken festgestellt, daß auch Ihre RAMs ihre Informationen nicht mehr halten können. Nachdem Sie nun die erste Diagnose kennen, werden Sie sich wohl freuen, daß eventuell doch noch Hoffnung besteht, die defekten RAMs reparieren zu können.

Von
Michael Rausch
Meinhard Ullrich

Denn es gibt mehrere Gründe, warum ein Speicherbaustein verrückt spielt. Inzwischen ist es wohl jedem bekannt, daß es sich zumeist um Timing-Probleme beim Refresh handelt. Wie wir inzwischen erfahren haben, tritt der Gedächtnisschwund mit Vorliebe auf der oberen Speicherbank der Expansionskarte von Commodore auf (ab $c80000). Das liegt oft daran, daß die Zeit zwischen fallender Flanke des -RAS-Signals und steigender -CAS-Flanke für die meisten eingesetzten Chips zu kurz ist. (Vielen Dank für Deine Erläuterung, Uwe!) Die fest eingelöteten Chips, die auf der unteren Speicherbank der Expansionskarte sitzen und Ihren AMIGA 2000 auf 1 Megabyte aufrüsten, vertragen diese kleine Zeitspanne "zufällig". Oft jedoch ist die Ursache für Speicherfehler auch woanders zu suchen. Und vor allem, wenn die untere Speicherbank oder das Chipmem Fehler eingestehen müssen, sollten Sie jetzt weiterlesen. Dann nämlich kann Ihnen die im folgenden vorgestellte Chipkur vermutlich helfen.

Profis vor

Bevor wir fortfahren, soll hier schon eine erste Warnung gegeben werden. Das Verfahren ist nicht ganz unkritisch. Deswegen möchten wir Sie jetzt schon darauf hinweisen, daß wir hierfür keine Haftung übernehmen können. Wer unsere Anleitung befolgt, sollte keine zwei linken Hände haben. Ziehen Sie einen Freund oder eine Freundin zu Rate, der/die mehr davon versteht als Sie. Oder befragen Sie Ihren Commodore-Händler, ob die Therapie für Ihre Chips sinnvoll ist. Am besten aber rufen Sie bei der Commodore-Entwicklungsabteilung in Braunschweig an und lassen sich unter genauer Angabe der Seriennummer und der Bezeichnung der RAMs dort beraten.

Was ist nun diese andere Ursache, die am Versagen Ihres Speichers schuld ist? Speicherbausteine sind zur Zeit sehr teuer, und deshalb werden sie auch immer rarer. Das hat zur Folge, daß in der Vergangenheit bei einigen Herstellern die Auswahltoleranz gestiegen ist, das heißt, es kommen RAMs in den Handel, die früher aussortiert worden wären. Dies geschieht, weil oft nicht mehr alle Kontrollen angewandt werden, oder weil die Ansprüche an das Produkt gesunken sind. Außerdem sind einige Chips auf den Markt gelangt, bei denen eine Kältebehandlung überhaupt nicht durchgeführt worden ist. Die Folge ist, daß Leiterbahnen oft mangelhaft ausgeführt sind. Es treten winzige Haarrisse in den Leiterbahnen auf, so daß oft sogar der Kontakt zumindest zeitweise unterbrochen ist. Dies hat nicht nur zur Folge, daß High-Signale manchmal nicht übertragen werden. Bei den geringen Abständen zwischen den Leiterbahnen kommt es bei einem Signalabstand von etwa 10 dB (Rechtecksignale!) zu Ohm-Sprüngen.

Teurer Rat ist gut

Schauen wir uns dazu zunächst Bild 2 an. An der dritten Leiterbahn von oben stellen wir einen besonders schwerwiegenden Materialfehler fest. Dieser Bereich, der durch eine Umrandung markiert ist, wird in Bild 3 noch genauer dargestellt. Es handelt sich hier um eine Spitze an der Bruchstelle mit einem Durchmesser von etwa 2 nm (2 x 109 m). Tritt hier ein Rechtecksignal auf, wie es in Bild 1 dargestellt ist, so kommt es in der Spitze zu einem Potentialstau. Über die Potentialgleichung

$$U = \frac{Q}{4\pi E_o} * \frac{1}{r}$$

können Sie auf die Ladung schließen, die sich hier bei einer Signalspannung von 5 Volt aufbaut. Durch diesen Potentialstau wird ein elektrisches Feld erzeugt, gemäß der Gleichung:

$$|E| = \left|-\frac{dU}{dr}\right| = \frac{U}{r} =$$

$$5V/2nm = 2{,}5 \times 10^9\ V/m.$$

Nun gibt es aber in Luft schon Durchbrüche für elektrische Feldstärken ab 2-3 x 10^6 V/m in alle Richtungen, so daß die Ladung in benachbarte Leiterbahnen übertragen werden kann (natürlich kann sie sich so auch meist in der betroffenen Leiterbahn fortsetzen).

Überdies erzeugt ein sich änderndes E-Feld nach Maxwell ein B-Feld:

1.Maxwell-Gleichung:

$$\oint \underline{E}\,d\underline{r} = -\frac{d}{dt}\iint \underline{B}\,dA = -\frac{d}{dt}\Phi m$$

folgt

$$U_{ind} = \oint \underline{E}\,d\underline{r} = -\frac{d}{dt}\iint \underline{B}\,dA = -\frac{d}{dt}\Phi m \Rightarrow U_{ind} = \dot{\Phi}_m$$

Im dynamischen Fall muß man das Linienintegral schreiben als

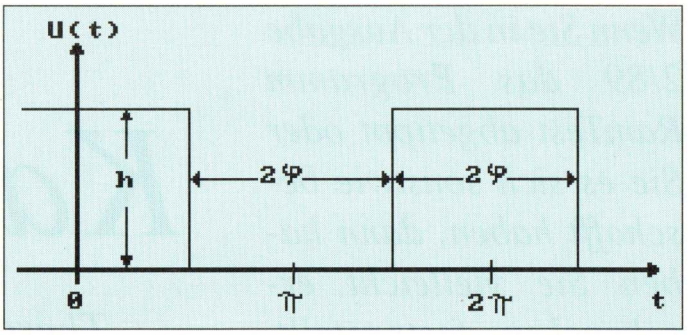
$$\oint \underline{E}\,d\underline{r}(t)$$

Wie man aus der Signalskizze in Bild 1 leicht einsehen kann, geht der Abstand dr(t) an den steilen Flanken des Rechtecksignals gegen unendlich. Dies deutet auf eine große E-Feld-Änderung hin, wodurch letztlich auch ein starkes B-Feld erzeugt wird. Dieses induziert nun zusätzlich einen Stromfluß in den eh schon gebeutelten Nachbarleiterbahnen. Man nennt das lose Kopplung. Nehmen wir für das Rechtecksignal (immer noch Bild 1) die Fourierreihe:

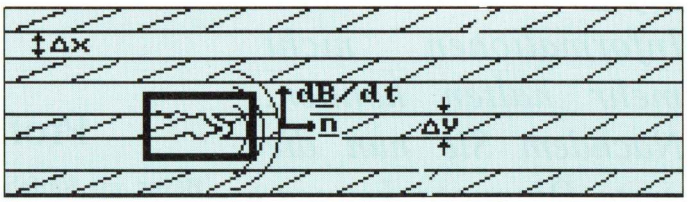
$$U(t) = \frac{4h}{\pi}\left(\frac{\cos\varphi}{1}\sin t + \frac{\cos 3\varphi}{3}\sin 3t + \right.$$
$$\left. + \frac{\cos 5\varphi}{5}\sin 5t + \frac{\cos 7\varphi}{7}\sin 7t + \cdots\right)$$

her, brechen nach fünf Gliedern ab, und setzen als Nebenbedingungen Arbeitstemperatur und Ausdehnungskoeffizient des Leitermaterials geeignet ein, so kommt man schlußendlich zu einem induzierten Potentialverlauf in den Nachbarleitern, wie er etwa in Bild 4 dargestellt ist. (Jeder Flankenwechsel des Originalsignals erzeugt eine Spannungsspitze im induzierten Leiter. Diese Signale im Nachbarn werden nun leider häufig als High-Signale mißinterpretiert.

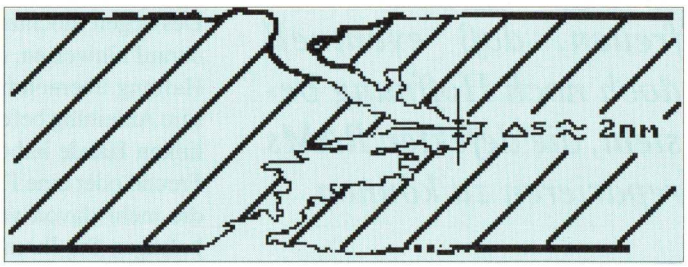
Alles in allem sehen Sie, wozu diese Leiterbrüche führen können. Gäbe es nur eine Möglichkeit, sie zu kitten.



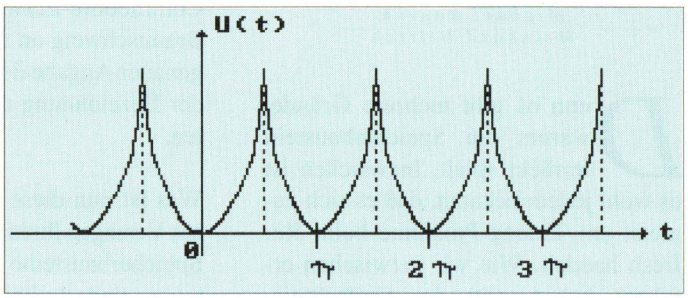
Signal in zeitlichen Verlauf



Leiterbahndurchbruch



Schemazeichnung des Leiterbruchs



Lose Kopplung einer Nachbarleiterbahn

Kleben, ohne zu tropfen

Treten die RAM-Fehler auf der Speicherkarte auf, schrauben Sie Ihren AMIGA auf. Achtung: Haben Sie noch einen Garantieanspruch, dann bringen Sie lieber Ihr Gerät zu Ihrem Systemhändler! Ziehen Sie nun Ihre Expansionskarte oder andere Speichererweiterungskarten. Wir holen damit nun das nach, was bei der Herstellung womöglich vergessen wurde. Wickeln Sie Ihre Karte in ein dünnes Handtuch ein und legen Sie sie in Ihren Kühlschrank bei etwa 4°C. Bitte NICHT ins Gefrierfach legen! Lassen sie die Karte etwa 4 - 5 Stunden im Kühlschrank. Danach wikkeln Sie die Karte aus und lassen sie bei Raumtemperatur (20°C) mindestens 12 Stunden liegen. Dies hat folgenden Grund: Legen Sie die Chips in die Kälte, ziehen sich die Leiterbahnen etwas zusammen; sobald sie danach wieder einer höheren Temperatur ausgesetzt werden, dehnen sie sich aus und verhaken oder verkleben miteinander. Kommen Sie trotzdem nicht auf die Idee, die Karte anzuwärmen! Sie würden ihr damit schaden. Der beschriebene Vorgang ist irreversibel, hat er funktioniert, treten diese Probleme nicht mehr auf. Trotzdem sollten Sie diesen Versuch nur einmal machen, denn hat es beim ersten Mal nicht geklappt, besteht kaum Hoffnung, daß es beim zweiten Mal besser wird. Das Handtuch ist nötig, um die Karte vor der erhöhten Luftfeuchtigkeit im Kühlschrank zu schützen und somit Kurzschlußschäden weitgehend auszuschließen. Daher sollten Sie die Karte danach auch lange genug im Warmen liegen lassen, so daß sie ausreichnend trocknen kann. Im übrigen brauchen Sie die RAMs nicht von der Karte zu lösen und allein kalt zu machen: Die beschriebenen Materialfehler können unter Umständen auch genausogut auf der Karte selbst auftreten.

Was tun wir, wenn sich der Fehler im Chipmem befindet? Es hilft nichts, es muß der ganze Rechner in den Kühlschrank. In Handtücher eingewickelt, sollte ihm nichts passieren, wenn Sie danach lange genug warten (hier bis zu 24 Stunden), bevor Sie ihn wieder benutzen. Sie sollten aber auf keinen Fall den AMIGA 2000 kaltstellen, vor allem dann nicht, wenn andere Karten, wie zum Beispiel die XT-Karte, oder eine Festplatte eingebaut sind. Überhaupt noch einmal die Warnung: Wenn Sie nichts davon verstehen: Bitte Finger weg!

Bei einigen hartnäckigen RAMs hatten wir auch schon Erfolg, nachdem wir sie mit 1,4-Dimethylbenzol besprüht hatten. Diese Methode können wir aber nicht weiterempfehlen, sie führt bei unsachgemäßer Anwendung zur Zerstörung der Chips.

Natürlich können Sie die Speicherkarte während des Betriebs in -270°C kalten Stickstoff tauchen; der erhöhte Widerstand an den Leitungsunterbrechungen entfällt wegen des Quanten-Hall-Effekts völlig, legt man die Ergebnisse von Klitzings zugrunde. Diese Methode behebt alle Ihre Probleme auf einmal. Aber Spaß beiseite. Bleiben Sie lieber bei oben vorgestellter Vorgehensweise. Sind Sie sich auch nur ein bißchen unsicher, so fragen Sie erst an kompetenter Stelle nach. Falls Sie aber unsere Methode, die wir nach dem Ort ihrer Entstehung Kneipenkur getauft haben, erfolgreich angewendet haben, so lassen Sie uns das bitte wissen.

Viel Erfolg!

ENDE

AMIGA 500 / 1000 / 2000

	DM		DM
IC 8362	39,10	Netzteil A-500	143,64
IC 8364	58,48	Maus A-500/1000/2000	106,70
IC 8520 A-1	29,98	Tastatur A-500	209,08
KICKST. V1.3	69,43	Einbaulaufw. A-500	283,18
IC 6242 B	17,90	Leergehäuse A-500	78,66
IC 6570-036	29,75	Digitalboard A-500 PAL	513,00
IC 68000-8	19,49	Tastatur-Kabel A-1000	30,55
IC 68000-10	33,74	Tastatur A-1000	311,22
IC 68010-8	54,72	Netzteil A-2000	414,22
IC 8371 FAT	80,03	Tastatur A-2000	297,54
IC 8367 PAL	94,28	1 MB Ram Erw. A-2000	513,00
IC 5719 GARY	23,60	Dram41256-15 (256Kx1)	19,95
IC 8361 NTSC	60,76	IC FE 2010 A	112,86
externes Tastaturgehäuse für A-500			117,08
passendes Verbindungskabel hierzu			38,42

sowie (fast) alle CBM-Chips für Reparatur + Service
Preisliste gegen adressierten Freiumschlag.

HÄNDLERANFRAGEN ERWÜNSCHT
(bitte Nachweis beifügen)
Preise sind unverbindlich empfohlene Verkaufspreise!

CIK-Computertechnik · Ingo Klepsch
Postfach 13 31 5828 Ennepetal 1
Tel. 0 23 33 / 8 02 02 Fax 0 23 33 / 7 03 45

AMIGA Public-Domain SUPERPREISE!!
Über 2000 Disks im Archiv!
Jede 3,5" PD-Disk. . . . . 2,70 DM
ab 50 Stück nur **2,60 DM**
Wir verwenden nur errorfreie Qualitätsdisketten!

Wir liefern: Fish,Chiron,RPD,Poseidon,Kickstart, Auge,Tornado,Panorama,Bordello,Amicus,Faug,Ruhr, Cactus,ACS,Taifun,Franz,RHS,PornoShow,TBAG,SACC und ca. 25 andere Serien !

Leerdisketten 3,5" 2DD 135 TPI 10 Stück DM 21.-

Spielepaket 10 Disketten=ca.40 PD-Spiele DM 40.-
Einsteigerpaket für Amiga-Anfänger (Utilities, CLI-Hilfen,Infos usw.) 10 Disk.DM 40.-
Das Superpaket bestehend aus Textverarbeitung, CAD,Haushaltsprogramm,Anti-Virus-Disk usw. alles mit deutscher Anleitung! 15 Disk.DM 55.-
Das Soundpaket 10 Disketten mit tollen Sonix-Super-Sounds,inkl. Sonix-Player-Disk DM 40.-
Das Super-Mix (je 5 Disketten aus vorgenannten Paketen) 20 Disk.zum SONDERPREIS von nur DM 69.-

Alle Preise zuzüglich Versandspesen.
Wir suchen laufend Programme aller Art ! Rufen Sie an !
Tel. 05202/71099

ABC-SOFT
I.Güldenpfennig,Postfach 1124,4811 Oerlinghausen

Peter Biet Computerdesign
Georg-Fischer-Str. 5, 6415 Petersberg 2
Telefon 06 61 / 60 12 63



PAL-RGB-MULTIPROZESSOR 2000 (Version I, II und III)
Die neue Gerätegeneration, mit noch mehr Einsatzmöglichkeiten. Durch die enorme Vielseitigkeit verknüpft mit guten Leistungsmerkmalen und einem bisher unschlagbaren Preis-Leistungs-Verhältnis stellt dieses neukonzipierte Gerät einen Meilenstein im Bereich der Desktop-Videobearbeitung mit dem Amiga dar.

Einsatzbereiche:
1.) Elektronischer **RGB-Farbsplitter** (automatisch und manuell!!). Separate Regelmöglichkeiten in den Bereichen **Helligkeit, Kontrast, Farbsättigung** und die einzelnen **RGB-Intensitätswerte**.
Damit ist es auch mit **DIGI-VIEW** und einer Farbcamera oder einem Videorecorder möglich, absolut professionelle Ergebnisse zu erzielen!
2.) **Video-Colorprozessor:** dient zur Veränderung bzw. Verfremdung des anliegenden Farbvideosignals in o. g. Bereichen zwecks Überwachung, Kontrolle und Überspielung
3.) **Videoüberspielverstärker:** mit Level und Konturregelung.
4.) **RGB-PALUMWANDLER:** erzeugt aus dem Computer-RGB-Signal ein PAL-Videosignal. Ebenfalls in o. g. Bereichen regelbar! (Ersetzt damit alle herkömmlichen PAL-Videokarten.)
neu 19 Zollgehäuse mit neuem Design, Digitastern, Anzeigendisplay und Chinchbuchsen.
neu Mehrfachüberspielmöglichkeit auf bis zu drei Videorecorder gleichzeitig.
neu Anschlußmöglichkeiten eines Multisynchronmonitors als Videomonitor über RGB-Ausgang.
neu Einspeisung des neuen **Super VHS** Signals und Umwandlung in ein normales FBAS-Signal. (Vollständig regelbar ohne Qualitätsverlust!)
neu DIGI-VIEW-Digitizer kann direkt intern eingesteckt werden. (Stromversorgung durch eigenes Netzteil, keine externe Kabelverbindung mehr nötig und Druckerumschaltmöglichkeiten!) **Preis auf Anfrage**

DVS-2000 System
Das bewährte Realtimefixing Digitizersystem in voller PAL- und Overscanauflösung.
Lieferumfang:
- **Digitizer** (alle Auflösungen incl. HAM 2-4096 Farben in PAL!)
- **Digitaler VHS-Videorecorder** (eigenständige Nutzung möglich!) (kann auch durch einen Super-VHS-Recorder ersetzt werden!)
- **PAL-RGB-MULTIPROZESSOR 2000** Version III
zusammen nur 2798,– DM (auch einzeln)
Demo-Dia-Show (DVS-2000 und PAL-RGB-MULTIPROZESSOR 2000) 10,– DM

PAL-RGB-GENLOCK
Verschiedene neue Typen und Versionen für den Amiga 2000.
Preise und Unterschiede auf Anfrage.

PRINT TECHNIK
8000 München 40 Phone (49) (89) 36 81 97
Nikolaustraße 2 FAX (49) (89) 39 97 70

N E U !! ELEKTRONISCHE FILTER!!!
RGB-TRENNER für DiggiView und Diamond Amiga Digitizer jetzt erhältlich
SENSATIONSPREIS DM 298,–
Legen Sie die Filter weg. Mit diesem Zusatz können sie die Bilder von Ihrer FARB-KAMERA digitalisieren, in den Computer ablegen, farbig auf dem Schirm darstellen und ausdrucken!!! Ein ideales Gerät für jeden Digitizer. Anschlußfertig zwischen VIDEOQUELLE und Digitizer einzusetzen.
AMIGA FLACHBETT A4 SCANNER
Print-Technik Universal DM 1.498,–
Der Scanner kann als BILDERFASSUNGS-GERÄT/Kopierer und Thermodrucker eingesetzt werden. Die Scanndichte beträgt 200 Punkte/Zoll, die Scannzeit 10 Sekunden. Die Ablage des GANZEN Bildes erfolgt im IFF, die Auflösungen 320 x 200 / 640 x 400 / 320 x 256 / 640 x 512 werden unterstützt. Ausschnittvergrößerungen sind möglich. Komplett mit Software. Binär + 16 Grau Darstellung. Demo DM 10,–
VIDEO TEXT (WELTNEUHEIT)
EMPFANGS-SPEICHER-MODUL DM 298,–
Endlich kann man das VIDEOSIGNAL eines Recorders, ScartTV oder Tuners dazu verwenden den freien Service des Teletextes im IFF oder ASCII Format abzulegen. Empfängt alle Programme und Sie sind über alle Teletextangebote in Europa informiert. Super Grafik-Darstellung.
NEU!!! EUROTIZER/RGB-DIGITIZER DM 498,–
Dieser Digitizer mit Software enthält einen RGB-Trenner. Sofort Farbbild auf dem Schirm. Sensationell
FARB CCD-KAMERA
mit ZOOM 12.5.-75 DM 1.998,–
Ideale Reprokamera für den Digi View und in Verbindung mit einem Frame Freezer ideal zur Echtzeit-Digitalisierung.
Benelux: 0 10-45 07 696 / NL: 0 79-41 25 63

OSTERREICH · 1060 WIEN · STUMPERGASSE 34 · TEL. 02 22 / 597 34 23 · TELEX 112 996
SCHWEIZ · MICROTRON · 2542 PIETERLEN · BAHNHOFSTR. 2 · TEL. 0 32 / 87 24 29

VON CHRISTIAN KELLER

Praxis

WordPerfect im Detail

Teil 2

Mailmerging

Heutiges Thema soll das Mergen von Texten sein. Die Hauptanwendung dabei ist das "MailMerging", also das Erstellen von Serienbriefen. Diese Funktion von WordPerfect kann aber zu weit mehr benutzt werden, als man denkt. So wären hier die Herstellung von Addreßaufklebern, Beschriftung von Briefumschlägen oder das Ausfüllen von Formularen zu nennen.

Was ist ein Serienbrief?

Zuerst soll einmal erläutert werden, was ein Serienbrief überhaupt ist. Darunter versteht man ein Dokument, das an mehrere Addressaten verschickt werden soll. Das Dokument unterscheidet sich nur durch seine unterschiedlichen Addressen und Anreden. Es wird nur einmal erstellt, wobei besondere Mischbefehle eingefügt werden, die die addressatenspezifischen Angaben ersetzen. Diese Angaben werden in einer zweiten, ebenfalls mit besonderen Steuerzeichen eingerichteten Datei gespeichert. Die Sekundärdatei enthält also diverse, gleichartig gestaltete Datensätze, die den Addressaten entsprechen. Die Anzahl der Datensätze stimmt mit der Zahl der erzeugten Texte überein.

Erstellen einer Sekundärdatei

Die Erläuterung der Sekundärdatei wird zuerst durchgeführt, da so das Verstehen der Primärdateibefehle wesentlich leichter ist. Zu Beginn öffnet man in WordPerfect ein leeres Dokument. In diesem werden nun die einzelnen Datensätze eingetragen. Jeder Datensatz ist in mehrere Datenfelder unterteilt. Ein Datenfeld wird mit dem Code "^R" abgeschlossen. Dabei ist es gleichgültig, ob man diesen Code durch die Betätigung der Funktionstaste "Mischen R" oder gleichzeitiges Drücken der CTRL- und R-Taste auf dem Bildschirm erzeugt. Über diese Möglichkeit, die manchem Benutzer angenehmer ist, schweigt sich das Handbuch leider aus. Das Ende eines Datensatzes wird mit "^E" gekennzeichnet. Zu den Formaten der Daten ist folgendes zu beachten: Jeder Datensatz muß vom Format her absolut identisch sein. Jede Veränderung im Aufbau schlägt sich, zumeist sehr negativ, im entsprechenden Mischergebnis nieder. Hier einige Tips, wie man Fehler vermeiden kann:

- Die Anzahl der Leerzeilen vor dem ersten Datensatz muß gleich der Anzahl der Leerzeilen nach dem Datensatzendezeichen bis zur ersten Zeile des nächsten Datensatzes sein. Diese Leerzeilen zählen zum ersten Datenfeld, werden also mit eingemischt.
- Zwischen dem letzten Zeichen des Datenfeldes und dessen Endekennung immer die gleiche Anzahl Leerzeichen setzen.
- Leere Datenfelder dürfen nicht übersprungen werden. Hier muß in einer Extrazeile ein alleiniges Datenfeldendezeichen erzeugt werden. Geschieht dies nicht, werden alle folgenden Datensätze um ein Feld verschoben und an der falschen Stelle eingemischt.
  Am Ende einer Sekundärdatei keine Leerzeilen einfügen. Diese werden als Anfang eines weiteren Datensatzes interpretiert. WordPerfect beginnt das Dokument von neuem und erzeugt es bis zum ersten Einfügezeichen. Dadurch erhält man am Ende des Mischprodukts eine halbe Version des Primärdokumentes.

Die Primärdatei

In die Primärdatei werden die Datensätze der Sekundärdatei eingefügt. Um WordPerfect mitzuteilen, an welchen Stellen dies zu geschehen hat, werden wiederum spezielle Steuerzeichen ein-

gefügt. Hierbei wird wie folgt vorgegangen:

WordPerfect numeriert die Datenfelder eines Datensatzes mit Eins beginnend durch. Will man nun die einzelnen Datenfelder abrufen, wird im Text an der entsprechenden Stelle der Code "^Fn^" eingefügt. "n" steht für das einzufügende Datenfeld und ist durch dessen Nummer zu ersetzen. Da diese Nummer nirgends von WordPerfect angezeigt wird, notiert man sich einen Datensatz und numeriert diesen. So hat man bei der Erstellung einer neuen Primärdatei immer eine Hilfe zu den Datensatznummern.
Den Leerschritten vor und nach den einzufügenden Datenfeldern sollte auch ein besonderes Augenmerk gelten. Hat man im Datenfeld schon Leerzeichen eingefügt und setzt in der Primärdatei ebenfalls welche, ergeben sich lange Zwischenräume, die unschön aussehen. Besonders wichtig ist dies bei Feldern, deren Anfang ein vorhergehendes Wort in der Primärdatei ergänzen soll. Hat man im Text folgendes stehen:

Sehr geehrte^F7^,

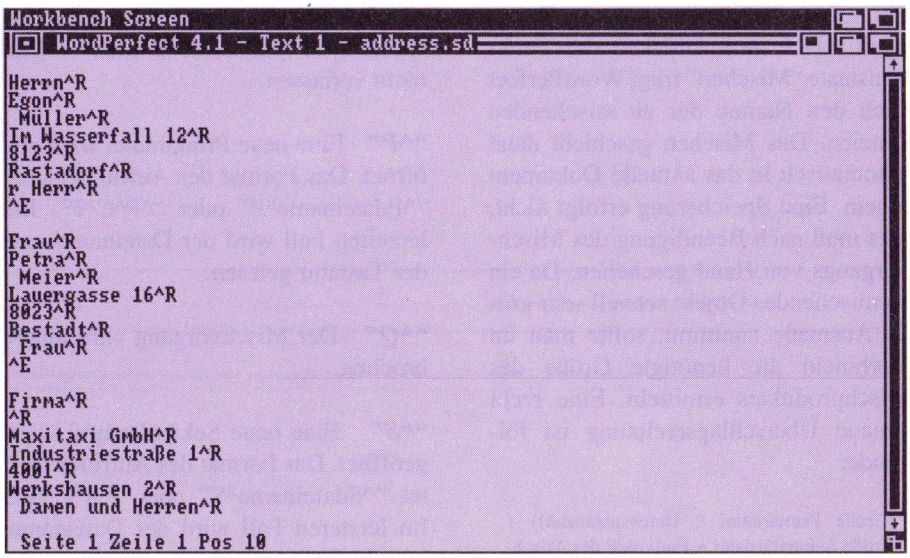
und will man dies durch geschlechts- oder familienstandsspezifische Anreden ergänzen, müssen die Datenfelder mit der Nummer 7 wie folgt aussehen:

"r Herr^R",
" Frau^R",
"s Fräulein" oder
" Damen und Herren"

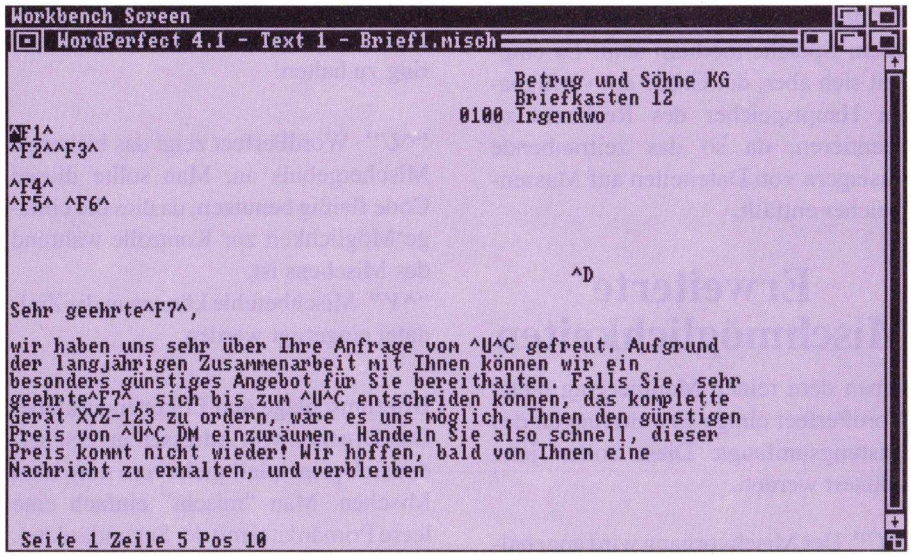
Die Anführungszeichen dienen nur zur Kenntlichmachung von Anfang und Ende der Zeile. In der Sekundärdatei dürfen diese nicht eingegeben werden. Fügt man nun vor "r Herr^R" oder "s Fräulein^R" ein Leerzeichen ein, wird der Datensatz wie "Sehr geehrter Herr" eingefügt. Läßt man hingegen bei " Frau^R" den Leerschritt wegfallen, werden die Worte "geehrte" und "Frau" aneinandergeschrieben. Beides ergibt wohl nicht den gewünschten Effekt.

Die Mail, die merge ich jetzt

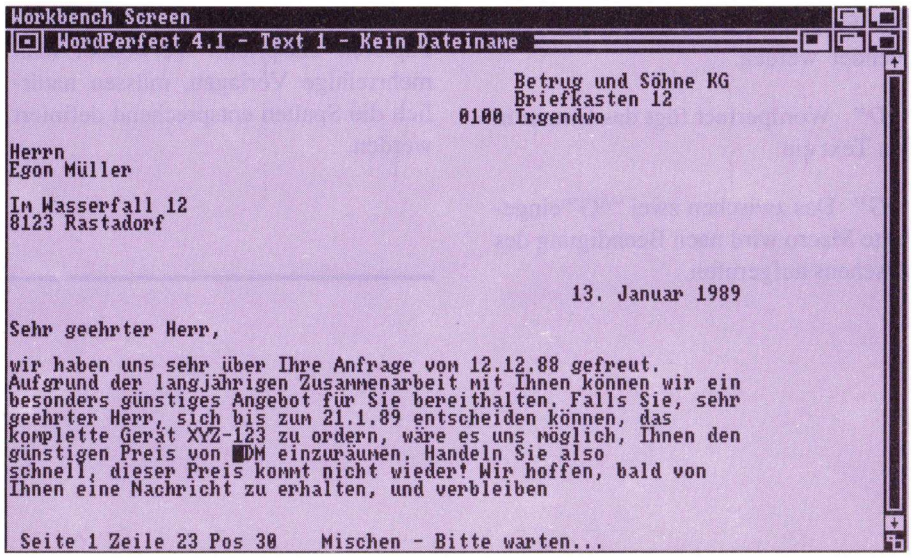
Hat man sowohl Primär- als auch Sekundärdatei fertiggestellt, kann der Mischvorgang begonnen werden. Dazu benö-



Die Sekundärdatei enthält die Datensätze die mit dem Serienbrief gemischt werden.



Die Primärdatei mit den Steuerzeichen zum Einfügen der Daten



Das Ergebnis des Mischvorganges. An verschiedenen Stellen können auch während des Mischens Eingaben gemacht werden

tigt man wiederum ein leeres Dokument. Nach dem Drücken der Funktionstaste "Mischen" fragt WordPerfect nach den Namen der zu mischenden Dateien. Das Mischen geschieht dann automatisch in das aktuelle Dokument hinein. Eine Speicherung erfolgt nicht, dies muß nach Beendigung des Mischvorgangs von Hand geschehen. Da ein zu mischendes Objekt schnell sehr große Ausmaße annimmt, sollte man im Vorhinein die benötigte Größe des Mischproduktes ermitteln. Eine recht genaue Überschlagsrechnung ist folgende:

(Größe Primärdatei * Datensatzanzahl) + Größe Sekundärdatei = Endgröße des Mischproduktes

Diese Größe muß auf jeden Fall kleiner als der verfügbare Speicherplatz auf Ihrem Speichermedium sein. Es empfielt sich aber, die Größe am verfügbaren Hauptspeicher des Rechners zu orientieren, da so das zeitraubende Auslagern von Dateiteilen auf Massenspeicher entfällt.

Erweiterte Mischmöglichkeiten

Neben dem reinen Mailmerging bietet WordPerfect einige Erweiterungen des Leistungsumfangs. Diese sollen jetzt erläutert werden.

"^C" Der Mischvorgang wird angehalten und eine Tastatureingaben erwartet. Dies kann dazu dienen, veränderliche Daten in die Serienbriefe einzugeben. Die Eingabe muß mit "Mischen R" beendet werden.

"^D" Wordperfect fügt das Datum in den Text ein.

"^G" Das zwischen zwei "^G" eingefügte Macro wird nach Beendigung des Mischens aufgerufen.

"^N" Lesen des nächsten Datensatzes. Dabei wird das aktuelle Mischobjekt nicht verlassen.

"^P" Eine neue Primärdatei wird geöffnet. Das Format des Aufrufes lautet "^Pdateiname^P" oder "^P^C^P". Im letzteren Fall wird der Dateiname von der Tastatur gelesen.

"^Q" Der Mischvorgang wird abgebrochen.

"^S" Eine neue Sekundärdatei wird geöffnet. Das Format des Aufrufes lautet "^Sdateiname^S" oder "^S^C^S". Im letzteren Fall wird der Dateiname von der Tastatur gelesen.

"^T" Das Mergeergebnis wird zum Drucker gesendet. Das ist eine Möglichkeit, den benötigten Speicherplatz gering zu halten.

"^U" WordPerfect zeigt das bisherige Mischergebnis an. Man sollte diesen Code fleißig benutzen, da dies die einzige Möglichkeit zur Kontrolle während des Mischens ist.

"^V" Mischbefehle können in die Zieldatei eingefügt werden.

Briefumschläge und Etikettenaufkleber
Das Drucken von Adressen unterscheidet sich prinzipiell nicht vom normalen Mischen. Man "mischt" einfach eine leere Primärdatei mit der Sekundärdatei. "Leer" bedeutet aber nicht ganz leer, es ist nur kein Text darin gespeichert. Einziger Inhalt der Primärdatei ist das Papierformat, das den zu bedruckenden Papieren entspricht. Verwendet man mehrreihige Vorlagen, müssen natürlich die Spalten entsprechend definiert werden.

```
Misch-Codes
C. Eingabe über Tastatur
D. Datum
E. Ende Datensatz
F. Feld aufrufen
G. Macro aufrufen
N. Nächster Datensatz
P. Primärdatei eröffnen
Q. Mischvorgang beenden
R. Feldende
S. Sekundärdatei eröffnen
T. Zum Drucker
U. Bildschirm neu anzeigen
V. Transfer-Codes
0. Storno
```

Besondere Controlcodes zum Mailmerging

Formulare

Bei Formularen wird eine Primärdatei verwendet, die an allen Stellen, die eine Eingabe erfordern, den Code "^C" enthalten. Der Mischvorgang wird so gestopt und der Benutzer kann seine Daten einfügen. Da WordPerfect immer eine Sekundärdatei verlangt, verwendet man am besten ein Dokument, das nur das Papierformat enthält. Natürlich kann die Primärdatei auch andere Codes enthalten, als Beispiel sei hier nur das Datum genannt. Beim Herausfinden der Positionen, an denen die Einfügecodes plaziert werden, kann nur eines geraten werden: Ausmessen und Probieren. Man kann hierzu ein Raster erstellen und auf das Formular drucken oder mit einem speziellen Lineal die Positionen messen.

Ich hoffe, Ihnen hiermit genügend Hilfestellungen zur Nutzung dieses WP-Features gegeben zu haben. Auch hier gilt wieder: Übung macht den Meister. Hat man sich einmal mit dem Mergen auseinandergesetzt, fällt es leichter, als es aussieht.

Viel Erfolg.

ENDE

RESETFEST

VON FRANK SCHÄFER

Unsterbliche Programme nicht nur für Freaks, sondern für Jedermann

In letzter Zeit machen "resetfeste" Programme auf dem AMIGA immer mehr Furore, Turboprint II, Grabbit 1.2, Card:, vd0: oder angekündigte Harddisktreiber wie A.L.F. 2.0 sind "resetfest". Eine weitere Klasse dieser Programme sind Viren wie der SCA-Virus, der Byte-Bandit-Virus oder ihre zahlreichen Mutanten. In diesem Aufsatz möchten wir zeigen, daß auch der Hobbyprogrammierer "resetfeste" Programme schreiben kann.

Definieren wir doch erst einmal den Begriff "resetfest": Ein Programm soll nach einem Reset, ausgelöst durch CTRL-AMIGA-AMIGA, automatisch wieder eingebunden und gestartet werden. Wenn möglich, soll dies auch bei Systemabstürzen geschehen. Wieso das nicht immer klappt, dazu weiter unten.

Die Möglichkeit, resetfeste Programme zu schreiben, ist schon im Betriebssystem vorgesehen. Grundsätzlich gibt es zwei Möglichkeiten, Programme resetfest zu machen.

1. Über die Pointer *ColdCapture* und/oder *CoolCapture*, um diese dreht sich dieser Aufsatz.

2. Über die Vektoren *KickTagPtr* und *KickMemPtr*, diese Möglichkeit wird heute nicht näher erläutert.

Bevor wir mit einem eigenen Programm beginnen, erst einmal ein Stück disassembliertes Betriebssystem, und zwar der Abschnitt, der bei einem Reset durchlaufen wird.

Alle hier und später benutzten Offsets findet man in den Includes, die bei jedem Assembler/Compiler mitgeliefert werden. Bei DevPac sind die Includes zu *ExecBase* im Direktory *Exec* und dort in dem File *'ExecBase.i'* zu finden.

In dem ROM/WOM-Listing sieht man einen Auschnitt der Resetroutine. Zuerst wird die Adresse von ExecBase auf ihre Konsistenz hin geprüft. Dies geschieht auf zwei Methoden: Zuerst wird geprüft, ob *ExecBase* an einer geraden Adresse liegt. (Bit Null gesetzt = ungerade Adresse). Im Anschluß daran wird *ExecBase* zu *ChkBase* addiert. In *ChkBase* steht die negierte Adresse von *ExecBase*. Falls das negierte Ergebnis ungleich Null ist, wird die Kaltstartroutine aufgerufen.

Der Start der Kickstart, Teil 1

```
FC0140    move.l  ExecBase,d0          * ExecBase nach d0
          btst.l  #0,d0                * Adresse gerade?
          bne.s   Kaltstart (FC01CE)   * Nein, -> Kaltstart
          movea.l d0,a6                * ExecBase nach a6
          add.l   ChkBase(a6),d0       * ChkBase + ExexBase
          not.l   d0                   * Summe negieren
          bne.s   Kaltstart (FC01CE)   * Ungleich Null ->
          moveq   #0,d1                * Summenregister = 0
          lea.l   $SoftVer(a6),a0      * Anfangsadresse
          moveq   #$18,d0              * über $18 Worte
Sum       add.w   (a0),+,d1            * Summe bilden
          dbra    d0,Sum               * Schleife
          not.w   d1                   * Ergebnis negieren
          bne.s   Kaltstart (FC01CE)   * # Null -> Reset
          move.l  ColdCapture(a6),d0   * ColdCapture laden
          bne.s   NoColdCapture        * Kein Vektor!
          movea.l d0,a0                * ColdCapture -> a0
          lea     NoColdCapture,a5     * Rückkehradr. -> a5
          clr.l   ColdCapture(a6)      * ColdCapture löschen
          jmp     (a0)                 * Programm starten

NoColdCapture
  .
  .
  .
```

Die Prüfsumme...

...über den ersten Teil von *ExecBase* wird im nächsten Teil des Listings berechnet. Der Prüfsummenalgorithmus selbst ist denkbar einfach. Über den Bereich $22 (*SoftVer*) bis $3A (*SysStkLower*) der *ExecBase* wird eine Word-Summe gebildet. Diese Summe wird negiert und dann in *LowMemCkhSum*, Offset $24, abgespeichert (der Bereich ist 18 Words lang). Beim Bilden der Prüfsumme wird das Word, in dem die Prüfsumme abgespeichert ist, als Null angenommen. Beim Testen der Summe auf ihre Richtigkeit wird *LowMemChkSum* (dort ist die Prüfsumme abgespeichert) zu der Gesamtsumme addiert. Man beachte, daß das *LowMemChkSum* vor dem Reset schon berechnet wurde. Die Gesamtsumme ist, wie schon gesagt, das Ergebnis der Addition der Words in dem oben angegebenen Bereich von *ExecBase*.
Das Ergebnis dieser Addition wird negiert. Falls bei der Negation Null herauskommt, stimmt die Prüfsumme.

Der Zweck der Prüfsumme ist einfach erklärt: Falls bei einem Absturz Daten an unbestimmte Adressen geschrieben werden, könnten auch in *ExecBase* Fehlinformationen auftauchen. Bei einem erneuten Hochfahren hätte dies fatale Folgen. Wenn bei den Tests eine solche Fehlinformation gefunden wird, folgt daraus ein Sprung in die Kaltstartroutine, die erst einmal die kompletten Systemdaten neu aufbaut. Bei diesem Vorgang werden zwangsläufig auch die Vektoren gelöscht, über die ein resetfestes Programm eingebunden wird. Aus diesem Grund klappt das Einbinden von Programmen nach Systemabstürzen auch nicht immer.

In der Systemroutine wird nun der ColdCapture-Pointer in das Register D0 geladen. Falls der Pointer gleich Null ist, wird das Label *NoColdCapture* angesprungen. Im anderen Fall wird die Rücksprungadresse nach A5 übertragen. Sie entspricht der Adresse des Labels *NoColdCapture*. Das Löschen des ColdCapture-Vektors ist die darauf folgende Aktion. Mit dem JMP-Befehl wird nun die im ColdCapture-Vektor eingetragene Routine angesprungen.

Aufgaben der angesprungenen Routine

Die angesprungene Routine muß das Register A5 retten, da in ihm die Adresse steht, die nach der Abarbeitung des eigenen Programmes angesprungen werden muß. Falls das Programm auch bei dem nächsten Reset angesprungen werden soll, muß das Programm den ColdCapture-Pointer neu setzen, der ja vom Betriebsystem gelöscht wurde.

Der ColdCapture-Vektor wird sehr früh in der Resetroutine angesprungen. Zu diesem Zeitpunkt ist im System noch nicht viel initialisiert (Stack, Libraries, Devices), wenn man mal von der ExecBase absieht. Ein Programm, das in den ColdCapture-Vektor eingebunden wird, sollte weder Libraries noch den Stack (!) benutzen. Aus diesem Grund wird die eigene Routine auch mittels "JMP" angesprungen und nicht mit dem Befehl "JSR", der ja die Rückkehradresse auf dem Stack ablegt.

Falls man in einem Bereich, über den eine Prüfsumme gebildet wird, einen Wert verändert, muß man folglich auch die Prüfsumme korrigieren. Zu diesem Zweck findet man in den Listings ein Macro mit dem Namen "PRUEFSUMME". Es ist nicht die übliche Version, die u.a. auch im Betriebssystem und in allen Büchern zu finden ist, sondern eine optimierte Version, die mit einigen Operationen weniger auskommt.

In dem Listing, in dem das PRUEFSUMMEN-Macro zu finden ist, befindet sich auch die Routine, die in allen Programmen als resetfestes Programm eingebunden wird. Diese kleine Spielerei macht nichts anderes, als laufend das Farbregister für die Hintergrundfarbe ($DFF180) mit einer neuen Farbe zu setzen. Und immer, wenn einmal die gesamte Palette durchlaufen wurde, wird die linke Maustaste überprüft. Falls das sechste Bit in der Speicherstelle $BFE001 gesetzt ist, ist die linke Maustaste gerade gedrückt. Als Folge davon wird die Schleife beendet.

Colorregister & Copper

Bei verschiedenen Gesprächen mit anderen Programmierern ergab sich die Frage, wieso man nicht einfach in ein Farbregister schreiben kann, bzw. wieso ein Wert, den man in ein solches Register schreibt, nicht erhalten bleibt. Die Antwort auf diese Frage gibt die Copperlist eines normalen Screens, hier einer Workbench. Wie man in dem Copperlist-Programm sieht, das bei jedem Aufbau des Bildschirms abgearbeitet wird, erfolgt in der Copperlist u.a. auch das Setzen der Farbregister.
Wenn man selbst die Farbe dauerhaft verändern möchte, kann man u.a. diese Copperlist verändern, oder man muß mit dem Prozessor recht häufig die Farbregister auffrischen. Sinnvollerweise sollte man aber den Weg über die Copperlist wählen. Da die Erstellung bzw. Veränderung einer Copperlist aber mit etwas Aufwand verbunden ist, und es hier nur um eine kleine Demo geht, wurde der "primitive" Weg über den Prozessor beschritten.

Besprechen wir aber nun das erste Listing, RESET 1.

Als Option für den DevPac-Assembler wird nur "d+" gewählt, was bedeutet, daß eine Symboltabelle zum Debuggen erzeugt wird. Es lassen sich im Endeffekt aber nur die Routinen debuggen, die zum Einbinden des resetfesten Programmes durchlaufen werden. Mit der Assemblerdirektive "incdir" wird festgelegt, von wo die Includes geladen werden sollen. In den nächsten Zeilen werden die Includes bestimmt, die der Assembler beim Assemblieren laden soll. Dieser Kopf ist übrigens bei allen Programmen praktisch gleich. Das eigentliche Programm hat nur 5 Zeilen. Zuerst wird die Adresse von *ExecBase* in das Register A6 geschrieben. Die Adresse wird benötigt, um mittels der Offsets *CoolCapture* und *ColdCapture* an die entsprechenden Variablen zu kommen. Der CoolCapture-Vektor wird sicherheitshalber gelöscht. In den ColdCapture-Vektor wird die Adresse der Resetroutine eingetragen. Mit dem Macro PRUEFSUMME wird noch die *LowMemChkSum* von *ExecBase* berechnet und in die entsprechende Variable eingetragen. Mit dem RTS-Befehl endet RESET 1. Nun ist ein resetfestes Programm über den ColdCapture-Vektor eingebunden.

Falls man nach dem Start von Reset1 nun einen Reset durchführt, wird man feststellen, daß der Bildschirm ziemlich wild und bunt zu flackern beginnt. Dieses Flackern wird bei allen hier vorgestellten resetfesten Programmen benutzt, um zu zeigen, daß jetzt eine eigene Routine aktiv ist.

In der Resetroutine wird zunächst A5 in eine eigene Variable gerettet (der Stack ist ja nicht verfügbar!), und anschießend wird die Adresse der Resetroutine wieder in den ColdCaptureVektor eingetragen. Nun wird die "DoSomething"-Routine solange durchlaufen, bis man die linke Maustaste drückt. Nachdem man die linke Mausstaste gedückt hat, wird die Rücksprungadresse in das Register A5 geladen. Mit dem JMP-Befehl kehrt die Routine in das Betriebssystem zurück.

Ein Test mit RESET1 kann hier und da auch schiefgehen. Ein wichtiger Grund hierfür ist, daß das Programm nicht in die Memory-Lists eingetragen wird.

Memory-Verwaltung, kurz und knapp

Wenn man ein Programm startet, wird dieses vom DOS geladen.
Hierfür fordert das DOS von *Exec* Speicher an. In diesen Speicher, der jetzt als belegt gekennzeichnet ist, kopiert DOS das soeben gestartete Programm, berechnet die Sprungtabellen und startet das Programm. Für die Dauer der Laufzeit ist der Speicher, in dem das Programm liegt, als belegt deklariert. Sowie das Programm aber beendet wird (üblicherweise durch RTS), wird der von dem Programmcode belegte Speicher wieder in die Liste des freien Speichers aufgenommen. Falls ein anderer Prozeß Speicher anfordert oder ein neues Progamm gestartet wird, ist die Wahrscheinlichkeit, daß der eben freigegebene Speicher wieder überschrieben wird, recht groß.
Mit einem *MemClear*-Programm (PD) kann man dieses Überschreiben absichtlich veranlassen.

Auf diese Art kann es passieren, daß unser Reset1-Programm gleich wieder gelöscht wird. Nun ist der Programmcode nicht mehr vorhanden, aber die Vektoren in *ExecBase* sind noch auf die alten

Die Copperlist einer gewöhnlichen Workbench

```
Begin
0001d058 WAIT (00,2b)          * Warten bis Elektronenstrahl
0001d05c MOVE 005a,color[0]    * oben ist; Farbe 0 setzen
0001d060 MOVE 0fff,color[1]    * Farbe 1 setzen
0001d064 MOVE 0002,color[2]    * Farbe 2
0001d068 MOVE 0f80,color[3]    * Farbe 3
0001d06c MOVE 0000,color[16]   * Mausfarbe 0
0001d070 MOVE 0d22,color[17]   * Mausfarbe 1
0001d074 MOVE 0000,color[18]   * Mausfarbe 2
0001d078 MOVE 0fca,color[19]   * Mausfarbe 3
0001d07c MOVE 0444,color[20]   * ..
0001d080 MOVE 0555,color[21]   * ..
0001d084 MOVE 0666,color[22]   *
0001d088 MOVE 0777,color[23]   *
0001d08c MOVE 0888,color[24]   *
0001d090 MOVE 0999,color[25]   *
0001d094 MOVE 0aaa,color[26]   *
0001d098 MOVE 0bbb,color[27]   *
0001d09c MOVE 0ccc,color[28]   *
0001d0a0 MOVE 0ddd,color[29]   *
0001d0a4 MOVE 0eee,color[30]   *
0001d0a8 MOVE 0fff,color[31]   *
0001d0ac MOVE 0581,diwstrt     *
....
....
0001d0e4 MOVE a200,bplcon0     *
0001d0e8 WAIT (6f,ff)          * Ende der Copperlist
```

Ende der disassemblierten Workbench-Copperlist

RESET, erster Akt

```
*********************************************************
*                      RESET 1                          *
* Resetroutine über den ColdCapture-Vektor von ExecBase.*
* Programm wird nicht in die Memorylist eingetragen.    *
* Frank Schäfer & Kickstart in 88/89                    *
* Für den DevPac-Assembler                              *
*********************************************************

        opt  d+           * Debugger möglich machen
        incdir "DevPac:Include/"   * Wo sind die Includes
        INCLUDE exec/execbase.i    * ExecBase wird benötigt
        INCLUDE exec/exec_lib.i    * SysBase von hier
        INCLUDE RESETMACROS        * Die eigenen Macros

* Das eigentliche Programm zum Eintragen der Resetroutine

        move.l  _SysBase,a6        * ExecBase nach a6
        clr.l   CoolCapture(a6)    * CoolCapture löschen
        move.l  #reset,ColdCapture(a6) * ResetRoutine
        PRUEFSUMME                 * Prüfsumme
        rts                        * Ins CLI zurück


* Die folgende Routine wird beim Reset angesprungen

reset:  move.l  a5,Zurück          * Hierhin zurück-

                                   * springen
        move.l  _SysBase,a6        * ExecBase nach a6
        move.l #reset,ColdCapture(a6) * Resetroutine

        DoSomething                * Eigenes Programm

        move.l  Zurück,a5          * Rücksprungadresse
        jmp     (a5)               * In Exec-Resetroutine

Zurück  dc.l    0                  * Rücksprungadresse

        END
```

Adressen gesetzt. Dort steht aber nicht mehr Reset1, sondern ein anderes Programm bzw. Teile davon. Bei einem Reset führt das mit Sicherheit zu einem Systemabsturz.
Mit den Memory-Lists gibt es aber noch ein weiteres Problem: Wenn unser Programm bis zum nächsten Reset "überlebt" und dann auch ordnungsgemäß abgearbeitet wird, ist der Speicher, in dem es liegt, immer noch freigegeben. Denn während des Resetvorganges wird der gesamte belegte Speicher wieder freigegeben. Bei nächster Gelegenheit kann es wieder überschrieben werden.

Es stellt sich nun die Frage, wie man nach dem Reset ein Programm wieder in die Memory-List eintragen kann. Hierzu schauen wir uns den Verlauf der Resetroutine des Betriebssystemes noch einmal an.

RESET 3 ist nun ein Programm, mit dem die Routine nach dem Reset wieder in die Memorylist eingetragen wird. Hierzu sind nur drei Zeilen notwenig. In der der Prozedur "memory" wird zuerst die Größe des zu belegenden Speicherbereiches nach D0 geschrieben.
Anschließend wird die Anfangsadresse des zu belegenden Bereiches nach a1 geladen. Die Routine *AllocAbs* belegt nun den gewünschten Speicherbereich, sofern dieser noch frei ist. Falls das Belegen nicht funktioniert, ist keine Fehlermeldung vorgesehen; dies hat einen ganz einfachen Grund, mir ist nichts eingefallen was man anstatt tun könnte, außer vielleicht einem Alert? Praktisch ist es auch nicht notwendig, eine Fehlerabfrage zu machen, da zu diesem Zeitpunkt im System noch der komplette Speicher frei ist, wenn man von dem Speicher absieht, der vom Betriebssystem benötigt wird. Falls das Belegen nicht funktioniert, gibt *Alloc Abs* in D0, Null zurück, eine kleine Fehlerroutine kann sich ja (fast) jeder selbst einbauen. Die Routine "memory" wird, wie schon gesagt, in den CoolCapture-Vektor eingebunden. Die Routine "reset", schon aus RESET 1 bekannt, wird durch den ColdCapture-Vektor angesprungen.

```
*************************************************************
*            Macros für die Resetroutinen                   *
* 1. PRUEFSUMME: Berechnet die LowMemChkSum für ExecBase    *
* 2. DoSomething: Eine kleine Farbspielerei, die als        *
*                 Programm genutzt wird, welches Reset      *
*                 gemacht werden soll                       *
* Frank Schäfer & Kickstart in 88/89                        *
* Für den DevPac-Assembler                                  *
*************************************************************


* Errechnen der ExecBase-Prüfsumme

PRUEFSUMME  macro

        movem.l d0-d1/a0/a6,-(sp)    * Register sichern
        moveq   #0,d1                * d1 für Summe löschen
        move.l  _SysBase,a6          * ExecBase nach a6
        lea.l   ChkBase(a6),a0       * Adresse von ChkBase -> a0
        move.w  SoftVer(a6),d1       * SoftVer erster Summand
        moveq   #$16,d0              * Über die nächsten 16 Worte
sum:    add.w   (a0)+,d1             * Prüfsumme bilden
        dbra    d0,sum               * Fertig?
        not.w   d1                   * Prüfsumme negieren
        move.w  d1,LowMemChkSum(a6)  * Prüfumme abspeichern
        movem.l (sp)+,d0-d1/a0/a6    * Register setzen

        endm


* Irgendeine kleine Spielerei oder ein eigenes Programm

DoSomething macro

        moveq   #0,d0           * d0 löschen
Maus:   move.w  d0,$dff180      * Hintergrundfarbe setzen
        add.w   #1,d0           * Nächste Farbe
        cmp.w   #$FFF,d0        * Alle 4096 Farben durch?
        bne.s   Maus            * Nein, dann zu weiter
        moveq   #0,d0           * Ja, -> d0 löschen
        btst    #6,$bfe001      * linke Maustaste?
        bne.s   Maus            * Nein, dann zu Label Maus
                                * Ja, weiter im Programm
        endm
```

Kickstart-Resetroutine, Teil 2

```
.
.
.
FC0500lea    ResTabelle,a0          * Suchtabelle
bsr    MakeResident_List            * Liste erstellen
move.l d0,ResModules(a6)            * Liste abspeichern
         bclr.b #1,BFE001           * LED an (hell)
         move.l CoolCapture(a6),d0  * CoolCapture -> D0
beq.s  NoCoolCapture                * Kein Vektor
         move.l d0,a0               * CoolCapture -> a0
         jsr     (a0)               * Anspringen CoolC.
NoCoolCapture
         moveq   #1,d0              * Flag = Kaltstart
         moveq   #0,d1              *. Keine best. Version
         bsr     InitCode           * DOS-hochfahren
         move.l  WarmCapture(a6),a0 * -> Fehler beim
         beq.s   NoWarmCapture      * DOS starten
         move.l  d0,a0              * hier weitermachen
         jsr     (a0)               * Warmcapture
NoWarmCapture                       * anspringen
         moveq   #D,d0              * Schleifenzähler
Clear    clr.l   -(pc)              * 14 Bytes auf
         dbra    d0,Clear           * dem Stack löschen
         movem.l (a7)+,d0-d7/a0-a5  * Register löschen
DebugSchleife
         jsr      Debug(a6)         * Debug anspringen
         move.l   ExecBase,a6       * Exec nach a6
         bra.s    DebugSchleife     * neu debuggen
```

Wie man sieht, wird im weiteren Verlauf der Resetroutine noch ein weiterer Vektor angesprungen, der CoolCapture-Vektor. Er wird am Ende der Resetroutine angesprungen. Zu diesem Zeitpunkt ist im System praktisch alles initialisiert (Stack, Libraries etc.). Es ist infolgedessen möglich, über diesen Vektor einen

Die Lösung heißt "kopieren"

Ein schon angesprochenes Problem wartet aber immer noch auf seine Lösung. Wie kann man verhindern, daß ein Programm, nachdem es beendet wurde, wieder überschrieben werden kann. Es wäre möglich, das Programm selbst in den Speicher zu laden (mit LoadSeg), oder man könnte das Programm gar nicht beenden und den Task weiterlaufen lassen. Eine weitere Lösung des Problems ist das Programmteil, welches resetfest gemacht werden soll, in einen Speicherbereich zu kopieren, den man vorher belegt hat. In dem Programm RESET 4 wird die zuletzt genannte Lösung angewendet.

Hierzu wird RESET 3 um "copy", die Kopierroutine, und um "Ziel", eine Variable, in der abgespeichert wird, wohin kopiert wurde, ergänzt. Weiterhin müssen die eingebundenen Programme jetzt PC-relativ programmiert sein. Hierzu noch ein paar Anmerkungen.
Normalerweise sind AMIGA-Programme 'relokatibel', was so viel wie verschiebbar bedeutet. Ein solches Programm kann an jeder beliebigen Speicherstelle beginnen. Der Assembler sorgt dafür, daß ein Programm relokatibel ist. Dies geschieht, indem er (zusammen mit dem Programm) eine Tabelle aller absoluten Adressen (Offsets) ablegt. Das DOS, genaugenommen der Loader, kann dann aus dieser Tabelle die aktuellen Adressen bestimmen. Da wir aber einen Teil des Programmes einfach im Speicher umkopieren, stimmen die berechneten Adressen nicht mehr. Folglich darf man in dem Teil des Programmes, der kopiert wird, keine Adressierung verwenden, die der DOS-Loader korrigiert hat. Dies geschieht, indem man die Variablen PC-relativ anspricht. Dies bedeutet, daß man zu dem aktuellen Stand des ProgrammCounters einen Offset addiert und so die entsprechende Adresse anspricht.

In "copy" wird zuerst mittels *AllocMem*, ein Speicherbereich belegt, in den die Routinen kopiert werden sollen. Falls das Speicherbelegen nicht erfolgreich war, läßt "NoMem" den Bildschirm kurz blinken und beendet dann das Programm; hier sollte man sinnvollerweise zweiten Einsprung vorzunehmen und die Memory-Liste entsprechend zu manipulieren.
Zuvor aber erst einmal ein Progamm, RESET2, das den CoolCapture-Vektor benutzt, um die Routine "DoSomething" in den Reset einzubinden.

```
*********************************************************
*                      RESET 2                          *
* Resetroutine über den CoolCapture-Vektor von ExecBase.*
* Programm wird nicht in die Memorylist eingetragen.    *
* Frank Schäfer & Kickstart in 88/89                    *
* Für den DevPac-Assembler                              *
*********************************************************

        opt  d+                 * Debugger möglich machen

        incdir "df1:Include/"   * Wo sind die Includes
        INCLUDE exec/execbase.i * ExecBase wird benötigt
        INCLUDE exec/exec_lib.i * SysBase von hier
        INCLUDE RESETMACROS     * Die eigenen Macros

* Das eigentliche Programm zum Eintragen der Resetroutine

        move.l  _SysBase,a6             * ExecBase nach a6
        clr.l   ColdCapture(a6)         * ColdCapture löschen
        move.l  #reset,CoolCapture(a6) * ResetRoutine
        PRUEFSUMME                      * Prüfsumme
        rts                             * Ins CLI zurück


* Die folgende Routine wird beim Reset angesprungen

reset:  movem.l  d0-d7/a0-a6,-(sp)      * Register retten

        DoSomething                     * Eigenes Programm

        movem.l (sp)+,d0-d7/a0-a6       * Register setzen
        rts                             * In Exec-Resetroutine
        END
```

Wie man sieht, entspricht RESET 2 in weiten Teilen RESET 1, nur der CoolCapture-Vektor wird gesetzt und ColdCapture wird gelöscht.
Die Prüfsumme wird wie in RESET 1 berechnet. In der Routine, auf die der Vektor zeigt, hat sich auch etwas geändert. Da der Stack verfügbar ist, werden die Register auf den Stack gelegt, *DoSomething* abgearbeitet und dann, nach dem Setzen der Register, mit RTS in das Betriebssystem zurückgesprungen.

```
*********************************************************
*                      RESET 3                          *
* Resetroutine über den ColdCapture & CoolCapture-Vektor*
* von ExecBase. Programm wird in die Memorylist         *
* eingetragen.                                          *
* Frank Schäfer & Kickstart in 88/89                    *
* Für den DevPac-Assembler                              *
*********************************************************

        opt  d+                 * Debugger möglich machen

        incdir "df1:Include/"   * Wo sind die Includes
        INCLUDE exec/execbase.i * ExecBase wird benötigt
        INCLUDE exec/exec_lib.i * SysBase von hier
        INCLUDE RESETMACROS     * Die eigenen Macros

* Das eigentliche Programm zum Eintragen der Resetroutine

        move.l  _SysBase,a6             * ExecBase nach a6
        move.l  #memory,CoolCapture(a6) * Memory belegen
        move.l  #reset,ColdCapture(a6)  * ResetRoutine
        PRUEFSUMME                      * Prüfsumme best.
        rts                             * Ins CLI zurück

* Die Memoryroutine wird vom CoolCapture-Pointer aufgerufen

memory: move.l   #resetEnd-memory,d0    * Speicherbereichgröße
        lea.l    memory,a1              * Adresse des Breiches
        CALLEXEC AllocAbs               * Speicher belegen
```

den Benutzer mit einem Alert warnen. Aber aus Komplexitätsgründen habe ich darauf verzichtet. Aus dem gleichen Grund wurde in den Programmen auch auf Dinge verzichtet wie lokale Labels oder komplexe Ablaufsteuerungen. Nun werden die Register mit den Startwerten für die CopyMem-Funktion geladen, die den Speicherbereich von der *"memory"*-Routine bis hin zu *"resetEnd"* in den soeben belegten Speicherbereicht kopiert. In Ziel ist die Startadresse dieses Bereiches abgelegt worden. Diese Adresse wird als CoolCapture-Adresse abgespeichert. Zu ihr wird dann der Offset der Cold-Capture-Routine *"reset"* addiert und als ColdCapture-Vektor abgespeichert. Anschließend wird wie gehabt die Prüfsumme berechnet und das Programm beendet.
Nun kann jeder seine eigenen resetfesten Programme schreiben.

Probleme...

... kann es natürlich auch geben, hier noch ein paar Hinweise dazu. Zur Zeit gibt es eine ganze Menge "AntiViren-Programme"; diese überprüfen meist auch die benutzten Vektoren und könnten der Meinung sein, daß ein eigenes Programm ein Virus ist. Bei nicht richtig in das System eingebundenen Speichererweiterungen gibt es ein weiteres Problem, bei Tests auf anderen Rechnern machte sich die sonst einwandfreie MEGABOX hier unangenehm bemerkbar. (Diese Erweiterung sollte den Benutzer dann auch vor dem SCA-Virus schützen, der Byte Bandit-Virus benutzt den *KickTagPtr*, der heute nicht besprochen wurde.) Ein Ausschalten der Erweiterung reichte aber aus, um das Programm zum Laufen zu bringen. Anstatt die Erweiterung auszuschalten, kann man das Programm ins CHIP-Memory kopieren. Dazu ist nicht mehr notwendig, als beim Speicherbelegen *"#2 = CHIP-Memory"* anzugeben. Bei manchen AMIGA 2000-Modellen gab es bei der Ausführung der Routine hier und da Abstürze, die aber nicht reproduzierbar waren. Auch traten diese Fehler nur sehr selten auf, dürften also folglich nicht von besonderer Bedeutung sein.

Systemkonform verhalten sich die Programme RESET 1 bis 4 nicht gerade, denn sie überschreiben einfach Vektoren, die vielleicht schon eingetragen

```
        rts                             * Nach Exec zurück

* Die folgende Routine wird beim Reset, ColdCapture-Vektor,
* angesprungen

reset:  move.l  a5,Zurück               * Hierhin zurückspringen
        move.l  _SysBase,a6             * ExecBase nach a6
        move.l  #reset,ColdCapture(a6)  * Resetroutine

        DoSomething                     * Eigenes Programm

        move.l  Zurück,a5               * Rücksprungadresse
        jmp     (a5)                    * In Exec-Reset-Routine

Zurück  dc.l    0                       * Rücksprungadresse

resetEnd:                               * Speicher bis hier belegen
```

```
*********************************************************
*                      RESET 4                          *
* Resetroutine über den ColdCapture & CoolCapture-Vektor *
* von ExecBase. Programm wird in die Memorylist          *
* eingetragen.                                           *
* Frank Schäfer & Kickstart in 88/89                     *
* Für den DevPac-Assembler                               *
*********************************************************

        opt  d+                     * Debugger möglich machen

        incdir "DevPac:Includes"    * Wo sind die Includes
        INCLUDE exec/execbase.i     * ExecBase wird benötigt
        INCLUDE exec/exec_lib.i     * SysBase von hier
        INCLUDE RESETMACROS         * Die eigenen Macros

* Das eigentliche Programm zum Eintragen der Resetroutine

        move.l  _SysBase,a6         * ExecBase nach a6
        jsr     copy                * Speicher belegen
        move.l  Ziel,a0             * Speicheradresse
        move.l  _SysBase,a6         * ExecBase nach a6
        move.l  a0,CoolCapture(a6)  * Memory belegen
        adda.l  #reset-memory,a0    * Offset bestimmen
        move.l  a0,ColdCapture(a6)  * Resetroutine
        PRUEFSUMME                  * Prüfsumme errechnen
        rts                         * Ins CLI zurück

NoMem:  addq.l  #4,a7               * Stack korrigieren
        move.w  #$FFF,$dff180       * Blinken
        rts     * zurück

* Die Memoryroutine wird vom CoolCapture-Pointer
* aufgerufen

copy:   move.l  #resetEnd-memory,d0 * Wie groß ist der
                                    * resetfeste Teil
        moveq   #2,d1               * Speicherart CHIP-Mem
        CALLEXEC AllocMem           * Speicher belegen
        move.l  d0,Ziel             * Adresse sichern
        beq     NoMem               * Kein Speicher verfügbar
        lea.l   memory,a0           * Von wo
        move.l  d0,a1               * Nach wo
        move.l  #resetEnd-memory,d0 * Wie groß
        CALLEXEC CopyMem            * Kopieren
        rts                         * Und fertig



* Die Memoryroutine wird vom CoolCapture-Vektor aus
* angesprungen und erklärt den benutzten Speicher
* als belegt.

memory: movem.l d0-d7/a0-a6,-(sp)   * Register retten
        move.l  #resetEnd-memory,d0 * Speicherbereichgröße
        lea.l   Ziel(pc),a1         * Adresse der Variable
        move.l  (a1),a1             * Adresse des Breiches
        CALLEXEC AllocAbs           * Speicher belegen
        movem.l (sp)+,d0-d7/a0-a6   * Register sichern
        rts                         * Nach Exec zurück
```

sind. Für kommerzielle Anwendungen sollte man sich den schon eingetragenen Vektor merken und nach der Ausführung der eigenen Routine mit "JSR" bzw. "JMP" (Rückkehradresse in A5!) anspringen.
So kann man einige Programme resetfest halten.

```
* Die folgende Routine wird beim Reset, ColdCapture-Vektor,
* angesprungen

reset:  lea.l   Zurück(pc),a0          * Adr von Zurück -> a0
        move.l  a5,(a0)                * Rücksprungadr -> Zurück
        move.l  _SysBase,a6            * ExecBase nach a6
        lea.l   reset(pc),a0           * Anfangsadr von reset!
        move.l  a0,ColdCapture(a6)     * Resetroutine einbinden!

        DoSomething                    * Eigenes Programm

        lea.l   Zurück(pc),a0          * Adr von Zurück
        move.l  (a0),a5                * Rücksprungadr -> a5
        jmp     (a5)                   * Zur Exec-Resetroutine

Zurück  dc.l    0                      * Rücksprungadresse
Ziel    dc.l    0                      * Nach wo wurde kopiert

resetEnd:                              * Speicher bis hier belegen

        END
```

ENDE

Animate-Turbo-Board III

Unser bestes,
ist das Animate-Turbo-Board III, ein Einsteckboard für den Amiga 2000.

Die Merkmale sprechen für sich.
- einsteckbar in den MMU-Slot.
- 32 Bit Kickstart-Port.
- 32 Bit Peripherie-Port.
- 32 Bit Static-RAM 256K-1M on board.

Turbo-Board I:

Animate-Turbo-Board I **mit** MC68020, MC68881 14 MHz	**1099,-**
Animate-Turbo-Board I **mit** MC68020, MC68881 16 MHz	**1249,-**
Animate-Turbo-Board I **mit** MC68020, MC68882 16 MHz	**1439,-**
Animate-Turbo-Board I **ohne** Prozessoren	**499,-**

Turbo-Board III:

Animate-Turbo-Board III **mit** MC68020, MC68881 14 MHz ohne Ram	**1649,-**
Animate-Turbo-Board III **mit** MC68020, MC68881 16 MHz ohne Ram	**1799,-**
Animate-Turbo-Board III **mit** MC68020, MC68882 16 MHz ohne Ram	**1999,-**
Animate-Turbo-Board III **ohne** Prozessoren ohne Ram	**999,-**
Animate-Turbo-Board III RAM-Chip Satz 256 K	auf Anfrage
Animate-Turbo-Board III Ram-Chip Satz 1 M	auf Anfrage

Prozessoren:

Prozessor MC68020-12E	**390,-**
Co-Prozessor MC68881-12B	**290,-**
Co-Prozessor MC68881-16B	**440,-**
Co-Prozessor MC68882-16A	**640,-**

Entwicklung – Beratung – Verkauf

HARMS Computer-Systeme
Harsefelder Str. 18 · 2800 Bremen 1 · Tel.: 04 21 / 44 47 90
Bestellservice: Montag – Freitag 9 – 12 und 15 – 18 Uhr

Norbert Dornhofer & Michael Böttcher GbR
DFÜ-SHOP
Discovery 1200 C+
Hayes-Kompatibles 1200 Baud-Modem, 300,1200 Baud Vollduplex, CCITT (V21,V22), Bell (103,212A), eingebauter Lautsprecher, Auto-Dial, -Answer, -Redial u.v.m.
Preis inkl. Steckernetzteil nur DM 279.-
Discovery 1200H (PC/IT-Karte) DM 198.-
Discovery 2400 C
2400 Baud-Modem, sonst wie 1200C+ DM 449.-
Ext. ATARI-Drive (NEC 1037A, abschaltbar) DM 279.-
Ext. AMIGA-Drive (NEC 1037A, abschaltbar,Bus) DM 259.-
AMIGA-Drive (NEC 1037A, f.A2000, intern) DM 229.-
VIRUS FORSCHER SET DM 39.-
PUBLIC-DOMAIN-SERVICE BERLIN
Trackanzeige DM 79.-
DFÜ-SHOP Norbert Dornhofer & Michael Böttcher GbR Kolonnenstraße 33
1000 Berlin 62 • Tel.: 030 782 71 18
(★) Anschluß ans öffentl. Telefonnetz ist strafbar!

PUBLIC DOMAIN CENTER
Postfach 3142
5840 Schwerte
über 2000 Disk vorrätig!
Jede nur:
2,60 DM
(3,5 Zoll, 2 DD)
Amiga
Info anfordern!

1 First Public Domain Shop Koblenz
Wir haben die größte Public Domain Software Serie für alle AMIGAs (über 1800 D.)

Die Lieferung erfolgt auf
SONY 3.5 MFD-2DD pro Stück 6,– DM
ab 10 pro Stück 5,50 DM
Auf NoName Disketten pro Stück 5,– DM
Auf 5.25" Disketten pro Stück 5,– DM
Rabatt bei höheren Stückabnahmen

Leerdisketten
10 SONY 3.5" MFD-1DD 25,– DM
10 SONY 3.5" MFD-2DD 35,– DM
10 SONY 5.25" MD-2D 25,– DM
10 SONY 5.25" MD-2DD 33,– DM
10 NoName 3.5" 2DD 24,– DM

Super Laufwerke für alle Amigas
3.5 Zoll AMIGA-farben 300,– DM
5.25 Zoll Laufwerk 350,– DM

Auch Versand möglich.
Public Domain für ATARI ST, MAC u. PC in Vorbereitung

First Public
Stegemannstr. 21
5400 Koblenz
Tel. 02 61 / 3 31 92

VON ANDREAS KRÄMER

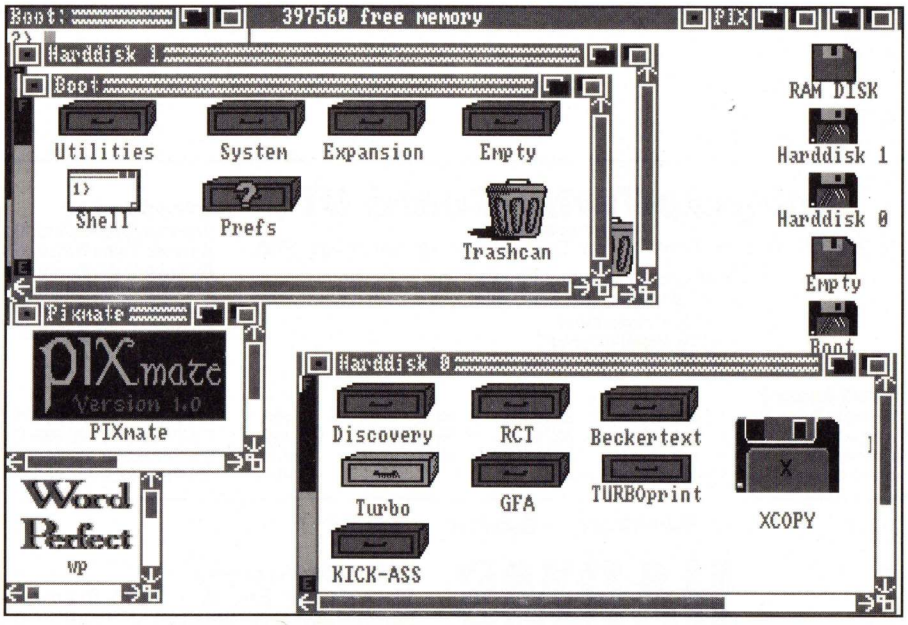
Die Oberflächen des AMIGA

Die verschiedenen Oberflächen

Jeder Computer braucht eine Schnittstelle, um mit einem Benutzer kommunizieren zu können, eine sogenannte Benutzeroberfläche. Grundlegend unterscheidet man zur Zeit zwei verschiedene Typen.

Auf der einen Seite sind die grafikorientierten Oberflächen zu nennen. Dazu zählt unter anderem die Workbench, aber auch GEM, das auf dem ATARI ST oder auch auf dem PC Verwendung findet. Das Haupteingabemedium stellt bei diesen Oberflächen die Maus dar, mit deren Hilfe Befehle an den Rechner weitergegeben werden können. Sie werden deshalb als grafikorientierte Oberflächen bezeichnet, weil viele grafische Elemente implementiert sind. Laufwerke sind beispielsweise als mehr oder weniger kleine Piktogramme auf dem Bildschirm zu sehen.

Auf der anderen Seite sind die konservativen Oberflächen zu nennen. Zu dieser Kategorie zählt das CLI, der Command Line Interpreter, zu deutsch Befehlszeilenübersetzer. Bei einer solchen Oberfläche werden Befehle über die Tastatur in einer Zeile eingegeben, der Computer interpretiert diese Zeile und führt daraufhin die nötigen Schritte aus. Für das CLI werden des weiteren noch etliche Shells angeboten. Shell heißt auf deutsch Muschel, aber treffender ist die Bezeichnung Umgebung. In der Regel machen die Shells die Bedienung der Oberfläche einfacher und komfortabler. Am Grundprinzip ändert sich hingegen



Die Workbench ist eine grafikorientierte Benutzeroberfläche. Das Grundeingabemedium ist die Maus.

Der AMIGA besitzt zwei unterschiedliche Benutzeroberflächen: dieWorkbench und das CLI - den Befehlszeilen-Interpreter. Beide Oberflächen haben sowohl Vor- als auch Nachteile. Besonders AMIGA-Anfänger haben so ihre Schwierigkeiten mit den beiden Oberflächen. Die Unterschiede und die verschiedenen Möglichkeiten möchte ich für Sie durchleuchten.

nichts. Befehle müssen dem Rechner weiterhin mit der Tastatur mitgeteilt werden. Die neue Workbench 1.3 beinhaltet eine solche Shell.

Die Sonderstellung des AMIGA

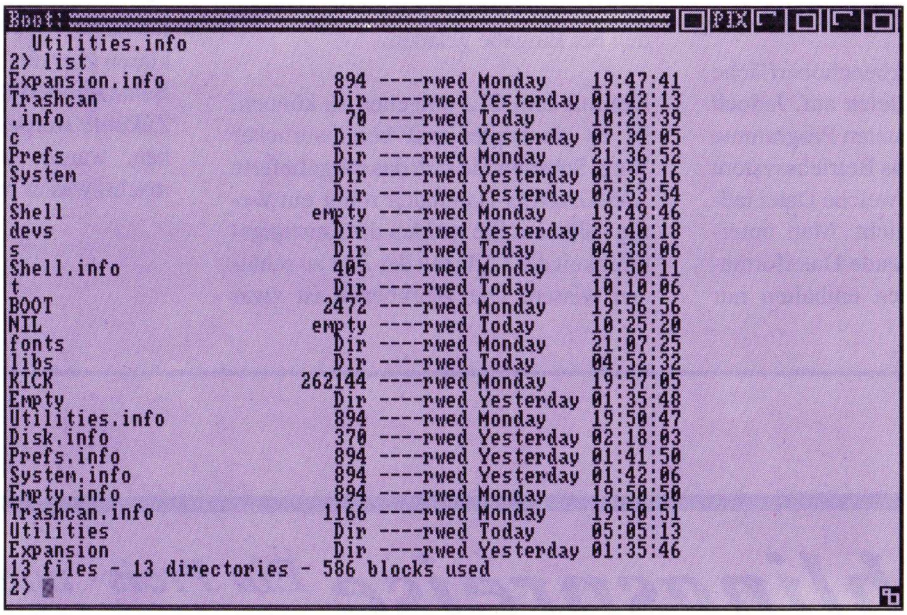
Das Betriebssystem des AMIGA ist multitaskingfähig. Das bedeutet, daß mehrere Programme gleichzeitig ablaufen können. Eine Oberfläche ist nichts anderes als ein Programm, und da Programme gleichzeitig ablaufen können, ist der AMIGA in der Lage beide Oberflächen in Betrieb zu haben, das CLI und die Workbench.
Der Anwender kann schnell und einfach zwischen beiden wechseln.

Die Workbench

Anfänger werden das CLI zunächst nur wenig beachten und ihr Augenmerk mehr auf die Workbench richten. Diese Oberfläche wird automatisch nach Einlegen der mitgelieferten Diskette mit dem Namen Workbench geladen. Beim Ladevorgang können Sie das CLI kurz betrachten, das aber nach geringer Ladezeit wieder geschlossen wird.

Die Workbench zeigt sich in der Regel auf blauem Untergrund. Eingelegte Disketten sind als Piktogramme (Icons, kleine Bilder) auf dem Bildschirm zu sehen. Drückt der Anwender die rechte Maustaste, erscheinen drei Pull-Down-Menüs. Durch Anwählen eines Menüpunktes kann Verschiedenes bewirkt werden. Drückt der Anwender zweimal die linke Maustaste über einem Piktogramm, wird der Disketteninhalt innerhalb eines Fensters angezeigt. Die einzelnen Programme oder Ordner sind jetzt wiederum als Piktogramme zu sehen. Durch einen doppelten Klick können die Programme dann gestartet werden, bei einem Ordner wird der Inhalt des Ordners angezeigt. Eines sollte sich der Computeranwender vor Augen halten: Piktogramme sind nicht gleichzusetzen mit startbaren Programmen. Viele Programme erzeugen Piktogramme für ihre Programmdaten. Solche Piktogramme sind meistens nicht startbar.

Einer der größten Nachteile der Workbench besteht darin, daß nur die Programme in einem Disketten- oder Ordnerfenster zu sehen sind, die eine bestimmte Datei besitzen. Die Datei beinhaltet alle Informationen des Piktogramms, also die Bilddaten und bestimmte Zuweisungen. Diese Dateien sind alle mit der Extansion '.info' versehen. Dateien, die kein dazugehöriges '.info' besitzen, sind demnach nicht auf der Workbench sichtbar, also auch nicht von dieser aufrufbar. Das führt leider oft zu der Annahme, daß eine Diskette leer sei, (weil kein Piktogramm im Fenster sichtbar ist), obwohl sie randvoll ist. Um an die Programme 'heranzukommen', muß der Weg über das CLI beschritten werden. Anwender tun sich aber mit dieser zweiten Oberfläche erheblich schwerer als mit der Workbench, besonders wenn nur ein Diskettenlaufwerk zur Verfügung steht. Warum? Das werden Sie gleich erfahren.



Das CLI ist eine konservative Oberfläche, die ausschließlich über die Tastatur bedient wird.

Das CLI

Der Command Line Interpreter, kurz CLI, ist wie schon erwähnt die zweite Schnittstelle des Computers mit dem Benutzer. Das CLI-Programm, das die zweite Oberfläche einlädt, befindet sich im Systemordner der Workbenchdiskette und wird wie gewohnt aufgerufen. Es erscheint ein Fenster auf den Bildschirm mit einem Prompt, hinter dem die Befehle eingegeben werden müssen. Die Maus kann zur Seite gelegt werden, da dem Rechner alle Kommandos über die Tastatur mitgeteilt werden. Das CLI hat gegenüber der Workbench entscheidende Vorteile. Es ist wesentlich flexibler und komplexer als die Workbench. Allerdings ist es dadurch für Einsteiger wesentlich schwerer zu beherrschen. Der Nachteil, zumindest auf den ersten Blick, besteht darin, daß immer nach der Workbenchdiskette verlangt wird. Daraus resultiert, daß Besitzer mit nur einem Laufwerk das Nachsehen haben und ständig am Wechseln der Disketten sind. Ein permanentes Arbeiten mit dem CLI wird so fast unmöglich gemacht und treibt den AMIGA-Anwender zur Verzweiflung. Der Kauf eines zweiten Laufwerkes ist unumgänglich und sollte in die Haushaltsplanung unbedingt mit einbezogen werden. Doch warum verlangt das Betriebssystem immer nach der Workbenchdiskette? Das liegt daran, daß die Befehle sich nicht im Arbeitsspeicher des AMIGA befinden, sondern von Diskette erst eingelesen werden müssen. Alle Befehle befinden sich im Ordner 'C'. Von der Workbench aus bleibt der Ordner 'unsichtbar', wie andere System-Ordner auch - 'L', 'T', 'Devs' usw. Da die Befehle eingelesen werden, ist es natürlich leicht möglich, die Befehlsvielfalt beliebig zu erweitern. Sie müssen lediglich den neuen Befehl in den Ordner 'C' kopieren.

Besitzer mit nur einem Laufwerk können sich natürlich auch helfen lassen. Sie können z.B. den Befehlsordner, also

den 'C'-Ordner, auf die RAM-Disk kopieren und eine Zuweisung *path ram: add* bzw. *assign c: ram: c* machen. Im mitgelieferten Handbuch wird dieser Vorgang ausführlich beschrieben.

Noch ein kleiner Hinweis: Das Betriebssystem sucht den eingegebenen Befehl zuerst im aktuellen Verzeichnis und dann erst im *'C'-Ordner*. Mit dem *'PATH'*-Befehl können Sie die Suche auf andere Ordner leicht erweitern.

Anders als die Workbenchoberfläche listet das CLI alle Dateien auf. Jedoch sind nicht alle aufgelisteten Programme startbar. Leider sagt das Betriebssystem dem Anwender nicht, welche Datei ladbar ist und welche nicht. Man unterscheidet mehrere folgende Dateiformate. Reine Datendateien enthalten nur Daten und können nicht gestartet werden. Solche Dateien sind beispielsweise Bilder, Sounddateien oder die Piktogrammdateien (.info). Stapeldateien sind Dateien, die nur mit dem *'EXECUTE'*-Befehl gestartet werden können. Eine typische Stapeldatei ist die Startup-Sequence, die sich im Ordner 'S' der Workbench-Diskette befindet. Diese Stapeldatei wird übrigens beim Booten immer ausgeführt. Die letzte Kategorie sind die startbaren Programme, sie werden bei Eingabe geladen.

Um mit dem CLI umgehen zu können, ist es ein weiter und beschwerlicher Weg. Schauen Sie sich das mitgelieferte AMIGADOS-Handbuch ruhig ein wenig näher an. Sie werden die Leistungsfähigkeit des CLIs mit der Zeit zu schätzen wissen. Die Workbench ist zwar wesentlich einfacher zu bedienen, der Anfänger arbeitet sich wesentlich schneller ein, aber die Einschränkungen veranlassen den Anwender früher oder später, auch das CLI zu nutzen. Unter der Workbench 1.3 ist das CLI noch wesentlich leistungsstärker geworden. Die Bedienung wurde beispielsweise durch eine neue Shell (bedienerfreundliche Arbeitsumgebung) enorm gesteigert.

Ich hoffe, ich konnte Sie ein wenig aufklären über die Unterschiede der beiden Benutzeroberflächen des AMIGA. In Zukunft werden Sie vielleicht verstehen, warum der AMIGA nicht das macht, was er machen soll.

ENDE

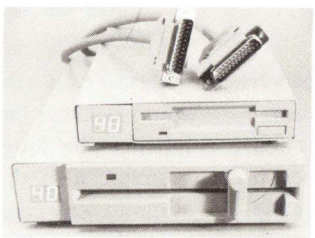
Frühlingspreise *Ein Preis- und Leistungsvergleich lohnt sich !*

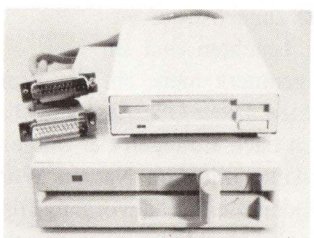
Laufwerke

komplett anschlußfertig mit Kabel, amigafarbenes Metallgehäuse, abschaltbar, 3 ms Stepprate, Busdurchführung bis df3:, DiskChange wird erkannt, 5.25"-TEAC-Laufwerke mit 40/80 Trackumschaltung,

SDN 3.5 mit NEC 1037A - Markenlaufwerk	229.-
SDN/DA 3.5 mit NEC 1037A und Busdurchführung	239.-
SDN 3.5 mit NEC 1036A - Markenlaufwerk	219.-
SDN 3.5 intern NEC 1036A für A2000, Komplettkit	219.-
SDN 5.25 mit TEAC FD 55 - Markenlaufwerk	269.-
SDN/DA 5.25 mit TEAC FD 55 und Busdurchführung	279.-

alle 5.25" TEAC-Drives können ohne Aufpreis auch mit 5.25" NEC-Drives ausgeliefert werden.





Festplatten

komplett anschlußfertig mit Controller, bereits formatiert und installiert, serienmäßige Auslieferung mit A.L.F. Wir verwenden ausschließlich Qualitätsfestplatten von SEAGATE und NEC, Fast-File-System fähig

30 MB A2000	745.-	50 MB FileCard	1295.-
60 MB A2000	1195.-	80 MB A500	1599.-
30 MB A500	945.-	60 MB A500	1399.-
80 MB A500 mit NEC Harddisk	1695.-	alle A500 Festplatten auch für A1000 lieferbar	

Sollten Sie spezielle Wünsche haben, rufen Sie uns an ! Wir liefern ganz nach Ihren Wünschen 5.25" und 3.5" Festplatten. Erfragen Sie die aktuellen Tagespreise !

Stalter Computerbedarf GmbH
Gartenstr. 17 - 6670 St. Ingbert

☎ **06894/2012**

Einkaufsführer

Hier finden Sie Ihren Commodore/Amiga Fachhändler

1000 Berlin

HD AMIGA Hardware Spezialist
Computertechnik
Sämtliche verfügbaren KICKSTART-PD's vorrätig!
Und über 1500 weitere PD's für den AMIGA!
1000 Berlin 65 · Pankstr. 42
Tel. 030/465 70 28-29

COMPUTER-STUDIO
Schlichting
... die etwas andere Computerei
ATARI-Fachmarkt
MS-DOS Fachmarkt · NEC-Fachhandel
Katzbachstraße 8 · 1000 Berlin 61
☎ 030/786 43 40

RUNOW
Büroelektronik
Keithstraße 26 · 1000 Berlin 30
☎ 26 111 26

SOFTPOWER
Das Software-Paradies im Norden Berlins!
1000 Berlin 65
Schwedenstr. 18c
Tel.: 030/492 20 56
Mo-Fr. 11-18.30 * Sa. 10-14
Daily News !!!
IBM

2000 Hamburg

Bit Computer Shop
Osterstraße 173 · 2000 Hamburg 20
Telefon: 040/494400
Createam
Computer Hard & Software
Bramfelder Chaussee 300 · 2000 Hamburg 71
Telefon: 040/6416473+6416861

GMA mbH
Systemhändler
Wandsbeker Chaussee 58
2000 Hamburg 76

2000 Hamburg

Computer
Hardware · Software · Zubehör
Lilienstraße 32
(beim Mönckebergbrunnen)
2000 Hamburg 1
Tel. (040) 33 67 08
SYSTEMSHOP ®

Computer & Zubehör-Shop
Gerhard u. Bernd Waller GbR
Kieler Straße 623
2000 Hamburg 54
☎ 040/570 60 07
BTX 040 570 52 75

2160 Stade

BERGAU
Büromaschinen · EDV-Systeme
Neue Straße 5 · 2160 Stade
Telefon (04141) 2364 u. 2384

2300 Kiel

Hardware
Software
Service
Home Computer Laden
Spezialisiert auf Public Domain
Immer die neueste Software auf Lager
Gutenbergstraße 5 · 2300 Kiel · Tel. (04 31) 55 55 55

2390 Flensburg

ecl electronic computer laden ohg
Norderstraße 94-96 · D-2390 Flensburg
☎ (0461) 28181 + 28193

2900 Oldenburg

GOLDT
Computerhaus
Donnerschweer Straße 127-129
(gegenüber Weser Ems Halle)
2900 Oldenburg
Telefon (04 41) 88 47 06

2940 Wilhelmshaven

Radio Tiemann
Commodore-Systemfachhändler
Marktstraße 52
2940 Wilhelmshaven
Telefon (0 44 21) 2 61 45

3000 Hannover

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

Fun Action **CSJ** Fun Action
Hannover's Softwarethek Nr. 1
DIE AUSWAHL an Software für C 16, C 64 / 128, Amiga, Atari ST u. IBM PC.
An der Tiefenriede 27 · 3000 Hannover 1
Tel. 05 11/88 63 83

3500 Kassel

Hermann Fischer GmbH
Commodore-Systemfachhändler
Rudolf-Schwander-Str. 5-13
3500 Kassel
Telefon (05 61) 70 00 00

4500 Osnabrück

B.O.S.S.
Computerhaus
Am Berliner Platz
Goethering 3
4500 Osnabrück
Telefon (05 41) 2 65 70

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

6000 Frankfurt

COMPUTER bco
Büro-Computer + Organisations GmbH

Oederweg 7-9
6000 Frankfurt/Main 1
☎ (0 69) 55 04 56 - 57

Commodore OKI ATARI TOSHIBA

6200 Wiesbaden

Luisenstraße 47
6200 Wiesbaden
Telefon (0 61 21) 56 20 91

DTM
Computershop GmbH

AUTORISIERTER
COMMODORE
SYSTEM-HÄNDLER
Commodore

COMPUTER TREFF

Computerbedarf, PD und
Software für
ATARI, AMIGA, PC

Nettelbeckstraße 12
6200 Wiesbaden
Tel. (06121) 40 43 02

6380 Bad-Homburg

PDC GmbH
Produkte u. Details Computerverband

Louisenstraße 115
Ladenpassage Alter Bahnhof
6380 Bad-Homburg
Telefon (0 61 72) 2 47 48

Autorisierter Commodore-Systemhändler!

6450 Hanau

JET Computersysteme

Amiga Hard- und Software
Business, BTX, Musik, Grafik und Spiele

JET Computersysteme
Inh. E. Hinrichsen
Hospitalstr. 6, 6450 Hanau, Tel. (06181) 25 23 81

6457 Maintal

Landolt-Computer
Beratung · Service · Verkauf · Leasing

Autorisierter Commodore-Händler

Wingertstr. 112 · 6457 Maintal/Dörnigheim
Telefon (0 61 81) 4 52 93

6551 Fürfeld

MICHAEL WEISGERBER

HARD + SOFT

Rathausstraße 2
6551 Fürfeld
Telefon (0 67 09) 7 78

6680 Neunkirchen

SHOP 64
Computer GmbH
Neunkirchen
Telefon (0 68 21) 2 37 13

Commodore
Systemhändler

Saarbrücken
Saarlouis
Homburg
St. Ingbert
Neustadt
Trier
Kaiserslautern

6800 Mannheim

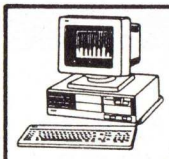
GAUCH+STURM
Computersysteme + Textsysteme

6800 Mannheim 24
Casterfeldstraße 74-76
Telefon (06 21) 85 00 40 · Teletex 6 211 912

7000 Stuttgart

»If AMIGA, go to Schreiber«
Stuttgart's starker Computer-Laden.
SCHREIBER COMPUTER
Rotebühlplatz 10
7000 Stuttgart-1
Tel.0711/227099
Neu · Neu · Neu · Neu
im SUBWAY
BREUNINGER City

7800 Freiburg



Comp.Z
Pochgasse 31
7800 Freiburg
T. 0761/554280

Commodore
AUTORISIERTER
COMMODORE
SYSTEM-HÄNDLER

7890 Waldshut-Tiengen

hettler-data
service gmbh

Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon (0 77 51) 30 94

8000 München

Commodore
AMIGA + PC, HARD- UND SOFTWARE
— BERATUNG / VORFÜHRUNG —
VIDEOBEARBEITUNG

MODL
FOTO · VIDEO · AUDIO · FERNSEH
COMPUTER · GELEGENHEITSMARKT
REPARATUREN · FOTOKOPIEN
FARBKOPIEN · PASSBILDER
MAX-WEBER-PLATZ · 8000 MÜNCHEN 80 · ☎ 4 80 16 50

8400 Regensburg

Zimmermann elektroland

8400 Regensburg
Dr.-Gessler-Str. 8
☎ 09 41 / 9 50 85

8390 Passau
Kohlbruck 2a
☎ 08 51 / 5 20 07

8700 Würzburg

SCHOLL BÜROTEAM
Hardware · Software
Service · Schulung

computer center
am Dominikanerplatz
Ruf (09 31) 3 08 08 - 0

Schweiz

Computer Trend

Ihr Computer Spezialist

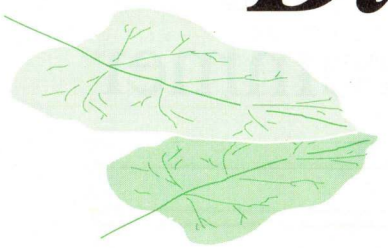
5000 Aarau, Bahnhofstrasse 86,
Tel. 064/22 78 40
4102 Basel-Binningen, Kronenplatz,
Tel. 061/47 88 64
5430 Wettingen, Zentralstrasse 93,
Tel. 056/27 16 60
8400 Winterthur, St. Gallerstrasse 41,
Tel. 052/27 96 96
8021 Zürich, Langstrasse 31,
Tel. 01/241 73 73

Grösste Auswahl an Peripherie, Software, Literatur und Zubehör.

ATARI
Commodore

von Gary Glendown

Biologische Hardware



Schon als wir in der letzten Juliausgabe von der sensationellen Entdeckung biologischer Chips berichtet haben, war die Frage offen, ob es gelingen würde, auch andere biologische Hardware zu erzeugen, z.B. einen Monitor, der in der Lage ist, die nahezu unendliche Anzahl von Farben zu erzeugen, die einer der neuen Biochips anzusteuern in der Lage ist.
Niemand ahnte damals, wie nahe man an der neuen Entdeckung war...

Als uns Herr X, wie wir Herrn Langholz im Laufe dieses Artikels nennen, um ihn zu schützen, beim ersten Mal anrief, um uns von den biologischen Chips zu erzählen, waren wir recht skeptisch. Als uns jedoch vor wenigen Tagen ein neuer Anruf von ihm erreichte, waren wir doch gespannt, was er uns wohl diesmal vorstellen würde. Und ein Besuch bei ihm eröffnete uns wahrlich einen Blick in das nächste Jahrtausend...

KICKSTART: *Herr X, Ihnen und Ihrem Team ist ein neuer Durchbruch in der biologischen Computerentwicklung gelungen. Können Sie uns und unseren Lesern diese Weltneuheit bitte einmal vorstellen?*

Herr X: Aber sicher, das mach' ich doch gern.

(2 Minuten Pause, Herr X sagt kein Wort, die Red.)

KICKSTART: Wollten Sie uns nicht von etwas berichten?

Herr X: Ach, jetzt gleich? Ich dachte, wir warten noch auf die Leser?

KICKSTART: Ja, bitte gleich, wir nehmen das doch auf und drucken es dann, das wissen Sie doch noch vom letzten Mal.

Herr X: Ach so, nun gut. Also, einige Tage nach unserem letzten Gespräch haben wir uns gedacht, wir sollten auch mal versuchen, größere Pflanzen mit den neuen Methoden zu modifizieren. Doch da entstand ein kleines Problem.

Während bei kleineren Pflanzen die Wachstumszeit recht kurz ist, steigt sie mit der Größe der Pflanze exponentiell an. Das würde dazu führen, daß man bei einer Eiche z.B. einige Jahrhunderte warten müßte, um wirklich konkrete Ergebnisse zu erhalten. Das war uns verständlicherweise zu lang.

KICKSTART: Und wie haben Sie dieses Problem gelöst?

Herr X: Ganz einfach. Wir haben neuen, biologisch-kristallinen Dünger entwickelt. Wir nahmen also eine Handvoll Chips...

KICKSTART: Er hat 'Chips' gesagt!!! (vgl. "Bio-Chips", Juli 88, die Red.)

Herr X: Oh, Verzeihung. Ich meinte natürlich Schipps.

KICKSTART: Hilfe!!!

Herr X: Also wie gesagt, wir nahmen eine Handvoll unserer selbstgezüchteten Schipps, zerkleinerten sie, vermischten sie mit etwas Silenzium...

KICKSTART: Silicium!

Herr X: ... und düngten damit die kleinen Pflänzchen. Dazu kam dann noch ein starkes Magnetfeld von der Decke aus und eine stärkere Wechselspannung, die am Boden jeweils auf den beiden Seiten der Pflanzen angelegt wurde.

Und den Erfolg sehen Sie hier. Was ist das Ihrer Meinung nach?

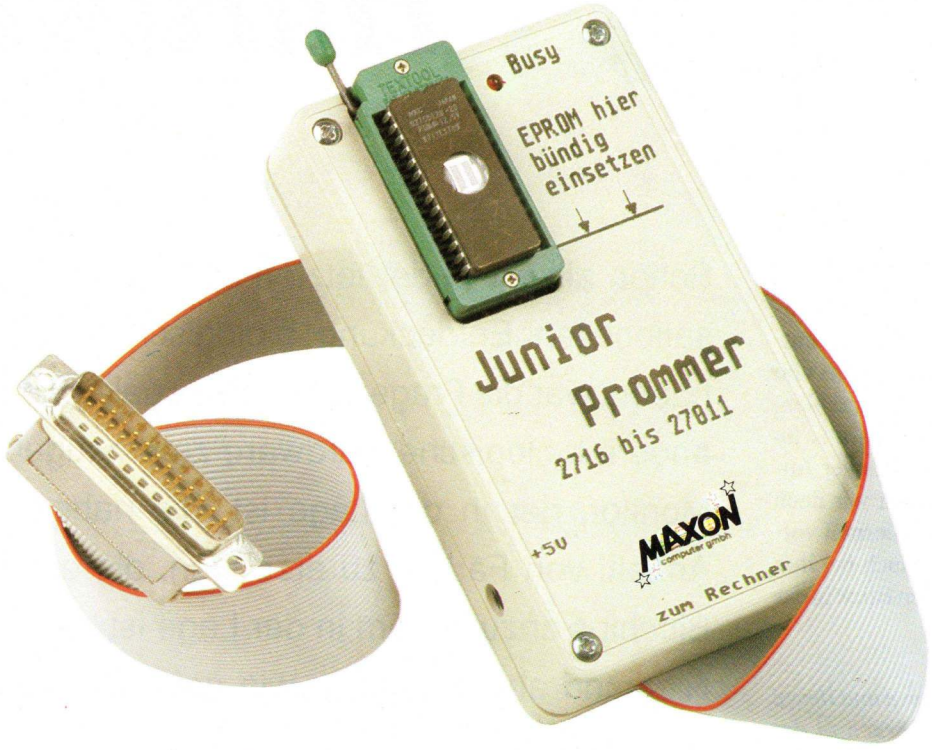
KICKSTART: Ein ... Kürbis?

Herr X: Kürbis? Sie Ignorant. Das ist unser neuer Super-Bildschirm. Haben Sie wenigstens eine Idee, wie lange es gedauert haben könnte, bis er so groß geworden ist?

KICKSTART: Nun, ich schätze, na, so etwa ... 4 Monate?

Herr X: Sie haben ja keine Ahnung! Dieser Kürbis -äh- Bildschirm ist innerhalb von 3 Wochen von einem Samen bis zu dieser Größe gewachsen!

Klein, kompakt und leistungsstark- der JuniorPrommer



Der JuniorPrommer programmiert alle gängigen EPROM-Typen, angefangen vom 2716 (2 KByte) bis zum modernen 27011 (1 MBit). Aber nicht nur EPROMS, sondern auch einige ROM- und EEPROM-Typen lassen sich lesen bzw. programmieren.

Zum Betrieb benötigt der Junior-Prommer nur +5V, die am Joystickport Ihres Amiga abgenommen werden; alle anderen Spannungen erzeugt die Elektronik des Junior-Prommers. Die sehr komfortable Software, natürlich Menü-unterstützt, erlaubt alle nur denkbaren-Manipulationen.

Fünf Programmieralgorithmen sorgen bei jedem EPROM-Typ für hohe Datensicherheit. Im eingebauten Hex-/ASCII-Monitor läßt sich der Inhalt eines EPROMS blitzschnell durchsuchen und auch ändern.

Alles dabei!

Bemerkenswert ist der Lieferumfang, so wird z.B. das Fertiggerät komplett aufgebaut und geprüft im Gehäuse mit allen Kabeln anschlußfertig geliefert.

Auf der Diskette mit der Treibersoftware befindet sich noch ein Programm, das die Kickstart-Diskette in vier EPROM-Dateien für 27512 (64 KByte EPROM) zerlegt, die dann mit dem JuniorPrommer gebrannt werden können. Ferner wird der Source-Code für die Lese- bzw. Programmierroutinen mitgeliefert. Und last but not least ist im Bedienungshandbuch (deutsch) der Schaltplan abgedruckt.

MAXON

Bestellcoupon MAXON Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Hiermit bestelle ich:

- ☐ Fertiggerät wie oben beschrieben DM 249,00
- ☐ Leerplatine (o. Bauteile) und Software DM 59,00
- ☐ Leergehäuse (gebohrt und bedruckt) DM 39,90

Versandkosten: Inland DM 7,50
Ausland DM 10,00
Auslandsbestellungen nur gegen Vorauskasse
Nachnahme zuzüglich DM 3,50 Nachnahmegebühr

KICKSTART: Nein!

Herr X: Doch!

KICKSTART: Toll. Aber sagen Sie, Sie meinten, das wäre kein Kürbis, sondern ein Bildschirm?

Herr X: Genau. Sehen Sie her. Ich nehme ein Messer und schneide ihn hier an... Sieht aus wie ein ganz normaler Kürbis! Aber jetzt: Ich schließe diesen Prototyp eines Bio-AMIGAs an und ...

KICKSTART: Wow! Das hat ja Fernsehbild-, nein - Fotoqualität! Diese Farben, diese Auflösung!

Herr X: Wir haben ähnliche Effekte auch schon bei kleineren Früchten entdeckt, z.B. bei Äpfeln oder Birnen. Wir haben zur Zeit nur noch ein, nein, eigentlich sind es zwei kleinere Probleme. Zum einen haben wir zwar durch den Schnitt einen total flachen Bildschirm, doch ist das Bild, schließlich ist es ja immer noch ein Kürbis, noch rund! Das wollen wir jedoch durch eine neue Züchtung verändern, bei der wir quaderförmige Kürbisse herstellen wollen.

KICKSTART: Und das zweite Problem?

Herr X: Nun, das zweite Problemchen... ist zur Zeit die Haltbarkeit.

KICKSTART: Haltbarkeit?

Herr X: Nun ja, wenn man den Kürbis anschneidet, dann trocknet die Oberfläche mit der Zeit, je nach Außentemperatur in 2 bis 5 Tagen, aus, wodurch dann natürlich kein ordentliches Bild mehr darstellbar ist. Man muß dann eine weitere Scheibe von etwa 1 bis 2 cm Dicke abschneiden.
Somit wird der Kürbis mit der Zeit immer dünner und das Bild immer kleiner. Außerdem muß man immer aufpassen, daß das Zimmer von Insekten befreit ist, da diese den Alterungsprozeß noch beschleunigen. Um das Problem der Austrocknung zu beseitigen, hatten wir schon versucht, die Bildfläche mit Wasser feucht zu halten, doch das führte dazu, daß er innerhalb kürzester Zeit zu schimmeln anfing. Der Benutzer muß sich also damit zufrieden geben, etwa jeden Monat einen neuen Bildschirm zu kaufen oder selber welche zu züchten.

KICKSTART: Doch die Entdeckung des Bio-Bildschirms ist nicht die einzige Sensation, die es aus Ihren Laboratorien neu gibt?

Herr X: Nein, wir haben unter anderem auch einen neuen Speicher entwickelt, der auf eine ganz neue Art und Weise funktioniert. Sehen Sie dieses Glas?

KICKSTART: Aber sicher, ein ganz normales Wasserglas. Nein, halt, ich glaub', ich weiß: Sie haben eine neue Glassorte entwickelt, die als Speicher funktioniert?!

Herr X: Ich meine nicht das Glas auf dem Tisch, das ist ein ganz normales Wasserglas... Ich meine das hier. Wie Sie sehen, befindet sich hierin eine leicht getrübte Flüssigkeit. Es handelt sich dabei um normales Wasser, das von Blaualgen, also einer der ältesten Lebensformen auf unserer Erde, wimmelt. Und nun passen Sie auf. Ich schließe den Algenspeicher an den BioAMIGA an...

KICKSTART: Was, nur ein kleiner Schlauch?

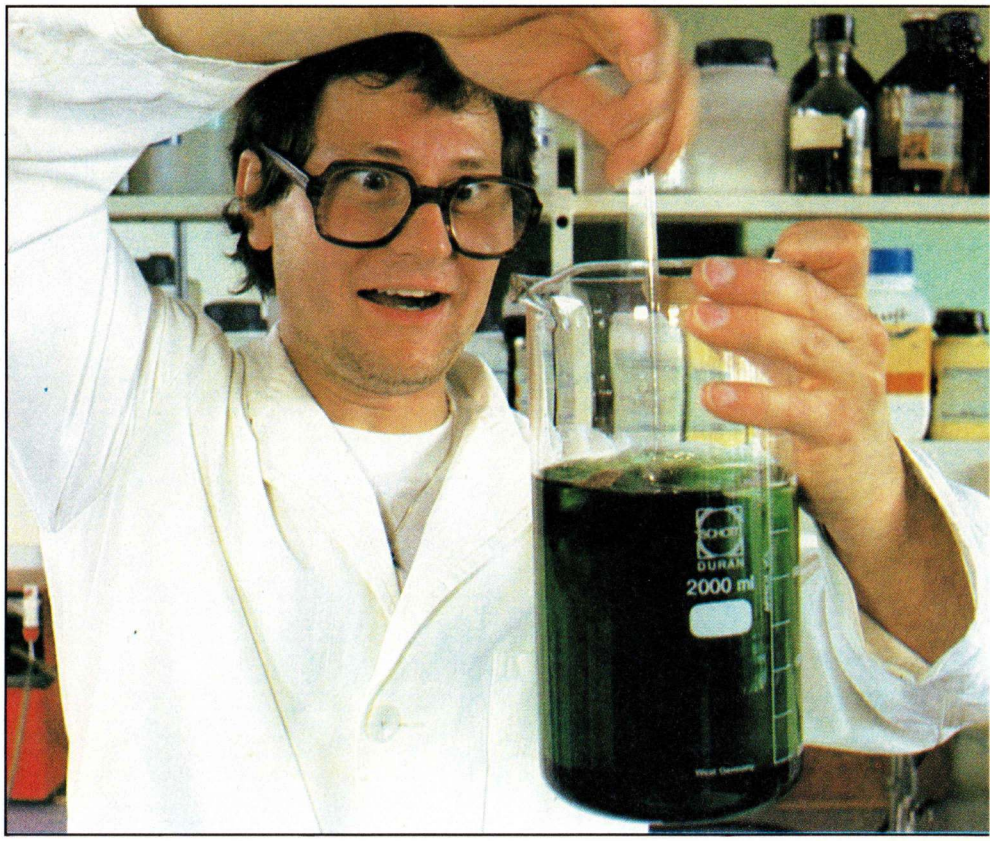
Herr X: Genau, man braucht nur eine Leitung. Wie gesagt, ich hänge den Schlauch in das Gefäß. Im AMIGA habe ich ein etwas modifiziertes Betriebssystem, das in der Lage ist, das neue Laufwerk DA0: anzusprechen (*Drive Algen 0, die Red.*) und von ihm zu booten. Passen Sie auf. Ich drücke Reset und... die Workbench ist da.

KICKSTART: Wahnsinn! Das waren keine 5 Sekunden!

Herr X: Warten Sie's ab! Ich öffne jetzt das Directory *DA0:*. Sehen Sie, das Window mitsamt Inhalt ist innerhalb von 10 Sekunden da. Und wenn Sie meinen, das sei nichts besonderes, dann schauen Sie sich doch mal das Info des Glases an.

KICKSTART: Aber das ist doch wirklich kein Problem, innerhalb von 10 Sekunden ein Paar Drawers anzuzeigen. Moment, DA0: anklicken, dann Info aufrufen und ... wow! Das Glas hat einen Speicher von über 700 MByte! Ich halt' s nicht aus. Wie haben Sie das geschafft? Moment, was ist das? Die Zahl wird ja immer größer? Was ist denn das? Die Zahl steigt ja in jeder Sekunde um einige Byte?

Herr X: Toll, was? Das ist der Vorteil bei der Verwendung von biologischen Spei-



Herr X mit seiner hochoptimierten Nährlösung. (Archivfoto '88)

chern! Dadurch, daß sich die Speicherzellen ständig teilen, steigt der gesamte zur Verfügung stehende Speicher auch ständig an. Zwar sterben auch einige der Zellen, doch ist die Vermehrungsrate etwa doppelt so groß wie die Sterberate.

KICKSTART: Wow! Und wie lange kann man ihn benutzen, bevor man ihn auswechseln muß?

Herr X: Das ist verschieden, bisher haben wir Haltbarkeiten bis zu 4 Wochen erreicht, d.h. man sollte alle 2 Wochen eine Sicherheitskopie anlegen.

KICKSTART: Na, das geht ja noch, äh Moment. Sicherheitskopie? Und worauf soll man das speichern?

Herr X: Nun, am besten wäre ein weiterer biologischer Speicher, doch da gibt es ein Problem... Ganz am Anfang besteht das Glas biologischer Speicher nur aus ca. 20 KByte. Man sollte also, auch der Sicherheit willen, auf normale Disketten oder Festplatten kopieren.

KICKSTART: Moment, ich will gerade mal was ausrechnen... Also: Bei 700 MByte sind das $e^{20.414024}$ Byte, mal sin 135°, durch 210, durch cos 405°, mal $e^{-6.7799219}$, das macht ... 814.5. Was? 815 Disketten, nur um eine Sicherheitskopie zu ziehen? Das ist ja Wahnsinn!

Herr X: Tja, das ist eben der Nachteil dieses Speichers. Aber bedenken Sie: Sie brauchen nur 35 Festplatten à 20 MByte, um eine Sicherheitskopie zu machen!

KICKSTART: Nun ja, wenn man nun 360 MByte-Platten nimmt, dann braucht man sogar nur 2 Stück... einfach toll!

Herr X: Na also. Hier sehen Sie übrigens unser neues Projekt: ein $5^{3}/_{4}$ Zoll Blatt-Laufwerk.

KICKSTART: Und wie funktioniert das?

Herr X: Genau wie jedes andere Laufwerk auch. Man schiebt einfach ein Blatt unseres speziellen Baums ein, und schon kann man genauso damit arbeiten wie mit jedem anderen Laufwerk auch. Zur Zeit haben wir allerdings noch eine recht niedrige Speicherkapazität, nur ca. 2.32 MByte, aber wir wollen das in Zukunft noch ändern. Es hat übrigens auch noch eine Besonderheit. Wie Sie sicher wissen, dreht sich in einem normalen Laufwerk die Diskette an dem Schreib-/Lesekopf vorbei...

KICKSTART: Das ist mir nicht unbekannt...

Herr X: Dann passen Sie bitte mal auf. Ich gebe hier auf dem AMIGA 'Dir DB0:' ein, und sehen Sie, was passiert!

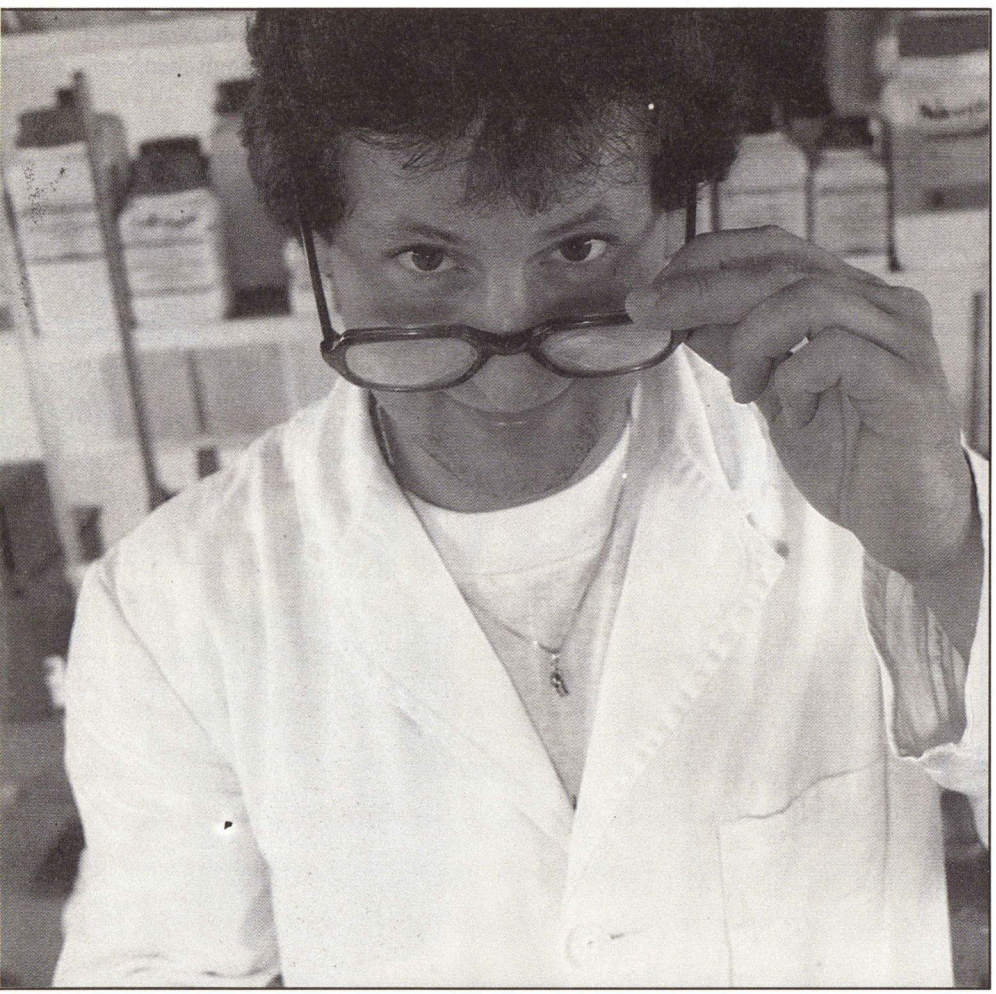
KICKSTART: Nein, das halt' ich nicht aus! Das ganze Laufwerk dreht sich!

Herr X: Na ? Lustig, gell! Wir haben uns gedacht, wir sollten mal was anderes machen, und das ist doch wirklich mal was anderes.

KICKSTART: Wirklich, etwas ganz anderes. Herr X, ich möchte Ihnen im Namen aller unserer Leser für diese Einblicke in die neue Computertechnologie danken und Sie bitten, auch weiterhin so weiterzumachen. Nur eine Frage zum Abschluß: Alle großen Firmen und Entwickler werden auf den großen Messen in diesem Jahr wieder Stände haben. Haben Sie auch vor, Ihre Produkte dort der Öffentlichkeit vorzustellen?

Herr X: Das kann ich im Moment noch nicht sagen, denn erstens weiß ich noch nicht, wie weit wir mit den zur Zeit laufenden Entwicklungen sein werden, und zweitens braucht man für eine solche Messe bekanntermaßen auch Zeit, die ich im Moment wirklich nicht habe. Doch seien Sie beruhigt, wenn wir weitere neue Entdeckungen machen, werden Sie wieder die Rechte zur Exklusivveröffentlichung bekommen. Leider ist es für die CeBIT schon zu spät, das heißt, wir müssen eventuell mal sehen, ob wir auf der CES in Chicago einen Stand bekommen können.

KICKSTART: Herr X, wir danken Ihnen für dieses Gespräch!



Was wird die Zukunft bringen? Herr X blickt ihr optimistisch entgegen.

Kurz & Klar

AMIGA GFA-BASIC 3.0

PETER FRITZEN
KLAUS SCHNEIDER
OLIVER STEINMEIER

Der GFA-BASIC 3.0-Interpreter stellt zur Zeit eines der leistungsfähigsten Entwicklungssysteme für den Commodore Amiga dar. Aufgrund seines sehr großen Befehlssatzes (mehr als 300 Befehle und Funktionen) kann man nahezu alle Probleme auf einfache Art und Weise lösen. Allerdings ist es fast unmöglich, die Syntax und exakte Funktionsweise sämtlicher Anweisungen auswendig zu kennen. „Kurz & Klar – Amiga GFA-BASIC 3.0" soll hier helfen. In kurzer, prägnanter Form werden die Kommandos des Interpreters (incl. der im Handbuch fehlenden Anweisungen wie die Bob-Steuerung) in alphabetischer Reihenfolge erklärt. Dabei ist jeweils die genaue Syntax angegeben. Kurze Beispiel beseitigen letzte Unklarheiten. Das Buch enthält weiterhin eine nach Sachgruppen geordnete Befehlsreferenz, einen ausführlichen Index sowie zahlreiche nützliche Tabellen und Übersichten, die bei der Programmierung in GFA-BASIC auf dem Amiga benötigt werden. Kurz gesagt enthält das „Kurz & Klar – Amiga GFA-BASIC 3.0" alles, was man zur Entwicklung eigener Programme benötigt, wenn grundlegende Kenntnisse der Sprache selbst bereits vorhanden sind.



AUS DEM INHALT:

- umfangreiche alphabetische Befehlsübersicht mit genauer Syntaxangabe, Befehlserklärung und kurzen Beispielen
- Sachgruppen-Übersicht aller Anweisungen des GFA-BASICs
- zahlreiche nützliche Tabellen und Übersichten (Füllmuster, ASCII, Scan-Codes, Schriftarten, CSI-Codes, etc.)
- umfangreiches Schlagwortregister

Das übersichtliche Nachschlagewerk

ca. 250 Seiten, Hardcover **DM 29,-**

Preise sind unverbindlich empfohlene Verkaufspreise

Heim Verlag

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

Schweiz: DataTrade AG
8021 Zürich
Langstr. 94

BESTELLCOUPON Kurz & Klar – Amiga GFA-BASIC 3.0

Bitte senden Sie mir _____ St. **Kurz & Klar – AMIGA GFA-BASIC 3.0**
zzgl. DM 5,- Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname _____

Straße, Hausnr. _____ PLZ/Ort _____

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte.

Linksverkehr

Emulation des BBC-Micro auf dem AMIGA

In den letzten Januartagen wurde in England "The Emulator" vorgestellt, ein Software-Emulator, der es ermöglicht, den AMIGA wie einen BBC-Micro zu betreiben. Der BBC-Micro der Firma Acorn, die mit dem Archimedes für eine Überraschung gut war, stellte vor sieben Jahren diesen Rechner vor, unterstützt von der englischen Regierung, die in ihrem Bemühen, Informatik und Computerwissen (vor allem an den Schulen) zu fördern, sich für den BBC-Micro stark machte.

VON WILFRIED SOLBACH



Die Rundfunkanstalt BBC war bei der Entwicklung des Rechners beteiligt und stand bei der Namensgebung Pate. Außerdem stellte sie kostenlose Werbezeiten zur Verfügung. Schnell eroberte sich der BBC-Micro den englischen Home- und Erziehungsmarkt.

Was kann der BBC-Micro-Emulator nun?

Ein Zitat aus einer englischen Computerzeitschrift (Your Computer, April 1984):

"A full colour 32K ROM computer with text and graphics modes up to 640 x 256 graphics and 80 column text screen, extended Microsoft Basic, built in assembler 1 MHZ and tube interface, sideways ROM, RS 423, A/D converter:"

Und das alles für 399 Pfund - der C64 kostete damals in England nur ca. 230 Pfund. "Und jetzt können wir diese Mühle also auf dem AMIGA emulieren...", höre ich es schon von allen Seiten maulen.

Warum denn nicht? Es gibt Zehntausende von Programmen, da wird wohl für jeden das Passende dabei sein. Nun zum Emulator selber: Ein schütteres Päckchen flatterte mir da aus England ins Haus - eine Diskette, ein 19seitiges Handbuch, eine Registrierkarte.

Nicht viel!?

Das Programm wird über die Workbench gestartet, es lädt die *'mathtrans-library'* von der Workbench nach. Öffnet man das Fenster des Diskicons, erblickt man das Programmicon, Basicicon und diverse andere zum direkten Start von insgesamt vier Laufwerken.

Die erste Überraschung: Das Programm ist multitaskingfähig, AMIGA-DOS verabschiedet sich also nicht auf Nim-

merwiedersehen wie beim MS-DOS-Emulator. Der Mauszeiger bleibt erhalten; klickt man irgendwo in den Bildschirm, erscheint am oberen Bildschirmrand eine "dragbar", und man kann "den BBC-Micro" herunterziehen und in der AMIGA-Umgebung weiterarbeiten. Die Hersteller versichern, daß man bei einem AMIGA 2000 mit 3MB RAM und einer PC-Brückenkarte BBC-DOS, MS-DOS und AMIGA-DOS parallel laufenlassen und zwischen allen drei Systemen Daten austauschen kann. Der Emulator benötigt 300 KByte Speicher und läuft auch mit 68010- und 68020-Prozessoren. Auch lassen sich gewisse Features der Workbench 1.3 (z.B.PIPE) miteinbeziehen, Keymap- und Preferenceeinstellungen werden vom Emulator respektiert.



Bild 1: Die Einschaltmeldung des Emulators

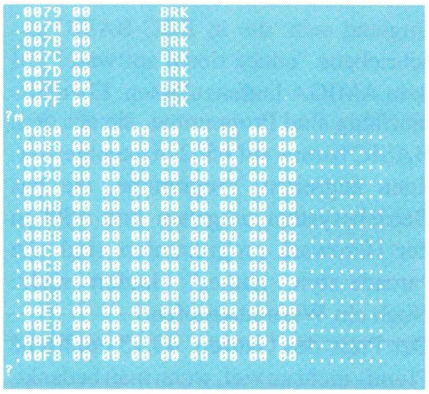


Bild 2: Der eingebaute Monitor des BBC-Micro.

```
1380 Pos%=FNPT(MX%(I%)+48,MY%(I%)-44):IFPos%=1ORPos%=3THEN1370
1390 PROCmonster(mx%,my%):PROCmonster(MX%(I%),MY%(I%))
1400 ENDPROC
1410 DEFPROCundoA
1420 IFdown%<>monstery%THEN1480
1430 IFSGN(HORZ%)=-1ANDlineacross%(HOR%,VER%)=1 lineacross%(HOR%,VER%)=0:GOTO14
70
1440 IFSGN(HORZ%)=-1THEN1480
1450 IFlineacross%(OLDX%,OLDY%)=1 lineacross%(OLDX%,OLDY%)=0:GOTO1470
1460 GOTO1480
1470 MOVEX%(OLDX%),Y%(OLDY%):DRAWX%(HOR%),Y%(VER%):GOTO1480
1480 ENDPROC
1490 DEFPROCundoU
1500 IFacross%<>monsterx%THEN1560
1510 IFSGN(VERT%)=-1ANDlinedown%(HOR%,VER%)=1 linedown%(HOR%,VER%)=0:GOTO1550
1520 IFSGN(VERT%)=1THEN1560
1530 IFlinedown%(OLDX%,OLDY%)=1 linedown%(OLDX%,OLDY%)=0:GOTO1550
1540 GOTO1560
1550 MOVEX%(OLDX%),Y%(OLDY%):DRAWX%(HOR%),Y%(VER%)
1560 ENDPROC
1570 DEFPROCcolour
1580 VDU19,11,7;0;:
1590 VDU19,7,1;0;19,9,1;0;19,15,1;0;
1600 VDU19,8,4;0;19,14,4;0;
1610 VDU19,5,3;0;
1620 VDU19,13,5;0;19,12,5;0;
1630 ENDPROC
1640 DEFFNPT(d%,e%)
1650 =POINT(d%,e%-1)
1660 DEFFNpt(f%,g%)
1670 =POINT(f%+32,g%)
>_
```

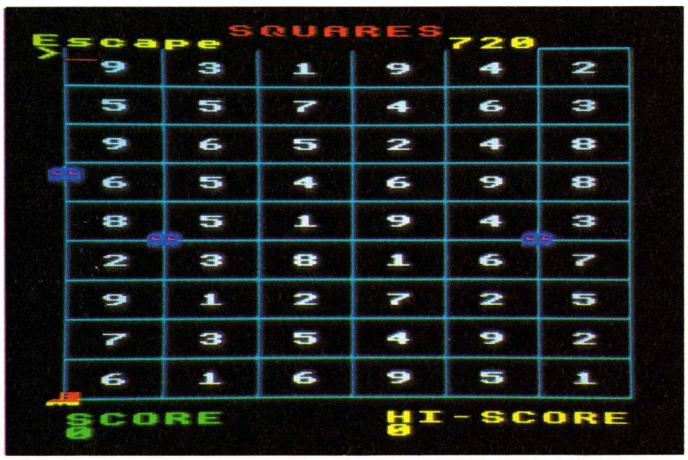
Bild 3: Im 80-Zeichen Modus kann am besten programmiert werden.

Die Auflösungen des Emulators

Das Programm unterstützt die folgenden Graphikauflösungen:

0	640 x 256	2 Farben	80x32
1	320 x 256	4 Farben	40x32
2	320 x 256	16 Farben	20x32
3	(nur Text)	2 Farben	80x25
4	320 x 256	2 Farben	40x32
5	320 x 256	4 Farben	20x32
6	(nur Text)	2 Farben	40x25
7	Teletextdarstellung		40x25

Durch die Eingabe von *MODE<x> wechselt der Emulator blitzartig seinen Graphikmodus. Ja, blitzartig, denn jetzt kommt die zweite Überraschung: Der Emulator ist rasant. Bei einer kleinen Rechenaufgabe in BASIC ("Zähle bis 100000") blieb das AMIGA-BASIC auf der Strecke: Der Emulator war etwa 30% schneller. Das BBC-BASIC 2 war schon immer dafür berühmt, eines der schnellsten BASIC-Varianten zu sein. Für den Emulator wurde das BASIC im 68000er Assemblercode völlig neu geschrieben und ist dementsprechend flott. Ein kleines Spielelisting wurde eingegeben - und siehe da, das Programm lief, schön bunt, mit Sound und sehr schnell. Ein erster Wermutstropfen: Die Tastaturabfrage bei einem Spiel klappte noch nicht so recht, nun ja...(RETURN wurde in diesem Fall nicht als RETURN erkannt, und auch die Cursortasten wollten nicht so, wie sie sollten).



Das abgebildete Spiel, eine PAC MAN-Variante, lief recht flott.

Laden und Speichern

Abspeichern kann man auf einer normalen AMIGA-DOS-Diskette, nur muß vorher ein winziges Programm (DFScat) abgespeichert sein, um dem Emulator zu ermöglichen, das Diskettenverzeichnis lesen zu können. Mit *ASSIGN 1 DF1: wird das externe Laufwerk angesprochen, dann folgt *DRIVE 1, und schon läßt sich mit SAVE "File" und LOAD "File" munter drauflosarbeiten.

Hauptzweck des Emulators soll nach den Vorstellungen von Commodore/

England sein, die in BBC-BASIC geschriebene "educa tional software" auf dem AMIGA laufenzulassen. Ebenfalls lauffähig sind Programme, die aus dem BASIC heraus auf 6502-Assemblerroutinen zugreifen. Programme, die das Betriebssystem umgehen und wild in der Hardware programmiert sind wie "arcade-style games", laufen nicht. Noch ein Wermutstropfen.
Der Preis des Ganzen: 49.95 englische Pfund - nicht zu viel, wenn man bedenkt, daß in der Tat jetzt Zehntausende von neuen Programmen verfügbar sind. Theoretisch zumindest, denn wie kommt man in Deutschland schon an BBC-Software? Dies ist wahrlich ein Problem, vielleicht besitzt ein interessierter AMIGA-Anwender einen guten Bekannten auf der Insel. Der Markt in England - auch was Public Domain angeht - ist auf jeden Fall riesig.

Für knapp 10 Pfund bietet James Associates, der "sole distributor" auch noch ein Kabel an, mit dem sich Daten und Programme direkt vom BBC-Micro zum AMIGA übertragen lassen. Eine andere Möglichkeit bietet sich natürlich per Modem, da die BBC-Programme alle auf 5 $^1/_4$ Zoll-Disketten laufen. So läßt sich alles direkt auf 3 $^1/_2$ Zoll abspeichern und ohne Probleme im AMIGA verwenden.

Minuspunkte

Nach so viel Lob muß auch gemeckert werden: Das Handbuch wendet sich offensichtlich an die Leute, die den Emulator programmiert haben, sich zumindest aber in BBC-BASIC auskennen. Die BASIC-Befehle müssen unbedingt ausführlicher dargestellt werden, die Probiererei nervt sonst zu sehr. Außerdem ist das Programm kopiergeschützt, mir jedenfalls ist es nicht gelungen, eine Sicherheitskopie anzufertigen. Ob das an einem Kopierschutz liegt oder daran, daß Teile des Emulators im BBC-Format vorliegen, sei dahingestellt.

Fazit

Mit dem BBC-Micro-Emulator eröffnet sich eine neue Welt, tausende von Programmen sind verfügbar. Für den deutschen Markt wird der Emulator aber nur eine zweitrangige Rolle spielen. Ist der BBC-Micro doch so gut wie unbekannt.

```
%,VER%)=0:GOTO1550
 1520 IFSGN(VERT%)=1
THEN1560
 1530 IFlinedown%(OL
DX%,OLDY%)=1 linedow
n%(OLDX%,OLDY%)=0:GO
TO1550
 1540 GOTO1560
 1550 MOVEX%(OLDX%),
Y%(OLDY%):DRAWX%(HOR
%),Y%(VER%)
 1560 ENDPROC
 1570 DEFPROCcolour
 1580 VDU19,11,7;0;:
 1590 VDU19,7,1;0;19
,9,1;0;19,15,1;0;
 1600 VDU19,8,4;0;19
,14,4;0;
 1610 VDU19,5,3;0;
 1620 VDU19,13,5;0;1
9,12,5;0;
 1630 ENDPROC
 1640 DEFFNPT(d%,e%)
 1650 =POINT(d%,e%-1
)
 1660 DEFFNpt(f%,g%)
 1670 =POINT(f%+32,g
%)
>
```

Bild 4: Der 20 Zeichen-Modus ist wohl eher für Kurzsichtige gedacht.

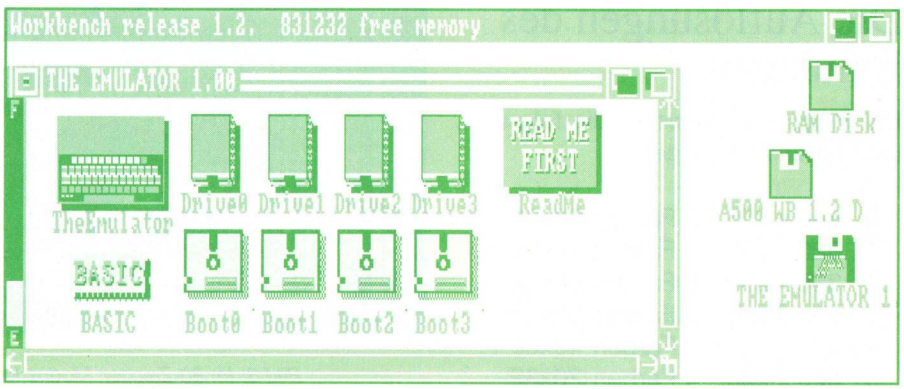


Bild 5: Von der Workbench aus wird der Emulator gestartet. Verschiedene Utiltyprogramme stehen zur Verfügung.

In England hingegen erfreut er sich großer Verbreitung. Erfreulich war die hohe Emulationgeschwindigkeit. AMIGABASIC kam zumindest ins Hintertreffen. Rein in Assembler geschriebene Programme werden langsamer emuliert, was zu erwarten war.

BBC Micro-Emulator

+ schnelle BBC Micro-BASIC-Emulation
+ BBC-Programme können in AMIGADOS-Format gespeichert werden
+ unterstützt alle Modi des BBC-Micro
+ Multitaskingfähig

- mangelhaftes Handbuch
- in Deutschland wenig BBC Micro-Programme erhältlich
- Emulator kopiergeschützt

Bezugsadresse:
englische Commodore-Fachhändler
Preis: 49.95 englische Pfund

ENDE

Btx/Vtx-Manager

Btx/Vtx.

Jetzt mit Grafik und Farbe auf dem Amiga!

Bildschirmtext als neue Welt der Telekommunikation läßt sich mit dem *Btx/Vtx-Manager V2.0* komfortabel nun auch auf dem Amiga handhaben. Dabei sind Grafik, Farbe und Maussteuerung so selbstverständlich wie der sichere Anschluß an das Postmodem DBT03. Weitere Informationen senden wir Ihnen auf Anfrage gerne zu.

(FTZ-Zulassung beantragt).

Drews EDV + Btx GmbH
Bergheimerstraße 134 b
D-6900 Heidelberg
Telefon (0 62 21)
2 99 00 und 2 99 44
Fax (0 62 21) 16 33 23
Btx-Nummer 0622129900
Btx-Leitseite *2 99 00 #

Drews

DONAU-SOFT

Ihr Public-Domain-Partner
mit ca. 2000 PD-Disk im Archiv
● ab 3,- DM ●

Alle gängigen Serien sind lieferbar

Preise:

Einzeldisk	4,50 DM
ab 10 Disk	4,— DM
ab 50 Disk	3,50 DM
ab 100 Disk	3,30 DM
ab 200 Disk	3,— DM

Leerdisketten 3,5'' 2 DD

NoName 100 %	ab 2,20 DM
Markendisk	ab 2,50 DM

Preise incl. 3,5''-2 DD-Disks
Mit Qualitätsgarantie
Wir kopieren nur mit doppeltem Verify.
— Alle Disks sind etikettiert —

3 ausführliche Katalogdisketten mit Kurzbeschreibung aller Programme gegen **10,—** DM (V-Scheck/Briefmarken) anfordern!
Viruskiller + CLJ-Wizard gratis

Nicht nur für Einsteiger:
Das große Amiga-PD-Handbuch

Band I, II, III	je 49,— DM
Kombi: I + II + III	139,— DM
10 (11) Disks zu I, II, III	je 55,— DM
alle 31 Disks	135,— DM
Ein Band + Disks	100,— DM
alle 3 Bände + alle Disks + 3 Katalogdisketten	255,— DM
Amiga Spielebuch	49,— DM

+ DM 5,— bei Vorkasse, + DM 8,— bei Nachnahme
Ausland: + DM 10,— (nur Vorkasse)

MAIK HAUER

Postfach 1401, 8858 Neuburg/Do. Tel. 08431/49798

PROBLEME MIT ENGLISCHER ANLEITUNG?

Übersetzen von englischen Bedienungsanleitungen, Computerhandbüchern und Programmen für AMIGA, Atari, Macintosh, MS-DOS, Unix-Xenix, Periferiegeräte und andere durch *Spezialisten.*

Sonderangebote deutscher Anleitungen für AMIGA

Aegis Sonix	**39,–**	**DBW Render**	**20,–**
Flight Simulator II	**29,–**	**Dr. Ts' KCS**	**49,–**
Aegis Draw plus	**49,–**	**Starglider II**	**20,–**
Aegis Impact	**39,–**	**CLI-Mate**	**20,–**
Aegis Audiomaster	**39,–**	**PrtDrvGen Druckertreibergenerator**	**15,–**
Sekaassambler	**29,–**	**Galileo**	**29,–**

Bei Versand im Inland berechnen wir DM 5,– für Porto Verpackung/NN.
Versand ins Ausland nur mit Vorauskasse + DM 4,– für Porto/Verpackung

0 83 74 - 98 73

T. Sonnenmoser · Hauptstraße 26 · D-8961 Haldenwang

POWERSOFT

S. Aulich & D. Johnson

Fantavision DM 89.– ★ Dragons Lair DM 95.–
X Copy DM 39.– ★ Viruskiller 2.0 DM 39.–
Superback DM 99.– ★ Kind Words DM 139.–
★ TV Sports Football dt.Anl. DM 85.– ★

Berliner Ladenlokal ★ Schwedenstraße 18 c ★ 1/65 ★ am U-Bahnhof Osloer Straße

Nachnahme 6,– Vorkasse 5,– ab 150,– frei/Ausland 8,–

POWERSOFT Versand
Hittenauer Str.7
1000 Berlin 26

♦ Täglich Neuheiten ♦
030 / 492 2056

Hotline 030 / 402 5941

49 MB Filecard für A-2000 (440 KB/sec.) **1398,- DM**
30 MB Filecard 998,- DM / 30 MB XT-Filecard 898,- / 49 MB 1298,-

20 MB Amigos-Festplatte für A-500 + 1000 **948,- DM**
30 MB 1098,- / 40 MB 1398,- / 65 MB 1548 DM
Alle Festplatten und Filecards sind softwaremäßig autobootfähig.

Modem Discovery 1200C+ incl. Software **279,- DM**
Discovery 2400C 448,- / Discovery 1200H 199,- /
Disc. 1200A 448,-
Anschluß an das Postnetz zur Zeit noch bei Strafe verboten.

Amigos-Drive 3.5 extern (für A-500 + 1000) **279,- DM**
Nec 1037A, abschaltbar, durchgef. Port, Metallgehäuse amigafarben.
Amigos-Drive 5.25 extern wie oben + 40/80 Tr. (Teac) **359,- DM**

A-2000 2.tes internes Laufwerk **199,- DM**
Nec 1036A Laufwerk kompl. modifiziert und anschlußfertig.

200 No Name 3.5 2DD Disks **400,- DM**
100 Verbatim 3.5 2DD 235,- DM / 100 No Name 5.25 2D 65,- DM

Public Domain: 100 Disks 3.5 2DD bespielt für **250,- DM**
IBM-PD auch vorrätig!

Star-Drucker: Der Amiga-Drucker!!
Star LC-10 548,- DM / Star LC-10 Color 698,- DM /
Star LC-24-10 998,- DM

Amiga 2000 + 1084S + 2.tes internes 2598,- DM
PC/XT-Karte 798,- DM / AT-Karte a. Anfrage
2 MB Erw. A-2000 (Aufrüstbar auf 8 MB) **1298,- DM**
Electronischer Bootselector (df0-df1-df2) **48,- DM**

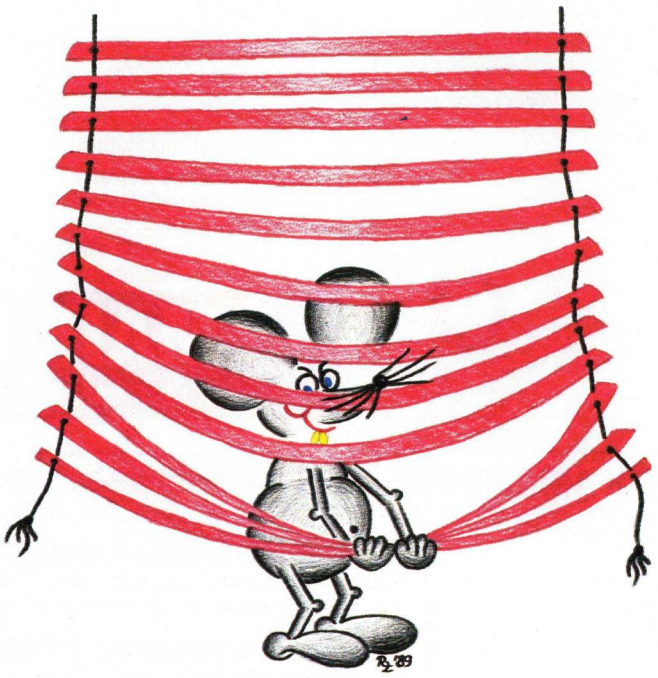
Versandkosten je nach Gewicht. Preisänderungen unter Vorbehalt.
Ladenzeiten: Mo – Fr von 10 – 13 und 14.30 – 18 Uhr. Samstag 10 – 13 Uhr.

Computer Müthing, Fliederstr. 27, 4370 Marl, Tel. 0 23 65 / 6 60 76

VON JÜRGEN MÜLBERT

MAUS AUS

Die Maus vom Bildschirm verbannt



Die Maus stellt ein sehr wichtiges Eingabemedium dar. Mit ihr können wir mit unserem AMIGA bequem und einfach kommunizieren. Also nur Vorteile? Leider nicht... Bekanntlich sollte man alles von zwei Seiten betrachten. Warum sollte das bei der Maus, die als Zeiger auf dem Bildschirm sichtbar ist, anders sein?

Der Mauszeiger kann auch Bereiche des Bildschirms überdecken, die man eigentlich sehen wollte. Das ist um so einfacher, je kleiner die auf dem Bildschirm dargestellte Information ist. Die kleinste auf dem Bildschirm darstellbare Information ist ein Pixel, also ein Bildpunkt. Ein Punkt gibt aber nicht sehr viel Information weiter. Die X-Auflösung oder die Breite beträgt beim AMIGA 320 bzw. 640 Bildpunkte. Der Mauszeiger überdeckt hiervon sechszehn, ist also 16 Bildpunkte breit. Die Y-Auflösung bzw. die Höhe des Bildschirms kann 256 und für Flimmerfans 512 Punkte (PAL) betragen. Die Höhe des Mauszeigers ist nicht begrenzt, beträgt aber in der Regel auch 16 Punkte.

Wenn Sie, was beim AMIGA Pflicht sein sollte, an einem Monitor arbeiten, haben Sie sicher den 80 Zeichen-Modus mit dem Preferences-Programm eingestellt. Der hierfür zuständige Zeichensatz heißt TOPAZ_8. An der acht am Ende läßt sich erkennen, daß ein Zeichen acht Punkte breit ist. Bei den meisten Zeichensätzen ist übrigens die Breite und die Höhe gleich. So auch in diesem Fall, aber auch bei dem anderen fest im ROM befindlichen Zeichensatz TOPAZ_9.

Hier kommt es zum Problem. Der Mauszeiger, der, wie wir wissen, in der Regel 16 Pixel hoch und breit ist, füllt eine Fläche von 16 x 16 Pixeln. So kann er, sonst so hilfreich, unter Umständen ein bis zweieinhalb Zeichen eines Textes überdecken, was mitunter sehr lästig werden kann, denn ausgerechnet da, wo man gerade lesen will, befindet sich der Mauszeiger. Der Griff zur Maus ist unumgänglich. Hier schafft das von mir erdachte Programm Abhilfe. Nach einer von Ihnen festgesetzten Zeit verschwindet der Zeiger vom Bildschirm. Er löst sich einfach auf. Ein kleines Verschieben der Maus oder der Druck auf eine der zwei Maustasten bringt den vermißten Zeiger wieder auf den Bildschirm zurück.

Drei Startmodi

Ich habe es möglich gemacht, das Programm auf drei verschiedene Arten zu starten. Zwei fallen Ihnen sicher sofort ein.

1. Vom CLI genügt die Eingabe von 'MausAus', gefolgt von einer Zeitangabe mit der von Ihnen gewünschten Verzögerung. Nach dieser Zeitverzögerung wird der Mauszeiger ausgeschaltet. Wenn Sie mit zehn Sekunden (vorbelegter Wert) einverstanden sein sollten, brauchen Sie keine Zeit anzugeben.

2. Der zweite Weg geht über die Workbench. Der Doppelklick mit der Maus auf das betreffende Programmicon startet das Programm, das die Maus verschwinden läßt. Doch Sie sind mit der Zeit nicht einverstanden. Sie wollen die Zeitverzögerung ändern, ob länger oder kürzer spielt keine Rolle. Sie brauchen das Programm jetzt nicht vom CLI aus zu starten. Klicken Sie das Icon einmal

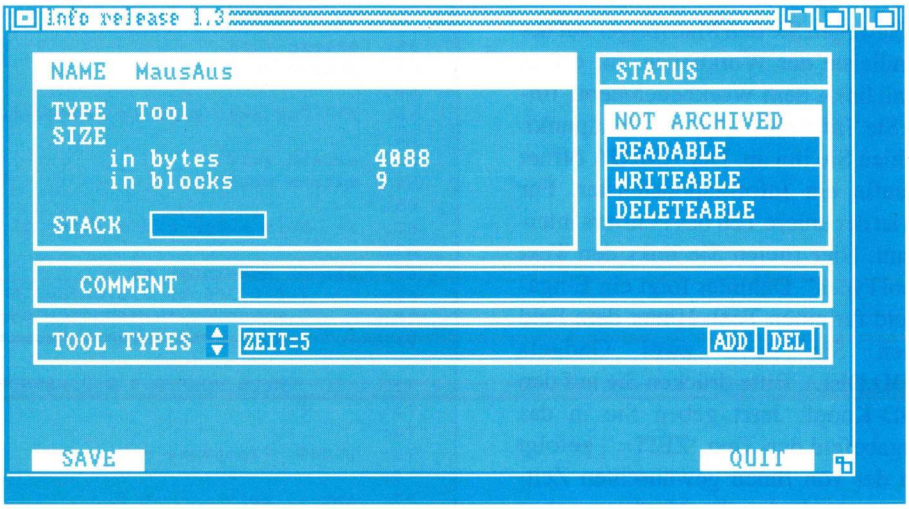


Bild: Im Info-Fenster können Sie die Zeitspanne, nach der die Maus ausgeschaltet wird, bestimmen.

```
 1:  /*              MausAus
 2:      die Maus wird aus dem Weg geräumt
 3:      Autor: Jürgen Mülbert
 4:      Compiler: AZTEC C V3.6
 5:      (C) MAXON Computer GmBH
 6:      KICKSTART 1989
 7:  */
 8:
 9:  /*  Viele, viele includes   */
10:
11:  #include <fcntl.h>
12:  #include <exec/types.h>
13:  #include <exec/lists.h>
14:  #include <exec/nodes.h>
15:  #include <exec/tasks.h>
16:  #include <exec/ports.h>
17:  #include <exec/memory.h>
18:  #include <exec/io.h>
19:  #include <exec/devices.h>
20:  #include <devices/input.h>
21:  #include <devices/inputevent.h>
22:  #include <devices/timer.h>
23:  #include <libraries/dos.h>
24:  #include <libraries/dosextens.h>
25:  #include <workbench/workbench.h>
26:  #include <workbench/startup.h>
27:  #include <hardware/custom.h>
28:  #include <hardware/dmabits.h>
29:  #include <hardware/intbits.h>
30:  #include <graphics/rastport.h>
31:  #include <graphics/gfx.h>
32:  #include <graphics/gfxmacros.h>
33:  #include <intuition/intuitionbase.h>
34:
35:  /*  Nur ein define! */
36:
37:  #define   AUSZEIT 10
38:
39:  /*  Die Externen. Ich hoffe die LATTICE-Besitzer
40:      freuen sich darüber. Ich hätte einfach '#include
41:      <functions.h>  mit einlesen können */
42:
43:  extern struct IntuitionBase *OpenLibrary();
44:  extern struct MsgPort       *CreatePort();
45:  extern struct Message       *GetMsg();
46:  extern struct IOStdReq      *CreateStdIO();
47:  extern struct Process       *FindTask();
48:  extern struct DiskObject    *GetDiskObject();
49:  extern struct Window        *OpenWindow();
50:  extern APTR                  AllocMem();
51:  extern STRPTR                FindToolType();
52:  extern long                  OpenDevice(), SetSignal(), Wait();
53:  extern int                   atoi(), strlen();
54:  extern void                  CloseLibrary(), CloseDevice();
55:  extern void                  FreeMem(), DoIO();
56:  extern void                  DeletePort(), DeleteStdIO();
57:  extern void                  ReplyMsg();
```

an. Bewegen Sie den Mauszeiger auf die Menüleiste der Workbench. Im ersten Menü links, dem Workbenchmenü, finden Sie 'Info' als vorletzten Menüpunkt. Wählen Sie ihn an. Der AMIGA öffnet daraufin ein Informationsfenster. Für uns ist der untere Teil des Fensters interessant. Hier finden Sie links den Text 'ToolTypes'. Dahinter folgt ein Eingabefeld für einen Text. Hinter dem Feld sehen Sie noch zwei Gadgets (ADD,DEL). Bitte drücken Sie auf den ADD-Knopf. Jetzt geben Sie in das Eingabefeld den Text 'ZEIT=', gefolgt von der von Ihnen gewünschten Zeitspanne, nach der die Maus ausgehen soll, ein. Der Text 'ZEIT' sollte in Großbuchstaben geschrieben sein. Wenn Sie das alles korrekt durchgeführt haben, betätigen Sie bitte das 'SAVE'-Feld (ganz unten links) mit der linken Maustaste. Die modifizierte Icon-Information wird jetzt auf die Diskette gespeichert. Wenn Sie jetzt das Programm von der Workbench aus starten, wird als Zeitverzögerung der von Ihnen zuvor definierte Wert genommen.

Ich kündigte weiter oben drei Möglichkeiten, das Programm zu starten, an. Die dritte ist mit der ersten Möglichkeit eng verwandt. Dennoch gibt es hier einen gewaltigen Unterschied. Durchleuchten wir doch noch einmal die erste Möglichkeit. Das Programm wird durch die Eingabe seines Namens gestartet. Der CLI-Process, in dem das Programm MausAus läuft, wartet solange, bis das Programm beendet wird. Was natürlich nie passiert, da es in einer Endlosschleife läuft. Schlecht daran ist, daß das CLI-Fenster unnütz 'herumliegt'. Es ist ja vom Programm MausAus belegt. Außer es in den Hintergrund zu klicken und in einem anderen Fenster tätig zu werden, bleibt uns anscheinend wenig übrig. Aber halt - da war doch noch was? Richtig, der CLI-Befehl, der einen neuen CLI-Task erzeugt, ihm aber kein eigenes Fenster schenkt. Das ist schließlich genau das, was wir hier brauchen. Aber auch das hat eine Schattenseite, die man aber mit dem Programm MausAus nicht sieht. Ich hoffe, Ihnen gefällt MausAus so gut, daß Sie es in der Startup-Sequence Ihrer Boot-Diskette einbauen. Doch ein Problem könnte auftreten: Sie haben beispielsweise ein Programm mit Run in der Startup-Sequence eingebunden. Die Startup führt es in gewohnter Manier

```
58:  extern void                    CloseWindow();
59:  extern void                    FreeDiskObject();
60:
61:  /*  Die zwei sind im Assemblerteil  */
62:
63:  extern void                    Handler();
64:  extern void                    InitHandler();
65:
66:  /*  Beim Amiga gibts immer globales */
67:
68:  struct NewWindow nw =
69:      { 0,0,300,11,0,1,CLOSEWINDOW,
70:        WINDOWDEPTH|WINDOWCLOSE|WINDOWDRAG,
71:  ~     NULL,NULL,(STRPTR)"MausAus von Jürgen Mülbert,
                                  KICKSTART 1989",
72:        NULL,NULL,0,0,0,0,WBENCHSCREEN
73:      };
74:
75:  ULONG  *IconBase = 0;           /* Die IconBase wird für die WB */
76:                                  /* gebraucht. */
77:  int    zeit      = 0;
78:  ULONG  windsig   = 0;
79:
80:  ULONG  ev_signal;               /* Die zwei werden im Assembler- */
81:  struct Process *meintask;       /* teil benötigt */
82:  struct Window  *win;
83:  struct MemEntry       me[10];   /* Etwas Platz für Daten der
                                       interrupt Routine   */
84:  struct MsgPort       *inp_msg, *tim_msg;
85:  struct IOStdReq      *inp_req;
86:  struct timerequest   *tim_req;
87:  struct Interrupt     i_handl;
88:  struct IntuitionBase *IntuitionBase;
89:
90:  /*
91:   *  In der Funktion ifback() wird festgestellt
92:   *  ob das Programm vom CLI aus gestartet wurde.
93:   *  Trifft das zu wird überprüft ob es als
94:   *  Hintergrund Process läuft (mit RUN gestartet)
95:   *  und das Ergebnis (TRUE wenn BACKGROUND) zu-
96:   *  rückgegeben
97:   */
98:
99:  BOOL ifback()
100: {
101:     register BOOL flag = 0;
102:     register struct CommandLineInterface *cli;
103:
104:     if (meintask->pr_CLI)
105:      {
106:       cli = (struct CommandLineInterface *)
107:             ((long)meintask->pr_CLI << 2);
108:       flag = cli->cli_Background;
109:      }
110:
111:     /* Das DOS hat eine etwas andere Aufassung von
112:        TRUE und FALSE. Deswegen die Umwandlung */
113:
114:     if (flag == DOSTRUE) return(TRUE);
115:     else                 return(FALSE);
116: }
117:
118: /*
119:  *  Die Routine _wb_parse() wird beim AZTEC im Startup-Teil
120:  *  aufgerufen. Allerdings überprüft die ob der USER wünscht
121:  *  das ein Fenster geöffnet wird. Für dieses Programm soll
122:  *  kein Fenster geöffnet werden.
123:  */
124:
125: void _wb_parse(pp,wbm)
126: register struct Process *pp;
127: struct WBStartup *wbm;
128: {
129:     void cleanup();
130:     register struct DiskObject *dop;
131:     register struct FileHandle *fhp;
132:     register STRPTR cp;
133:     register long wind;
134:     if  (IconBase = (ULONG *)OpenLibrary("icon.library",0L)) == 0)
135:         return;
136:     if ((dop = GetDiskObject(wbm->sm_ArgList->wa_Name)) == 0)
137:         goto fertig;
138:     if (cp = FindToolType(dop->do_ToolTypes,"ZEIT"))
```

aus. Aber warum geht das verflixte CLI-Fenster nicht zu, obwohl 'endcli' aufgerufen wurde? Ich erkläre es Ihnen. Das AmigaDos verwaltet zu jedem Fenster eine Liste der in ihm laufenden Programme. Wenn Sie ein Programm in einem CLI-Task normal starten, werden alle normalen Ein- und Ausgaben in diesem Fenster stattfinden. Deswegen können Sie jetzt nichts mehr eingeben, bzw. Ihre Eingaben werden erst beachtet, wenn das Programm beendet wurde. Anders ist es aber, wenn Sie ein Programm mit RUN starten. Jetzt erzeugt das DOS einen neuen Task, wie an der Ausgabe [CLI 2] unschwer festzustellen ist. Für jeden Task braucht das AmigaDos einen Ein- und Ausgabebereich. Da bei dem Befehl RUN kein Fenster geöffnet wird, haben die Programmierer hier getrickst. Dem im Hintergrund laufenden Task wird das Window, aus dem er gestartet wurde, als Ein- und Ausgabemedium zur Verfügung gestellt. Damit aber die Zeiger nicht versehentlich verschwinden, wird notiert, daß der 'Background-Task' jetzt berechtigt ist, hier seine Eingaben zu erhalten und seine Ausgaben zu machen.

Also steht das Fenster jetzt für zwei Jobs zur Verfügung. Es kann nur geschlossen werden, wenn der im Hintergrund laufende Task nicht mehr darauf zugreifen muß. Dazu muß er ganz einfach beendet werden. Da nützt es gar nichts wenn, Sie ganz lässig ENDCLI eingeben, oder wenn in der Batchdatei der Befehl ENDCLI steht. Aber wie heißt es doch so schön: "Gefahr erkannt, Gefahr gebannt!". Wenn Sie das Fenster schließen wollen, nachdem Sie ein Programm mit RUN gestartet haben, müssen Sie nur den In- und den Output, der vom DOS zur Verfügung gestellt wurde, schließen. Schon läßt sich auch das Fenster bei Bedarf schließen. Toll, was?!. Aber wehe Ihnen, Sie beenden danach auch noch das Programm, mit dem Sie solche Sachen gemacht haben. Der GURU wird über Sie kommen.

MausAus - MausEnde

Wenn Sie MausAus ganz normal vom CLI aus starten, können Sie es mit der Tastenkombination CTRL-C beenden. Haben Sie es von der Workbench aus gestartet, wird das auch erkannt.

```
139:        zeit = atoi(cp);
140:      FreeDiskObject(dop);
141:
142:  fertig:
143:      CloseLibrary(IconBase);
144:
145:      /*  Wenn das Programm von der WB gestartet wurde
146:          wird die Intuition geöffnet und ein Fenster
147:          das nur so hoch wie die Titelleiste ist. Dieses
148:          Fenster hat den Ausschaltknopt für das Programm
149:          zu beenden
150:      */
151:
152:      if ((IntuitionBase = OpenLibrary("intuition.library",0L))
                                                        == NULL)
153:       exit(20);
154:
155:      if ((win = (struct Window *)OpenWindow(&nw)) == NULL)
156:        cleanup(1);
157:
158:      windsig = 1L <<win->UserPort->mp_SigBit;
159:  }
160:
161:  /*
162:   *  Alles zurück zum AMIGA
163:   */
164:
165:  void cleanup(fehler)
166:  int fehler;
167:  {
168:      switch(fehler)
169:       {
170:        case  0:
171:          AbortIO(tim_req);
172:          CloseDevice(tim_req);
173:        case  6:
174:           FreeMem(tim_req,(long)sizeof(struct timerequest));
175:        case  5:
176:          DeletePort(tim_msg);
177:        case  4:
178:          inp_req->io_Command = IND_REMHANDLER;
179:          DoIO(inp_req);
180:          CloseDevice(inp_req);
181:        case 3:
182:          DeleteStdIO(inp_req);
183:        case 2:
184:          if (win) CloseWindow(win);
185:          DeletePort(inp_msg);
186:        case 1:
187:          FreeSignal(ev_signal);
188:          if (IntuitionBase) CloseLibrary(IntuitionBase);
189:       }
190:
191:      exit(fehler);
192:  }
193:
194:  /*
195:   * Alle Resourcen allokieren
196:   */
197:
198:  void openall()
199:  {
200:      InitHandler();
201:
202:      if ((inp_msg = CreatePort(0L,0L)) == NULL)
203:       cleanup(1);
204:
205:      if ((inp_req = CreateStdIO(inp_msg)) == NULL)
206:       cleanup(2);
207:
208:      i_handl.is_Data = (APTR)&me[0];
209:      i_handl.is_Code = Handler;
210:      i_handl.is_Node.ln_Pri = 51;
211:
212:      if (OpenDevice("input.device",0L,inp_req,0L))
213:       cleanup(3);
214:
215:      if ((tim_msg = CreatePort(0L,0L)) == NULL)
216:       cleanup(4);
217:
218:      if ((tim_req = (struct timerequest *)
219:            AllocMem((long)sizeof(struct timerequest),
                              MEMF_PUBLIC|MEMF_CLEAR))
```

Vier Kontrollen

Vom Programm werden jetzt noch vier zusätzliche Aktionen durchgeführt. Komfort fordert seinen Preis. Zuerst wird abgecheckt, ob Sie eine andere Zeitspanne wünschen, und entsprechend reagiert. Dann werden die Intuition Library und ein Fenster geöffnet. Dieses Fenster meldet sich, wenn Sie es ausschalten. Wie Sie sich denken können, ist das das Startzeichen, das Programm zu beenden. Wenn Sie das Programm mit RUN gestartet haben, können Sie es nicht beenden. Keine Sorge, auch nicht versehentlich. Dafür habe ich gesorgt.

Lattice C

Damit die Leute, die meinen, der LATTICE-Compiler wäre besser, es leichter mit der Umstellung haben, habe ich alle externen Funktionen explizit definiert. Ein paar Dinge gibt es hier noch besonders zu beachten. Wenn Sie das Programm mit dem LATTICE compilieren wollen, müssen Sie im Maschinenspracheteil den Befehl JSR _geta4# ersatzlos streichen. Diesen Befehl finden Sie am Beginn der zwei Subroutinen. Außerdem müssen die 'MOVEMs' an den Einsprung und Ausstiegspunkten überprüft werden. Auch der Fehlerausstieg in der Subroutine InitHandler muß korrigiert werden.

Beim AZTEC sind die ints normal 16 Bit lang. Deswegen müssen die LATTICE-Benutzer nach dem LABEL 'ende' den MOVE.W und MOVE.L umändern. Die anschließende Stackkorrektur können Sie so lassen.

Der C-Teil

Soviel zum Assemblerteil. Beim C-Teil gibt es eine Routine, die nicht aufgerufen wird. So sieht es auf jeden Fall aus. Daß ich mir aber nicht nur so die Arbeit mache und so einen Teil eingebe, ist ja auch klar. Der Teil *_wb_parse()* wird beim Startup des Programms vom AZTEC in der Routine *_main()* aufgerufen, falls das Programm von der Workbench gestartet wurde. Hier müssen sich die LATTICE-Benutzer etwas einfallen lassen. Auch nach der Überprüfung, ob das Programm als Background-Process läuft *[ifback()]*, sind Änderungen notwendig. Vielleicht ge-

```
220:          == NULL)
221:        cleanup(5);
222:
223:      tim_req->tr_node.io_Message.mn_Node.ln_Type = NT_MESSAGE;
224:      tim_req->tr_node.io_Message.mn_ReplyPort    = tim_msg;
225:
226:      if (OpenDevice(TIMERNAME,UNIT_VBLANK,tim_req,0L))
227:        cleanup(6);
228:
229:      inp_req->io_Command = IND_ADDHANDLER;
230:      inp_req->io_Data    = (APTR)&i_handl;
231:
232:      DoIO(inp_req);
233:  }
234:
235:  /*
236:   *  Hier wird der Timer jede Sekunde neu gestartet
237:   */
238:
239:  void QueueTimer(timer)
240:  register struct timerequest *timer;
241:  {
242:      timer->tr_node.io_Command  = TR_ADDREQUEST;
243:      timer->tr_time.tv_secs     = 1;
244:      timer->tr_time.tv_micro    = 0;
245:      SendIO(timer);
246:  }
247:
248:  /*
249:   *  Ein kurze Hinweis für die Benutzung
250:   */
251:
252:  void usage()
253:  {
254:      static char U[] ="MausAus\nUSAGE: MausAus [Zeit]
                                          default 10 Sekunden\n";
255:      Write(_devtab[1].fd,U,(long)strlen(U));
256:      exit(1);
257:  }
258:
259:  /*  Hier gehts los */
260:
261:  main(argc,argv)
262:  int  argc;
263:  TEXT *argv[];
264:  {
265:      register ULONG sigs,timesig,sig;
266:      register BOOL  Maus = TRUE;
267:      register UWORD interval = 0;
268:
269:      if (argc > 1)
270:       {
271:        if (*argv[1] == '?') usage();
272:        else     zeit = atoi(argv[1]);
273:       }
274:
275:      if (zeit == 0) zeit = AUSZEIT;
276:
277:      openall();
278:
279:      /* Kein wichtiger Task  */
280:      SetTaskPri(meintask,-1L);
281:
282:  /*  Bei RUN werden alle geöffneten Ein und Ausgabe Kanäle
                                                  geschlossen */
283:
284:      if (ifback() == TRUE)
285:       {
286:        _devtab[0].mode &= ~O_STDIO;
287:        _devtab[1].mode &= ~O_STDIO;
288:        close(0);
289:        close(1);
290:        close(2);
291:        sigs = 0; /* Und CONTROL_C wird gesperrt */
292:       }
293:      else sigs = SIGBREAKF_CTRL_C;
294:
295:      QueueTimer(tim_req);
296:
297:      timesig = 1L <<tim_msg->mp_SigBit;
298:      sigs    |= ev_signal | timesig | windsig;
299:
300:      FOREVER
301:       {
```

AZTEC-C FÜR AMIGA VERSION 3.6

Wußten Sie, daß eines der verbreitesten und komplexesten Betriebssysteme - UNIX - in C geschrieben ist ?



Wußten Sie, daß auch das Betriebssystem des AMIGA größtenteils in C geschrieben wurde ?

Wußten Sie, daß C eine der wichtigsten und modernsten Programmiersprachen ist ?

Wußten Sie, daß Aztec-C einer der schnellsten und leistungsfähigsten Compiler für den AMIGA ist ?

Wußten Sie, daß jetzt Aztec-C in der Version 3.6 verfügbar ist ?

Möchten Sie mehr darüber wissen? Dann schicken Sie uns einen ausreichend frankierten Rückumschlag und Sie erhalten ausführliche Information.

HIERMIT BESTELLE ICH:

- ☐ AZTEC-C68K/AM-P PROFESSIONAL SYSTEM FÜR DM 298,-
- ☐ AZTEC-C68K/AM-D DEVELOPER SYSTEM FÜR DM 598.-
- ☐ AZTEC-SDB SOURCE LEVEL DEBUGGER FÜR DM 149.-

Versandkosten: Inland DM 7,50 Ausland DM 10,-
Auslandbestellungen nur gegen Vorauskasse
Nachnahmegebühr DM 3,70

- ☐ Vorauskasse
- ☐ Nachnahme

NAME: _____

VORNAME: _____

STRAßE: _____

ORT: _____

UNTERSCHRIFT: _____

AZTEC-C IST EIN EINGETRAGENES WARENZEICHEN VON MANX SOFTWARE SYSTEMS

MAXON

MAXON COMPUTER GMBH
INDUSTRIESTRAßE 26
6236 ESCHBORN
TEL. 06196/481811

nügt es hier einfach, *Close[Input()]* bzw. *Close[Output()]* einzugeben. Ansonsten müßte alles klargehen.

Um dem Programm MausAus ein Icon zu geben, kopieren Sie einfach eine Datei, die die Endung '.info' besitzt, um.

Beispiel:

```
copy CLI.info to MausAus.info
```

Das eigentliche Programm MausAus muß sich natürlich im betreffenden Verzeichnis befinden.

Den Nutzen von MausAus werden Sie schnell selber feststellen können. Für Tippfaule steht unser Diskettenservice zur Verfügung.

```
302:        sig = Wait(sigs);
303:
304:        if (sig & SIGBREAKF_CTRL_C) break;
305:        if (sig & windsig)
306:         {
307:          ReplyMsg(GetMsg(win->UserPort));
308:          goto fertig;
309:         }
310:        else if (sig & timesig)
311:         {
312:          GetMsg(tim_msg);
313:          if (Maus == TRUE && ++interval > zeit)
314:           {
315:            OFF_SPRITE;
316:            Maus = FALSE;
317:           }
318:          if (Maus == TRUE) QueueTimer(tim_req);
319:
320:     }
321:        else if (sig & ev_signal)
322:         {
323:          if (Maus == FALSE)
324:           {
325:            Maus = TRUE;
326:            ON_SPRITE;
327:            interval = 0;
328:            QueueTimer(tim_req);
329:           }
330:         }
331:       }
332:  }
333:  fertig:
334:      /* Falls die Maus noch versteckt ist wird
335:         sie wieder enttarnt.     */
336:      if (Maus == FALSE) ON_SPRITE;
337:      cleanup(0);
338:  } /* LISTING MAUSAUS Ende */
```

```
 1:  ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
 2:  ;   Handler für die InputEvents
 3:  ;   Autor: Jürgen Mülbert
 4:  ;   (c) MAXON Computer GmbH 1989
 5:  ;   KICKSTART 1989
 6:  ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
 7:
 8:  ;   INCLUDES nicht Listen
 9:
10:   NOLIST
11:
12:       include "exec/types.i"
13:       include "exec/lists.i"
14:       include "exec/nodes.i"
15:       include "exec/tasks.i"
16:       include "exec/ports.i"
17:       include "exec/io.i"
18:       include "exec/libraries.i"
19:       include "devices/timer.i"
20:       include "devices/inputevent.i"
21:
22:       LIST
23:
24:  ;   Die benötigten Bibliotheks Funktionen
25:
26:  Signal         EQU -324
27:  AllocSignal    EQU -330
28:  FindTask       EQU -294
29:
30:       SECTION Handler,CODE
```

```
31:
32:       XDEF    _Handler
33:       XDEF    _InitHandler
34:
35:  ;    Die sind alle beim C-Teil definiert
36:
37:       XREF    _SysBase
38:       XREF    _IntuitionBase
39:       XREF    _cleanup
40:       XREF    _ev_signal
41:       XREF    _meintask
42:
43:  ;    Hier geht los
44:
45:  _InitHandler:
46:
47:       movem.l d2/d3/a4,-(sp)
48:       jsr     _geta4#
49:       move.l  _SysBase,a6
50:       move.l  #0,a1
51:       CALLLIB FindTask
52:       move.l  d0,_meintask
53:       moveq   #-1,d0
54:       CALLLIB AllocSignal
55:       cmpi    #-1,d0          Wenn kein Signal
                                  mehr frei ist
56:       beq     ende            wird das Programm
                                  beendet
57:       clr.l   d1
58:       bset    d0,d1
59:       move.l  d1,_ev_signal
60:       movem.l (sp)+,d2/d3/a4
61:       rts
62:
63:  ende:
64:       move.w #20,-(sp)
65:       jsr _cleanup
66:       add.w   #2,sp
67:       rts             Wird nie erreicht!
68:
69:  ;    Hier ist der wichtigste Teil!!
70:
71:  _Handler:
72:       movem.l d2/d3/a4,-(sp)
73:       jsr     _geta4#
74:       move.l  a0,-(sp)
75:       clr.l   d0
76:       move.b  ie_Class(a0),d0
77:       cmpi    #IECLASS_RAWMOUSE,d0  Wenn was mit
                                        der Maus passiert
78:       bne.s   1$                    den Task wecken
79:       move.l  _ev_signal,d0
80:       move.l  _meintask,a1
81:       move.l  _SysBase,a6
82:       CALLLIB Signal
83:  1$:
84:       move.l  (sp)+,d0          Event weitermelden
85:       movem.l (sp)+,d2/d3/a4
86:       rts
87:  ;Ende Listing Handler
```

Der Assemblerhandler für die Input Events

END of KICK

VON SVEN OBER

Dem C-Compiler Beine gemacht



Das Programm 'Fasm' ist ein Utility für den Aztec C-Compiler. Bei vielen Programmen, die mit Fließkommaarithmetik arbeiten, ist die Rechengeschwindigkeit ein Problem. Sie läßt sich um 10-20 % steigern, wenn man sich nicht mit dem C-Programm begnügt.

Doch zunächst ein paar Grundlagen. Der Aztec C-Compiler übergibt die Floats zwar in Registern - aber leider nicht an die Rechenroutine, sondern an eine Routine, die erst den Offset der Rechenroutine auf den Stack schreibt.

Dann wird überprüft, ob die *mathffp.-library* schon offen ist.

Ist sie geöffnet, wird der Library-Pointer in das Register A6 geschrieben und mit dem zuvor geholten Offset angesprungen.

Diese Prozedur benötigt viel kostbare Zeit.

Mit *Fasm* ist es nun möglich, das vom C-Compiler erzeugte Assemblerlisting nachzubehandeln. Das Programm ruft dann ohne Umweg über die Compilerroutinen die Systemroutinen der *mathffp.library* auf und steigert so die Ablaufgeschwindigkeit.

Man muß in seinem C-Programm nur die *mathffp.library* öffnen, da der Compiler nicht mehr abfragen kann, ob sie schon offen ist.

```
MathBase=(struct Math Base*)
OpenLibrary("mathffp.library,0L")
```

wobei '*MathBase*' global als Zeiger auf die Mathbase-Struktur deklariert sein muß. Nun übersetzt man nur noch sein C-Programm mit der -a-Option, um das '*.asm*'-*File* zu erhalten. Mit dem Aufruf

```
Fasm name.asm
```

ist die Arbeit auch schon vorbei. Es hat sich scheinbar nichts geändert, sieht man sich aber das '*.asm*'-*File* an, fallen die neuen Aufrufe sofort ins Auge.

Das erhaltene File muß noch assembliert und gelinkt werden:

```
as name
ln name.o -lc
```

Die *m.lib* kann man sich jetzt sparen, wenn man nur mit den Floats rechnen möchte. Will man aber die Zahlen mit *printf()* ausgeben, muß die *m.lib* nach wie vor hinzugelinkt werden.

Das fertige Programm ist deutlich schneller als das vorherige.

Die Geschwindigkeitssteigerung beträgt 20%, sind komplexe Floatrechenoperationen angesagt, machen sie sich bemerkbar.

In dem vorliegenden Programm werden nur die Aufrufe der *mathffp.library* geändert. Mit einigen kleinen Änderungen ist es leicht möglich, auch die Aufrufe für andere Libraries zu beschleunigen.

Nun noch ein Tip für bestimmte, häufig vorkommende Rechenoperationen.

Diese funktionieren aber nur mit FFP-Zahlen des AMIGA. Man kann bei Multiplikationen und Divisionen durch Potenzen von 2 einen kompletten Rechenvorgang sparen!!

Dieses kleine Programm wird das verdeutlichen.

```
main()
{
  float a,b;
  a=123.323;
  b=a/2;
}
```

Der C-Compiler macht aus dem Ausdruck *a/2*

```
move.l  #$80000042,d1
move.l  .x,d0       .x ist eine lokale Variable
                    (z.B.4)
jsr     .Fdiv#
move.l  d0,.y
```

Für den speziellen Fall der Zweierpotenzen geht es aber einfacher und schneller:

```
move.l  .x,d0
sub.l   #1,d0
move.l  d0,.y
```

Dieses ergibt eine Division durch 2. Eine Division durch 4 erhält man, wenn man 2 subtrahiert. Bei Multiplikationen muß man nicht subtrahieren, sondern addieren. Das funktioniert wegen des Formats der AMIGA-FFP-Zahlen. Eine FFP-Zahl ist 32 Bit breit. Die Bits 0-6 enthalten den Exponenten, Bit Nr. 7 gibt das Vorzeichen an (0 = positiv, 1 = negativ). Die eigentliche Zahl ist in den Bits 8-31 codiert. Mit der Subtraktion bez. Addition vom Exponenten wird nur das "Komma" um die subtrahierte/addierte Anzahl von Bits verschoben. Damit erreicht man eine Beschleunigung des Rechenvorganges um ein Vielfaches, weil jetzt nur noch ein *add.l*- bzw. *sub.l*-Befehl benötigt wird, und keine Fließkommaroutine mehr aufgerufen werden muß.

Der Aufruf zum Compilieren von *Fasm* ist für den Aztec C-Compiler:

```
cc Fasm
ln Fasm.o -lc
```

Das kleine C-Programm können Sie als Testdatei ausprobieren. Sie werden überrascht sein über die Geschwindigkeitssteigerung.

```
 1:  /* Fasm.c   von Sven Ober November 88 *
 2:  Ein Utility für den Aztec C Compiler */
 3:
 4:  #include <exec/types.h>
 5:  #include <exec/memory.h>
 6:  #include <functions.h>
 7:  #include <libraries/dos.h>
 8:
 9:  long filegroesse;
10:  char *namzei;
11:  char *dread();
12:  long checksize();
13:  void checkfile();
14:  char mathb[]={"move.l  _MathBase,a6"};
15:  char js[]={" jsr "};
16:  char mul[]={"-78(a6)"};
17:  /* offsets der FFP Routinen */
18:  char add[]={"-66(a6)"};
19:  char ab[]={"-54(a6)"};
20:  char cmp[]={"-42(a6)"};
21:  char div[]={"-84(a6)"};
22:  char fix[]={"-30(a6)"};
23:  char flt[]={"-36(a6)"};
```

```
24:  char neg[]={"-60(a6)"};
25:  char sub[]={"-72(a6)"};
26:  char tst[]={"-48(a6)"};
27:
28:  main(argc,argv)
29:  int argc;
30:  char *argv[];
31:   {
32:  if(argc!=2) /* Filenamen angegeben*/
33:
34: {
35:    Write(Output(),"Usage Fasm <file>\n",18L);
                    /*Ausgabe auf das CLI- */
36:    cleanup();     /*Fenster ohne printf()*/
37:   }
38:  namzei=dread(argv[1]);
39: /* File öffnen und in den Speicher laden*/
40:   checkfile(argv[1]);
41:  }
42:
43:  char *dread(file)
44:    char *file;
45:     {
```

```
 46:       struct FileHandle *fh;
 47:       char *bp;
 48:       long len;
 49:
 50:       bp = (char *)AllocMem((LONG)
           filegroesse=checksize(file)),MEMF_CLEAR);
 51:       if((fh=(struct FileHandle
                 *)Open(file,MODE_OLDFILE))==0L)
 52:         {
 53:           Write(Output(),"File nicht gefunden \
           n",21L);
 54:           cleanup();
 55:         }
 56:       len = Read(fh,bp,filegroesse);
 57:       Close(fh);
 58:       return(bp);
 59:     }
 60:
 61:  void checkfile(file)
 62:   char *file;
 63:   {
 64:    register char *unters;  /* unters ist der
       zeiger auf das */
 65:    register char *rep; /* zu untersuchende
       File  */
 66:    register long b=0;
 67:    register long f,err;
 68:
 69:    struct FileHandle *fh;
 70:
 71:    char ret=10;/* 10 ist der code für return*/
 72:     unters=namzei;
 73:     f=0;
 74:     if(!(fh=Open(file,MODE_NEWFILE)))
 75:      cleanup();
 76:
 77:  /* Die folgende Schleife durchsucht das File
       nach z.B.
 78:     Fmul# es wird dabei aber zunächst nur auf
       F und # geprüft */
 79:     do
 80:      {
 81:       if((*unters == 'F') && (unters[4]=='#'))
 82:        {
 83:   unters++;
 84:   switch(unters[0])
 85:    {
 86:     case 'm':                    /* Fmul */
 87:      {rep=mul;break;}
 88:     case 'd':                    /* Fdiv */
 89:      {rep=div;break;}
 90:     case 's':                    /* Fsub */
 91:      {rep=sub;break;}
 92:     case 'c':                    /* Fcmp */
 93:
 94:  {rep=cmp;break;}
 95:     case 't':                    /* Ftst */
 96:
 97:  {rep=tst;break;}
 98:     case 'n':                    /* Fneg */
 99:
100:  {rep=neg;break;}
101:     case 'a':                    /* Fadd oder
      Fabs */
102:
103:  {
104:       switch(unters[1])
105:        {
106:         case 'd':                /* Fadd */
107:          {rep=add;break;}
108:         case 'b':                /* Fabs */
109:          {rep=ab;break;}
110:              }
111:       break;
112:      }
113:     case 'f':
114:      {
115:       switch(*(unters+1))
116:        {
117:         case 'i':                /* Ffix */
118:          {rep=fix;break;}

119:         case 'l':                /* Fflt */
```

```
120:           {rep=flt;break;}
121:         }
122:       break;
123:      }
124:    }
125:   err=Write(fh,&namzei[b],(long)(f-b-5L));
126:   err=Write(fh,mathb,20L);
127:   err=Write(fh,&ret,1L);
128:   err=Write(fh,js,16L);
129:   err=Write(fh,rep,7L);
130:   f++;
131:   b=f+4;
132:        }
133:       else
134:        {
135:   unters++;f++;
136:        }
137:      }while(*unters);
138:   err=Write(fh,&namzei[b],(long)(filegroesseb));
139:    Close(fh);
140:   }
141:
142:  cleanup()
143:   {
144:    if(namzei) FreeMem(namzei,filegroesse);
145:    Exit();
146:   }
147:
148:  LONG checksize(name)
149:   /* Bestimmt die grösse eines Files   */
150:   char *name;   /* die Grösse wird in err
      zurückgegeben */
151:  {
152:   register long err;
153:
154:   struct FileHandle *fh;
155:   if((fh=(struct FileHandle *) Open (name,
      MODE_OLDFILE))==0L)
156:        {
157:          Write(Output(),"File nicht gefunden \
           n",21L);
158:          cleanup();
159:        }
160:   err=Seek(fh,0L,OFFSET_END);
161:   err=Seek(fh,0L,OFFSET_END);
162:   Close(fh);
163:   return(err);
164:  } /* LISTING ENDE */
```

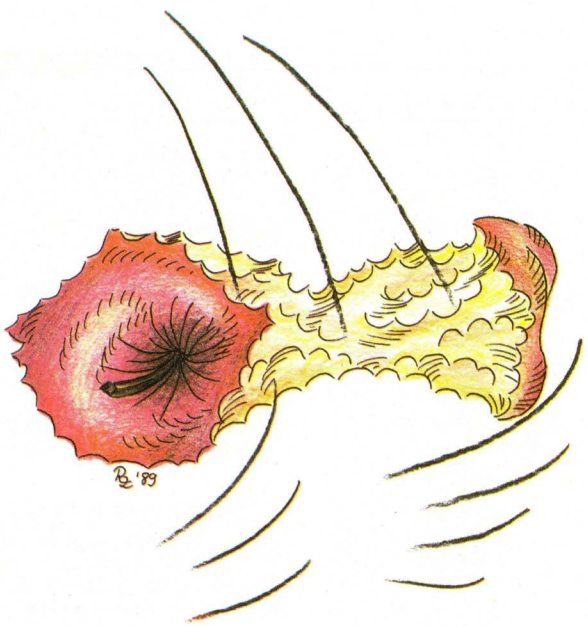
```
  1:  /* Mit diesem kleinen Beispielprogramm können
         Sie die
  2:     Geschwindigkeitssteigerung leicht selbst
         feststellen */
  3:
  4:  #include <functions.h>
  5:
  6:  struct MathBase *MathBase;
  7:
  8:  main()
  9:  {
 10:   static  float a,b,c=5;
 11:   register long k;
 12:   MathBase=(struct MathBase *) OpenLibrary
       ("mathffp.library",0L);
 13:     c=a=400.44;
 14:     k=20000;
 15:     do{
 16:     b=a/c;           /* ca 8 sec */
 17:     b=a*c;
 18:     b=a-c;
 19:     b=a+c;
 20:     k--;}while(k);
 21:    printf("%f\n",b);
 22:   CloseLibrary(MathBase);
 23:  }
 24:
```

END of KICK

VON ACHIM KÄMMLER

Bis an´s bittere Ende

Apfelmännchen in Assembler



*Schon wieder Apfelmännchen... Doch diesmal in 100 % Assembler (KICK-ASS) mit direkter Ansprache der Hardware (wo's was bringt). Das ist schnell: Das Standardbild der Mandelbrotmenge (xmin = -0,7; xmax = 2,1; ymin = -1; ymax = 1; Tiefe 32; 320 * 200 Punkte) wird in 55 Sekunden berechnet. Wer interessiert ist, der lese weiter.*

Die relativ hohe Geschwindigkeit des Programms liegt darin begründet, daß es nur mit Longwords rechnet und den Speicher der Bitplanes direkt anspricht. Aus der Verwendung von Longwords folgt jedoch der größte Nachteil der Routine, nämlich daß der Ausschnitt der komplexen Zahlenebene, der dargestellt werden kann, derart beschränkt ist, daß Höhe und Breite des Bildes auf mindestens 1 (max - min > 1) beschränkt sind, das heißt, es können kaum Ausschnittsvergrößerungen vorgenommen werden. Der Hauptzweck dieses Programms soll

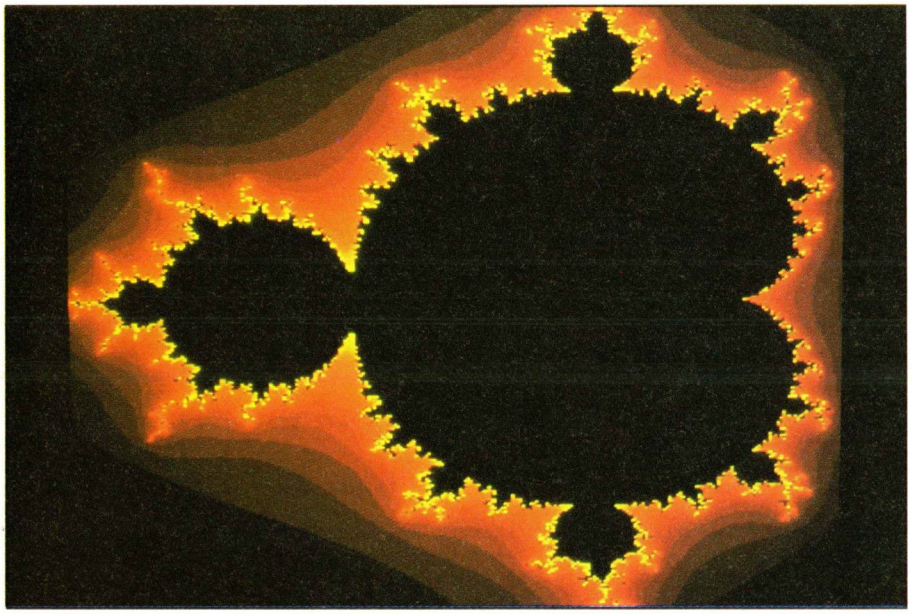
jedoch darin bestehen, das direkte Ansprechen des Speichers für farbige Grafiken zu demonstrieren.

Das kleine Programm öffnet einen Lo-Res-Screen (320x256 Punkte) mit fünf Bitplanes, demnach sind 32 Farben verfügbar. Die Abmessungen der auszugebenden Grafik kann aber geändert werden.

Nach Initialisierung der Farben werden in den Zeilen 70-75 die Zeiger auf die Bitplanes zwischengespeichert, da sie später verändert werden. In den Zeilen 133-137 werden dann die jeweiligen Punkte Plane für Plane in den Speicher 'geschoben'. Die Zeilen 144-148 sorgen dafür, daß (nach 16 Punkten) die Zeiger auf die Planes erhöht werden, d.h. sie zeigen dann immer auf das jeweils berechnete Word. Beendet wird das Programm durch Betätigen der linken Maustaste am Zeilenende oder nach Berechnung des Bildes. Das Programm läßt sich auch über die Workbench starten. Von der Verwendung von FFP-Zahlen wurde abgesehen, da dadurch die Geschwindigkeit der Berechnung in schlafende Sphären dahingleiten würde.

Ein Hi-Res-Screen kann schnell definiert werden, dazu müssen Sie lediglich die Variable 'Hires' bei der Bildschirmdefinition (Screendef) einsetzen, vergessen Sie aber nicht die Tiefe des Bildschirms um eins zu vermindern, die Bildschirmgröße können, aber müssen Sie nicht anpassen.

Natürlich können Sie diese Methode für vielfältige Anwendungen verwenden. Das Programm wurde mit dem KICK-ASS erstellt und besitzt in der startbaren Version eine Länge von 626 Bytes. Sie sehen, die Routine ist nicht sehr lang. Die Anpassung auf andere Assembler dürfte keine Schwierigkeiten bereiten, beim SEKA-Assembler beispielsweise müssen Sie lediglich das Anhängsel '.s' bei einigen Befehlen weglassen.



Das Apfelmännchen wird in 55 Sekunden berechnet.

```
 1:  ; Apfelmännchen
 2:
 3:  ExecBase = 4
 4:  OldOpenLibrary = -408
 5:  OpenScreen = -198
 6:
 7:  ShowTitle = -282
 8:  SetRGB4 = -288
 9:  CloseScreen = -66
10:  CloseLibrary = -414
11:
12:  xmin   = -700
13:  xmax   = 2100
14:  ymin   = -1000
15:  ymax   = 1000
16:  tiefe  = 32
17:  faktor = 1024
18:  xsize  = 352 ;Breite des Screens
19:  ysize  = 282 ;Höhe des Screens
20:  BPlanes = 5
21:  Hires  = -32764
22:
23:  start:
24:         move.l ExecBase.s,a6
25:         lea Graphics(pc),a1 ;Graphics.Library öffnen
26:         jsr OldOpenLibrary(a6)
27:         move.l d0,GFXbase
28:         beq Ende
29:
30:         lea Intuition(pc),a1 ;Intuition.Library öffnen
31:         jsr OldOpenLibrary(a6)
32:         move.l d0,IntuitionBase
33:         beq Ende
34:         move.l d0,a6
35:
36:         lea Screendef(pc),a0 ;Screen öffnen
37:         jsr OpenScreen(a6)
38:         move.l d0,Screen
39:         beq Ende
40:         add.l #44,d0 ;ViewPort ermitteln
41:         move.l d0,ViewPort
42:         add.l #40,d0 ;RastPort ermitteln
43:         move.l d0,RastPort
44:         move.l Screen(pc),a0
45:         moveq #0,d0 ;kein Titel
46:         jsr ShowTitle(a6)
47:         move.l GFXbase(pc),a6
48:         moveq #15,d5 ;Schleife 16 mal durchlaufen
49:  farben:
50:         move.l ViewPort(pc),a0
51:         moveq #0,d2
52:         moveq #0,d3
```

```
 53:         move.l d5,d1
 54:         move.l d5,d0
 55:         jsr SetRGB4(a6) ;Farben 0-15 mit r=d5,
                             g=0, b=0
 56:         move.l ViewPort(pc),a0
 57:         moveq #15,d1
 58:         move.l d5,d2
 59:         move.l d5,d0
 60:         add.l #15,d0
 61:         jsr SetRGB4(a6) ;Farben 15-30 mit r=15,
                             g=d5, b=0
 62:         dbf d5,farben
 63:         move.l ViewPort(pc),a0
 64:         moveq #31,d0
 65:         moveq #15,d1
 66:         move.l d1,d2
 67:         moveq #1,d3
 68:         jsr SetRGB4(a6) ;Farbe 31 mit r=15, g=15,
                             b=1
 69:         move.l Screen(pc),a0
 70:         add.l #$C0,a0   ;Hier steht Zeiger auf
                             Bitplane 0
 71:         lea Plane0(pc),a1 ;Anfang Zwischenspeicher
                               für Bitplanes
 72:         moveq #BPlanes-1,d0
 73:  planes:
 74:         move.l (a0)+,(a1)+ ;Zeiger auf Planes
                                speichern
 75:         dbf d0,planes
 76:
 77:         ; Hier beginnt Algorithmus zur Berechnung
               der Apfelmännchen
 78:
 79:         move.l #xmax-xmin,d0
 80:         move.l #xsize,d1
 81:         divs d1,d0
 82:         move.w d0,a2
 83:         move.l #ymax-ymin,d0
 84:         move.l #ysize,d1
 85:         divs d1,d0
 86:         move.w d0,a3
 87:         move.l #xmin,d6
 88:         move.l #ymax,d7
 89:         lea 0.s,a4
 90:  forz:
 91:         lea 0.s,a1
 92:         btst #6,$BFE001 ;Linke Maustaste gedrückt?
 93:         beq End ;wenn ja, Sprung nach End
 94:  forss:
 95:         lea 0.s,a5
 96:  fors:
 97:         moveq #0,d0
 98:         moveq #0,d1
 99:         moveq #0,d2
100:         moveq #0,d3
101:         moveq #0,d4
102:  while:
103:         move.l #faktor,d5
104:         muls d0,d1
105:         add.l d1,d1
106:         asr.l #5,d1
107:         asr.l #5,d1
108:         sub.l d7,d1
109:         move.l d2,d0
110:         sub.l d3,d0
111:         sub.l d6,d0
112:         move.l d0,d2
113:         muls d2,d2
114:         asr.l #5,d2
115:         asr.l #5,d2
116:         move.l d1,d3
117:         muls d3,d3
118:         asr.l #5,d3
119:         asr.l #5,d3
120:         addq.l #1,d4 ;LrPunktfarbe (=Tiefe des
                          Punktes) erhöhen
121:         cmp.b #tiefe,d4 ;mit max. Tiefe vergl.
122:         bge.s weiter1
123:         asl.l #3,d5
```

```
124:         sub.l d3,d5
125:         sub.l d2,d5
126:         bge.s while
127:         bra.s weiter2
128:  weiter1:
129:         clr.w d4 ;Farbe auf 0 setzen,
                      da tiefe > maxtiefe
130:  weiter2:
131:         lea Plane0(pc),a0 ;Zeiger auf
                                Bitplane 0 in a0
132:         moveq #BPlanes-1,d0
133:  next:
134:         move.l (a0)+,a6 ;Pos. der Bitplane in a6
135:         asr.w #1,d4 ;Farbwert 1 Bit
                         rechts rausschieben
136:         roxl (a6) ;und direkt in Bitplane schieben
137:         dbf d0,next ;nächste Plane
138:         add.l a2,d6
139:         addq.l #1,a5
140:         cmp.l #15,a5
141:         ble.s fors
142:         lea Plane0(pc),a0
143:         moveq #BPlanes-1,d0
144:  next1:
145:         addq.l #2,(a0)+ ;Alle Zeiger auf Planes
                             um 2 (1 word) erhöhen
146:         dbf d0,next1
147:         addq.l #1,a1
148:         cmp.l #xsize/16-1,a1
149:         ble forss
150:         move.l #xmin,d6
151:         sub.l a3,d7
152:         addq.l #1,a4
153:         cmp.l #ysize-1,a4
154:         ble forz
155:
156:  maus:
157:         btst #6,$BFE001 ;Linke Maustaste gedrückt
158:         bne.s maus ; nein, dann Sprung
159:
160:  End:
161:         move.l IntuitionBase(pc),a6
162:         move.l Screen(pc),a0
163:         jsr CloseScreen(a6)
164:         move.l ExecBase.s,a6
165:         move.l IntuitionBase(pc),a1
166:         jsr CloseLibrary(a6)
167:         move.l GFXbase(pc),a1
168:         jsr CloseLibrary(a6)
169:
170:  Ende:
171:         rts
172:
173:         even
174:  Intuition: dc.b 'intuition.library',0
175:         even
176:  Graphics: dc.b 'graphics.library',0
177:         even
178:  Screendef:
179:         dc.w 0,0,xsize,ysize,BPlanes
180:         dc.b 0,1
181:         dc.w 0,15
182:         dc.l 0,0,0,0
183:
184:  Screen: dc.l 0,0
185:         dc.w 0,0,0,0,15
186:  IntuitionBase: dc.l 0
187:  GFXbase: dc.l 0
188:  RastPort: dc.l 0
189:  ViewPort: dc.l 0
190:  Plane0: dc.l 0
191:          dc.l 0
192:          dc.l 0
193:          dc.l 0
194:          dc.l 0
195:          dc.l 0
196:  ;Ende
197:
```

END of KICK

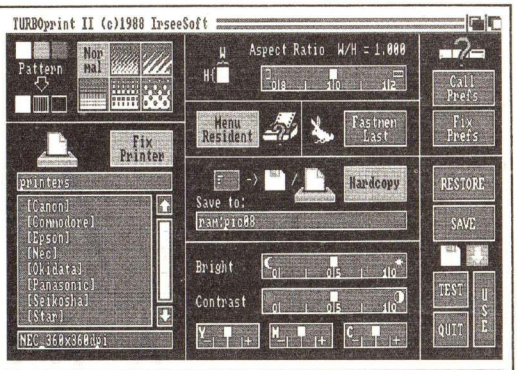
Optimal für Ihren Drucker: **TURBOprint II**



- bessere **Druckqualität** durch **Helligkeits-, Kontrast-** und **Farbregler**
- schneller als Workbench 1.3
- Höhere Auflösung: bis **360x360dpi** bei 24-Nadel Matrix
- **Hardcopy** auf Tastendruck, jetzt auch Ausschnitte
- **Bildabspeicherfunktion** im IFF Format
- schnelle **Glättefunktion** (Antialiasing) beseitigt unschöne Kanten, besonders bei Schriften
- optimale Wiedergabe durch **sechs wählbare Grafikraster**
- komfortable Einstellmöglichkeiten der Ausdruckgröße, auch in Millimetern oder Pixel
- jederzeitiger **Abbruch** des Ausdrucks möglich
- voll **Kompatibilität** zur Amiga-Software
- **superschnelle Übertragung** zum Drucker
- Turbotreiber für alle führenden Druckerfabrikate
- ausführliches deutsches Handbuch

Tunen Sie Ihren Drucker - mit Turboprint

DM 98,–



IRSEE SOFT

IrseeSoft SPCS
Grüntenstraße 6
8951 Irsee
Bestelltelefon: 08341/74327

NN 6,- DM. Vorkasse 4,- DM. Ausland: nur gegen Vorkasse 10,- DM
Händleranfragen erwünscht

Schweiz:	Österreich:
Microtron	**Intercomp**
Bahnhofstr. 2	Heldendankstr. 24
CH-2542 Pieterlen	A-6900 Bregenz
Tel. 032 87 2429	Tel. 05574/27345

Diskettenlaufwerke

vollkompatibel, anschlußfertig, Steuerplatine in SMD-Bautechnik, 2 x 80 Spuren, 1 MB unformatiert, 880 KB formatiert, abschaltbar, amigafarben (durchgeführter Bus: + 10,– DM).

NEC oder TEAC

3,5“ **239.-**

NEC 1037 oder TEAC FD 235 FN abschaltbar, vollkompatibel, Disk-Change-Erkennung.

IBM kompatibel

5,25“ **289.-**

TEAC FD 55 FR, 40/80 Tr. schaltbar, abschaltbar, 1 Jahr Garantie, vollkompatibel, Disk-Change-Erkennung.

Disketten

3,5“ MF 2 DD ab 10 Stück **25.-**
ab 100 Stück **23.-**
3 M oder Fuji **35.-**

NEC 1037 a

oder TEAC FD 235 FN **179.-**

1“ Bauhöhe, sehr leise, mod. Technik, Linearmotor, 3 MS Stepr., 5 V Stromversorg.

Festplatten ▶

FSE Frank Strauß Elektronik

Schmiedstr. 11 6750 Kaiserslautern
Tel. 0631/67096 Fax 0631/60697

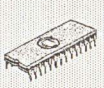
Versandbedingungen: Lieferung erfolgt mit UPS oder DBP per Nachnahme.
Versandkosten: DM 12.- inkl. Transportvers.
Unverbindliche Lieferzeit: 2 Tage

Anschlußfertige Seagate Festplatte mit OMTI Controller im Gehäuse, 1 Jahr Garantie.

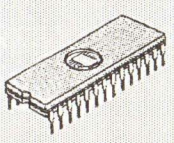
20 MB: 868.-
30 MB: 898.-
40 MB: 1198.-
65 MB: 1498.-

EPEX
DER EPROMEXPRESS

Die Leistungsdaten in Stichworten



- zwei Textoolsockel
- brennt Eproms von 2732-27011 (auch A-Typen)
- fünf Programmieralgorithmen
- eingebauter Monitor
- wortweises brennen oder kopieren möglich
- einstellbares Zahlensystem
- sehr komfortables Steuerprogramm
- Kickstartbrennoption



Preise

Amiga 2000 DM 348.--
Amiga 500 DM 298.--

Tel.: 02232/45018

Tröps + Hierl
Computertechnik GmbH
Jordanstr.3 - 5040 Brühl

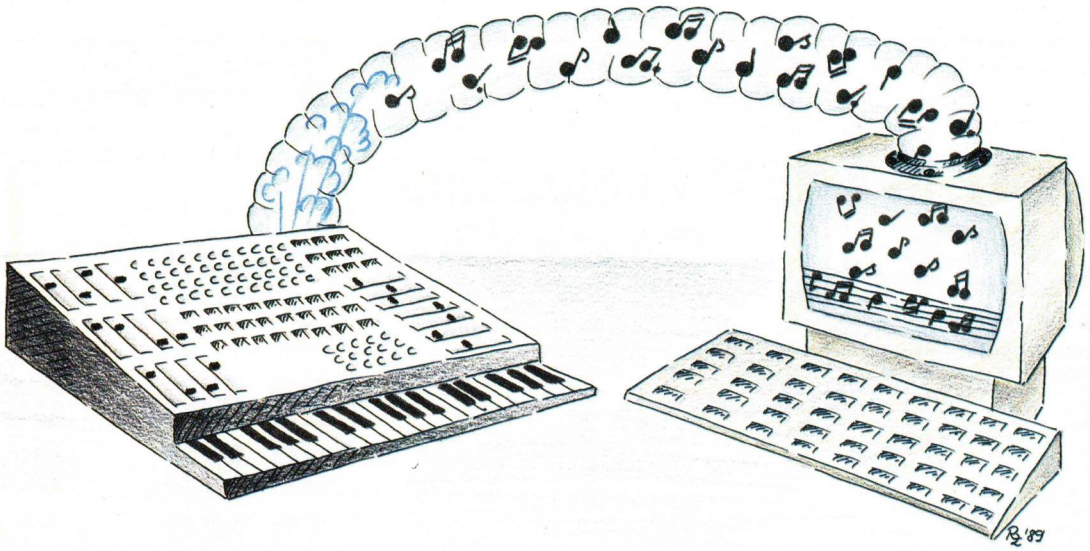
Wir führen die gesamte Amiga Produktpalette.
Fordern Sie unseren Katalog an.
Händleranfragen erwünscht.

tröps+ hierl Computer Technik

VON OLIVER WAGNER

MIDI IN MIDI OUT

CZGET & CZPUT, Casio Library Utility



Die Synthesizer der Casio CZ-Serie sind hervorragende Geräte mit einem gut durchdachten Konzept und angenehmem, "undigitalem" Klang. Doch leider haben die Entwickler den Geräten nur armselige 16 Speicherplätze spendiert (plus 16 Presets, aber lassen wir das), was an Synthesizer der ersten Generation (Promars) erinnert. Dem armen Spieler bleibt also nur übrig, haufenweise Cartridges zu kaufen (sehr im Sinne des Erfinders) oder die Sounds jedesmal per Hand einzugeben. Da man einen Computer zur Verfügung hat, der MIDI-fähig ist (egal, was ST-User auch sagen), bietet sich an, die Sounds mit dem AMIGA zu verwalten.

Das Prinzip

Midi-Interfaces für den AMIGA arbeiten alle nach demselben Prinzip: Man verwandelt die serielle Schnittstelle des AMIGA mit einem Optokoppler, einem 74LS00 (integrierter Chipbaustein) und drei Widerständen in eine Stromschleifenschnittstelle. Das *"serial.device"* des AMIGA-DOS kann dann einfach zur Ansteuerung der Schnittstelle verwendet werden.

Um den Umgang damit zu erleichtern, wurden die beiden Programme *"midi.c"*

und *"midi.h"* entworfen, die Routinen zum einfachen Gebrauch von Midi-Daten enthalten. Auf diese Bibliotheksroutinen greifen "CZGET" und "CZPUT" zurück, die eigentlichen Archivierprogramme. Alle Programme sind in "C" geschrieben und sollten mit dem Aztec C 3.4a oder 3.6 compiliert werden (16 Bit Integer).

CZGET

"CZGET" dient zum Einlesen eines Sounds vom CZ (egal ob CZ-101, CZ-1000 oder CZ-5000) in den Rechner und für Speicherung auf Diskette. Zur einwandfreien Funktion ist das *"serial.device"* im "DEVS"-Ordner der Workbench erforderlich. Der Aufruf für CZGET lautet:

```
CZGET <filename> [<midikanal>
      [<soundnummer>]]
```

Diese Schablone wird auch bei "CZGET ?" oder falschen Parametern ausgegeben. *"<filename>"* ist dabei ein gültiger AMIGA-DOS Dateiname (an die Gänsefüßchen bei Leerzeichen denken!), *"<midikanal>"* ist eine Zahl zwischen 1 und 16 und gibt den Midi-Datenkanal an, auf dem der CZ eingestellt ist (bei Mono-Mode immer der Basiskanal). *"<soundnummer>"* ist die interne Soundnummer des CZ, die bei den verschiedenen Modellen differiert:

CZ-101/1000:	0-15	Preset	1-16
	32-47	Internal	1-16
	64-79	Cartridge	1-16
CZ-5000 :	0- 7	Preset	A1-A8
	8-15	Preset	B1-B8
	16-23	Preset	C1-C8
	24-31	Preset	D1-D8
	32-39	Memory	A1-A8
	40-47	Memory	B1-B8
	48-55	Memory	C1-C8
	56-71	Memory	D1-D8

96 ist bei allen drei Geräten der Sound-Buffer, in den der Synthi bei Anwahl eines Speichers den Sound hereinkopiert, der dort verändert werden kann [ergo der Sound, den man bei Drücken einer weißen (oder auch schwarzen) Taste hören kann]. Wird *"soundnummer"* weggelassen, wird 96 angenommen, wird auch *"midikanal"* ausgelassen, wird Kanal 1 benutzt. "CZGET" liest nun den Sound über die Midi-Schnittstelle ein und speichert ihn auf Diskette (256 Bytes). Evtl. Fehler wer-

```
 1:  /*  CZPUT.C
 2:      Program zum senden von Sound-Daten an Casio CZ
 3:      Synthesizer.
 4:      Autor: Oliver Wagner
 5:      (C) Maxon Computer GmbH 1989
 6:      KICKSTART 1989
 7:      Version für Aztec 3.4a oder 3.6
 8:      CC czput
 9:      CC midi
10:      LN czput.o midi.o -lc
11:  */
12:
13:  #include <midi.h>
14:  #include <libraries/dos.h>
15:
16:  unsigned char buff[256];
17:  unsigned char rr1[] = {
18:   BSYSTEM_EXCLUSIVE,ID_CASIO,
19:   0x00,0x00,
20:   0x6f, /* channel */
21:   0x20,
22:   0x00
23:  };
24:
25:  main(argc,argv)
26:  register int argc;
27:  char **argv;
28:  {
29:   register int x,ch=1,sn=96;
30:   register long f,c;
31:
32:   puts("\x9b1mCASIO CZ-SERIES SOUND PUT UTILITY  Written 1988 by
                  O.Wagner (C)MAXON Computer\x9b0m");
33:   if(argc<2 || argc>4 || *argv[1]=='?') {
34:    puts("Usage: CZPUT filename [midichannel(1-16,def 1)
                                    [soundnr(32-96,def 96)] ]\n\
35:  Soundnr. assignment:\n\ CZ 101/1000: 32-47 Internal / 64-79
                                          Cartridge / 96
36:  Buffer\n\ CZ 5000    : 32-63 Memory / 96 Buffer");
37:    exit(20);
38:   }
39:
40:  if(argc>2) {
41:    if((ch=atoi(argv[2]))<1 || ch>16) {
42:     puts("Illegal Midi channel"); exit(20L);
43:    }
44:   }
45:   if(argc==4) {
46:    if((sn=atoi(argv[3]))<32 || sn>96) {
47:     puts("Illegal sound number"); exit(20L);
48:    }
49:   }
50:
51:  if(!(f=Open(argv[1],MODE_OLDFILE))) {
52:    puts("Can't open file"); exit(20);
53:   }
54:   Read(f,buff,256L);
55:   Close(f);
56:   if(midiopen()) {
57:    puts("Can't open file");
58:    exit(20);
59:   }
60:   rr1[4]+=ch; /* channel */
61:   rr1[6]=sn; /* sound nr */
62:  /* Send  "RECEIVE REQUEST 1" */
63:   midiwrite(rr1,7);
64:  /* auf korrekte antwort prüfen */
65:   if(midiget()!=BSYSTEM_EXCLUSIVE  || midiget()!=ID_CASIO
66:   || midiget()
67:   || midiget()
68:   || midiget()!=0x6f+ch
69:   || midiget()!=0x30) {
70:     puts("Incorrect Request answer (perhaps wrong keyboard)");
71:     midiclose(); exit(20);
72:   }
73:  /* daten senden data */
74:   midiwrite(buff,256);
75:  /* end of transmission */
76:   END_OF_EXCLUSIVE;
77:   midiclose();
78:   exit(0);
79:  }
80:  /* Listing Ende */
```

den abgefangen und angezeigt (z.B. wenn der falsche 'Synthi' angeschlossen ist). Ist kein Gerät angeschlossen, hängt der Computer sich auf (siehe Murphys Axiome, *"elektrischer Suizid"*).

CZPUT

"CZPUT" ist das Gegenstück zu "CZGET" und schreibt einen Sound von Diskette zurück in den Speicher des Synthi. Die Aufrufparameter und Bedingungen sind dieselben wie bei "CZGET", nur ist es natürlich nicht möglich, in ein "Preset" zu schreiben (eigentlich schade...). Zu beachten ist noch, daß man beim Einschreiben in den Soundbuffer (96) keinen Speicherplatz überschreibt. Es ist also sicherer, erst einen Sound in den Buffer zu schreiben und dann erst im Synthesizer dauerhaft zu speichern.

Wer näher an der Thematik der Midi-Kommunikation zwischen Computer und Synthesizer interessiert ist, sollte sich das "GUIDEBOOK FOR MIDI" von Casio besorgen, in dem alle Sys-Ex der CZ-Synthis genau beschrieben sind.

PS. Der Autor ist übrigens sehr am Tausch von CZ-Sounds interessiert, also Diskette mit den schärfsten, schrägsten und grausamsten Sounds an:

Oliver Wagner,
Landsberge 5
4322 Sprockhövel 2.

```
/* CZGET.C
   Program zum einlesen von Sounddaten von Casio
   CZ-Synthesizern über die Midi-Schnittstelle
   Autor: Oliver Wagner
   (C) MAXON Computer GmbH 1989
    KICKSTART 1989
   Version für Aztec C 3.4a oder 3.6
   CC czget
   CC midi
   LN czget.o midi.o -lc
*/

#include <midi.h>
#include <libraries/dos.h>

/* Send-Request 1 */
static unsigned char sr1[] = {
 BSYSTEM_EXCLUSIVE,ID_CASIO,
 0x00,0x00,
 0x6f, /* Channel */
 0x10,
 0x00 /* Sound-Nr */
};
static unsigned char buff[256];

main(argc,argv)
register int argc;
char **argv;
{
 register int ch=1,sn=96,c;
 register long *f,*Open();
 puts("\x9b1mCASIO CZ-SERIES SOUND GET UTILITY
      Written 1988 by O.Wagner\x9b0m");
 if(argc<2 || argc>4 || *argv[1]=='?') {
  puts("Usage: CZGET filename [midichannel
       (1-16,def 1) [soundnr(0-96,def 96)] ]\n\
        Soundnr. assignm ent:\n\
  CZ 101/1000: 0-15 Preset / 32-47 Internal /
  64-79 Cartridge / 96 Buffer\n\
  CZ 5000    : 0-31 Preset / 32-63 Memory /
  96 Buffer");
  exit(20);
 }
 if(argc>2) {
  if((ch=atoi(argv[2]))<1 || ch>16) {
   puts("Illegal Midi channel"); exit(20L);
  }
 }
 if(argc==4) {
  if((sn=atoi(argv[3]))<0 || sn>96) {
   puts("Illegal sound number");
exit(20L);
  }
 }
 if(midiopen()) {
  puts("Can't open serial device");
  exit(20); }
 sr1[4]+=ch;
 sr1[6]=sn;
/* Send  "SEND REQUEST 1" */

midiwrite(sr1,7);
/* auf richtige antwort prüfen */

if(midiget()!=BSYSTEM_EXCLUSIVE || midiget()!
                                   =ID_CASIO
 || midiget()
 || midiget()
 || midiget()!=0x6f+ch /* Channel */
 || midiget()!=0x30) {
   puts("Incorrect Request answer (perhaps
                                wrong keyboard)");
   midiclose();
   exit(20);
 }
 midiput(0x6f+ch);
 midiput(0x31);
/* daten einlesen */
 for(c=0; c<256; c++)   buff[c]=midiget();
 END_OF_EXCLUSIVE;

midiclose();
/* auf disk schreiben */
 if(!(f=Open(argv[1],MODE_NEWFILE)))
{
  puts("Can't open file");
  exit(20);
 }
 Write(f,buff,256L);
 Close(f);
 exit(0);
}
/* LISTING ENDE */
```

```
/*      MIDI.C
   Bibliothek zur Benutzung der
   Midi-Schnittstelle des Amiga
   Version für Aztec C3.4, 2.6 16Bit
   Autor: Oliver Wagner
   (c) Maxon Computer GmbH 1989
   KICKSTART 1989
*/

#include "midi.h"
#include <functions.h>

static char *sername=SERIALNAME;
static struct MsgPort *ReadPort,*WritePort,
                                  *CreatePort();
static struct IOExtSer *Read_Request,
                  *Write_Request,*CreateExtIO();
static unsigned char ReadB,WriteB;

/* (int)midiopen() öffnet und initialisiert die
   serielle Schnittstelle zum Midi-Betrieb */

int midiopen()
```

```
22:  {
23:   Read_Request=CreateExtIO((ReadPort=CreatePort
          (0L,0L)),(LONG)sizeof(struct IOExtSer));
24:   Read_Request->io_SerFlags=SERF_SHARED|
                                  SERF_RAD_BOOGIE;
25:   if(OpenDevice(sername,NULL,Read_Request,NULL))
26:   {DeleteExtIO(Read_Request);
27:    DeletePort(ReadPort);
28:    return(ERROR_NO_READ_DEVICE);
29:   }
30:   Read_Request->IOSer.io_Length =1;
31:   Read_Request->IOSer.io_Data   =(APTR)&ReadB;
32:   Read_Request->io_Baud         =31250;
33:   Read_Request->io_ReadLen      =8;
34:   Read_Request->io_WriteLen     =8;
35:   Read_Request->io_CtlChar      =0xf4f4f4f4L;
36:   Read_Request->IOSer.io_Command=SDCMD_SETPARAMS;
37:   DoIO(Read_Request);
38:   Read_Request->IOSer.io_Command=CMD_READ;
39:   Write_Request=CreateExtIO((WritePort=CreatePort
          (0L,0L)),(LONG)sizeof(struct IOExtSer));
40:   Write_Request->io_SerFlags=SERF_SHARED|
                                  SERF_RAD_BOOGIE;
41:   Write_Request->io_StopBits=1;
42:   if(OpenDevice(sername,NULL,Write_Request,
                                          NULL)) {
43:    DeleteExtIO(Read_Request);
44:    DeletePort(ReadPort);
45:    DeleteExtIO(Write_Request);
46:    DeletePort(WritePort);
47:    return(ERROR_NO_WRITE_DEVICE);
48:   }
49:   Write_Request->IOSer.io_Command=CMD_WRITE;
50:   Write_Request->IOSer.io_Length =1;
51:   Write_Request->IOSer.io_Data   =(APTR)&WriteB;
52:   Write_Request->io_SerFlags     =SERF_SHARED|
                                   SERF_XDISABLED;
53:   BeginIO(Read_Request);
54:   return(ERROR_OK);
55:  }
56:
57:  /* schliesst die Schnittstelle */
58:  void midiclose()
59:  {
60:   CloseDevice(Read_Request);
61:   CloseDevice(Write_Request);
62:   DeleteExtIO(Read_Request);
63:   DeletePort(ReadPort);
64:   DeleteExtIO(Write_Request);
65:   DeletePort(WritePort);
66:  }
67:
68:  /* gibt ein Zeichen (unsigned, int) aus */
69:  void midiput(c)
70:  register int c;
71:  {
72:   WriteB=c;
73:   DoIO(Write_Request);
74:  }
75:
76:  /* gibt einen string bekannter länge aus
77:     midiwrite(string,(int)länge) */
78:  void midiwrite(s,l)
79:  register char *s;
80:   register int l;
81:  {
82:   Write_Request->IOSer.io_Length=l;
83:   Write_Request->IOSer.io_Data=(APTR)s;
84:   DoIO(Write_Request);
85:   Write_Request->IOSer.io_Length=1;
86:   Write_Request->IOSer.io_Data=(APTR)&WriteB;
87:  }
88:
89:  /* wartet und empfängt ein zeichen
                                  (unsigned, int) */
90:  int midiget()
91:  {
92:   register int c;
93:   WaitIO(Read_Request);
94:   c=ReadB;
95:   BeginIO(Read_Request);
96:   return(c);
97:  }
```

```
98:
99:  /* testet, ob Zeichen empfangen, ansonsten -1 */
100: int midimayget()
101: {
102:  register int c;
103:
104:  if(!CheckIO(Read_Request)) return(-1);
105:  WaitIO(Read_Request);
106:  c=ReadB;
107:  BeginIO(Read_Request);
108:  return(c);
109: }
110: /* LISTING ENDE */
```

```
 1:  /* MIDI.h
 2:     Headerfile für MIDI.C
 3:     Autor: Oliver Wagner
 4:     (C) Maxon Computer GmbH 1989
 5:     KICKSTART 1989
 6:  */
 7:
 8:  #ifndef MIDI_H
 9:  #define MIDI_H
10:  #ifndef DEVICES_SERIAL_H
11:  #include <devices/serial.h>
12:  #endif !DEVICES_SERIAL_H
13:
14:  /* returnwerte von "MIDIOPEN" */
15:  #define ERROR_OK 0
16:  #define ERROR_NO_READ_DEVICE 1
17:  #define ERROR_NO_WRITE_DEVICE 2
18:
19:  /* makro für die ausgabe eines mit null
                                 beendeten strings */
20:  #define midiwrites(s) midiwrite(s,strlen(s))
21:
22:  /* makros für midi-befehle */
23:  /* Voice Messages */
24:  /* c=channel(0-15) */
25:  #define BNOTE_ON 0x80
26:  #define NOTE_ON(c,n,v) { midiput(0x80+c);
                              midiput(n); midiput(v); }
27:  #define NOTE_OFF(c,n) { midiput(0x80+c);
                              midiput(n); midiput(0); }
28:  #define BPOLYPRESSURE 0xa0
29:  #define POLYPRESSURE(c,n,a) { midiput(0xa0+c);
                              midiput(n); midiput(a); }
30:  #define BCONTROL_CHANGE 0xb0
31:  #define CONTROL_CHANGE(c,d,v) { midiput(0xb0+c);
                              midiput(d); midiput(v); }
32:  }
33:  #define BPROGRAM_CHANGE 0xc0
34:  #define PROGRAM_CHANGE(c,p) { midiput(0xc0+c);
                                    midiput(p); }
35:  #define BPITCH 0xe0
36:  #define PITCH(c,v) { midiput(0xe0+c);
                    midiput(v&255); midiput(v<<8); }
37:
38:  /* Mode Messages */
39:  #define LOCAL_ON(c) CONTROL_CHANGE(c,0x7a,0)
40:  #define LOCAL_OFF(c) CONTROL_CHANGE(c,0x7a,0x7f)
41:  #define ALL_NOTES_OFF(c) CONTROL_CHANGE(c,0x7b,0)
42:  #define OMNI_OFF(c) CONTROL_CHANGE(c,0x7c,0)
43:  #define OMNI_ON(c) CONTROL_CHANGE(c,0x7d,0)
44:  #define MONO_ON(c,v) CONTROL_CHANGE(c,0x7e,v)
45:  #define POLY_ON(c) CONTROL_CHANGE(c,0x7f,0)
46:
47:  /* Common Messages */
48:  #define BSONG_POSITION 0xf2
49:  #define SONG_POSITION(p) { midiput(0xf2);
                    midiput(p&255); midiput(p<<8); }
50:  #define BSONG_SELECT 0xf3
51:  #define SONG_SELECT(n) { midiput(0xf3);
                                  midiput(n); }
52:  #define BTUNE_REQUEST 0xf6
53:  #define TUNE_REQUEST midiput(0xf6)
54:  #define BSYSTEM_RESET 0xff
55:  #define SYSTEM_RESET midiput(0xff)
56:
```

```
57:  /* "Realtime" Messages  */
58:  #define BTIMING_CLOCK 0xf8
59:  #define TIMING_CLOCK midiput(0xf8)
60:  #define BSTART 0xfa
61:  #define START midiput(0xfa)
62:  #define BCONTINUE 0xfb
63:  #define CONTINUE midiput(0xfb)
64:  #define BSTOP 0xfc
65:  #define STOP midiput(0xfc)
66:  #define BACTIVE_SENSING 0xfe
67:  #define ACTIVE_SENSING midiput(0xfe)
68:
69:  /* system exclusives */
70:  #define BSYSTEM_EXCLUSIVE 0xf0
71:  #define SYSTEM_EXCLUSIVE midiput(0xf0)
72:  #define BEND_OF_EXCLUSIVE 0xf7
73:  #define END_OF_EXCLUSIVE midiput(0xf7)
74:
75:  /* sys-ex ID's */
```

```
76:  #define ID_CASIO 0x44
77:  #define ID_YAMAHA 0x43
78:  #define ID_KORG 0x42
79:  #define ID_ROLAND 0x41
80:  #define ID_OBERHEIM 0x10
81:  #define ID_MOOG 0x4
82:  #define ID_KURZWEIL 0x7
83:  #define ID_BONTEMPI 0x20
84:  #define ID_STEINWAY 0x9
85:  #define ID_SEQUENTIAL 0x1
86:  #define ID_BIGBRIAR 0x2
87:  #define ID_OCTAVE 0x3
88:  #define ID_PASSPORT 0x5
89:  #define ID_LEXICON 0x6
90:  #define ID_CBS 0x08
91:  #define ID_SIEL  0x21
92:  #endif!MIDI_H
93:  /* Ende Header */
```

END of KICK

KICKS FÜR INSIDER

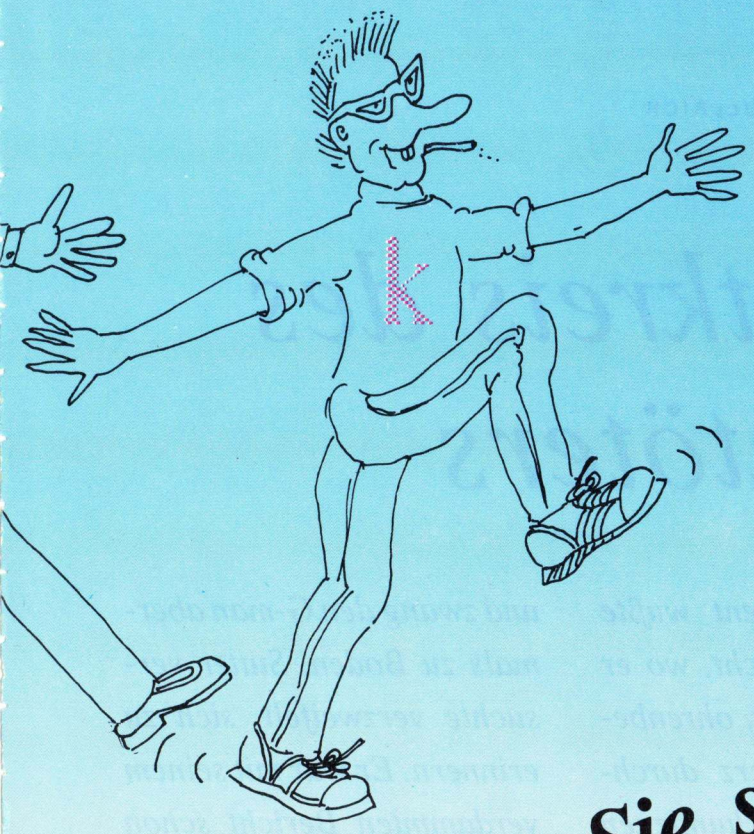
Liebe Leser, KICKS für INSIDER ist eine Rubrik, die für interessante und lehrreiche Programme geschaffen wurde, denn was nutzt das schönste Programm, wenn man es nur abtippen, aber nicht verstehen kann.
Um jeden Monat neue KICKS zu veröffentlichen, sind wir natürlich auch von unseren Lesern abhängig, denn irgendwann einmal sind die Einfälle jedes Programmierers erschöpft. Also, nehmen Sie an dieser Rubrik aktiv teil, ergänzen Sie sie mit Ihren eigenen Programmen. Werden Sie Autor in der KICKSTART.
Dabei sind alle auf dem AMIGA existierenden Programmiersprachen gewünscht. Egal, ob Basic, C, Assembler, Modula 2, Pascal oder andere.

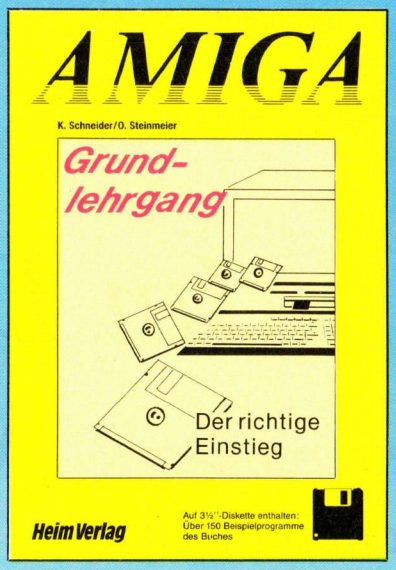
WIE DAS GEHT?

Ganz einfach. Schicken Sie uns das Programm, samt dem zugehörigen Artikel, eventuellen Bildern und Hardcopies (nach Möglichkeit "gegrabbt") auf Diskette zu. Wir werden uns dann bei Ihnen melden. Ein interessantes Honorar wartet bei Veröffentlichung natürlich auch auf Sie.
Wenn Sie Fragen haben, rufen Sie doch einfach bei uns an.

Tel. 06196/481811
Mo-Fr. 11 bis 17 Uhr

Ihre KICKSTART-Redaktion





Holen Sie sich auch den neuen AMIGA-GRUNDLEHRGANG

DM 59,-

Buch und Diskette

unverbindlich empfohlener Verkaufspreis

WICHTIGE MERKMALE:

★ Das Buch für den richtigen Einstieg mit dem Commodore AMIGA ★ Auf über 400 Seiten werden dem Leser leicht verständlich die Grundlagen der Computertechnik und der Umgang mit Hardware erklärt ★ Ein ausführlicher Hauptteil ist dem Einsatz der grafischen Benutzeroberfläche des Betriebssystems gewidmet. Hier erläutert das Buch Fenster, Pulldown-Menüs und die vielen anderen Teile der ***Workbench*** ★ Wer die Maus nicht mag, der kann aus dem Kapitel über den ***Command Line Interpreter (CLI)*** entnehmen, wie man den AMIGA auch ohne Maus einsetzen kann ★ Ein weiterer Bereich des Buches ist die Einführung in die Programmiersprache ***BASIC***. Eine umfangreiche Befehlsübersicht sowie einige interessante Programme dienen der Erlernung und dem guten Training von BASIC ★ Anhänge wie z. B. ein ***Index*** und eine ***Sachworterklärung*** bieten das schnelle Nachschlagen und Auffinden wichtiger Punkte ★ ***Mit dem Buch erhalten Sie eine Programmdiskette mit allen abgedruckten Listings.*** Damit können die Beispielprogramme ohne die Mühe und Arbeit des Eintippens auf dem Computer nachvollzogen werden.

AUS DEM INHALT:

1. Die Hardware des AMIGA
★ die versch. AMIGA-Modelle ★ die Disketenstation ★ Anschluß eines Druckers ★ Monitore am AMIGA ★ Erweiterung des AMIGA-Systems ★ Einstieg in die MS-DOS Welt mit dem AMIGA ★ Die „Innereien" des AMIGA (RAM, ROM u. Prozessoren)

2. Das Betriebssystem des AMIGA
★ Betriebssysteme und ihre Bedeutung ★ Die Benutzeroberfläche des AMIGA ★ Steuerung der Workbench ★ Arbeiten mit Maus, Fenstern und Pull-down-Menüs ★ Verwendung von Disketten, Dateien, Directory ★ Die Programme der Workbench Diskette im Einzelnen ★ Der CLI und seine Bedienung ★ Kopieren, Löschen und Batch-Bearbeitung im CLI

3. Programmieren in Amiga-Basic
★ Die Bedienung des Basic-Interpreters ★ Variable in Basic ★ Schleifenstrukturen ★ Die IF-Abfrage ★ Proceduren zur Programmstrukturierung ★ Graphik-Programmierung in AMIGA-BASIC ★ Dateiverwaltung ★ ausführliche Befehlsübersicht mit detaillierten Erklärungen

4. Zum Training
★ Programm-Diskette mit allen abgedruckten Listings ★ Sachworterklärung (Fachwörter-Lexikon) ★ Ausführlicher Index (Stichwortverzeichnis mit entspr. Verweisen)

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Ich bestelle _____
zzgl. DM 5,– Versandkosten (unabhängig von der bestellten Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname _____

Straße, Hausnummer _____

PLZ, Ort _____

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte

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

VON MEINHARD ULLRICH

Im Dunstkreis des Neuntöters

Steuercodes machen auch Ihren Drucker gefügig

Für einen Moment wußte Donald Sutter nicht, wo er war. Ein schriller, ohrenbetäubender Schmerz durchzuckte seinen geschundenen Körper. Wieder zerhackte dieses unmenschliche Tak-Tak-Tak-Tak die brüllende Stille der samtblauen Nacht und zwang den G-man abermals zu Boden. Sutter versuchte verzweifelt, sich zu erinnern. Er war mit seinem verdammten Bericht schon wieder überfällig. In einer halben Stunde war letzter Termin, sonst...

Tak-Tak-Tak-Tak! Sutter heulte auf. Er fingerte in ohnmächtiger Wut in die Nähe seines Schulterhalfters, wo sich sonst der Deoroller befand. Aber die Umstände hatten ihn entwaffnet.

Erneut peitschte giftig zischend eine Salve durch den Raum. Sutter federte hoch, wollte sich überschlagen, um - wie sie es an der Akademie damals tausendmal geübt hatten - sich geschickt mit einer Rolle vorwärts aus der Gefahrenzone zu bringen. Er schaffte es nicht ganz und schlug mit einem dumpfen, trockenen Schlag mit dem Kopf auf dem Natursteinboden auf. Die schmerzverzerrten Züge unterstrichen den indianischen Einschlag seines kantigen, etwas grobschlächtigen Gesichts.

Tak-Tak-Tak-Tak spuckte es aus allen neun Nadeln. Gellender Schmerz erfaßte ihn noch einmal und kroch bis zur äußersten Faser seiner Nerven.

Doch Sutter konnte sich nicht mehr bewegen. Er konnte nur noch denken. 50 Kilobyte mußte er noch ausdrucken. Er hatte den DIP-Schalter auf NLQ umgelegt, den Ausdruck gestartet und war noch einmal telefonieren gegangen.

Als er zurückgekommen war, mußte er mit schierem Entsetzen feststellen, daß seine 9-Nadel-Automatik vom Typ Star SG 10 seinem erklärten Willen zum Trotz den Bericht in "Draft" ausgedruckt hatte. Mit ungläubigen, glasigen Augen sah Sutter in den Mündungsblitz, oder was er in seinem Wahnsinn für einen solchen hielt. Hatte ihm nicht erst kürzlich sein Freund und V-Mann Peter Hampton von diesem Phänomen bei seiner schallgedämpften Epson LQ 850 erzählt? Richtig! Beim ersten Ausdruck ist immer die Einstellung wirksam, die in den Preferences eingestellt ist.

Tak-Tak-Tak-Tak. Sutter verspürte keinen Schmerz mehr. Ein dichter, schmutziggrauer Schleier umgab ihn, hüllte ihn immer mehr ein. Der 35jährige G-man Donald Sutter war bereits tot, als der Drucker seinen letzten kreischenden Wagenrücklauf erledigte.

Ich war am nächsten Tag schon sehr früh im Büro. Der Tod unseres Kollegen hatte uns alle sehr betroffen gemacht. Warum mußte ausgerechnet er, der alle Himmelfahrtskommandos unbeschadet überstanden hatte, am Formularkrieg zugrunde gehen? Obwohl niemand mir den Auftrag erteilte, spürte ich, daß ich einen solchen Vorfall für alle Zukunft verhindern mußte. Ich schrieb *Codes*.

Codes, oder wie immer Sie das Programm nennen wollen, läßt zwei verschiedene Aufrufe zu. Entweder Sie wählen eine Reihe von Optionen, die voreingestellt sind. Zum Beispiel: *Codes 2 9 17* stellt Kursiv, Breitdruck und Unterstreichen ein. Wollen Sie eine Auflistung der Möglichkeiten, dann geben Sie nur *"Codes"* ein. Die andere Alternative ist die Direkteingabe von Steuersequenzen. In diesem Fall muß den weiteren Argumenten, die ASCII-Codes darstellen, ein kleines *d* vorangehen. *"Codes d 27 66 4"* beispielsweise stellt bei meinem Drucker Schönschrift ein (27 steht für ESC, 66 ist der ASCII-Wert von B!).

Die Steuersequenzen, die im unten abgedruckten Programm stehen, sind auf den STAR SG 10 angepaßt. Die Änderung für einen anderen Drucker ist sehr einfach: Schlagen Sie in Ihrem Druckerhandbuch die Steuercodes für verschie-

dene Schriftarten oder auch andere Funktionen nach, wie zum Beispiel Farbdruck oder zweiminütigen Dauerton, und tragen Sie diese nach dem gleichen Muster in die Struktur Befehle[] ein. Zählen Sie dann einfach durch, wieviele Befehle sie definiert haben, und tragen diese Zahl in die Definition *MAXBefehle* ein (hier: 22).

Das Programm ist mit dem Aztec-Compiler Version 3.6 entstanden und erfordert keine besonderen Optionen beim Kompilieren, Assemblieren und Binden.

In der nächsten Ausgabe lesen Sie:

Gemeine Rowdies hielten die einzige Telefonzelle besetzt.
...Doch ich hatte Sprechfunk.

```
/*
** Codes.c
** Autor Meinhard Ullrich
** (c) Maxon Computer GmbH 1988/89
*/

/* Achtung: Dieses Programm ist nicht vollkommen
   "idiotensicher".
** Beispielsweise würde ein Aufruf 'codes 0' oder
   'codes x' mit
** x > MAXBefehle Unsinn liefern. Dies allerdings
   alles zu
** berücksichtigen, macht das Programm als "Kick"
   zu lang.
*/

#include "exec/types.h"
#include "libraries/dosextens.h"
#include "functions.h"

#define MAXBefehle 22
#define CODELAENGE 3

struct Kodierung {
   STRPTR Befehl;
   BYTE   Code[CODELAENGE];
   };

struct Kodierung  Befehle[MAXBefehle] =
{
   {"Druckerinitialisierung",       {27, '@', 0 }},
                                    /* '@' = 64 */
   {"Kursiv ein",                   {27, '4', 0 }},
   {"Kursiv aus",                   {27, '5', 0 }},
   {"NLQ ein",                      {27, 'B', 4 }},
   {"NLQ aus",                      {27, 'B', 5 }},
   {"Pica ein",                     {27, 'B', 1 }},
   {"Elite ein",                    {27, 'B', 2 }},
   {"Schmalschrift ein",            {27, 'B', 3 }},
   {"Breitdruck ein",               {27, 'W', 1 }},
   {"Breitdruck aus",               {27, 'W', 0 }},
   {"Doppelter Anschlag ein",       {27, 'G', 0 }},
   {"Doppelter Anschlag aus",       {27, 'H', 0 }},
   {"Fettdruck ein",                {27, 'E', 0 }},
   {"Fettdruck aus",                {27, 'F', 0 }},
   {"Proportionalschrift ein",      {27, 'p', 1 }},
   {"Proportionalschrift aus",      {27, 'p', 0 }},
   {"Unterstreichen ein",           {27, '-', 1 }},
   {"Unterstreichen aus",           {27, '-', 0 }},
   {"Zeilenvorschub/Wagenrücklauf", {10, 13 , 0 }},
   {"Blattvorschub",                {12,  0 , 0 }},
   {"oberen Rand setzen",           {27, 'R', 2 }},
   {"unteren Rand setzen",          {27, 'N', 3 }}
};
/* Die beiden letzten Steuercodes bewirken zusammen,
   daß die Perforation
** übersprungen wird. Es sollten beide zusammen
   angewendet werden, denn
** im unteren Code ist der Offset aus dem oberen
   mitberücksichtigt.
*/

struct FileHandle    * Drucker, * Parallel;
```

```
VOID  Gebrauchsanweisung(Programmname)

   STRPTR  Programmname;

{
   int i;

   printf("\n");
   printf("Schreibweise: %s ", Programmname);
   printf("d <Code_1> <Code_2> ... <Code_n>
                                 (Direkteingabe)\n");
   printf("oder       : %s ", Programmname);
   printf("<N_1> <N_2> ... <N-n>, mit <N_i> aus
                  [1..%ld].\n", (LONG)MAXBefehle);
   printf("Bedeutung von <N_i>:\n");
   for (i = 1; i <= MAXBefehle; i++)
      printf("%3d)...%s\n", i, Befehle[i-1].Befehl);
}

main(argc, argv)
   int     argc;
   STRPTR  argv[];

{
   int Auswahl, i;
   BYTE Wert;

   if (argc == 1)
      Gebrauchsanweisung(argv[0]);
   else
   {
      Drucker = Open("PRT:", MODE_OLDFILE);
      if ((Write(Drucker, "\0", 1L)) == -1L)
      {
         printf("Drucker aus, eh?\n");
         Close(Drucker);
         exit(FALSE);
      }
      Close(Drucker);

      Parallel = Open("PAR:", MODE_OLDFILE);

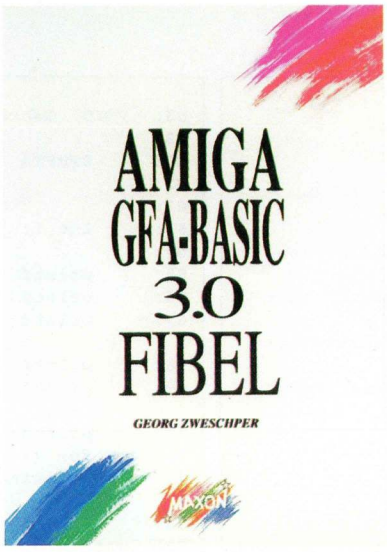
      if (strcmp(argv[1], "d")!= 0) /* Direkte
                     Anwahl eines Druckercodes   */
         for (i=1; i<argc; i++)
         {
            Auswahl = atoi(argv[i]);
            Write(Parallel, Befehle[Auswahl - 1].
                           Code, (LONG)CODELAENGE);
         }

      else  /* Direkte Eingabe einer Steuersequ. */
         for (i=2; i<argc; i++)
         {
            Wert = (BYTE)atoi(argv[i]);
            Write(Parallel, &Wert, 1L);
         }

      Close(Parallel);
   }
}
```

ENDE

AMIGA GFA-BASIC 3.0 Fibel



NEU!

Alles

Editor, Variablentypen, Befehle, Funktionen und Operatoren sind vollständig aufgeführt und beschrieben. Tastaturbelegung, Füllmuster und Fehlermeldungen sowie eine Liste aller implementierten Betriebssystem-Funktionen des AMIGA befinden sich im Anhang.
Dieses Buch eignet sich deshalb für alle Programmierer, die in diesen neuen und einzigartigen BASIC-Dialekt einsteigen möchten.

Ausführlich

Zu jedem Befehl wird eine genaue Beschreibung der Syntax und der Parameter gegeben. Die detaillierten, aber kompakten Erklärungen lassen keine Unklarheiten bestehen. Zusätzliche Informationen und Hinweise machen die Lektüre komplett und das Verstehen einfach.
Die Programmbeispiele zeigen den richtigen Umgang mit dieser mächtigen Programmiersprache, für den BASIC-Einsteiger wie für den AMIGA-Insider. Das Buch für jeden GFA-BASIC-Programmierer.

Alphabetisch

Wenn Sie mit den fast 400 Befehlen und Funktionen von GFA-BASIC arbeiten und die Syntax oder Bedeutung einzelner Befehle nachschlagen möchten, dann werden Sie diese Vorzüge, die Ihnen eine alphabetische Sortierung bietet, nicht mehr missen wollen.
Befehle, Funktionen, Operatoren und Systemvariablen sind deshalb streng alphabetisch geordnet und vereinfachen das Auffinden der gewünschten Informationen erheblich, denn Sie benötigen kein Inhaltsverzeichnis und keinen Index.

MAXON

Bestellcoupon MAXON Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Name: _____
Vorname: _____
Straße: _____
Ort: _____
Unterschrift: _____

Hiermit bestelle ich:

☐ Exemplar(e) "AMIGA GFA-BASIC 3.0 Fibel" DM 39,00

Versandkosten: DM 7,50
Nachnahme zuzgl. DM 3,50 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
Vertrtieb in Holland: JOTKA COMPUTING Postbus 8183 NL-6710 AD Ede Tel.: 08380/38731 Fax.: 08380/21675

VON ANDREAS BIERNAT

Die Floppy im Griff ! ! !

Teil 2:

Rund um das Trackdisk-Device

Im zweiten Kursteil wollen wir uns der praktischen Arbeit widmen und uns anschauen, wie man das Trackdisk-Device öffnet. Voraussetzung dafür ist eine initialisierte IO-Request-Struktur. Wie muß diese Initialisierung nun genau aussehen? Bei der Größe der IOStdReq-Struktur könnte man meinen, daß diese Aufgabe ziemlich aufwendig wäre. Aber genau das Gegenteil ist der Fall.



Initialisierung

Um das Trackdisk-Device zu öffnen, brauchen wir zunächst nur einen Reply-Port, über den das Device Messages an uns senden kann. Die Message, die wir an das Device senden, ist ja Teil der IOStdReq-Struktur. Der Reply-Port wird ebenfalls durch eine Struktur dargestellt, in der nur eingetragen sein muß, wer die Nachrichten erhalten soll. Da unser eigenes Programm, also unser eigener Task, die Nachrichten empfangen soll, müssen wir einen Zeiger auf unseren eigenen Task in die Reply-Port-Struktur eintragen. Wenn wir dann unseren Reply-Port in die globale Liste der Message-Ports aufnehmen (mit der AddPort-Funktion der Exec-Library), übernimmt das System die restliche Verwaltung des Ports.

Damit ist die notwendige Initialisierung der IOStdReq-Struktur erledigt, und das Trackdisk-Device kann geöffnet werden. Das Ganze wird nochmal an einem Beispiel in Listing 1 gezeigt. Das Programm öffnet das Trackdisk-Device und schließt es direkt danach wieder. Es ist zu empfehlen, sich die IOStdReq-Struktur mit *'q diskio'* anzuschauen. Vielmehr als wir hat das System auch nicht in die Struktur eingetragen. Was sofort ins Auge fällt, sind die zwei Langwörter, die jeweils -1 (=$FFFFFFFF) beinhalten. Laut Struktur sollten dort Zeiger auf die Device- und die Unit-Struktur stehen. Die Unit-Struktur kümmert uns ja wenig, aber es scheint zuerst so, als ob uns der Einblick in die Device-Struktur versagt bleiben würde. Wenn wir aber im Listing direkt hinter dem OpenDevice-Aufruf ein MOVE.L 20(A1),A5 plazieren, haben wir nach Beendigung des Programms den Zeiger auf die Device-Struktur in A5. Dort stehen z.B. ein Zeiger auf die *GfxBase* (Offset 56), ein Zeiger auf die *Disk-Resource* (Offset 60) oder einer auf die *CIAB-Resource* (Offset 94).

Das erste Kommando

So, nun sind wir soweit, daß wir die ersten Kommandos ausprobieren können. Die ersten drei Kommandos sind folgende:

1) Read

Mit diesem Kommando liest man einen oder mehrere Sektoren von Diskette. Das Command- Wort der IOStdReq-Struktur wird auf den Wert Zwei gesetzt und der Length-Eintrag auf die Anzahl der zu lesenden Bytes, wobei die Länge ein ganzzahliges Vielfaches von 512 sein muß. *Data* zeigt auf einen Speicherblock, in den die gelesenen Daten hineingeschrieben werden sollen. Mit dem Offset legt man fest, an welcher Stelle der Diskette (von welchem Sektor an) die Datenübertragung beginnen soll. Auch hier muß der Wert ein ganzzahliges Vielfaches von 512 sein. Aus dem Error-Byte kann man einen eventuell aufgetretenen Fehler ablesen.

2) Write

Dieses Kommando wird genauso benutzt wie *Read*, nur mit dem Unterschied, daß geschrieben und nicht gelesen wird. Das Command-Wort muß hier mit dem Wert Drei beschrieben werden. Sonst können alle Angaben von oben übernommen werden.

3) Motor

Mit diesem Befehl kann der Motor ein- und ausgeschaltet werden. Sinnvoll ist aber nur das Ausschalten, da der Motor automatisch eingeschaltet wird, wenn auf die Diskette zugegriffen werden muß. Das Command-Word wird in diesem Fall mit dem Wert 9 beschrieben. Bei diesem Kommando muß dann lediglich das Längenfeld initialisiert werden: 0 bedeutet ausschalten ; 1 einschalten.

Eine eventuelle Fehlermeldung läßt sich aus dem Error-Byte ablesen. Anhand des Actual-Langwortes kann man den Zustand des Motors vor Ausführung des Befehls erkennen. Null bedeutet, daß der Motor vorher aus war; eine Eins bedeutet demnach, daß er vorher an war. Listing 2 demonstriert die ersten drei Befehle. Da das Programm die Diskette beschreibt, sollte man keine Disketten mit wichtigen Daten im Laufwerk lassen, da diese unter Umständen zerstört werden könnten. Das Programm öffnet

```
execbase =      4
findtask =   -294
addport =    -354
remport =    -360
opendevice = -444
closedevice = -450

run:
 move.l   execbase,a6          ;Zeiger auf Exec-Basisadresse
 sub.l    a1,a1                ;0 als Argument => eigener Task
 jsr      findtask(a6)         ;Eigene Taskadresse suchen
 move.l   d0,replyport+16      ;Eigenen Task als Empfänger eintragen

 lea      replyport,a1         ;Zeiger auf Reply-Port
 jsr      addport(a6)          ;Reply-Port in globale Liste aufnehmen

 lea      diskio,a1            ;Zeiger auf IOStdReq-Struktur
 move.l   #replyport,14(a1)    ;Reply-Port eintragen
 move.l   #0,d0                ;Laufwerk: df0:
 clr.l  d1                     ;Keine Flags
 lea      trdname,a0           ;Zeiger auf Devicenamen
 jsr      opendevice(a6)       ;Trackdisk-Device öffnen
 tst.l  d0                     ;Fehler aufgetreten?
 bne      error                ;Ja! Programm beenden

 lea      replyport,a1         ;Zeiger auf Replyport
 jsr      remport(a6)          ;Reply-Port aus globaler Liste entfernen

 lea      diskio,a1            ;Zeiger auf IOStdReq-Struktur
 jsr      closedevice(a6)      ;Trackdisk-Device schließen

error:
 rts                           ;Programmende

trdname: dc.b "trackdisk.device",0
even
diskio:    blk.l 20,0
replyport: blk.l 8,0
```

Abbildung 1: Das Anmelden eines Diskettenzugriffs

```
#include <exec/types.h>
#include <devices/trackdisk.h>

main()
{
  struct MsgPort *mport;
  struct IOStdReq *iorequest;

  mport = CreatePort(0L,0L);
  iorequest = CreateStdIO(mport);
  OpenDevice(TD_NAME,0L,iorequest,0L);   /* Laufwerk df0: */
    /* Hier stehen die zukünftigen Arbeiten mit dem Device */
  CloseDevice(iorequest);
  DeleteStdIO(iorequest);
  DeletePort(mport);
}
```

Listing 1: Das Anmelden eines Diskettenzugriffs in C

zuerst das Trackdisk-Device. Dann werden sieben Sektoren gelesen, und zwar ab Cylinder 18 Head 0 Sektor 4. Diese Werte entsprechen der Blocknummer 400. Danach wartet das Programm auf die Betätigung der linken Maustaste, um mit dem Schreiben von vier Sektoren, ebenfalls ab Block 400 (um die Diskette nicht zu verwüsten), fortzufahren. Nun wird nur noch der Laufwerksmotor ausgeschaltet und das Trackdisk-Device geschlossen.

Vom RAM auf Diskette

Aufmerksamen Leuten wird aufgefallen sein, daß das Programm direkt nach dem Drücken der Maustaste seine Arbeit beendet. Man hört von der Floppy gar nichts, das Schreiben scheint verdammt schnell gegangen zu sein. Dieser Effekt tritt auf, weil das Trackdisk-Device immer mit ganzen Tracks arbeitet.

Will man also den ersten Sektor des Root-Directories lesen, wird der ganze Track 40 ins RAM geladen. Von dort wird der Sektor ausgelesen und in den festgelegten Puffer kopiert. Soll nun der gleiche Sektor wieder mit der WRITE-Anweisung zurückgeschrieben werden, werden die Daten aus dem Puffer an die Stelle ins RAM geschrieben, wo sie herkamen, nämlich in den Trackpuffer im RAM. Die WRITE-Anweisung schreibt also nicht immer direkt auf Diskette, sondern in den Trackpuffer im RAM. Wenn aber der erste Sektor von Cylinder 79 gelesen auf Cylinder 3 kopiert werden soll, bleibt der WRITE-Anweisung nichts anderes übrig, als direkt auf Diskette zu schreiben.

Was aber muß man unternehmen, um die Daten aus dem Trackpuffer wieder auf Diskette zu bekommen? Dafür gibt es ein spezielles Kommando namens UPDATE.

Der Disketten-Update

Insgesamt gibt es zwei Befehle, die im Zusammenhang mit dem Trackpuffer benutzt werden können:

```
execbase =      4
findtask =    -294
addport =     -354
remport =     -360
opendevice =  -444
closedevice = -450
doio =        -456

run:
 move.l   execbase,a6          ;Zeiger auf Exec-Basisadresse
 sub.l    a1,a1                ;0 als Argument => eigener Task
 jsr      findtask(a6)         ;Task suchen
 move.l   d0,replyport+16      ;Eigener Task als Empfängertask

 lea      replyport,a1         ;Zeiger auf Reply-Port
 jsr      addport(a6)          ;In globale Liste der MsgPorts eintragen

 lea      diskio,a1            ;Zeiger auf IOStdReq-Struktur
 move.l   #replyport,14(a1)    ;Reply-Port in Message eintragen
 move.l   #0,d0                ;Laufwerksnummer (hier df0:)
 clr.l    d1                   ;keine Flags
 lea      trdname,a0           ;Zeiger auf Devicename
 jsr      opendevice(a6)       ;Device öffnen
 tst.l    d0                   ;Fehler?
 bne      error                ;Ja! Programm beenden

 lea      diskio,a1            ;Zeiger auf IOStdReq-Struktur
 move     #2,28(a1)            ;Kommando: READ
 move.l   #diskbuff,40(a1)     ;Puffer für Zieldaten
 move.l   #7*512,36(a1)        ;Länge: 7 Sektoren
 move.l   #400*512,44(a1)      ;Offset: 400 Sektoren
 jsr      doio(a6)             ;Sektoren lesen
 move.l   32(a1),d7            ;Anzahl gelesener Bytes in D7
 move.b   31(a1),d6            ;Eventueller Fehler in D6

 bsr      wait                 ;Auf linke Maustaste warten

 lea      diskio,a1            ;Zeiger auf IOStdReq-Struktur
 move     #3,28(a1)            ;Kommando: WRITE
 move.l   #diskbuff,40(a1)     ;Adresse des Datenpuffers
 move.l   #4*512,36(a1)        ;Länge: 4 Sektoren
 move.l   #400*512,44(a1)      ;Offset: 400 Sektoren
 jsr      doio(a6)             ;Sektoren schreiben
 move.l   32(a1),d7            ;Anzahl geschriebener Bytes in D7
 move.b   31(a1),d6            ;Eventueller Fehler in D6

 lea      diskio,a1            ;Zeiger auf IOStdReq-Struktur
 move     #9,28(a1)            ;Kommando: MOTOR
 move.l   #0,36(a1)            ;0:=Ausschalten / 1:=Einschalten
 jsr      doio(a6)             ;Motor ausschalten

 lea      replyport,a1         ;Zeiger auf Reply-Port
 jsr      remport(a6)          ;Port aus globaler Liste entfernen

 lea      diskio,a1            ;Zeiger auf IOStdReq-Struktur
 jsr      closedevice(a6)      ;Trackdisk-Device schließen

error:
 rts                           ;Programm beenden

wait:
 btst     #6,$bfe001           ;Linke Maustaste gedrückt?
 bne      wait                 ;Nein, warten
 rts                           ;Zurück zum Hauptprogramm
```

1) Update

Dieser Befehl bewirkt ein sofortiges Schreiben des Trackpuffers auf die Diskette. Dabei braucht nur das Command-Wort mit dem Wert 4 initialisiert zu werden. Eine eventuelle Fehlermeldung kann man aus dem Error-Byte ersehen.

2) Clear

Der Trackpuffer des Trackdisk-Devices wird für ungültig erklärt, so daß beim nächsten Zugriff der Track noch einmal gelesen wird. Wie beim Kommando Update braucht nur das Command-Wort mit der Wert 5 beschrieben zu werden. Daneben gibt es noch einen weiteren Befehl, der zwar auch mit ganzen Tracks arbeitet, jedoch mit dem Trackpuffer weniger zu tun hat: den FORMAT-Befehl. In das Command-Feld muß hierfür eine 11 eingetragen werden. Das *Length*-Feld wird mit der Anzahl der Cylinder in Byte beschrieben, d.h. ein Cylinder entspricht 22*512 Bytes. Der Inhalt dieses Feldes errechnet sich also:

*Length = AnzahlCylinder * 22*512.*

Genauso wird der Offset gehandhabt: Auch er muß in Bytes angegeben werden:

*Offset = AnfangsCylinder * 22*512.*

Nun muß nur noch ein Zeiger auf den Datenbereich zeigen, der auf Diskette geschrieben werden soll. Dieser Puffer sollte logischerweise mindestens 22*512 Bytes lang sein (ein Cylinder). Listing 3 demonstriert die Benutzung des Trackpuffers und das Format-Kommando. Es werden zuerst zwei Sektoren in zwei verschiedene Puffer geladen. Dann wird der zweite Sektor an die Stelle des ersten zurückgeschrieben (in den Trackpuffer). Damit keine Unordnung auf der Diskette entsteht, wird der Trackpuffer für ungültig erklärt und nochmal eingelesen. Das geschieht, indem einfach der zweite Puffer nochmals mit dem READ-Befehl geladen wird. Nun ändert man den Diskettennamen etwas und schreibt die beiden Sektoren in den Trackpuffer. Durch das UPDATE-Kommando wird der Trackpuffer auf Diskette geschrieben. Danach liest man noch Cylinder drei ein und schreibt ihn mit dem FORMAT-Kommando zurück. Mit dem FORMAT-Befehl können also zwei Sachen auf einmal erledigt werden. Es werden

```
trdname:  dc.b "trackdisk.device",0
even
diskio:   blk.l 20,0
replyport: blk.l 8,0
diskbuff:  blk.b 7*512,0
```

Abbildung 2: Die Befehle Read, Write und Motor

```
#include <exec/types.h>
#include <exec/memory.h>
#include <devices/trackdisk.h>

main()
{
  struct MsgPort *mport;
  struct IOStdReq *iorequest;
  LONG *buffer;
  char taste;

  buffer = (LONG*)AllocMem(7*512L,MEMF_CHIP);
  mport = CreatePort (0L,0L);
  iorequest = CreateStdIO(mport);
  OpenDevice(TD_NAME,0L,iorequest,0L);
/* Zuerst muß der Motor eingeschaltet werden */
    iorequest->io_Command = TD_MOTOR;
    iorequest->io_Length = 1L;          /* 1=an  0=aus */
    DoIO(iorequest);
/* Nun sieben Blocks lesen */
    iorequest->io_Command = CMD_READ;
    iorequest->io_Length = TD_SECTOR * 7;
    iorequest->io_Data = buffer;
    iorequest->io_Offset = TD_SECTOR * 400;
    DoIO(iorequest);
/* Auf eine Taste warten */
    printf("Bitte eine Taste drücken...\n");
    scanf("%c",&taste);
/* Die gelesenen Blocks zurückschreiben */
    iorequest->io_Command = CMD_WRITE;
    iorequest->io_Length = TD_SECTOR * 7;
    iorequest->io_Data = buffer;
    iorequest->io_Offset = TD_SECTOR * 400;
    DoIO(iorequest);
/* Zum Schluß den Motor ausschalten */
    iorequest->io_Command = TD_MOTOR;
    iorequest->io_Length = 0L;
    DoIO(iorequest);
/* Device schließen und Speicher freigeben */
  CloseDevice(iorequest);
  DeleteStdIO(iorequest);
  DeletePort(mport);
  FreeMem(buffer,7*512L);
}
```

Listing 2: Die gleichen Befehle in C

```
execbase =      4
findtask =   -294
addport =    -354
remport =    -360
opendevice = -444
closedevice = -450
doio =       -456
copymem =    -624

run:
 move.l   execbase,a6          ;Zeiger auf Exec-Basisadresse
 sub.l    a1,a1                ;0 als Argument => eigener Task
 jsr      findtask(a6)         ;Eigenen Task suchen
 move.l   d0,replyport+16      ;Unser Task als Empfänger eintragen

 lea      replyport,a1         ;Zeiger auf Reply-Port
```

Jetzt für AMIGA

PICTURE-DISKS GRAFIKSAMMLUNG

Die Picture-Disks-Serie ist die erste Graphik-Sammlung mit hochauflösenden Graphiken für Commodore Amiga Computer. Die Picture-Disks sind von der Qualität einzigartig und decken alle Themenbereiche ab.

500 Graphiken DM 59,–
1000 Graphiken DM 98,–
2000 Graphiken DM 189,–

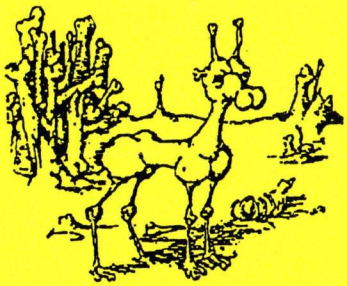
Die hochauflösenden Graphiken im IFF-Format können ohne Probleme in allen gängigen Graphik-, Mal- oder DTP-Programme der AMIGA Welt übernommen und dort weiterverarbeitet werden. Ein schnelles Auffinden der einzelnen Graphiken garantiert ein mitgeliefertes Handbuch. Picture-Disks sind ein Muß für alle professionellen DTP-, Mal- und Textverarbeitungssysteme mit Graphikeinbindung.

Unverbindlich empfohlene Verkaufspreise



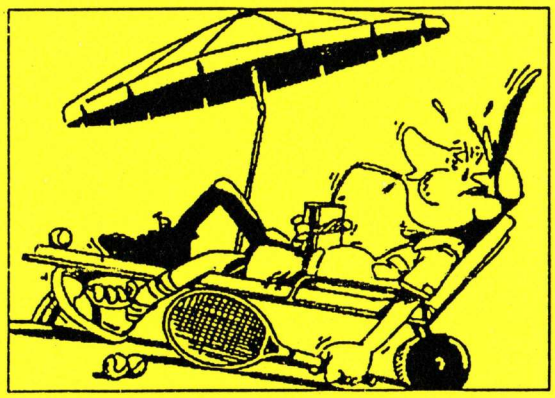














PICTURE DISKS erhalten Sie im guten Fachhandel oder direkt beim Heim-Verlag

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

Schweiz: DataTrade AG
Langstr. 94
CH-8021 Zürich

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Ich bestelle ☐ 500 AMIGA PICTURE-DISKS zum Preis von DM 59,–
☐ 1000 AMIGA PICTURE-DISKS zum Preis von DM 98,–
☐ 2000 AMIGA PICTURE-DISKS zum Preis von DM 189,–
zzgl. DM 5,– Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname _____

Straße, Hausnr. _____

PLZ, Ort _____

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte.

Tracks initialisiert und direkt beim Initialisieren schon beschrieben.

Diskette schreibgeschützt?

Wie jeder wohl schon irgendwann bemerkt hat, wird der Benutzer, falls eine Diskette schreibgeschützt ist und ein Schreibzugriff stattfinden soll, durch ein Requester über den Mißstand informiert. Wir wollen nun mal mit Listing 2 ausprobieren, was passiert, wenn die Diskette im Laufwerk schreibgeschützt ist. Bei Ausführung der WRITE-Anweisung merkt das Betriebssystem diesen Fehler, und der geplante Schreibvorgang wird gar nicht erst angefangen. Ein Fehlercode (hier 28) wird in das Error-Byte der IOStdReq-Struktur eingetragen und zum Programm zurückgekehrt. Wenn man nun im Programm feststellen möchte, ob eine Diskette schreibgeschützt oder überhaupt eingelegt ist, muß man die entsprechenden Befehle des Trackdisk-Devices benutzen:

1) Changestate

Diese Funktion prüft, ob eine Diskette im Laufwerk eingelegt ist. Es muß nur das Command-Wort mit dem Wert 14 beschrieben werden. Anhand des Actual-Langwortes kann man den Status ablesen. Null bedeutet, daß sich eine Disk im Laufwerk befindet, ein Wert ungleich Null das Gegenteil.

2) Protstatus

Hier wird geprüft, ob die eingelegte Diskette schreibgeschützt ist. Wie bei obigem Kommando muß nur das Command-Wort initialisiert werden. In diesem Fall gehört dorthin der Wert 15. Das Actual-Langwort enthält eine Null, wenn die Diskette nicht schreibgeschützt ist, sonst einen Wert ungleich Null. Am Error-Byte kann man eine eventuelle Fehlermeldung ablesen (wenn z.B. keine Diskette eingelegt war).

3) GetDriveType

Dieser Befehl stellt den Typ des angeschlossenen Laufwerks fest. Dabei sind zwei Typen zu unterscheiden, und zwar 3.5 Zoll-Laufwerke und 5.25 Zoll-Laufwerke. In das Command-Feld muß der Wert 18 eingetragen werden. Nach Beendigung der Funktion enthält das Actual-Langwort entweder eine Eins

```
jsr     addport(a6)             ;MsgPort in globale Liste eintragen

lea     diskio,a1               ;Zeiger auf IOStdReq-Struktur
move.l  #replyport,14(a1)       ;Reply-Port in Message eintragen
move.l  #0,d0                   ;Laufwerksnummer (hier df0:)
clr.l   d1                      ;keine Flags
lea     trdname,a0              ;Zeiger auf Devicename
jsr     opendevice(a6)          ;Trackdisk-Device öffnen
tst.l   d0                      ;Fehler?
bne     error                   ;Ja! Programm beenden

move    #2,28(a1)               ;Kommando: READ
move.l  #puffer1,40(a1)         ;Adresse des Puffers
move.l  #512,36(a1)             ;Länge: 1 Sektor
move.l  #880*512,44(a1)         ;Offset: 880 Sektoren
jsr     doio(a6)                ;Sektor lesen
move.l  #puffer2,40(a1)         ;Adresse des 2. Puffers
move.l  #512,36(a1)             ;Länge: 1 Sektor
move.l  #881*512,44(a1)         ;Offset: 881 Sektoren
jsr     doio(a6)                ;Sektor lesen

move    #3,28(a1)               ;Kommando: WRITE
move.l  #puffer2,40(a1)         ;Adresse des 2. Puffers
move.l  #512,36(a1)             ;Länge: 1 Sektor
move.l  #880*512,44(a1)         ;Offset: 880 Sektoren
jsr     doio(a6)                ;Sektor schreiben

;Block 881 steht nun an der Stelle, wo normalerweise Block 880 stehen
;sollte, jedoch nur im RAM und nicht auf der Diskette

move    #5,28(a1)               ;Kommando: CLEAR
jsr     doio(a6)                ;Trackpuffer ungültig machen

move    #2,28(a1)               ;Kommando: READ
move.l  #puffer2,40(a1)         ;Adresse des 2. Puffers
move.l  #512,36(a1)             ;Länge: 1 Sektor
move.l  #881*512,44(a1)         ;Offset: 881 Sektoren
jsr     doio(a6)                ;Trackpuffer neu einlesen

lea     puffer1,a2              ;Zeiger auf 1. Puffer
move.l  434(a2),d0              ;2. bis 5. Buchstabe des Disknamens in D0
move.l  438(a2),434(a2)         ;6. bis 9. Buchstabe als 2. bis 5.
move.l  d0,438(a2)              ;2. bis 5. Buchstabe als 6. bis 9.

;Hier hat eine kleine Veränderung des Diskettennamens stattgefunden.
;Aus 'Kickstart' würde z.B. 'Ktarticks' werden.

move    #3,28(a1)               ;Kommando: WRITE
move.l  #puffer1,40(a1)         ;Adresse des 1. Puffers
move.l  #512,36(a1)             ;Länge: 1 Sektor
move.l  #880*512,44(a1)         ;Offset: 880 Sektoren
jsr     doio(a6)                ;Sektor in Trackpuffer schreiben

move    #4,28(a1)               ;Kommando: UPDATE
jsr     doio(a6)                ;Trackpuffer zurückschreiben

move    #2,28(a1)               ;Kommando: READ
move.l  #trackmem,40(a1)        ;Adresse eines eigenen Trackpuffers
move.l  #22*512,36(a1)          ;Länge: 22 Sektoren (1 Cylinder)
move.l  #3*22*512,44(a1)        ;Offset: 3 Cylinder
jsr     doio(a6)                ;Cylinder lesen

move    #11,28(a1)              ;Kommando: FORMAT
move.l  #trackmem,40(a1)        ;Adresse des eigenen Trackpuffers
```

oder eine Zwei, wobei die Eins für ein 3.5 Zoll-Laufwerk und die Zwei für ein 5.25 Zoll-Laufwerk steht.

4) GetNumTracks

GetNumTracks ermittelt die Anzahl Tracks, die das Laufwerk unter normalen Umständen ansprechen kann. Man sollte hierbei unterscheiden zwischen den Begriffen Track und Cylinder. Ein Track geht nur über eine Seite der Diskette und hat 11 Sektoren, während ein Cylinder zwei Tracks enthält, nämlich den auf der Ober- und den auf der Unterseite. Ein Cylinder hat somit 22 Sektoren.

Der Rückgabewert aus dem Actual-Langwort enthält also die Zahl 160 ($A0), nachdem das Command-Feld mit 19 initialisiert und die Funktion aufgerufen wurde. Listing 4 enthält diese vier Kommandos. Nach dem Start werden mit deren Hilfe die Diskette und das Laufwerk untersucht und entsprechende Meldungen auf dem Bildschirm ausgegeben. Diese stehen dabei nicht im SEKA-, sondern im CLI-Fenster. Man muß also unter Umständen das SEKA-Fenster in den Hintergrund klicken, um die Meldungen sehen zu können.

Der Rest

Es gibt nun noch fünf Befehle des Trackdisk-Devices, die wir noch nicht besprochen haben. Zwei davon sind die RAW-Befehle, die es ermöglichen, ein Track ohne Decodierung zu lesen bzw. zu schreiben. Doch auf diese Befehle komme ich zu sprechen, wenn ich mit der Einführung in den direkten Floppyzugriff beginne. Es bleiben also noch drei Befehle, die eigentlich nichts miteinander zu tun haben, hier jedoch trotzdem in einem Abschnitt behandelt werden sollen.

1) Seek

Dieser Befehl dient zum Bewegen des Schreib-/Lesekopfes. Er wird zu dem Track bewegt, den der Byte-Offset kennzeichnet. Der Byte-Offset muß wieder ein ganzzahliges Vielfaches von 512 sein, sonst bleibt der Kopf dort stehen, wo er bisher stand. Schreib- oder Leseoperationen finden bei diesem Kommando nicht statt. Zu initialisieren sind nur das Command-Wort mit dem Wert 10 und der Byte-Offset.

```
move.l   #22*512,36(a1)      ;Länge: 1 Cylinder
move.l   #3*22*512,44(a1)    ;Offset: 3 Cylinder
jsr      doio(a6)            ;Track 3 formatieren und zurückschreiben

move     #9,28(a1)           ;Kommando: MOTOR
move.l   #0,36(a1)           ;Ausschalten
jsr      doio(a6)            ;Motor ausschalten

lea      replyport,a1        ;Zeiger auf Reply-Port
jsr      remport(a6)         ;Reply-Port aus globaler Liste entfernen

lea      diskio,a1           ;Zeiger auf IOStdReq-Struktur
jsr      closedevice(a6)     ;Trackdisk-Device schließen

error:
rts                          ;Programmende

trdname:   dc.b "trackdisk.device",0
even
diskio:    blk.l 20,0
replyport: blk.l 8,0
puffer1:   blk.b 512,0
puffer2:   blk.b 512,0
trackmem:  blk.b 22*512,0
```

Abbildung 3: Das Programm erklärt die Nutzung des Trackpuffers und des Format-Kommandos

```
#include <exec/types.h>
#include <exec/memory.h>
#include <devices/trackdisk.h>

main()
{
  struct MsgPort *mport;
  struct IOStdReq *iorequest;
  LONG *puffer1,*puffer2,*trackmem,hilf;

  puffer1 = (LONG*)AllocMem(512L,MEMF_CHIP);
  puffer2 = (LONG*)AllocMem(512L,MEMF_CHIP);
  trackmem = (LONG*)AllocMem(22*512L,MEMF_CHIP);
  mport = CreatePort(0L,0L);
  iorequest = CreateStdIO(mport);
  OpenDevice(TD_NAME,0L,iorequest,0L);
/* Motor wird beim Lesen automatisch eingeschaltet */
    iorequest->io_Command = CMD_READ;
    iorequest->io_Length = TD_SECTOR;
    iorequest->io_Data = puffer1;
    iorequest->io_Offset = TD_SECTOR * 880;
    DoIO(iorequest);
    iorequest->io_Data = puffer2;
    iorequest->io_Offset = TD_SECTOR * 881;
    DoIO(iorequest);
/* 2.Puffer an falsche Adresse zurückschreiben (nur im RAM) */
    iorequest->io_Command = CMD_WRITE;
    iorequest->io_Length = TD_SECTOR;
    iorequest->io_Data = puffer2;
    iorequest->io_Offset = TD_SECTOR * 880;
    DoIO(iorequest);
/* Trackpuffer ungültig machen und neu lesen */
    iorequest->io_Command = CMD_CLEAR;
    DoIO(iorequest);
    iorequest->io_Command = CMD_READ;
    iorequest->io_Length = TD_SECTOR;
    iorequest->io_Data = puffer2;
    iorequest->io_Offset = TD_SECTOR * 881;
    DoIO(iorequest);
/* Trackpuffer zurückschreiben */
    iorequest->io_Command = CMD_UPDATE;
    DoIO(iorequest);
/* Cylinder 3 einlesen */
    iorequest->io_Command = CMD_READ;
```

2) ChangeNum

Diese Funktion ermittelt einen Zählerstand, der angibt, wie oft eine Diskette ins Laufwerk gelegt oder entnommen wurde. Nachdem man mit dem Befehl CHANGESTATE feststellen kann, ob eine Diskette eingelegt war, kann man nun auch feststellen, ob sie zwischenzeitlich entfernt wurde. Das Command-Wort muß hierbei mit 13 beschrieben werden. Den Zählerstand bekommt man im Actual-Langwort übergeben.

3) Remove

Mit Hilfe dieser Anweisung wird ein Interrupt installiert, der jedesmal auftritt, wenn eine Diskette entnommen oder eingelegt wird. Eine ähnliche Aufgabe sollte auch der Befehl ADDCHANGEINT (siehe Tabelle im Einführungsteil) erfüllen. Da aber bei ADDCHANGEINT keine Interrupt-Struktur übergeben werden kann, wird die IOReq-Struktur als Interrupt-Struktur angenommen. Durch das Einfügen der IOReq-Struktur in die Liste der Interrupt-Strukturen kommt es zum Absturz des Rechners. Demzufolge ist auch die Funktion REMCHANGEINT nutzlos. Man kann also mit der REMOVE-Anweisung einen Interrupt installieren, aber es treten noch zwei Probleme auf, die ich bis jetzt noch nicht lösen konnte. Erstens wird der Interrupt nicht nur aufgerufen, wenn eine Diskette entnommen wird, sondern auch die ganze Zeit, wenn keine Diskette im Laufwerk ist, dann etwa einmal pro Sekunde. Zweitens suche ich noch nach einer Möglichkeit, den alten Interrupt wieder herzustellen. Wenn er nämlich nur abgeschaltet wird (indem man eine Null anstelle der Interrupt-Struktur übergibt), erkennt das Betriebssystem danach die Disketten nicht mehr, weil deren Name nicht mehr per Interrupt eingelesen wird. Für die Leute, die dennoch mit dem Interrupt etwas experimentieren wollen, hier seine Struktur:

```
    iorequest->io_Length = TD_SECTOR * 22;
    iorequest->io_Data = trackmem;
    iorequest->io_Offset = TD_SECTOR * 3 * 22;
    DoIO(iorequest);
/* Cylinder 3 formatieren und zurückschreiben */
    iorequest->io_Command = TD_FORMAT;
    iorequest->io_Length = TD_SECTOR * 22;
    iorequest->io_Data = trackmem;
    iorequest->io_Offset = TD_SECTOR * 3 * 22;
    DoIO(iorequest);
/* Zum Schluß Motor ausschalten */
    iorequest->io_Command = TD_MOTOR;
    iorequest->io_Length = 0L;
    DoIO(iorequest);
/* Device schließen und Speicher freigeben */
  CloseDevice(iorequest);
  DeleteStdIO(iorequest);
  DeletePort(mport);
  FreeMem(trackmem,22*512L);
  FreeMem(puffer1,512L);
  FreeMem(puffer2,512L);
}
```

Listing 3: Die Nutzung des Trackpuffers und des Format-Kommandos in C

```
execbase =      4
findtask =   -294
addport =    -354
remport =    -360
opendevice = -444
closedevice = -450
doio =       -456
openlib =    -408
closelib =   -414
write =       -48
output =      -60

run:
 move.l     execbase,a6        ;Zeiger auf Exec-Basisadresse
 sub.l      a1,a1              ;0 als Argument => eigener Task
 jsr        findtask(a6)       ;Eigenen Task suchen
 move.l     d0,replyport+16    ;Task als Empfänger in MsgPort eintragen

 lea        replyport,a1       ;Zeiger auf Reply-Port
 jsr        addport(a6)        ;Reply-Port in globale Liste eintragen

 lea        diskio,a1          ;Zeiger auf IOStdReq-Struktur
 move.l     #replyport,14(a1)  ;Reply-Port in Message eintragen
 move.l     #0,d0              ;Laufwerksnummer (hier df0:)
 clr.l      d1                 ;keine Flags
 lea        trdname,a0         ;Zeiger auf Devicename
 jsr        opendevice(a6)     ;Trackdisk-Device öffnen
 tst.l      d0                 ;Fehler?
 bne        error              ;Ja! Programm beenden

 lea        dosname,a1         ;Zeiger auf Libraryname
```

Offset	Größe	Name	Beschreibung
0	--	Node	übliche Node-Struktur
14	dc.l	Data	Zeiger auf Datenspeicher
18	dc.l	Code	Zeiger auf Interrupt-Programm

Die Node-Struktur haben wir schon im Einführungsteil kennengelernt. Sie braucht uns eigentlich nicht zu interessieren. *Data* ist ein Zeiger auf einen Datenspeicher, der vom Interrupt beliebig genutzt werden kann. Da die Größe des Datenspeichers beliebig wählbar ist, heißt das, daß wir die Größe einfach auf Null setzen und somit gar keinen Daten-

KICK ME, AMIGA

Die Software mit dem gewissen KICK



KICK-ASS

Der besondere Assembler

KICK-ASS ist im Prinzip mehr als ein Assembler. Durch seinen integrierten EDITOR, MONITOR, DEBUGGER und LINKER ist KICK-ASS ein leistungsstarkes Entwicklungssystem, das die Entwicklung von Assembler-Programmen einfach, bequem und schnell macht. KICK-ASS wurde für all die entwickelt, die viel Wert auf Geschwindigkeit und Komfort legen.
Von den unzähligen Features, die KICK-ASS bietet, möchten wir nur einige nennen:

- Full-Screen-Editor mit vielen komfortablen Befehlen
- blitzschnelle Assemblierung schon während der Eingabe
- Der Quelltext wird stark komprimiert, so daß er sehr wenig Speicherplatz beansprucht.
- automatisches Einrücken der Zeilen, was gerade bei längeren Programmen eine hohe Lesbarkeit bewirkt
- automatische Syntaxüberprüfung schon bei der Eingabe
- Funktionen sind wahlweise durch Pull-Down-Menüs oder über Tastatur aufrufbar.
- Der integrierte LINKER fügt mehrere Programmodule zusammen. Sogar die Verbindung von Assembler mit Kompilaten von höheren Programmiersprachen ist möglich.
- Der DEBUGGER dient der Programmanalyse und Fehlersuche und ist eine unentbehrliche Hilfe beim Hineinschauen in fremde Programme.

KICK-ASS DM 89.-



KICK-ED

Der universelle Editor

KICK-ED können wir allen, die gern programmieren, sei es in 'C', Assembler, Pascal oder Modula-2, wärmstens empfehlen. Aber auch, 'um 'mal schnell' einen Brief zu schreiben und auszudrucken eignet sich dieser Editor.
KICK-ED ist ein professionelles Werkzeug für Programmierer. Die Bemühung um eine möglichst einleuchtende und angenehme Benutzerführung läßt sich an vielen Details aufzeigen:

- gleichzeitge Verwaltung mehrerer Texte, die mit Hilfe der Funktionstasten angewählt werden können
- Ein Hilfs-Menü, das jederzeit aufgerufen werden kann, gibt vollständige Information über die verschiedenen Befehle.
- bequemes Laden, Speichern und Anhängen von Texten über die integrierte Datei-Auswahl-Box
- Volle Multitasking-Unterstützung des AMIGA.
- superschnelles Softscrolling

KICK-ED DM 49.-

Bestellcoupon

Bitte senden Sie mir:

☐ KICK-ASS DM 89.-
☐ KICK-ED DM 49.-

zuzüglich Versandkosten:
Inland DM 7.50
Ausland DM 10.-

Name _____

Straße _____

Ort _____

Datum _____ Unterschrift _____

Auslandsbestellungen nur gegen Vorauskasse
Bei Nachnahme zuzüglich DM 3.50 Nachnahmegebühr

MAXON-Computer / Industriestr. 26 / 6236 Eschborn

speicher benötigen. Wir brauchen in dieses Feld also nur eine Null einzutragen. Code ist ein Zeiger auf unsere Interrupt-Routine, die ausgeführt werden soll, wenn ein Diskettenwechsel stattfindet. Wenn wir einen Zeiger auf diese Interrupt-Struktur in dem Data-Feld der IOStdReq-Struktur übergeben und dann noch das Command-Wort mit dem Wert 12 beschreiben, kann der REMOVE-Befehl ausgeführt werden, und unsere Interrupt-Routine ist installiert. Listing 5 zeigt die praktische Anwendung dieser drei Befehle. Zuerst wird die Interrupt-Routine installiert. Sie ist so ausgelegt, daß beim Diskettenwechsel (und wenn keine Diskette im Laufwerk ist) die Hintergrundfarbe 9999 mal verändert wird, wovon man nur ein kurzes Farbflimmern mitbekommt. Nun wartet das Programm auf die linke Maustaste. Dabei kann die Diskette gewechselt werden, um das Ergebnis der CHANGENUM-Funktion zu variieren. Das Ergebnis steht nach Beendigung des Programms in D7. Allerdings stellt die Zahl in D7 nicht die Anzahl der Diskettenwechsel dar.
Zuletzt wird noch der Schreib-/Lesekopf nach Cylinder 0 und dann nach Cylinder 79 bewegt.

Genug für diese Ausgabe,
weiter geht es im nächsten Heft.

```
 jsr        openlib(a6)        ;Dos-Library öffnen
 move.l     d0,dosbase         ;Dos-Basisadresse speichern
 move.l     d0,a6              ;Dos-Basisadresse in A6
 jsr        output(a6)         ;Handle des CLI-Fensters ermittels
 move.l     d0,handle          ;und speichern
 move.l     execbase,a6        ;A6 wieder auf ExecBase zeigen lassen

 lea        diskio,a1          ;Zeiger auf IOStdReq-Struktur
 move       #19,28(a1)         ;Kommando: GETNUMTRACKS
 jsr        doio(a6)           ;Kommando ausfÜhren
 cmp.l      #$a0,32(a1)        ;160 Tracks (normaler Wert)?
 bne        extratracks        ;Nein, Trackzahl <> 160
 move.l     #normtext,d2       ;Adresse des Textes
 move.l     #43,d3             ;Länge des Textes in Zeichen
 jmp        marke              ;folgenden Abschnitt überspringen
extratracks:
 move.l     #extratext,d2      ;Adresse des Textes
 move.l     #58,d3             ;Länge des Textes in Zeichen
marke:
 move.l     dosbase,a6         ;Dos-Basisadresse
 move.l     handle,d1          ;Handle des CLI-Fensters
 jsr        write(a6)          ;Text ausgeben
 move.l     execbase,a6        ;A6 wieder auf ExecBase zeigen lassen

 lea        diskio,a1          ;Zeiger auf IOStdReq-Struktur
 move       #18,28(a1)<        ;Kommando: GETDRIVETYPE
 jsr        doio(a6)           ;Befehl ausfÜhren
 cmp.l      #1,32(a1)          ;3.5 Zoll?
 bne        zoll5              ;Nein, 5.25 Zoll
 move.l     #text35,d2         ;Adresse des Textes
 move.l     #40,d3             ;Länge des Textes in Zeichen
 jmp        print              ;folgenden Abschnitt Überspringen
zoll5:
 move.l     #text525,d2        ;Adresse des Textes
 move.l     #41,d3             ;Länge des Textes in Zeichen
print:
 move.l     dosbase,a6         ;Dos-Basisadresse
 move.l     handle,d1          ;Handle des CLI-Fensters
 jsr        write(a6)          ;Text ausgeben
 move.l     execbase,a6        ;A6 wieder auf ExecBase zeigen lassen

 lea        diskio,a1          ;Zeiger auf IOStdReq-Struktur
 move       #14,28(a1)         ;Kommando: CHANGESTATE
 jsr        doio(a6)           ;Befehl ausfÜhren
 cmp.l      #0,32(a1)          ;Diskette eingelegt?
 beq        diskok             ;Ja, dann verzweige
 move.l     dosbase,a6         ;Dos-Basisadresse
 move.l     handle,d1          ;Handle des CLI-Fensters
 move.l     #nodisktext,d2     ;Adresse des Textes
 move.l     #28,d3             ;Länge des Textes in Zeichen
 jsr        write(a6)          ;Text ausgeben
 jmp        ende               ;Programm beenden

diskok:
 move       #15,28(a1)         ;Kommando: PROTSTATUS
 jsr        doio(a6)           ;Befehl ausfÜhren
 cmp.l      #0,32(a1)          ;Diskette schreibgeschÜtzt?
 beq        ende               ;Nein, dann verzweige
 move.l     dosbase,a6         ;Dos-Basisadresse
 move.l     handle,d1          ;Handle des CLI-Fensters
 move.l     #prottext,d2       ;Adresse des Textes
 move.l     #31,d3             ;Länge des Textes in Zeichen
 jsr        write(a6)          ;Text ausgeben

ende:
```

```
 move.l     execbase,a6      ;A6 wieder auf ExecBase
                              zeigen lassen
 lea        replyport,a1     ;Zeiger auf Reply-Port
 jsr        remport(a6)      ;Msgport aus globaler Liste
                              entfernen

 lea        diskio,a1        ;Zeiger auf IOStdReq-Struktur
 jsr        closedevice(a6)  ;Trackdisk-Device schließen

error:
 rts                         ;Programmende

trdname:    dc.b "trackdisk.device",0
dosname:    dc.b "dos.library",0
normtext:   dc.b "Die Diskette hat 160 Tracks (80 Cylinder).",10
extratext:  dc.b "Die Diskette hat nicht die normale "
            dc.b "Anzahl von 160 Tracks!",10
text35:     dc.b "Das Laufwerk ist ein 3.5 Zoll-Laufwerk.",10
text525:    dc.b "Das Laufwerk ist ein 5.25 Zoll-Laufwerk.",10
nodisktext: dc.b "Keine Diskette im Laufwerk!",10
prottext:   dc.b "Diskette ist schreibgeschützt!",10
even
handle:     dc.l 0
dosbase:    dc.l 0
diskio:     blk.l 20,0
replyport:  blk.l 8,0
```

Abbildung 4: Das Programm untersucht eine Diskette und gibt anschließend einen Informationstext aus

```
#include <exec/types.h>
#include <exec/memory.h>
#include <devices/trackdisk.h>

main()
{
  struct MsgPort *mport;
  struct IOStdReq *iorequest;

  mport = CreatePort(0L,0L);
  iorequest = CreateStdIO(mport);
  OpenDevice(TD_NAME,0L,iorequest,0L);
/* Prüfen, ob Diskette im Laufwerk */
    iorequest->io_Command = TD_CHANGESTATE;
    DoIO(iorequest);
    if (iorequest->io_Actual!= 0)
      printf("Keine Diskette im Laufwerk!\n");
    else
      printf("Diskette ist eingelegt.\n");
/* Prüfen auf Schreibschutz */
    iorequest->io_Command = TD_PROTSTATUS;
    DoIO(iorequest);
    if (iorequest->io_Actual == 0)
      printf("Diskette ist NICHT schreibgeschützt!\n");
    else
      printf("Diskette ist schreibgeschützt!\n");
/* Laufwerkstyp ermitteln */
    iorequest->io_Command = TD_GETDRIVETYPE;
    DoIO(iorequest);
    if (iorequest->io_Actual == 1)
      printf ("Laufwerkstyp: 3.25' (logisch, beim
                              internen Laufwerk!?)\n");
    if (iorequest->io_Actual == 2)
      printf ("Laufwerkstyp: 5.25' (wird bei df0: nie
                                    auftreten!!!)\n");
/* Anzahl der Tracks ermitteln */
    iorequest->io_Command = TD_GETNUMTRACKS;
    DoIO(iorequest);
    if (iorequest->io_Actual == NUMTRACKS)
      printf ("Normale Anzahl Tracks (160 Tracks oder 80
                                      Cylinder)\n");
    else
      printf ("Gesamtzahl der Tracks ist NICHT 160
                                      (komisch)\n");
/* Zählstand für Diskettenwechsel ermitteln */
    iorequest->io_Command = TD_CHANGENUM;
    DoIO(iorequest);
    printf ("Zählstand: %ld\n",iorequest->io_Actual);
/* Schreib-/Lesekopf bewegen */
    iorequest->io_Command = TD_SEEK;
    iorequest->io_Offset = 0L;    /* Zu Spur 0 bewegen */
    DoIO(iorequest);
/* Device schließen und Speicher freigeben */
  CloseDevice(iorequest);
  DeleteStdIO(iorequest);
  DeletePort(mport);
}
```

Listing 4: Ausgabe der Diskinfo in C

```
execbase =      4
findtask =   -294
addport =    -354
remport =    -360
opendevice = -444
closedevice = -450
doio =       -456

run:
 move.l     execbase,a6      ;Exec-Basisadresse
 sub.l      a1,a1            ;0 als Argument => eigener Task
 jsr        findtask(a6)     ;Task suchen
 move.l     d0,replyport+16  ;Task als Empfänger in
                              MsgPort eintragen

 lea        replyport,a1     ;Zeiger auf Reply-Port
 jsr        addport(a6)      ;Reply-Port in globale Liste
                              eintragen

 lea        diskio,a1        ;Zeiger auf IOStdReq-Struktur
 move.l     #replyport,14(a1) ;Reply-Port in Message
                              eintragen
 move.l     #0,d0            ;Laufwerksnummer (hier df0:)
 clr.l      d1               ;keine Flags
 lea        trdname,a0       ;Zeiger auf Devicename
 jsr        opendevice(a6)   ;Trackdisk-Device öffnen
 tst.l      d0               ;Fehler?
 bne        error            ;Ja! Programm beenden

 lea        irqstruct,a0     ;Zeiger auf Interrupt-Struktur
 move.l     #0,14(a0)        ;kein Datenspeicher
 move.l     #routine,18(a0)  ;Adresse der Irq-Routine

 move       #12,28(a1)       ;Kommando: REMOVE
 move.l     #irqstruct,40(a1) ;Adresse der Interrupt-Struktur
 jsr        doio(a6)         ;Interrupt installieren

wait:
 btst       #6,$bfe001       ;Auf linke Maustaste warten
 bne        wait

 move       #13,28(a1)       ;Kommando: CHANGENUM
 jsr        doio(a6)         ;Befehl ausführen
 move.l     32(a1),d7        ;Zählerstand in D7

 move       #10,28(a1)       ;Kommando: SEEK
 move.l     #0,44(a1)        ;Zuerst nach Cylinder 0
 jsr        doio(a6)         ;Kopf bewegen
```

```
move    #1759*512,44(a1)    ;dann nach Cylinder 79
jsr     doio(a6)            ;Kopf bewegen

lea     replyport,a1        ;Zeiger auf Reply-Port
jsr     remport(a6)         ;MsgPort aus globaler Liste
                             entfernen

lea     diskio,a1       ;Zeiger auf IOStdReq-Struktur
jsr     closedevice(a6) ;Trackdisk-Device schließen

error:
 rts                        ;Programmende
```

```
routine:
 lea     $dff180,a2         ;Hintergrundfarbregister
 move.l  #10000,d0          ;9999 Schleifendurchläufe
loop:
 add.w   #1,(a2)            ;Hintergrundfarbe ändern
 dbra    d0,loop
rts                         ;Ende der Interrupt-Routine

trdname:  dc.b "trackdisk.device",0
even
replyport:  blk.l 8,0
diskio:     blk.l 20,0
irqstruct:  blk.l 6,0
```

Abbildung 5: Der Schreib/Lesekopf wird von Spur 0 bis Spur 79 gestept

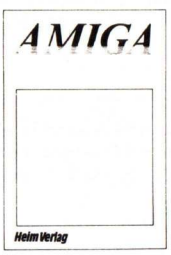
ENDE

★ ★ AUTOREN GESUCHT

Sie
. . . haben eine gute Programmidee
. . . wollen ein Buch schreiben
. . . kennen eine Menge Tips u. Tricks
. . . möchten Ihre Erfahrungen weitergeben

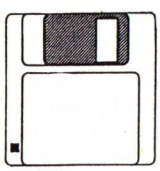
Wir
. . . bieten Ihnen unsere Erfahrung
. . . unterstützen Ihre Ideen
. . . sind ein leistungsstarker Verlag
. . . freuen uns von Ihnen zu hören

Buch



+

Programm



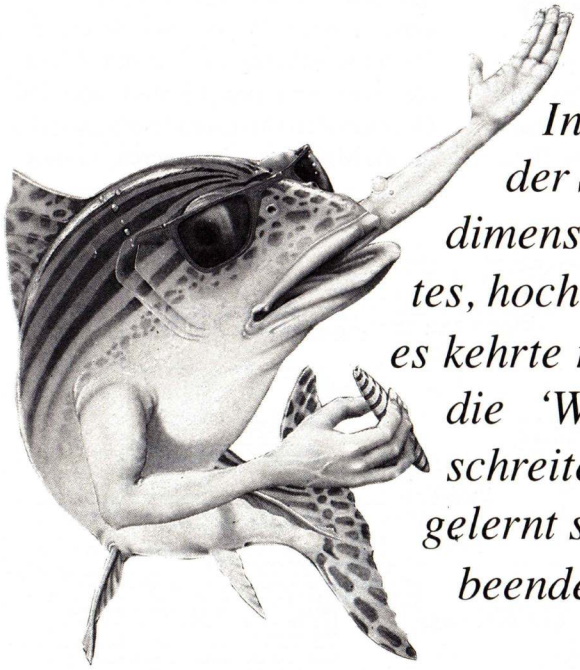
Schreiben Sie uns

Heim-Verlag
Kennwort: Autor
Heidelberger Landstr. 194
6100 Da.-Eberstadt
Tel.: 06151/56057

VON ANDREAS PAUL

FISH

Internationale Agentenjagd



In diesem Spiel geht es um die Bekämpfung der Seven deadly Fins, einer gefährlichen interdimensionalen Anarchistengruppe. Ein eingesetztes, hochklassiges Agententeam hatte keinen Erfolg, es kehrte nie wieder zurück. Als letzte Chance bleibt die 'Warp'-Technik, eine Art dimensionsüberschreitender Geistestransfer. Doch 'warpen' will gelernt sein und nur ein guter Agent kann das Spiel beenden.

Nachdem man sich gedreht hat, kann man in das Wasserschlößchen schwimmen. Dort findet man drei 'warps', die man in folgender Reihenfolge betreten muß:

smooth warp

Hier muß man sich zuerst die '*mould*' besorgen. Die erhält man allerdings nur, wenn man den Vogel aus dem Käfig entfernt. Das schafft man, indem man den Käfig öffnet und einfach den Raum verläßt. Danach muß man sich mit Micky über den Vogel unterhalten, und wenn er dann weggeht, sich die goldene Scheibe schnappen. Mit der Scheibe geht man zu der Schmiede (ohne Micky wiederzubegegnen). Dort nimmt man sich alles aus dem Schrank. Dann legt man die Scheibe in den Schmelztiegel und hält diesen mit der Zange ins Feuer. Natürlich sollte man auch die Asbesthandschuhe tragen. Das geschmolzene Gold schüttet man in die Form. Mit ihr geht man zur '*cool glade*' und wartet solange, bis sich das '*mould*' abgekühlt hat, um es ausschließend mit dem Hammer zu zerschlagen. Das alles muß sehr schnell gehen, da Micky sonst den Verlust der Scheibe bemerkt und Sie ins Jenseits befördert.

jagged warp

Zuerst sollte man sich anständig anziehen. Die Decke braucht man dazu nicht. In der Fahrerkabine findet man eine Taschenlampe (search). Mit dem Licht kann man sich auf nach Osten machen. In der Abtei gibt es unter den Katakomben noch einen Altarraum, wo man ein Seil findet. Mit diesem und dem Stuhl kann man es schaffen, den Bogen zu erklettern. Den Stuhl kann man nur mitnehmen, wenn man (nachdem man ihn hat) das Licht löscht. Auf dem Stuhl stehend, kommt man auf den Bogen, wo man das Seil am '*gargoyle*' befestigt. Wieder unten angekommen, läßt sich das Ganze durch einen kräftigen Ruck zum Einsturz bringen. Das '*gargoyle*' muß man nur noch in den Altar einsetzen, worauf ein Portal erscheint, auf dem ein Kelch steht, in welchem ein '*grommet*' liegt. Diese Aktion schafft man aber erst auf den zweiten Anlauf, da die Zeit, die man zur Verfügung hat, einfach zu knapp ist.

small warp

Um nicht dem Produzenten zum Opfer zu fallen, muß man regelmäßig Kaffee machen lassen. Rod erledigt das prompt, wenn er gefragt wird. Dann nimmt man sich die Kassette aus dem Abfalleimer und besorgt sich auch die beiden aus dem Store (gegenüber vom zweiten Aufnahmestudio). Im zweiten Aufnahmestudio reinigt man zuerst den Recorder mit der Reinigungskassette, bevor man sich die anderen Kassetten anhört. Dazu muß man natürlich die Lautstärke aufdrehen. Auf der letzten Kassette findet man eine Taktangabe (z.B. 5-4-7,5-4-7 ...). Das ist die Kombination, mit der man im Office den Tresor öffnen kann.

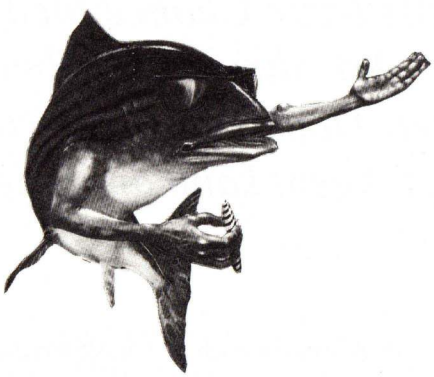
big warp

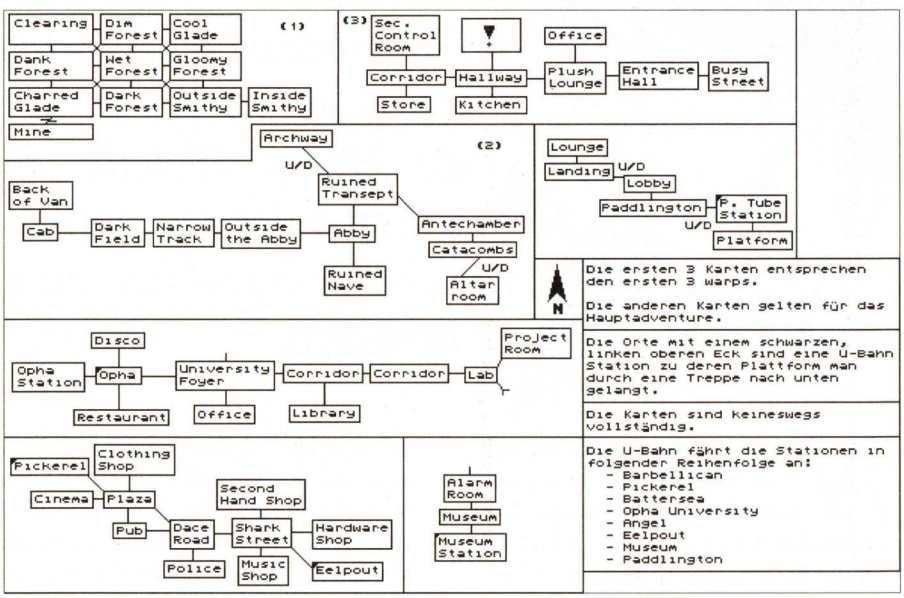
Erst jetzt ist es möglich, das '*big warp*' zu betreten, worin einen die eigentliche Aufgabe erwartet. Es geht also erst jetzt richtig los. In der Wohnung das *fishofax* (liegt auf dem Regal) und die *travel card* (liegt unterm *fishton*) mitnehmen. Bevor man die Rektorin trifft, kann man noch ein paar Besorgungen erledigen. Also in Pickerel aussteigen und im *clothing shop* eine Fliege und eine Sonnenbrille zulegen, im Second Hand-Shop das (eigene) *fishton*, im Musikladen die Ohrstöpsel einkaufen und last but not least den *Hyperdriver* und den *Screwdriver* inklusive Plastiktüte (sehr wichtig) besorgen. Dann geht man noch schnell ins Pub, um von dem Betrunkenen die ID card zu bekommen. Man gibt ihm erst einen *snifter* und fragt ihn dann. Den Zylinder bekommt man angeboten, wenn man sich drei *snifters* gekauft hat. Die soll man sich natürlich nicht selber reinziehen, sondern am besten dem Betrunkenen geben (echt ein dankbarer Abnehmer). Dann muß man sich aber hurtig nach Pickerle aufmachen, um noch rechtzeitig zu der Besprechung mit der Rektorin zu kommen. Von dort begibt man sich in die Bücherei, um sich die Seite mit dem *Focus Wheel* herauszureißen. Diese steckt man dann in den Fotokopier, den man am besten nach Gebrauch sofort ausschaltet. Jetzt ist es wichtig, den *Hyperdriver*, das *focus wheel* zu droppen, da es sonst mit der Tragekapazität des Fisches etwas knapp wird. Auf dem Weg zur U-Bahn geht man noch ins Restaurant, um sich einen Happen Plankton zu kaufen. Mit diesem kann man die Alarmanlage im Museum geschickt umgehen, da die Lichtschranken sichtbar werden. Vom Museum aus geht man dann zu seinem eigenen Appartement, wo man eine weitere Warp findet.

Appartement warp

Am besten hält man sich immer auf den diagonalen Himmelsrichtungen. So ganz klar ist mir das System selbst nach dem ausführlichen Studium der Hints nicht gewesen. Am besten einfach auf gut Glück immer die Diagonalen nehmen (soweit möglich). Wenn es endlich mal geklappt hat, am besten sofort abspeichern.

Dann geht es weiter nach Battersea, wo sich jetzt die Türe öffnen müßte, wenn man den Fotokopierer und das Licht im Appartment ausgeschaltet hat, nachdem man den Hebel umlegt. Jetzt wird Timing wichtig. Man hat genau Zeit, um zweimal nach Westen zugehen, den Filter abzumontieren und wieder herauszukommen, ohne als Scheiblettenfisch zu enden. Jetzt geht man ins Pub, um den Verräter zu entlarven. Zu diesem Zweck lädt man ihn einfach in die Disco ein. Selber muß man sich durch Sonnenbrille und Ohrstöpsel schützen, die Fliege benötigt man, um den werten Kollegen überhaupt discotauglich zu machen. Am einfachsten ist es, wenn man im Pub speichert und solange einen der drei Kollegen einlädt, bis man den richtigen erwischt. Mit dem Kristall aus dem Museum, der Plastiktüte und dem Zylinder ausgerüstet geht man jetzt in Saloms Labor, um sein Kristall zu tunen. Um hineinzugelagen, benötigt man natürlich seine ID-card. Mit der Plastiktüte und dem Zylinder baut man sich jetzt einen kleinen Ballon, um durch die Dachluke zu gelangen. Hier nimmt man die *bridge* und springt einfach über das Geländer. Jetzt hat man alles zusammen, um die Maschine zu reparieren. Also zurück ins Labor und versuchen die Maschine zu reparieren. Hier wird man dann vom reumütigen Verräter gewarnt. Wenn man es dann noch schafft, das Hackerspielchen zweimal zu gewinnen, kann man den *modulator* gegen den *accelerator* austauschen. Den *modulator* kann man bauen, wenn man entsprechend dem Bauplan, der im Computer gespeichert ist, vorgeht. Wenn man den *modulator* auch noch gut festgemacht hat (möglichst mit einer Schraube), hat man das Adventure gelöst.





ENDE

Wichtig ist, daß ein Spieler, der im Level gestiegen ist, mit dem Lord British reden muß. Erst dadurch kann er die gewonnene Erfahrung in "Hit Points"umsetzen. Hit Points bekommt man außerdem, wenn man zum Dungeon des Feuers geht. Dort befindet sich in der zweiten Etage eine Quelle, von der man trinken sollte. Die Charaktergruppe sollte aus verschiedenen Charakteren bestehen. Die ideale Gruppenzusammenstellung sollte sich aus Thief, Wizzard, Fighter und Clericer zusammensetzen. Man kann aber auch anstelle des Clericers oder des Fighters einen Paladin nehmen, der Clericersprüche benutzen kann und im Kampf stark ist.

Der Dieb hat den Vorteil, in den Dungeons Fallen aufzuspüren und stellt sich mit "*Powders*" geschickter an.

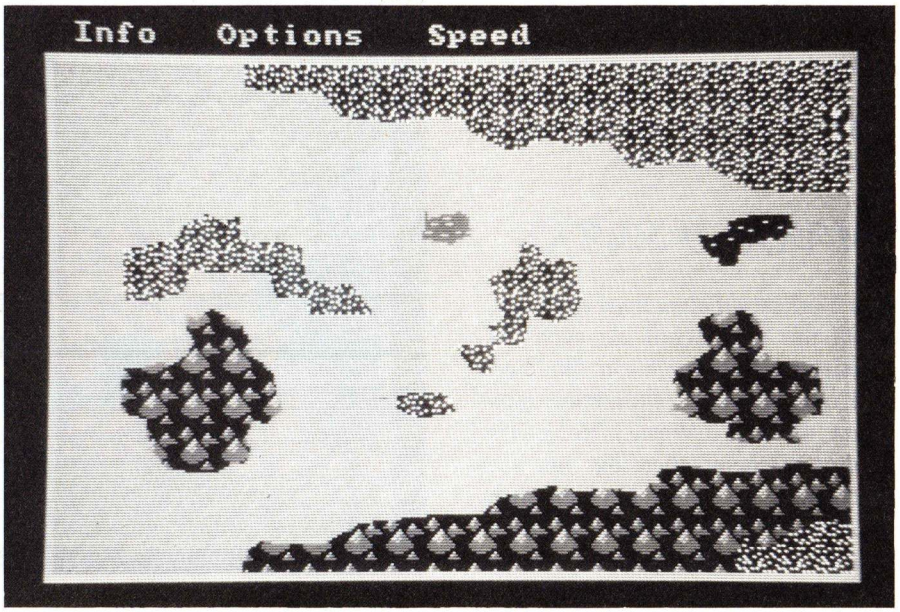
Die Clericersprüche sind hilfreich beim Heilen, Schatzkisten öffnen und anderen Dingen. Der Wizzard dagegen kann die Zaubersprüche besser im Kampf mit den Gegner einsetzen.

Die Koordinatenangaben beziehen sich alle auf das Schloß des Lord British. Sind die Koordinaten 8 W und 35 S angegeben, müssen Sie von Lord Britishs Schloß aus 8 Schritte nach Westen (links) und 35 Schritte nach Süden (unten) gehen.

Gegen sehr starke Gegner, zum Beispiel Drachen, Balrons oder größere Ansammlungen von Wachen empfiehlt es sich, *Powders* einzusetzen. *Powders* bewirkt, daß der Bewegungsablauf des Gegners sich verlangsamt.

Gegen Skelette hilft am besten der *Clericerspruch A (Pontori)*. Aber nur dann, wenn Sie nicht angegriffen werden. Ebenso gilt der Zauberspruch *A (Repond)* gegen Orcs und ähnliche Monster. Sobald Sie den Gegner besiegt haben, können Sie all sein Gold und sonstige Besitztümer übernehmen. Wenn Sie aber Schatztruhen öffnen, sollten Sie unbedingt den *Clericerspruch B (Apparunem)* verwenden, weil die Schatztruhen meistens mit hinterlistigen Fallen belegt sind.

Es gibt eine Stadt namens Dawn, in welcher man so ziemlich alles bekommt. Dort gibt es Powders, Gems, bessere Rüstungen und Waffen als in den anderen Städten. Die Stadt Dawn ist allerdings nur zu erreichen, wenn Sie die Mondstellung 0/0 und 8 West, 35 Süd gehen.



ULTIMA III

Lösung

Um ein Schiff zu bekommen, muß die ganze Mannschaft vernichtet werden. Aber achten Sie auf die Kanonen. Wenn Sie ein Schiff bekommen haben, müssen Sie aufpassen, wo Sie es vor Anker legen.

Ein schnell herankommender Strudel könnte das Schiff zerstören. Wenn Sie sich allerdings auf einer Insel befinden, auf welcher es keine Mondtore gibt, haben Sie Pech gehabt.

Bevor Sie in die Dungeons gehen, sollten Sie ein paar Gems besorgen. Davon brauchen Sie mindestens acht Stück. Gems bekommen Sie bei den Städten Dawn, Devil Guard oder Grey. Mit Hilfe von Gems lassen sich die einzelnen Dungeon-Levels kartographieren. Natürlich brauchen Sie für die Dungeons auch Fackeln. Diese bekommen Sie ebenfalls in den genannten Städten.

Wichtig sind auch die *Marks*, insgesamt vier Stück, die sich in Dungeons zumeist im 8.Level befinden. Jeder der Spieler muß je eine der Marks besitzen.

MARK OF FORCE

Dieses finden Sie in der Höhle des Feuers, wie schon erwähnt, im 8.Level. Die Koordinaten sind 19 Süd, 3 Ost, 1 Nord, 1 Ost, 2 Nord. Mit Hilfe der Mark Of Force kann man unbeschadet durch Kraftfelder gehen.

MARK OF FIRE

Diese finden Sie in der Höhle im Nordwesten, ebenfalls im 8.Level. Die Koordinaten sind 2 Süd, 12 Ost, 1 West, 2 Nord. Die *Marks Of Fire* ermöglichen es, ungehindert die Lava zu durchqueren.

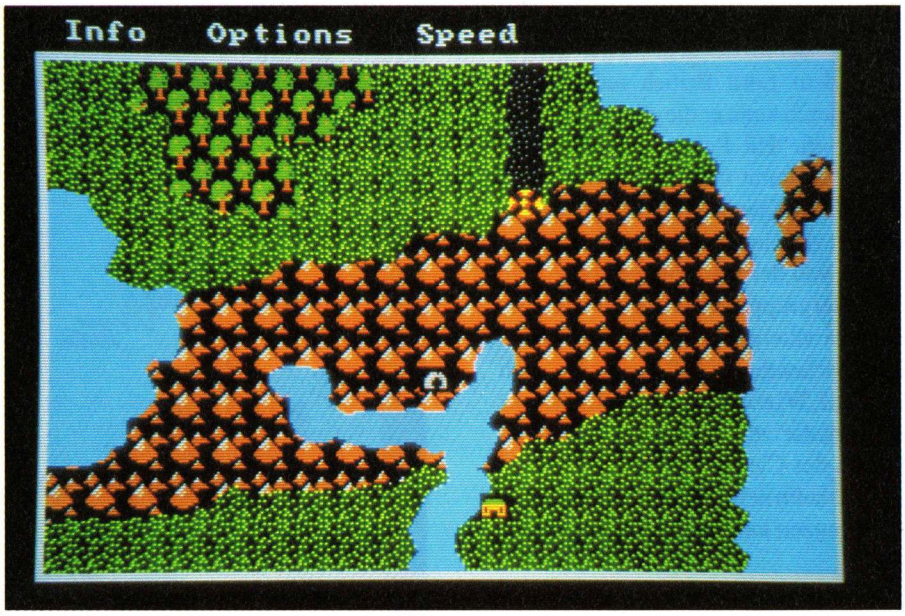
MARK OF KINGS

Die *Mark Of Kings* ist im selben Dungeon wie die *Mark Of Fire*, aber schon im 1.Level. Mit ihrer Hilfe kann man ein bestimmtes Schloß betreten.

MARK OF SNAKE

Diese Mark ist auf einer Insel zu finden, sowohl mit dem Schiff als auch auf schnellerem Weg durch ein Mondtor. Die Koordinaten des Mondtores sind 6 West, 41 Süd, 3 West, 1 Nord. Das Mondtor erscheint allerdings nur zur Mondstelung 3/1. Verlassen kann man die Insel bei 1/3. Die Mark Of Snake brauchen Sie, um zum Schloß von Exodus zu kommen. Denn eine Schlange bewacht den Eingang. Stellen Sie sich mit Ihrem Schiff direkt unter die Schlange, und geben Sie jetzt "*Yell evocare*" ein. Um aus der Bucht herauszukommen, müssen Sie genau das gleiche tun. Um zu dem Land "Ambrosia" zu gelangen, müssen Sie mit dem Schiff in einen Strudel fahren. Dies ist aber nur dann ratsam, wenn alle vier Charakter alle Marks besitzen. Sollte eine der Marks fehlen, sterben Sie. Ambrosia ist ein dunkles Land. Deshalb ist es besser, wenn Sie einen Gem einsetzen, um Ambrosia zu kartographieren. Sie finden auf Ambrosia vier *Shrines*, in denen Sie mit Geld Persönlichkeitsattribute aufstocken können, z.B. Intelligenz, Magic Points und andere.

Möchten Sie die Insel verlassen, finden Sie im Südwesten ein Schiff. Am Shrine geben Sie "*Other Cmd: Search*" ein, dann erhalten Sie bei jedem *Shrine* eine



Karte. Es gibt insgesamt vier verschiedene Karten. Diese brauchen Sie, um das Abenteuer zu lösen.

Wichtig sind auch Exotic-Rüstungen und Exotic-Waffen, welche Sie unbedingt beim Kampf im Schloß des bösen Exodus brauchen. Aber vergessen Sie nicht, bevor Sie das Schloß betreten, die Exotic-Ausrüstung anzulegen. Die Exotics können Sie auf kleinen Inseln finden, indem Sie "*Other Cmd: Suite*" eingeben. Denken Sie aber daran, daß Sie für alle vier Charakter Exotics brauchen. Wenn Sie das Schloß Exodus betreten haben, halten Sie sich links. Die *Powders* sollten Sie jetzt gezielt einsetzen, denn es kommt Ihnen einiges entgegen. Folgen Sie dem langen Gang bis ganz nach oben, dann biegen Sie rechts ein. In der Mitte des Schlosses gehen Sie wieder nach oben. Dann kämpfen Sie gegen den Fußboden und zwar vier Kämpfe, dann stehen Sie vor Exodus. Sie sehen jetzt vier Slots. Treten Sie an den ersten Slot von links. Geben Sie jetzt die "*Card Of Love*" ein. Tun Sie das mit dem Befehl "*Other Cmd: Insert*", nach der Eingabe der Richtung und *"L"* für *Card Of Love*. Jetzt gehen Sie zum zweiten Slot und tun das gleiche, diesmal allerdings mit einem *"S"* für *Card Of Sol*. In den dritten Slot stecken Sie die *Card Of Moon "M"*. Jetzt kommt der letzte Slot dran. Stecken Sie die *Card Of Death "D"* in den Schlitz. Das war der Tod von Exodus.

Gleichzeitig haben Sie auch das Abenteuer gelöst.

H. GUMPRECHT

ENDE

KICK START
AMIGA ZEITSCHRIFT

ABO

KICK START
AMIGA ZEITSCHRIFT

ABO

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Postkarte

Bitte mit 60 Pf. frankieren

Heim Verlag
Heidelberger Landstr. 194

6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57

KICK START
AMIGA ZEITSCHRIFT

Einzelheft- u. Monatsdisketten Bestellung

KICK START
AMIGA ZEITSCHRIFT

Einzelheft- u.
Disketten Service

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Postkarte

Bitte mit 60 Pf. frankieren

Heim Verlag
Heidelberger Landstr. 194

6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57

KICK START
AMIGA ZEITSCHRIFT

Kontaktkarte

KICK START
AMIGA ZEITSCHRIFT

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

KICK START
AMIGA ZEITSCHRIFT

Abonnement

Ja, bitte senden Sie mir die Amiga-Computer Fachzeitschrift ab _____
für mindestens 1 Jahr (11 Hefte) zum ermäßigsten Preis von jährlich DM 70,– frei Haus.
(Ausland: Nur gegen Scheck-Voreinsendung DM 90,– Normalpost.)
Der Bezugszeitraum verlängert sich nur dann um ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekündigt wird.

Name

Vorname

Straße/Nr.

PLZ Ort

Gewünschte Zahlungsweise bitte ankreuzen

☐ Bequem und bargeldlos durch Bankeinzug

Konto-Nr. BLZ

Institut Ort

☐ Ein Verrechnungsscheck über DM_____ liegt bei.

Garantie:
Diese Bestellung kann ich schriftlich innerhalb einer Woche (rechtzeitige Absendung genügt) widerrufen. Dies bestätige ich durch meine 2. Unterschrift.

Datum Unterschrift

Datum Unterschrift

KICK START
AMIGA ZEITSCHRIFT

ABO

KICK START
AMIGA ZEITSCHRIFT

Einzelheft- u. Monatsdisketten

Bestellung

KICKSTART können Sie direkt beim HEIM-VERLAG zum Einzelheft-Preis von DM 7,– (zuzüglich Gebühr für Porto und Verpackung) nachbestellen. Bearbeitung nur gegen beigefügten Scheck über den entsprechenden Betrag (keine Überweisung).

Jan.	Febr.	März	April	Mai	Juni	Juli	Aug.	Sept.	Okt.	Nov.	Dez.
■	■	■	■	■	■	■	■		■	■	
			■			■	■	■	■	■	

1987 = DM
1988 = DM
1989 = DM

+ Gebühr für Porto u. Verp.

☐ Scheck in Höhe zus. DM liegt bei

Disketten Service

Alle Programme, die in KICKSTART veröffentlicht wurden, sind auf Disketten erhältlich. Die Disketten enthalten die Programme von jeweils 2 KICKSTART-Ausgaben. Bestellen Sie durch ankreuzen die gewünschten Disketten

Preis je Diskette 19,– DM	Juli/Aug.	Sept./Okt.	Nov./Dez.	Jan./Febr.	März/April	Mai/Juni	Juli/Aug.	Sept./Okt.
	87	87	87	88	88	88	88	88
	Nov./Dez.	Jan./Febr.	März/April	Mai/Juni	Juli/Aug.	Sept./Okt.	Nov./Dez.	Jan./Febr.
	88	89	89	89	89	89	89	90

Lieferung: gegen beigefügten Scheck zuzügl. 5,– DM Versandkosten (unabhängig von bestellter Stückzahl)

KICK START
AMIGA ZEITSCHRIFT

Einzelheft- u. Monatsdisketten Bestellung

KICK START
AMIGA ZEITSCHRIFT

☐ Ich bitte um weitere Informationen

☐ Ich gebe folgende Bestellung auf

in Bezug auf Ihre Anzeige in Kickstart Heft _____ Seite ____

Menge	Produkt/Bestellnummer	DM	gesamt DM

Datum, Unterschrift (für Jugendliche unter 18 Jahren der Erziehungsberechtigte)

Abgesandt am:

KICK START
AMIGA ZEITSCHRIFT

Firma:

Bemerkungen:

KICK START
AMIGA ZEITSCHRIFT

'TOP 12'
Mein Lieblingsspiel

'TOP 12'
Mein Lieblingsspiel

Wettbewerbsbedingungen siehe TOP 12 in diesem Heft

Bitte frei-machen

MAXON - Computer GmbH
Redaktion KICKSTART
'TOP 12'
Industriestraße 26

D-6236 Eschborn

KICK START
AMIGA ZEITSCHRIFT

Kurzmitteilung

KICK START
AMIGA ZEITSCHRIFT
Kurzmitteilung

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Telefon

Postkarte

Bitte freimachen

MAXON Computer GmbH
KICKSTART Redaktion
Industriestraße 26

6236 Eschborn

KICK START
AMIGA ZEITSCHRIFT

PD Bestellung

KICK START
AMIGA ZEITSCHRIFT
PD Bestellung

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

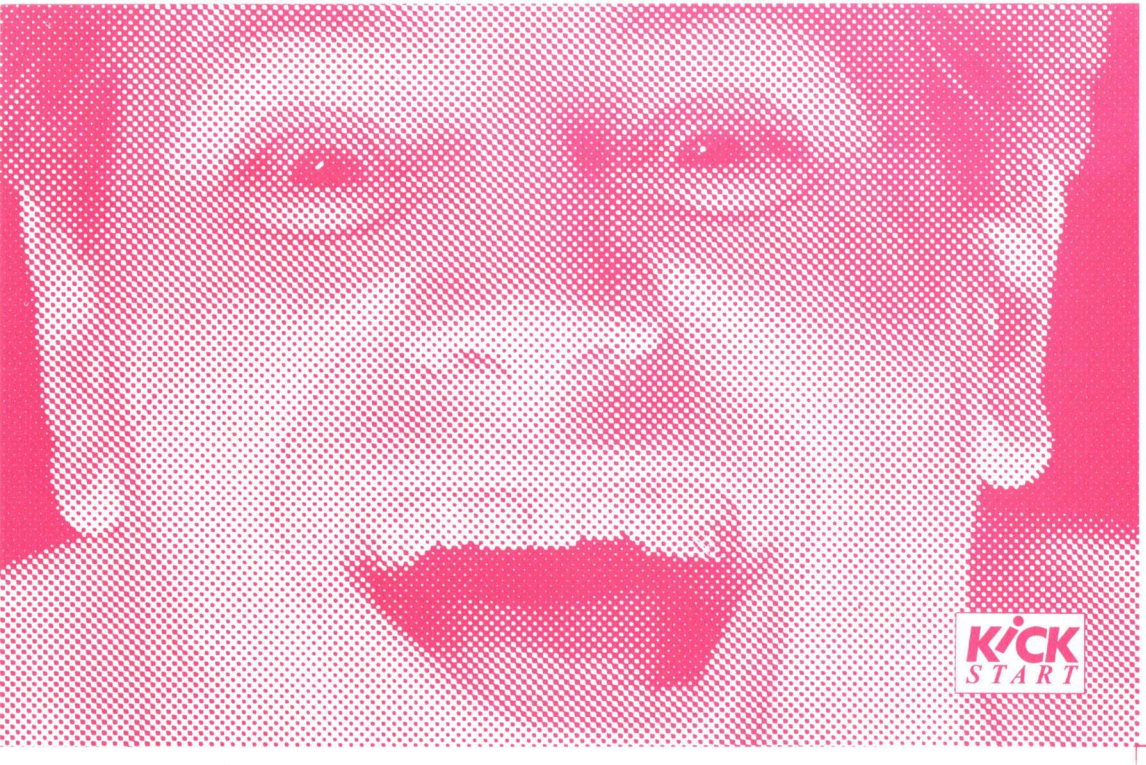
PLZ/Ort

Postkarte

Bitte freimachen

MAXON Computer GmbH
KICKSTART Redaktion
Industriestraße 26

6236 Eschborn



KICK START
AMIGA ZEITSCHRIFT

'TOP 12'
Mein Lieblingsspiel

KICK START
AMIGA ZEITSCHRIFT

Meine Meinung

Zu dem Artikel _____ in Heft ____, Seite ____ hätte ich folgendes zu bemerken:

☐ Ich möchte Ihnen folgendes Programm zur Veröffentlichung anbieten: (Kurzbeschreibung, Sprache, Länge in Druckerseiten)

☐ Ich kann über folgendes Thema berichten: (Tips & Tricks am AMIGA, Hardware, Software, etc.)

☐ Ich möchte gerne Autor in der KICKSTART werden. Meine Fachgebiete: (z.B. LISP, Modula-2, Assembler...)

☐ Ich möchte, daß folgendes Public-Domain Programm in Ihre Sammlung aufgenommen wird.

☐ Sonstiges

Bei weiteren Angaben oder Fragen wenden Sie sich bitte schriftlich oder telefonisch an die Redaktion. Tel. 0 61 96/48 18 11

KICK START
AMIGA ZEITSCHRIFT

Kurzmitteilung

KICK START
AMIGA ZEITSCHRIFT

PUBLIC DOMAIN SERVICE

Ich bestelle folgende PD-Disketten:
(Siehe PD Service in dieser Ausgabe)

_____

_____

_____

_____

_____

Zahlung erfolgt:

☐ per Scheck

☐ per Nachnahme

(nur Inland) zuzüglich DM 3,70 Nachnahmegebühr

Ab 5 Disketten Versandkosten frei

Je Diskette fügen Sie bitte einen Betrag von DM 8,— bei, für Porto und Verpackung je Sendung DM 5,– (Ausland DM 10,–)

_____ Datum

_____ Unterschrift

KICK START
AMIGA ZEITSCHRIFT

PD Bestellung

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____ Datum

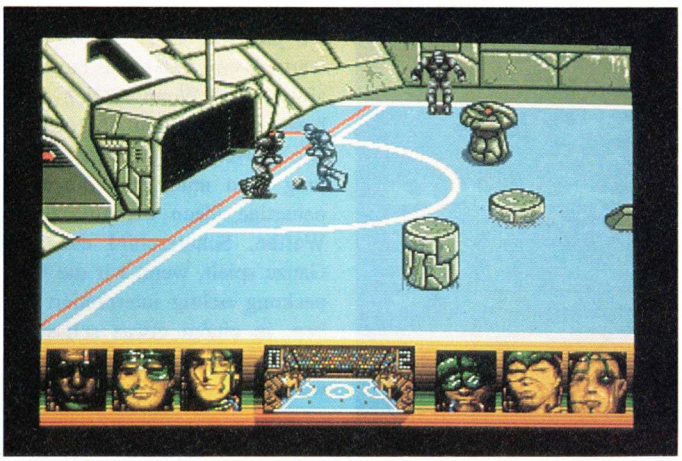
DAS NEUESTE VOM SPIELEMARKT

von ddf und cpl

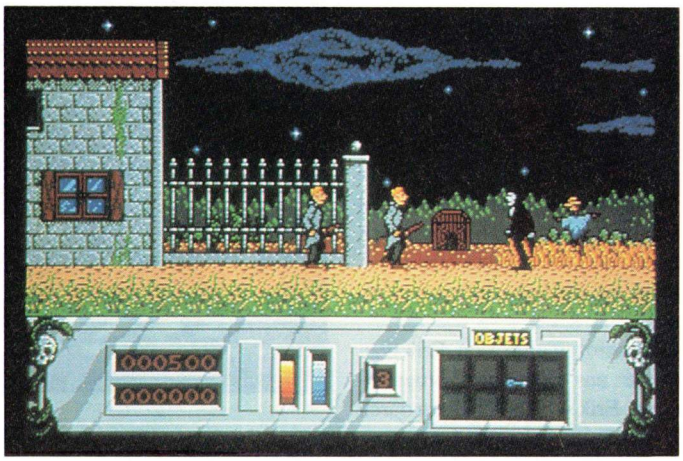
Skateball und Dracula

"Ubi Soft", in Frankreich ein bekannter Softwaredistributor, mischt seit einigen Monaten auch kräftig bei der Spieleproduktion mit. Als erstes Produkt entstand der Gauntletverschnitt *"Puffys Saga"* und auch die beiden Neuheiten *"Skateball"* und *"Nighthunter"*, die bald auf den Markt kommen werden. Unser Mitarbeiter Carsten Borgmeier warf einen kritischen Blick auf Vorabversionen der neuen Ubi-Spiele:

Auf der PCS-Show bekam ich ein Kribbeln in den Joystickfingern, als ich Videoaufzeichnungen von *"Skate Ball"* sah. Deshalb war ich überglücklich, als ich das Spiel endlich in die Finger bekam. Ein oder zwei Spieler liefern sich mit ihrem aus fünf Athleten bestehenden Team eine sportliche Auseinandersetzung, in der es darum geht, eine Stahlkugel in das gegnerische Tor zu schießen. Noch ein futuristisches Fußballspiel? Ja, aber eins mit knallharter Action. Auf dem Spielfeld befinden sich einige linke Hindernisse. Da gibt es Minen, Gräben, Bumpers und ein Hindernis, das über so spitz geschliffene Klingen verfügt, daß es tödlich ist, wenn ein Athlet mit seinen Rollerskates dagegen fährt. Erzielt eine Mannschaft vier bzw. acht Tore, gelangen beide Teams in eine neue Kampfarena, in der noch gefährlichere Hindernisse das Fußballspiel auf Rollerskates erschweren. Grafik und Sound sehen sehr vielversprechend aus.



Rasante Action wird bei Skateboard zu erwarten sein.



Graf Dracula erobert demnächst auch den AMIGA-Bildschirm.

In *"Night Hunter"*, das schon bald auf den Markt kommen soll, spielt Graf Dracula die Hauptrolle. Nachts steigt der berühmte Vampir aus seiner Gruft und saugt seinen Opfern Blut aus der Halsschlagader. Niemand kann ihm entrinnen. Nur Weihwasser, Kruzifixe, Tageslicht oder ein Holzpfahl können den Blutsauger töten. Jahrelang lag Dracula in seinem Sarg und träumte von Blutorgien. Doch durch eine Gleichgewichtsstörung von Gut und Böse konnte es Dracula gelingen, seinen Sarg zu verlassen. Eine Horde von Vampirbekämpfern macht sich auf den Weg, um Dracula zu vernichten.

Der Spieler steuert den Vampir durch 30 verschiedene Level, die jeweils aus 20 verschiedenen Bildern bestehen. In jedem Level gilt es, fünf Schlüssel und drei Urkunden einzusammeln und eine magische Pforte zu finden, um in den nächsten Spielabschnitt zu gelangen. Eine sehr amüsante Sache, zumal der Spieler Dracula mit dem Joystick in drei Rollen schlüpfen lassen kann. Um schneller vorwärtszukommen, verwandelt man Dracula in eine fliegende Fledermaus; muß er sich energisch gegen Angreifer wehren, in einen Werwolf. Um Lebensenergie in Form von Blut zu bekommen, verweilt man in der Gestalt von Dracula. Nur so kann man die Angreifer in die Halsschlagader beißen. *"Night Hunter"* überzeugt durch hervorragende Grafik, zum Spiel passende Soundeffekte und eine lustige Spielhandlung. ■

CBO

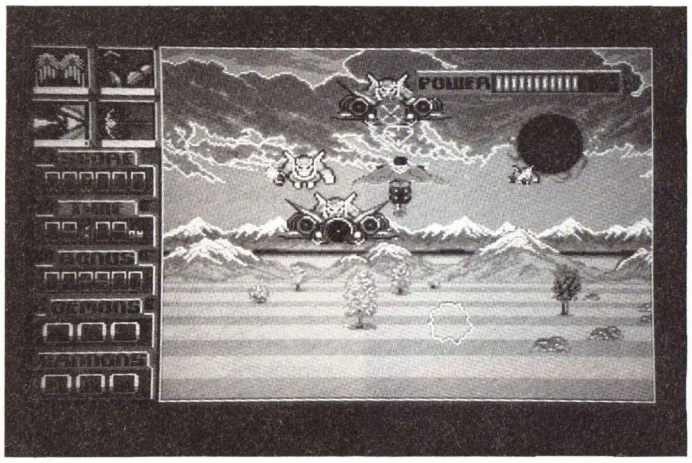
SUPERMAN

Ist es ein Vogel? Ist es ein Flugzeug? ... Nein, es ist Superman, der durch das neueste Spiel von Tynesoft fliegt.

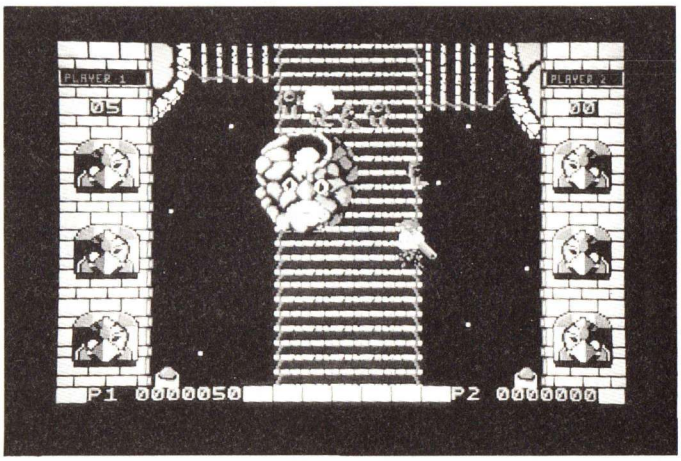
Mit einem wohlbekannten Superhelden als Spielfigur gilt es, acht Missionen zu erfüllen. Vor dem Beginn jeder Episode erscheint ein Blatt im Comic-Stil und veranschaulicht die zu lösende Aufgabe. Superman, der Held aus Stahl, besitzt Wunderkräfte, mit deren Hilfe er jeder Gefahr tapfer ins Auge blickt.

Am Anfang jedes Spielabschnitts wählen Sie die jeweiligen Kräfte, derer sich ihr Held gerade bedient. Angeboten werden Fähigkeiten wie Hitzeblick, Superschlagkraft, Fliegen, Teleskopblick und Superpuste (paß auf Supie, Mundgeruch macht einsam).

Gleich die erste Aufgabe ist eine der schwersten: Superman, der mit Lichtgeschwindigkeit durch die Lüfte braust, entdeckt eine Invasion von Paradämonen, die er zerstören muß. Gleichzeitig darf er aber nicht mit ihren todbringenden Wolken in Berührung kommen. Das zweite SUB-GAME bringt ihn zu dem vor den Toren von Metropolis liegenden "Atlantis", wo er seine große Liebe, Lois Lane, und den Governor aus den Händen der Terroristen befreit. Der Bildschirm wird richtig lebendig, wenn Superman seine Gegner mit dem berühmten Superkick (ähnlich wie Lothar Matthäus seine Elfmeter) in die Wolken kickt. Die Szenarios des Spiels, die graphischen Effekte und die Gegner sind abwechslungsreich gestaltet. Trotzdem raten wir davon ab, die Stunts aus dem Spiel im echten Leben nachzumachen. ■



VICTORY ROAD



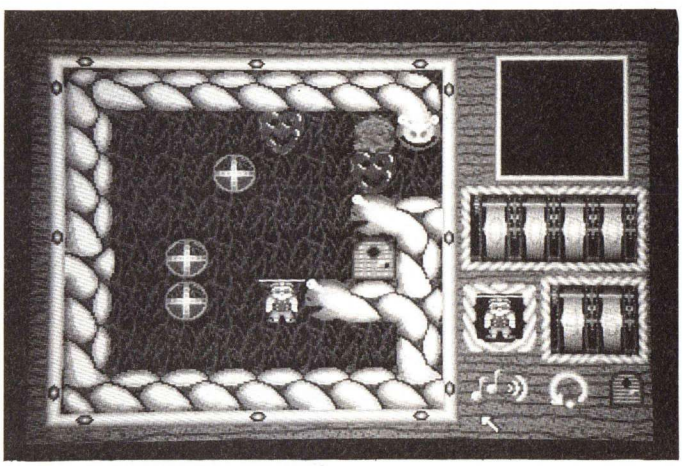
Victory Road ist ein weiteres, vertikal scrollendes Shoot-Em-Up. Die ungewöhnliche Spielidee paßt perfekt zu der Neuartigkeit der Spielhandlung: Bewaffnet mit einem Flammenwerfer marschieren ein oder zwei Helden vorwärts. Wenn sich etwas bewegt - abschießen! Wenn es sich nicht bewegt - trotzdem abschießen! Die abwechslungsreiche Handlung wird unterstützt durch so neuartige Ideen wie zusätzliche Waffen, Schutzschild etc. Das Ganze spielt, wenn wir die Verpackung richtig interpretiert haben, in einem etwas seltsamen ägyptischen Land. Wir sind auch weit davon entfernt, diese Aussage in Zweifel zu ziehen, besonders da Graphik und Sprites nicht detailliert genug sind, um anderes zu erkennen. Das fehlende Konzept harmonisiert dazu vorzüglich. So wird die Frage, was die einzelnen Figuren eigentlich darstellen sollen, letztendlich interessanter als das, was diese auf dem Bildschirm tun.

Um VICTORY RAOD aber wenigstens in irgendeine Schublade zu stecken, könnte man es als eine Art IKARI WARRIOR-Verschnitt bezeichnen. Dies soll aber nicht bedeuten, daß es diesem Standard auch nur irgendwie nahekäme. ■

PROSPECTOR

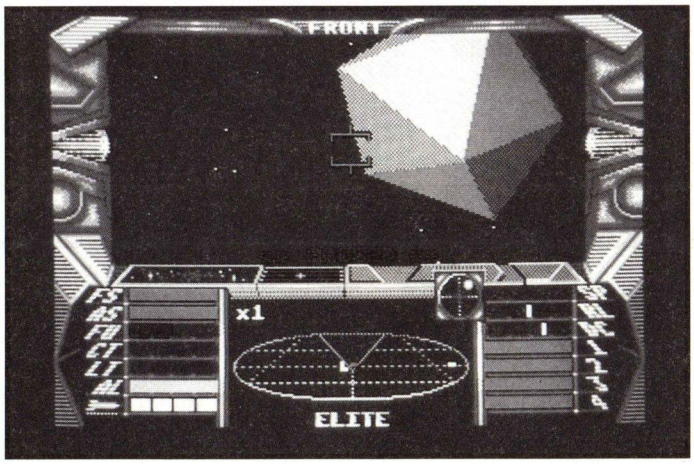
LOGOTRON befindet sich auf Schatzsuche. Gesucht wird, wie in der Softwarebranche üblich, das Geld, das die Computerbesitzer in der Tasche haben. Die beiden Helden von PROSPECTOR sollen Andrew Wright dabei helfen, seinem Traum vom Reichsein näherzukommen. In den Labyrinthen von XOR irren die Zwei umher, auf der Suche nach Kleinodien. Hilfreich ist dabei ein Lageplan. Dieser befindet sich ebenfalls im Irrgarten. Um die Suche zu erschweren, wurde die Karte allerdings in vier Teile zerrissen. Monster, Bomben, fallende Steine und andere Utensilien erschweren die Jagd nach dem Reichtum. In höheren Leveln ist es notwendig, sich den Weg freizusprengen, indem man die Bomben mit Steinen bewirft und sie so zur Explosion bringt. Die meisten Punkte gibt es nicht für den, der die größte Verwüstung anrichtet. Das Ziel ist, mit der kürzesten Anzahl von Schritten alle Schätze einzusammeln und das Labyrinth zu verlassen.

Prospector wurde von Erik van Hesse aus Hannover programmiert. Erik wurde vor allem durch STAR RAY bekannt. Auch diesmal sind Spielsteuerung und Graphik sehr annehmbar. Besonders witzig ist, den Spieler gleich zwei Figuren steuern zu lassen, zwischen denen per Feuertaste hin und her geschaltet wird. Mitgeliefert wird auch noch ein Construction-Kit. Fans von BOULDER DASH und EMERALD MINE finden hier ein weiteres Spiel dieses Genres. Das Geld, das die Prospectoren in die Schatzkammern von Logotron schleppen werden, ist verdient. ■



TOP 12

1. (1) Elite



2. (4) Dungeon Master
3. (2) Interceptor
4. (3) Ports of Call
5. (-) Falcon
6. (8) Dragon's Lair
7. (9) Shanghai
8. (10) Katakis
9. (7) Carrier Command
10. (12) Holiday Maker
11. (5) Starglider II
12. (6) Bard's Tale

In der Spitzengruppe tut sich was - **DUNGEON MASTER** ist auf Platz zwei vorgerückt und macht **ELITE** ernsthaft Konkurrenz. Ansonsten läßt sich feststellen, daß langsam neue Spiele in der Gunst der TOP 12-Wähler steigen und die Veteranen von den Plätzen verdrängen.

BONUS: 20 Spiele von RUSHWARE

Als Bonus verlosen wir diesen Monat 20 Spiele extra, die uns von RUSHWARE zur Verfügung gestellt wurden. Auch im nächsten Monat werden wieder 20 Extraspiele verlost, deshalb lohnt es sich besonders, diesmal an der Wahl teilzunehmen.

Mitmachen kann jeder, der die TOP 12-Postkarte (bitte keine andere!) ausfüllt und an uns zurückschickt. Der Rechtsweg ist dabei ausgeschlossen. Einsendeschluß ist der **30. März 1989**. Karten, die nach diesem Termin eintreffen, werden im nächsten Monat berücksichtigt.

Wir gratulieren den Gewinnern dieses Monats, die jeweils einmal den **TOP 12-TIP COSMIC PIRATE** gewonnen haben:

Philipp Zimmermann, Klingnau (Schweiz) / Andreas Meier, Lage / Michael Gollub, Leverkusen / Hermann Cuylen, Krefeld / Christian Winkler, Augsburg / Alexander Schmidt, Hückelhoven / Michael Merz, Frankfurt / Peter Lorenz, Groszpetersdorf (Österreich) / Gerd Plescher, Eichenzell / Jörg Brack, Herten / Oliver Huthoff, Bremen / Michael Maußner, Reichenschwand

Scary mutant space aliens from mars



Eine Schwalbe macht noch keinen Sommer. In diesem Sinn war Dragons Lair wohl eher der Ausrutscher als der neue Standard von ReadySoft. Nach dem phänomenalen Erfolg des Drachentöters Dirk tummeln sich jetzt all die bisher unverkäuflichen Produkte der in Lewistown N.Y. ansässigen Firma auf dem europäischen Markt. Vor *"Scary Mutant Space Aliens From Mars"* können wir nur warnen. Geplant war es wohl als Adventure. Das einzige Abenteuer besteht allerdings darin, herauszufinden, welche Wörter das Programm überhaupt versteht. Die meistgegebenen Antworten waren mit großem Abstand entweder: "I do not understand" oder "Answer incorrect". Besonders peinlich ist, daß verschiedene Gegenstände zwar gezeigt und auch im Text aufgeführt werden, der Parser von ihrer Existenz aber absolut keine Ahnung hat. Strafmildernd könnte man immerhin vermerken, daß die Graphik akzeptabel und die sehr mageren Bedienungsmöglichkeiten benutzerfreundlich sind. Das abschließende Urteil für den Käufer lautet aber eindeutig: Geldverschwendung. ■

CHASE

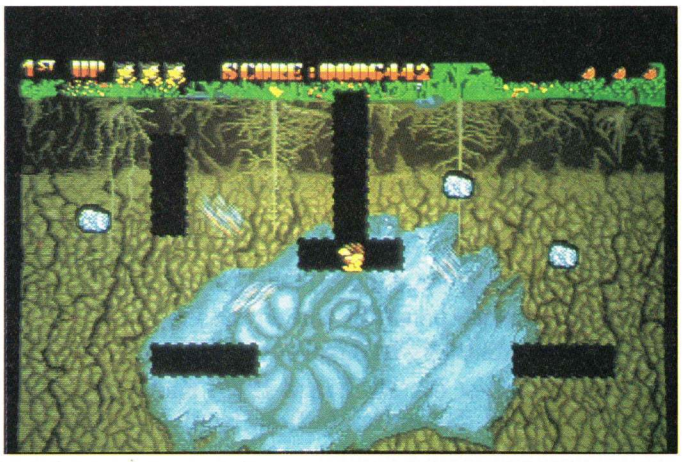
Mastertronic bietet mit *CHASE* ein sehr schnelles - fast hektisch wirkendes - Spiel mit Vektorgraphik an. Die dazu mitgelieferte Story erzählt von "...einem Kampf um Freiheit, Galanterie und wahre Liebe, den nur ein couragierter Held bestehen kann. Ein eiserner Wille und blitzschnelle Reflexe sind hierfür von Nöten...". Irgendwie muß dabei allerdings das falsche Spiel in die Verpackung gerutscht sein. Alles, was wir gesehen haben, sind Striche und Dreiecke. Eintönig wurde CHASE eigentlich erst nach circa 2 Minuten, als uns auffiel, daß sich beim besten Willen nichts ändert. Auch die unpassendste Story kann aus einem schlechten 3D-Vektorspiel kein gutes Programm machen. Hinzu kommt, daß Steuerung und Spielidee einfach lausig sind.
Alles in allem eine Mischung aus "The Empire strikes back" und Einfallslosigkeit. Die in der Anleitung geforderten "...lightning reflexes..." sollten Sie allenfalls dazu benutzen, diesem Programm auszuweichen, falls es Ihren Weg kreuzt. ■



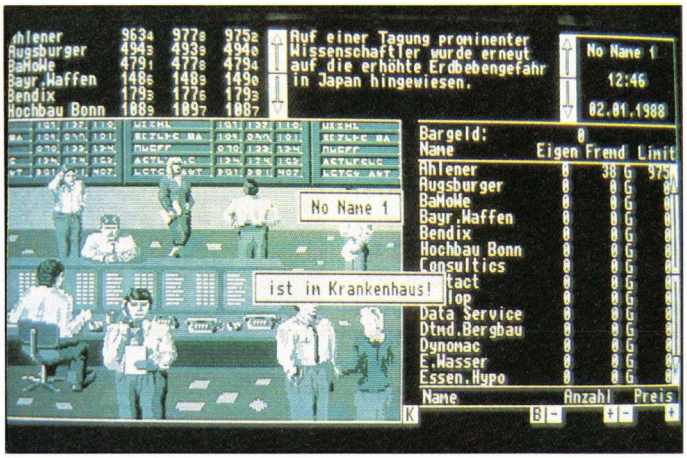
DUGGER

Herbie Stone, der Höhlenmensch aus dem Spiel CRACK, erscheint erneut in Linels jüngstem Werk *"DUGGER"*. Die Spielidee selbst zeigt erstaunliche Übereinstimmungen mit der des Spielhallenautomten "DIG-DUG".
Der Vorspann ist etwas ganz Besonderes, eine Art Trickfilm, wie wir ihn von *"ZOOM!"* kennen. Mit dem eigentlichen Spiel hat er nichts zu tun.
In DUGGER sieht man ein unterirdisches Höhlensystem, das von der Seite angeschnitten wurde. Sie müssen Tunnel durch die Erde graben, um zu den anderen Kreaturen vorzudringen. Wenn Sie nahe genug an eines der Biester herangekommen sind, drücken Sie den Feuerknopf. Als Resultat bläht sich das Monster immer mehr auf, bis - BANG! (another one bites the dust). Direkte Berührung der Geschöpfe ist fatal. Wahren Sie besser etwas Distanz, besonders bei den feuerspeienden Drachen. Eine Möglichkeit, ihn auszutricksen, ist, sich an ihn heranzupirschen, während er einen vertikalen Gang hinaufstürmt. Eine andere Möglichkeit, ihn zu erlegen, bieten kleine Felsbrocken. Wenn man unter diesen die Erde weggräbt, stürzen sie hinab und erschlagen alles darunter Befindliche - also Vorsicht!
Das Spiel wäre recht einfach, wenn die Biester in den bereits gegrabenen Tunneln bleiben würden. Sie haben aber die unangenehme Eigenschaft, sich in Geister zu verwandeln, davonzuschweben und an einem völlig anderen Platz wieder aufzutauchen.
Die Levels werden in ansteigender Reihenfolge immer schwieriger, es erscheinen immer mehr Monster, deren Reaktionen schneller werden.
Insgesamt ist DUGGER recht spielbar. Kritikpunkte wären allenfalls der Qualitätsabfall zwischen der Graphik des Vorspanns und der des Spieles. Das dafür geeignete Publikum sind wohl die etwas jüngeren Computerbesitzer.

■



Wall Street Wizard



Für Spekulanten genau das Richtige - Wall Street Wizard.

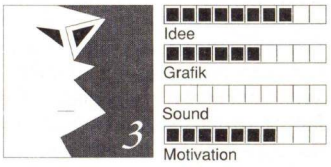
Es klingelt das Telefon, hier tackert der Fernschreiber, dort brüllt jemand "Kaufe!!!". Hektik und Streß begleiten die Arbeit an der Börse. Das wird auch in *"Wall Street Wizard"*, dem neuen Börsenspiel von Expert Software, deutlich. Doch bevor man sich ins komplexe Spielgetümmel wagt, sollte man erst einmal das fünfzigseitige deutschsprachige Handbuch studieren, bietet "Wall Street Wizard" doch eine Vielzahl von Spielmöglichkeiten und Optionen. So gibt es beispielsweise 10 Schwierigkeiten und fünf Zeitstufen. Maximal können sechs Spieler gegen zwölf computerverwaltete Börsenspekulanten antreten. Die Programmierer haben sogar eine Modemoption ins Programm eingebaut (wenn man oft davon Gebrauch macht, muß man tatsächlich an der Börse spekulieren, um die teuren Telefonrechnungen zu bezahlen!).

Ziel des Börsenspektakels ist es, die Gegner nach Punkten, Zeit oder Geld zu übertrumpfen. Zu Beginn des Spiels muß jeder Spieler auf Kundenaufträge warten. Mit einem Auftrag in der Tasche geht es dann in den Börsensaal. Hier ist gutes Kalkül gefragt. Wertvolle Aktien sollten so günstig wie möglich erworben werden. Während der Arbeit im Börsensaal erscheinen Mitteilungen, welche Aktien gerade steigen und welche fallen. Es müssen permanent Aktien gekauft und verkauft werden. Das Kaufverhalten sollte sich den vom Computer beschriebenen Marktsituationen anpassen.

Während des Spielverlaufs muß man durch zahlreiche Menüs toben. Neben den Möglichkeiten, die das Spekulieren an der Börse bietet, kann der Spieler aber auch noch andere Dinge unternehmen: Zum Beispiel kann er seinem Konkurrenten Schlägertruppen auf den Hals hetzen, Statistiken betrachten, Informationen kaufen, Detektive anheuern oder seine Gesundheit beim Arzt wieder auf Vordermann bringen. Alles in allem hat Expert Software die Börsenatmosphäre gut eingefangen. Die vielen Optionen und Funktionen halten das Spiel interessant. Die Grafiken sind für Strategiespielverhältnisse gut gelungen. Nur schade, daß es keinen Sound gibt.

CBO

Info: Profisoft



Cosmopiraten



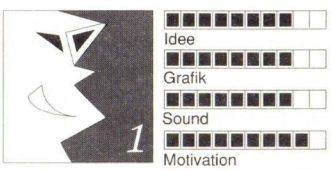
Cosmic Pirates zeichnet sich durch ausgezeichnete Grafik, guten Sound und einen hohen Motivationsgrad aus.

Guy Manly ist ein cleverer Weltraumbandit. Mit einem schwer bewaffneten Raumschiff, das er irgendwann einmal gekapert hat, fliegt er durch die unendlichen Weiten des Alls und überfällt Transportschiffe. Auch wenn er einen Großteil der Beute an das sogenannte Council abgeben muß, geht es dem räudigen Piraten finanziell ausgezeichnet. Ohne das Council (eine Art Vermittlungsbüro, das gegen entsprechendes Entgelt Informationen über Position und Bewaffnung der Transportschiffe herausrückt) könnte Manly nicht so effektiv die Transportschiffe (Space Trucks) ausfindig machen, ihre Bewaffnung zerstören und sie kapern.

Hat Guy einen Space Truck gekapert, muß er ihn zum Hauptquartier des Councils schleppen, und dann wird abgerechnet: 90 Prozent bekommt die kriminelle Agentur, den Rest Guy. Als Pirat läßt sich mit diesem Geschäftsverfahren 'ne Menge Kohle machen. Doch dummerweise läßt das Council nicht jeden für sich arbeiten. Bevor man die Positionen von Frachtschiffen etc. bekommt, muß man sich erst einmal in einigen Simulatoren verdient machen. In den Simulatoren sieht man genau wie in den späteren Missionen das Geschehen aus der Vogelperspektive, mit dem Unterschied, daß in den Simulatoren die Action in einer Art Vektorgrafik dargestellt ist. Verschiedene Feindschiffe sind zu zerstören, Überfälle und das Kapern von Frachtschiffen können ebenfalls geübt werden. Je mehr Punkte man im Simulator macht, desto beutereichere Missionen verschafft einem das Council. Fliegt man eine Mission, sieht man einen Pfeil in der Mitte des Bildschirms, dem man nachfliegen muß. Irgendwann erscheint ein Space Truck, den man kräftig beschießen sollte. Währenddessen wird man auch von anderen Raumschiffen angegriffen, gegen die es sich zur Wehr zu setzen gilt. Hat man die Verteidigungsanlagen auf dem Frachtschiff zerstört, kann man auf einer Plattform landen und dann das Frachtschiff zum Council ziehen. Nach der Abrechnung läßt sich das Geld in anspruchsvollere Simulatoren investieren. Danach vergibt das Council noch profitablere Missionen, in denen die Space Trucks zwar schwieriger zu bezwingen sind, dafür aber mehr Geld bringen. **Fazit:** Super! Tolle Sounds von David Whitaker, rasend schnelle Grafik, heiße Action mit einer Prise Handelssimulation und einer eigenständigen Spielidee. Cosmic Pirate von Palace Software ist kein Remake eines schon vorhandenen Spiels, Cosmic Pirate ist ein Original.

CBO

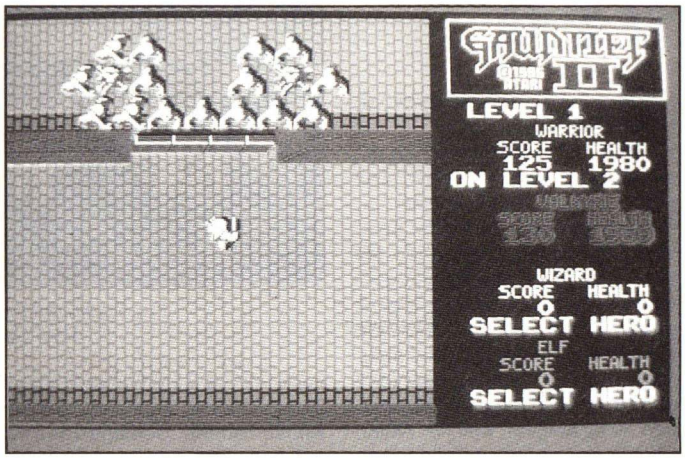
Hersteller:
Palace/Outlaw/Zippo Games



Gauntlet II

Fortsetzung folgt...

Darauf haben die Gauntlet-Fans lange gewartet. "Gauntlet II" ist da. Hersteller "U.S Gold" hat einen würdigen Nachfolger zum Spieleklassiker Gauntlet kreiert. Bis zu vier Spieler können sich gleichzeitig an Gauntlet II erfreuen. Bei drei oder mehr Spielern benötigt man allerdings ein spezielles Joystickinterface, das man bei U.S Gold bestellen kann.

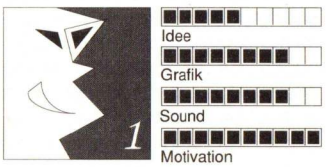


Unzählige Monster wollen einem ans Leder.

Das Spielprinzip der oft kopierten Automatenumsetzung ist denkbar einfach. Jeder Spieler steuert seinen Charakter durch Labyrinthe und muß dabei ein Feld erreichen, auf dem "Exit" steht. Die vier Helden von Gauntlet II sind genau dieselben wie beim Vorgänger, nämlich Krieger Thor, Walküre Thyra, Kobold Questor und Hexenmeister Merlin. Der Spieler steuert seinen Charakter durch ein Labyrinth und versucht, den Ausgang zu finden. Währenddessen begegnet man Monstern und Gespenstern, die an der wertvollen Lebensenergie nagen. Um neue Energie zu bekommen, braucht man nur kraftspendende Speisen einzusammeln. Neben den Speisen gibt es noch viele andere Gegenstände wie Zaubertränke, Amulette oder Schatzkisten im Labyrinth, die alle besondere Vorzüge und Nachteile für den Spieler bieten. Insgesamt gibt es 10 verschiedene Monster, die dem Helden ans Leder wollen. Von primitiven Gespenstern bis zu Giftpfützen und steinewerfenden Kobolden ist alles vertreten.
Das Geschehen präsentiert sich aus der Vogelperspektive, so daß der Bildschirm in alle acht Richtungen scrollt. Die Grafik der Charaktere und des Labyrinths ist nicht sonderlich spektakulär. Bewunderswert bei der Grafik ist eigentlich nur die Tatsache, daß unzählige Sprites gleichzeitig auf dem Bildschirm dargestellt sind. Darin liegt auch der Reiz des Spiels. Was glauben Sie, wieviel Spaß es macht, gegen 30 Monster gleichzeitig mit Joystick und Feuerknopf vorzugehen? Beim Soundtest fiel die Sprachausgabe auf, die originalgetreu vom Automaten übernommen wurde.
Fazit: Gauntlet II ist ein unterhaltsames Game, mit dem mehrere Spieler zusammen viel Spaß haben können. Der zweite Teil bietet mehr Monster, Gegenstände und Schikanen als das Vorgängerprogramm. Gauntlet II ist meiner Meinung nach noch um ein paar Klassen besser als der beliebte Vorgänger. Ich kann Ihnen diese sehr gute Automatenumsetzung nur wärmstens empfehlen.

CBO

Info: Rushware



Double Dragon

Wilde Prügelei!!!

In amerikanischen Großstädten sollte man ständig eine Waffe bei sich tragen. In dunklen Gassen lauern nämlich oft Straßenbanden, die für eine Handvoll Dollar mit Baseballschlägern auf friedliche Passanten losgehen und ihnen gar die Kehle durchschnibbeln. In dieser gefährlichen Umgebung spielt *"Double Dragon"* von "Melbourne House".

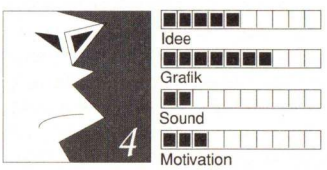


Es gilt, wilde Schlägertypen aus dem Weg zu räumen, um seine Angebetete zu befreien.

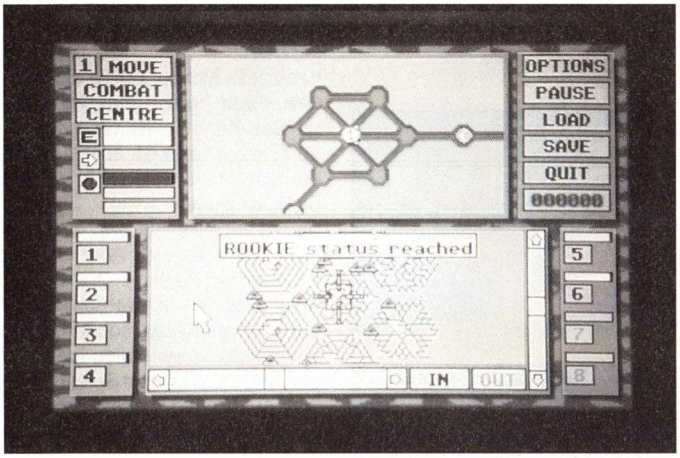
Shadow Boss, der Chef einer solchen Straßenbande, hat die Freundin eines jungen Amerikaners entführen lassen. Billy macht sich jetzt mit seinem Zwillingsbruder Jimmy auf den Weg, um das hübsche Mädel aus den Klauen der Bande zu befreien. Ein beschwerlicher Weg, denn überall lauern Schlägertypen, die nicht lange fackeln und sofort angreifen. Mit Karatetritten, Baseballschlägern, Peitschen, Messern und Handgranaten versuchen sie die beiden Helden daran zu hindern, in das Haupquartier der Bande vorzudringen. Glücklicherweise sind Jimmy und Billy nicht wehrlos. Mit Fausthieben und Tritten können sie ihre Angreifer außer Gefecht setzen. Haben sie einen bewaffneten Angreifer zu Boden gestreckt, läßt sich die Waffe durch Druck auf den Feuerknopf einsammeln und gegen die Gang einsetzen.
Manchmal greifen mehrere Bandenmitglieder gleichzeitig an. Der eine wirft mit Messern, der andere schwingt seinen Baseballschläger und ein dritter wirft mit Kisten und Fässern. In solch einem Falle sind schnelle Reaktionen gefragt, um seinen Weg ohne Repressalien fortsetzen zu können.
Die beiden Helden bewegen sich bei vertikalem Scrolling durch Fabrikhallen, Wälder und Felsenlandschaften. Insgesamt gilt es, sich durch fünf Landschaften voller Gefahren zu kämpfen, um dann im letzten Level gegen den Boß der Bande anzutreten. Fazit: Grafisch ist "Melbourne House" die Automatenumsetzung zu "Double Dragon" einigermaßen gut gelungen. Alle fünf Levels verfügen über ansehnliche Sprites und Hintergründe. Der Sound kommt bei der Bewertung schlecht weg, da einerseits viel zu wenig davon vorhanden ist und andererseits die Effekte sich auf simple Geräusche beschränken. Gut finde ich, daß man mit zwei Spielern gleichzeitig gegen die vielen Angreifer antreten kann. Wovon ich allerdings sehr enttäuscht war, das ist die geringe Kampfstärke der Angreifer. Alle greifen immer nach dem gleichen Muster an. Für einen geübten Spieler ist es deshalb überhaupt kein Problem, das Spiel gleich beim ersten Mal durchzuspielen, zumal die Anzahl der Bildschirmleben recht hoch bemessen ist.

CBO

Info: Leisuresoft

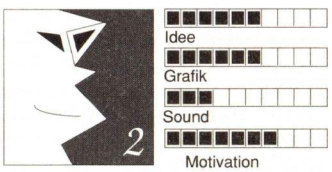


Tracker



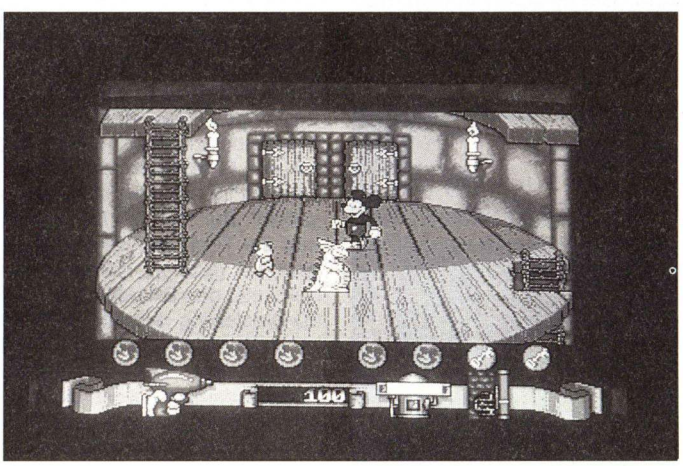
Bei dem Spielfeld handelt es sich um ein riesiges Labyrinth, das aus vielen einzelnen Korridoren besteht. Jeder dieser Korridore endet an sogenannten "Knotenpunkten", von denen man wieder in einen oder mehrere andere Korridore gelangen kann. Zur Fortbewegung stehen dem Spieler bis zu acht Gleiter zur Verfügung. Dieser Komplex ist in sieben sechseckige Teillabyrinthe unterteilt, die alle miteinander verbunden sind. In der Mitte eines solchen Teilkomplexes befindet sich jeweils ein Computer. Um den Zentralcomputer vernichten zu können, werden spezielle Bomben benötigt, die der Spieler nur erhält, wenn er die Computer der anderen sechs Komplexe schon zerstört hat. Bei jedem Komplex sind die Korridore mit ihren Knotenpunkten anders angelegt, so daß immer neue Wege gefunden werden müssen, um zu dem jeweiligen Computer zu kommen. Aber einfach nur den richtigen Weg zu den Computern zu finden, wäre zu einfach. Deshalb befinden sich in jedem Labyrinthkomplex vom Zentralcomputer gesteuerte Fahrzeuge, durch die dem Spieler der Zugang zu den Computern erschwert wird. Um das aktuelle Geschehen beobachten zu können, werden auf dem Bildschirm Landkarten dargestellt, auf denen nicht nur die Korridore, sondern auch die Positionen der feindlichen sowie der eigenen Gleiter zu sehen sind. Für die Flugruten der eigenen Gleiter muß nur ein Punkt angegeben werden, da die Arbeit auf Wunsch vom Autopilot übernommen werden kann. Natürlich ist auch manuelle Steuerung möglich, sie hat sogar entscheidende Vorteile.
Wurde der Zentralcomputer zerstört, ist das Spiel noch lange nicht geschafft, denn es folgen dann noch viele schwierigere Levels.
Bei der Graphik muß zwischen der Vektorgraphik und der sonstigen unterschieden werden. Wird auf manuelle Steuerung umgeschaltet, sieht der Spieler aus der Sicht des Piloten eines Gleiters die Korridore in einer Vektorgraphik, bei der die einzelnen Flächen leider nicht ausgefüllt sind, was man eigentlich von Vektorgraphiken auf dem AMIGA schon erwarten kann. Allerdings sind die Korridore eines solchen Labyrinths wahrscheinlich sowieso nicht sehr abwechslungsreich. Lediglich die Computer der einzelnen Komplexe sowie die feindlichen Gleiter haben ausgefüllte Flächen und sind dadurch rechtzeitig zu erkennen. Die sonstige Graphik ist durchschnittlich. Der Sound wurde so gut wie vergessen, spielt bei Strategiespielen aber auch eine nicht so große Rolle. Die Bedienung des Programms läßt sich schnell erlernen, und der Bildschirm ist übersichtlich aufgebaut. Insgesamt gesehen ist Tracker ein gutes, actiongeladenes Strategiespiel, das auch nach längerer Zeit noch Spaß macht.

Florian Du Bois



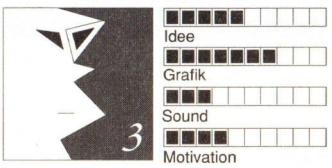
Mickey Mouse

Das Geschäft mit den großen Namen



Es begab sich zu einer Zeit, in der nicht Geschäftemacher, sondern Ungeheuer, Magier und Hexen die Welt beherrschten. Eben diese dreisten Hexen haben Merlins Zauberstab gestohlen und herrschen nun über Disneyland. Nur Mickey Mouse (das heißt der Spieler) ist in der Lage, die vier Stücke des Zauberstabs (gehütet in vier Türmen) zurückzuerobern. In eben diese Rahmenhandlung ist das neue Spiel des Softwarehauses GREMLIN gebettet, das jetzt nach dem ATARI ST auch auf dem AMIGA erschienen ist. Das Spiel ist im großen und ganzen so gestaltet, daß man in einem Turm immer höher klettern muß, um oben der Hexe einen Teil des Zauberstabs abzunehmen. Hat man alle vier Teile des Zauberstabes, trifft man auf den "Chef" der Hexen und muß ihn in einem Endkampf besiegen. Geister, Skelette und andere Gegner versuchen, dies Mickey zu erschweren. Doch unsere Maus ist nicht allein auf sich selbst gestellt, nein, unser Merlin (woher kenne ich nur diesen Namen?) hat Mikkey einen Hammer und eine Wasserpistole mitgegeben, die mit Zaubertrank gefüllt ist und aus den damit beschossenen Monstern kleine Bonusgegenstände (Flaschen, Schlüssel,...) macht. Das eigentliche Spiel teilt sich noch in einige Unterspiele, in denen es darum geht, die Türen, aus denen einige der Gegner kommen, zu verbarrikadieren. Doch nun zum Eindruck, den das Spiel bei mir hinterlassen hat: Die Grafik ist detailliert, die Animation ist im großen und ganzen disneymäßig, und der Sound ... nun, man MUSS ihn nicht abdrehen (ganz im Vertrauen: er nervt). Eigentlich wären mit Mickey Mouse als Charakter die Voraussetzungen für ein interessantes Spiel dagewesen, bei dieser Umsetzung fehlt mir jedoch einfach der Spielwitz. Nachdem man sich an der guten Animation und an den (zweifellos lustigen) Details sattgesehen hat, macht sich Langeweile und Routine breit. Doch in unserer Zeit, in der nicht Ungeheuer, Magier und Hexen, sondern Geschäftemacher die Welt beherrschen, ist die Verlockung groß, mit einem laschen Spiel (ABER mit dem Namen MICKEY MOUSE) eine schnelle Mark zu verdienen. Schade, aber wir sind gespannt, was uns BATMAN, TIM UND STRUPPIE usw. noch alles bescheren werden.

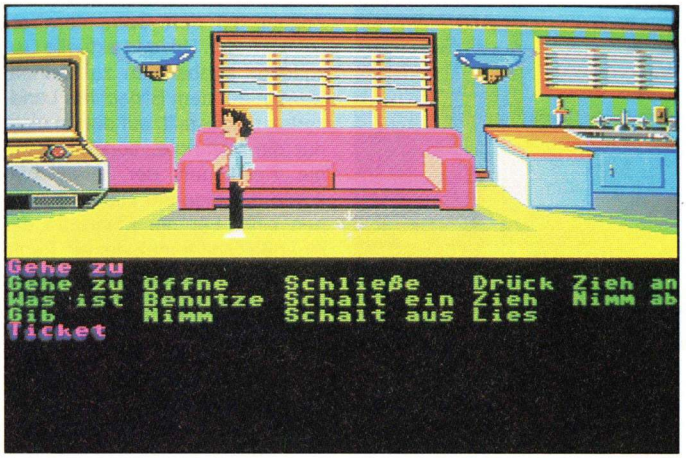
Sven Stillich



Zak McKracken!

Wo gibt's denn sowas? Außerirdische haben eine Verdummungsmaschine gebaut und wollen damit die Welt beherrschen. Das Schlimme ist, nur ein verrückter Klatschreporter namens Zak McKracken kann das Unheil von der Welt abwenden. Nur, Zak ist total unfähig, deshalb muß ihm ein begnadeter AMIGA-User unter die Arme greifen, um die Aliens aufzuspüren und ihre schreckliche Maschine zu zerstören. Soweit zur Hintergrundgeschichte von "Zak McKracken", dem neuesten Adventure aus dem Hause Lucasfilms, das "Softgold" vollkommen ins Deutsche übersetzt hat.

Dieses Abenteuerspiel ist kinderleicht zu bedienen: Der Spieler gibt alle Kommandos mit der Maus ein, indem er aus aufgelisteten Befehlen ein Verb heraussucht und anschließend einen Gegenstand im Grafikfenster ansteuert. Ich möchte das an einem Beispiel verdeutlichen: Wählt man "Gehe zu" und bewegt dann den Mauszeiger zur Tür, watschelt Zak in diese Richtung. Gibt man auf diese Weise "öffne" ein und zeigt wieder auf die Tür, öffnet sich die Tür. Kinderleicht! Man kann Zak überall in der Stadt bewegen, trifft auf viele Gestalten, die Hinweise geben oder wichtige Gegenstände besitzen, die man zur Lösung von Puzzles braucht.



Ausgerechnet Zak McKracken, der unfähige Reporter, muß die Welt vor dem großem Unheil retten, denn die Verdummungsmaschine droht.

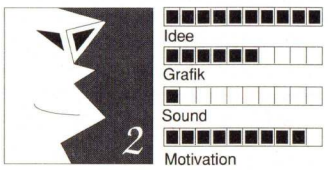
Der Spieler ist mit Hilfe der Funktionstasten F2-F4 Herr über drei weitere Hauptfiguren, nämlich Annie, Leslie und Melissa. Zak muß die drei Mädels aber erst einmal finden, bevor der Spieler mit den Funktionstasten zwischen den Charakteren hin- und herschalten kann. Zur Auflockerung der Spielhandlung zeigt der AMIGA ab und zu kleine Filmszenen, in die der Spieler nicht eingreifen kann. Jedoch sollte man diese Szenen nicht als Ruhepause nutzen, sondern stattdessen aufpassen. Aus diesen Filmszenen lassen sich nämlich versteckte Hinweise zur Lösung des Spiels entnehmen.

Fazit: Zak McKracken ist ein originelles und humorvolles Adventure, das durch einfache Handhabung in der Lage ist, auch Liebhaber von Actionspielen in seinen Bann zu ziehen.

Sound gibt es bis auf ein paar Geräusche keinen. Aber das ist nicht weiter tragisch. Bei diesem Spiel liegt der Schwerpunkt nicht auf Spezialeffekten, sondern mehr auf Handlung, der tollen Idee und der lang anhaltenden Spielmotivation. Auch wenn "Zak Mc Kracken" leicht zu bedienen ist, leicht zu lösen ist es sich nicht

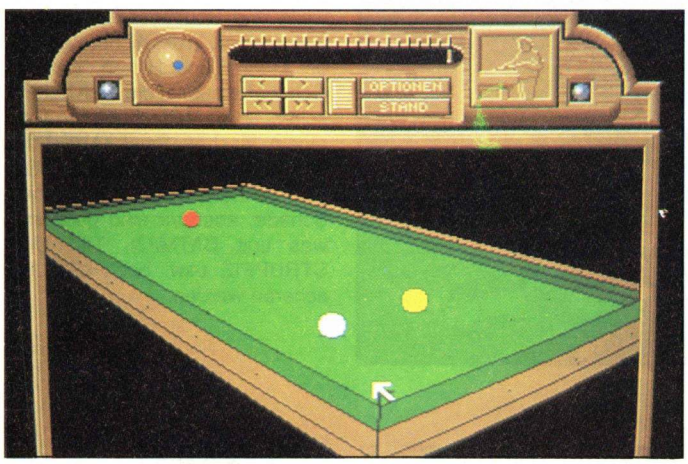
CBO

Hersteller:
Lucasfilms/ Softgold
Info: Rushware



Billard Simulator

Um Mißverständnissen vorzubeugen: Es handelt sich bei *"Billard Simulator"* von "Ere International" nicht um eine Simulation des Pool Billard-Spiels, wie es auch in den meisten Kneipen anzutreffen ist.



Der Billardtisch kann wahlweise in 3 oder 2D betrachtet werden.

Nein, Ziel von Eres Billardvariante ist es, mit einer Kugel und einem Stoß zwei andere Kugeln zu treffen. Zu Beginn des Spiels hat der Spieler die Möglichkeit, das Game seinen Wünschen anzupassen: Es stehen drei Queuetypen zur Wahl, 1 oder 2 Spieler können sich eine Billardpartie liefern, 2D- oder 3D-Darstellung sowie unterschiedliche Schwerkräfte sind ebenfalls vor Beginn des Spiels einstellbar. Mit einer anderen Schwerkraft als der auf der Erde Billard zu spielen, halte ich für unsinnig, aber wem's gefällt? Bei der 2D-Darstellung wird der Tisch aus der Vogelperspektive gezeigt, während der Computer den Billardtisch bei der 3D-Darstellung räumlich zeigt. Es ist übrigens möglich, den Tisch während des Spiels zu drehen. Sogar Bildausschnitt und Blickwinkel sind variabel. Am oberen Bildrand werden die Stoßkraft und der Ansatzpunkt an der Kugel bestimmt. Klickt man die Kugel an, erscheint eine Linie, die nun durch Mausbewegungen um die Kugel bewegt werden kann. Diese Linie stellt das Queue dar. Hat sich der Spieler die richtige Richtung und Geschwindigkeit ausgesucht, klickt er ein großes Icon an, und die Kugel setzt sich in Bewegung. Auf diese Art und Weise müssen die beiden anderen Kugeln berührt werden. Mit steigendem Schwierigkeitsgrad ist das gar nicht so leicht.

Fazit: Alles in allem ist "Billard Simulator" eine gelungene Simulation. Der Tisch dreht sich schnell, die Kugeln rollen den physikalischen Grundsätzen entsprechend, und die Tatsache, daß man alle erdenklichen Effekte und Eigenschaften einstellen kann, zeugt von gekonnter Programmierung. Sound kommt bis auf ein Klick und Klack nur in der Titelmelodie vor (Klaviermusik!).

Für Billardfans, denen ein Tisch zu teuer ist, bietet das Game eine Alternative. Als günstige Alternative läßt sich Billard Simulator allerdings nicht bezeichnen. 80 DM ist eine Menge Holz!

CBO

Info: Bomico



Hollywood Poker Pro

Nackte Tatsachen!

Ich war sehr skeptisch, als mir Holger und Uwe von Reline Software ihr neues Strip Poker-Spiel ankündigten: Diese Spielidee hat doch mittlerweile einen langen Bart, und alle Strip Pokers, die bisher auf den Markt kamen, verfügten über eine schlampig digitalisierte Grafik und einen schwachen Sound (man denke da nur an "Strip Poker II Plus" von Anco.)

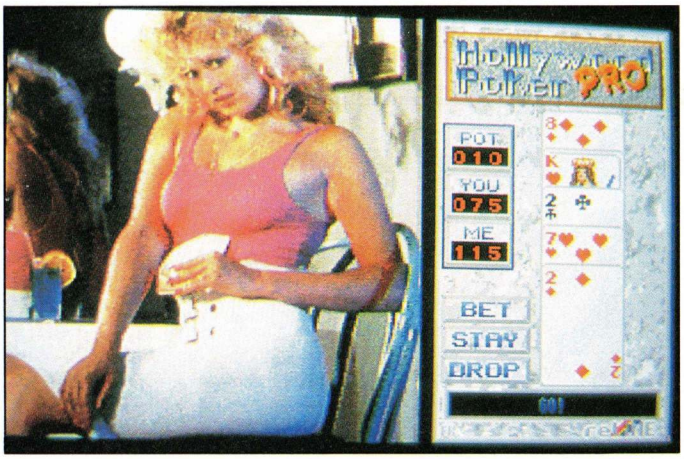
Doch meine anfängliche Skepsis verging wie im Fluge, als ich das Spiel geladen hatte. Im Vorspann zeigt der AMIGA die schemenhafte Darstellung eines tanzenden Mädchens. Durch die Verwendung von 150 digitalisierten Bildern wird eine sehr realistische Animation erreicht - eine echte Entzückung fürs Auge! Doch damit nicht genug: Die Bilder der Pokerpartnerinnen sind im HAM-Modus dargestellt, was den Grafiken die Qualität von gestochen scharfen(!) Farbfotografien verleiht.

Soundliebhaber kommen ebenfalls auf ihre Kosten: Die Effekte und die Sounds, die während der Pokerpartie erklingen, stammen vom Computersoundmagier Chris Hülsbeck, der, wie sich hier zeigt, nicht nur den Soundchip des C64 in- und auswendig kennt. In puncto Grafik und Sound ist Hollywood Poker Pro also wesentlich besser gelungen als vergleichbare Programme. Wie sieht es mit der Spielbarkeit aus?

Anstelle von zwei Mädchen wie bei Strip Poker II Plus hat der Spieler die Wahl zwischen vier Pokerpartnerinnen. Das Spielprinzip hat sich natürlich nicht geändert. Der AMIGA zeigt immer noch eine Grafik eines hübschen Mädchens, das sich nach und nach auszieht, wenn es sein Geld bei der Pokerpartie gegen den Spieler verloren hat. Im Gegensatz zu den bisherigen Strip Poker-Spielen verfügen die Mädchen über eine hohe Spielstärke. Die Strategie der Hollywood Poker Pro Damen ist nicht so leicht zu durchschauen wie die der Strip Poker II Plus-Mädels. War es mir bei Ancos Spiel möglich, durch Bluffs innerhalb von 15 Minuten ein Mädchen völlig auszuziehen, so brauchte ich für ein Mädchen bei Hollywood Poker Pro etwa zwei Stunden. Fazit: Mit Hollywood Poker Pro liegt mit Sicherheit das beste Strip Poker-Spiel für den AMIGA vor. In den Bereichen Grafik, Sound und Schwierigkeitsgrad ist es wesentlich überzeugender als vergleichbare Produkte. Zudem besitzt das Reline Spiel wenigstens ein gewisses Maß an Eigenständigkeit.

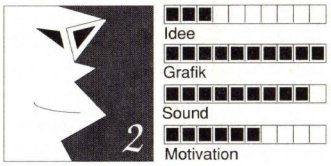
Hollywood Poker Pro ist das erste Strip Poker-Spiel mit einer Zoomfunktion, mit deren Hilfe sich einzelne Bildausschnitte vergrößern lassen (Wozu denn das?, die Red). So haben Voyeure die Möglichkeit, bestimmte Körperteile in doppelter Größe anzustarren (Ach so, die Red.). Lang anhaltende Motivation kann leider auch "Hollywood Poker Pro" nicht bieten. Hat man einmal alle Mädchen besiegt, wird das Spiel langweilig. Aber Fans dieses Spielgenres wird das vermutlich nicht abschrecken.

CBO



Die Mädels werden im HAM-Modus dargestellt, mit einer Zoomoption können bestimmte Körperteile sogar vergrößert werden.

Hersteller: Reline Software



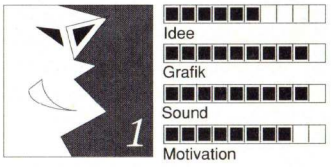
International Karate Plus



Vor einem Sonnenuntergang gilt es, den schwarzen Gürtel zu erringen. Doch die Kampfrichter sind streng.

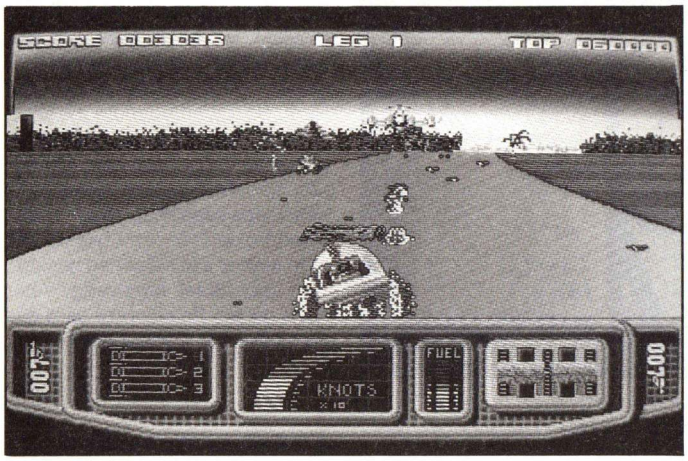
Wow, endlich einmal eine dem AMIGA angemessene Umsetzung eines guten alten 64er-Klassikers. Wie beim Vorbild versuchen sich immer drei Kämpfer gleichzeitig daran, den schwarzen Gürtel zu erringen. Diese sind wirklich überdurchschnittlich gut animiert, es macht schon allein das Zusehen Spaß. Bei diesem Spiel lohnt sich das genauere Betrachten des Hintergrunds sowieso, da man dort - vor der romantischen Kulisse eines Sonnenuntergangs - immer wieder überraschende Details entdecken kann. So irrt manchmal ein PacMan durchs Bild, ein U-Boot taucht auf, oder ein Wurm kriecht vorbei. Doch zurück zum Spiel. Hat man seine Gegner ein paarmal besiegt, kommt man in verschiedene Bonusrunden, in denen man Punkte hamstern kann, sofern man sie beherrscht. In der ersten muß man mit einem Schild heranhüpfende Bälle abwehren, in der anderen soll man Bomben aus dem Bildschirm heraustreten, bevor diese explodieren, oder deren Splittern ausweichen. Diese Bonusrunden sind nach einiger Zeit (wenn man allein spielt) ziemlich ermüdend und werden schnell langweilig. Der Spieler wird jedoch durch die gut gelungene Musik und die perfekt digitalisierten Kampfschreie immer wieder zu einem neuen Spiel motiviert. Farbenfrohe Grafik, viele Details (die Schatten der drei Kämpfer bewegen sich relativ zur untergehenden Sonne) sowie ein paar Gags (Karatekämpfer ohne Hosen!?) runden das an sich schon ansprechende Spielprinzip ab. Wie steht es doch so richtig auf der Packung: "International Karate - The Ultimate Martial Arts Simulation".

SV



Leben und sterben lassen

James Bond 007, die absolute Doppelnull, hüpft mal wieder von der Leinwand auf den Bildschirm. Nach den mißlungenen Bondspielen *"A view to a kill"* und *"Der Hauch des Todes"* schmeißt DOMARK mit *"Leben und sterben lassen"* (diesmal in einer Gemeinschaftsproduktion mit ELITE) schon das dritte Bond-Game auf den Markt. Die Hintergrundgeschichte ist zwar ebenso hintergründig wie die Handlung in den Filmen, aber darüber können wir ja hinwegsehen.



007, bekannt auch unter dem Namen James Bond, hat es diesmal auf einen skrupellosen Gangsterboß abgesehen.

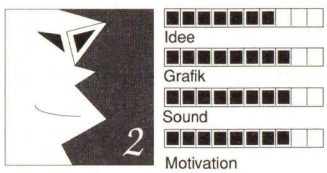
Dr. Kananga, ein rücksichtsloser Premierminister einer karibischen Insel, besitzt riesige Mohnfelder. Tonnenweise bringt er das daraus gewonnene Heroin auf den amerikanischen Markt. Das Schlimme ist, er läßt es dort kostenlos verteilen, damit möglichst viele Leute davon abhängig werden und dann für teures Geld Heroin bei ihm kaufen müssen. Mr. Big, wie der niederträchtige Minister in Unterweltkreisen auch genannt wird, verschifft das Heroin von einem geheimen Hafen einer Karibikinsel.

Klar, daß Bond nicht mit dem Treiben des windigen Geschäftemachers einverstanden sein kann. Er wird von seinem Boß "M" ins Feld geschickt, um dem bösen Treiben ein Ende zu bereiten. Dazu muß er mit einem Schnellboot durch eine Wasserstraße brausen und mit der Bordkanone auf die Verteidigungsanlagen des Heroinherstellers ballern. Dieser hält sich eine Privatarmee, um alle Eindringlinge abzuschrecken. So greifen Bond bei seiner rasanten Fahrt Flugzeuge und andere Schnellboote an. Vorsicht, im Wasser liegen explosive Fässer, Minen und natürliche Hindernisse wie Felsen und Eisblöcke.

Die Hindernisse, die der Feind in den Weg stellt, sind aber nicht Bonds einzige Sorge. Sein Motorboot frißt Sprit wie ein Esel Hafer. Deshalb muß er darauf achten, im Wasser herumschwimmende Benzinfässer einzusammeln. Auf einer Anzeige im Cockpit sieht man, wie es um den Tankinhalt bestellt ist. Wenn das Rennboot vollgetankt ist, kann man wieder Vollgas geben. Dazu drückt man den Joystick nach vorne, und die Kiste düst los. In superschneller 3D-Grafik rast die Landschaft auf den Spieler zu. Eine solche rasante und flüssige Grafik habe ich auf dem AMIGA selten gesehen. Das Durchfahren von Tunneln und Schleusen macht die Motorbootfahrt sehr realistisch und vor allem schwierig. Damit sich der Spieler an sein Motorboot und die Verteidigungsanlagen gewöhnen kann, hat Domark an drei Trainingsmissionen gedacht. In der ersten Mission düst man über die Wasserstraße und schießt auf Zielscheiben, während man sich in der zweiten und dritten Episode mit den Verteidigungsanlagen vertraut machen kann. In der eigentlichen Mission muß man die Wasserstraße so weit wie möglich durchfahren, um am Ende des Flusses auf die heroinverarbeitenden Fabriken von Mr. Big zu stoßen, die Sie mit einem gezielten Raketenschuß zerstören müssen. Fazit: "Leben und sterben lassen" ist ein gutes Game mit hervorragender Grafik, einem fetzigen Titelsound und einem süchtig machenden Spielprinzip.

CBO

Info: Bomico



Schiifoahn

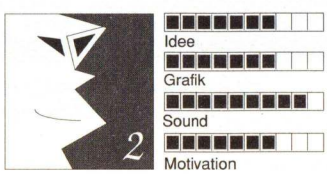


Die Skipisten sind mit allerlei Hindernissen bestückt, die gekonnt umkurvt werden müssen.

Nicht überall, wo Simulator draufsteht, ist auch Simulator drin!. Das ist die Weisheit, zu der man gelangt, wenn man *"Advanced Ski Simulator"* geladen hat. Bei Codemasters neuestem Spiel handelt es sich nämlich nicht um eine Simulation, sondern um ein Actionspiel. Ein oder zwei Spieler brausen über sieben verschiedene Skipisten im Slalom herunter, krachen gegen Hindernisse, springen über Schanzen und versuchen, die Strecke in der vom Computer vorgegebenen Zeit zu meistern. Da die Landschaften räumlich dargestellt und reichlich mit Bäumen, Schneehügeln und Steinen bestückt sind, wird von den Spielern Fingerspitzengefühl verlangt, damit die Athleten sicher ans Ziel kommen. Hat man einen Kurs durchfahren, lädt der AMIGA den nächsten, und die Action geht wieder von vorne los. Natürlich mit einer anderen Strecke und anderen Hindernissen.

Alleine gegen einen computergesteuerten Skifahrer zu spielen, bereitet eine Menge Freude. Mehr Spaß macht das Spiel aber, wenn zwei Skifreaks simultan spielen. Da wird gedrängelt und geschubst, was die Bretter halten. Wenn einer der beiden Spieler seinen Athleten aus dem vertikal scrollenden Bild fährt, kann er seinen Skifahrer über einen Radarschirm steuern. Fazit: Für mich ist Advanced Ski Simulator eine positive Überraschung. Ein solches Spiel hätte ich den Jungs von Codemasters gar nicht zugetraut. Boten Sie uns doch bisher nur "Billigspiele". Doch Advanced Ski Simulator kann sich sehen lassen. Ansehnliche Hintergrundgrafiken, ein toller Titelsound und sieben aufreibende Slalompisten- was will man mehr? Ach ja, da war noch was! Die Animation der Skifahrer ist nicht flüssig genug. Sieht man über diesen Schönheitsfehler hinweg, ist Advanced Ski Simulator ein nettes Actionspiel.

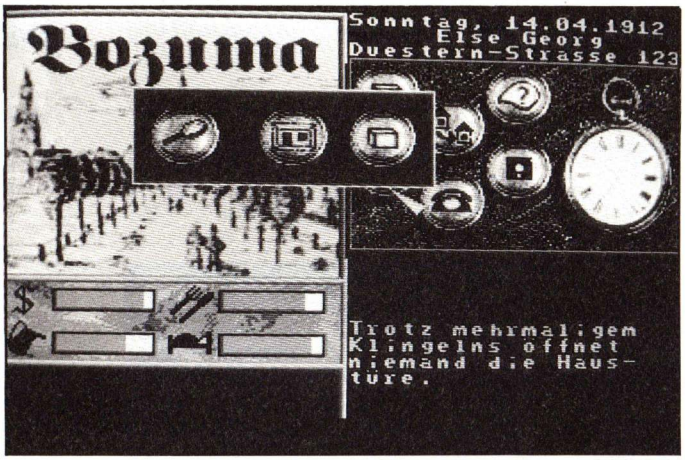
CBO



"Bozuma"

Kriminalroman in Schwarzweiß

Detektive leben gefährlich. Sie werden in Schießereien verwickelt und müssen andauernd um ihr Leben fürchten. Wer sich nicht ständig in Lebensgefahr begeben und trotzdem Detektivarbeit leisten möchte, sollte sich einmal *"Bozuma"* ansehen.



Das neue Kriminaladventure des Bremerhavener Softwarehauses versetzt Sie ins Jahr 1912. Ort des Szenarios ist Hamburg. Auf einem Zimmer im Interhotel befinden sich ein cleverer Anwalt, der auch schon beim Lösen einiger Kriminalfälle sehr erfolgreich war, und sein getreuer Butler Wilhelm. Plötzlich klingelt das Telefon. Der Portier kündigt einen Herrn Rabenberg an, der dringend den cleveren Anwalt sprechen möchte. Wenig später klopft es an der Tür, und ein beleibter Mann um die Fünfzig tritt ins Zimmer. Er gibt sich als Herr Rabenberg zu erkennen und beginnt sogleich, sein Anliegen vorzutragen: Vor etwa 30 Jahren sei Rabenbergs Vater zu einer Kolonisationsexpedition nach Kamerun aufgebrochen. Auf einem Erkundungsgang in einem Höhlengrab stieß er auf eine Mumie, die er mitnahm, um sie in einem Museum in Deutschland der Öffentlichkeit zugänglich zu machen. Die Eingeborenen weigerten sich jedoch, die Mumie zu tragen, da sie angeblich verflucht sei. Tatsächlich, wenige Tage später erkrankten die Träger an Malaria. Als man die Mumie nach Deutschland verschiffte und in ein Museum brachte, wurde sie in einer Nacht- und Nebelaktion von Kunstdieben gestohlen. Seitdem ist sie spurlos verschwunden. Rabensberg beauftragt Sie und Ihren getreuen Butler, die Mumie wiederzufinden, da sie für ihn ein schönes Erinnerungsstück an seinen Vater darstellt.

Mit dieser Hintergrundgeschichte wird der Spieler in die Handlung eingeführt. Dann beginnt das Detektivabenteuer. Keine Panik! Sie brauchen nicht wie ein Wilder auf der Tastatur herumzuhämmern, um den Helden zu bestimmten Handlungen zu bewegen. Es reicht, ein Icon anzuwählen, und schon macht der Protagonist, was man will. Man kann den Anwalt mit einem Auto an jeden Ort der Stadt fahren lassen. Dazu braucht man nur die Hausnummer des Zielortes einzugeben. Der Zeiger der Uhr am rechten Bildschirmrand rutscht einige Minuten nach vorne, und schon ist man am gewünschten Ort und kann Nachforschungen anstellen. Mittels Symbolsteuerung können Sie den Helden dazu veranlassen, Gegenstände zu untersuchen oder aufzusammeln. Natürlich sind auch Verhöre möglich. Mit der Wahl des entsprechenden Symbols kein Problem! Wichtige Aussagen können sogar in einem Notizblock festgehalten werden. Sie müssen Ihren ganzen Scharfsinn und Ihre Beobachtungsgabe einsetzen, um die Mumie zu finden. Während der Mumiensuche werden Sie übrigens in ein anderes Verbrechen verwickelt, aber schauen Sie selbst.

Neben der Adventurehandlung sind noch einige Extras im Programm integriert. So gibt es eine Actionszene, in der Sie in der Alster nach Beweisstücken tauchen müssen. In einem Labor haben Sie ebenfalls zu tun. Blut- und Giftnachweise sind dort zu erbringen. Während des Spiels können Sie sogar an kulturellen Genüssen wie einem klassischen Konzert in der Staatsoper teilnehmen oder einem Vortrag über Mumien lauschen. Für Observationsarbeiten besteht die Möglichkeit, einen anderen Detektiv zu engagieren, der Ihnen lästige Routinearbeiten abnimmt.

Man sieht, "Bozuma" verfügt über viele Features, die Fans von Detektivspielen begeistern dürften. Raubkopierer werden keinen Spaß an "Bozuma" finden. Ihnen fehlen nämlich wichtige Adressen und Telefonnummern, die sich in einem extra beigelegtem Telefonbuch befinden. Ohne dieses Buch, in dem sämtliche Orte vermerkt sind, die man bei der Fahrt mit dem Auto angeben muß, ist das Spiel völlig wertlos. Über die Qualität der Grafiken kann man streiten. Einerseits sind sie teilweise sehr detailliert, andererseits dafür nur in Schwarzweiß. "Bozuma" wird nicht so schnell langweilig. Es gibt viele Rätsel zu entdecken und zu lösen, auch wenn in der Anleitung die Angabe mit 186 verschiedenen Orten und Personen ein bißchen übertrieben ist. Viele Häuser haben für den Spielverlauf überhaupt keine Bedeutung.

CBO

Hersteller: Time Warp Software

Info: Rushware



INSERENTEN-VERZEICHNIS

	Seite
ABC-Soft	51
ALCOMP	131
AMIGAOBERLAND	35
A.P.S.	24
Ait-Rönn	24
CIK-COMPUTERTECHNIK	51
CWTG	15
CSS	15
CREATIVE VIDEO	24
COMBITEC	9
COMBIMATE	15
COMPUTER MÜTHING	75
DFÜ-SHOP	61
DTM	2
DONAU-SOFT	75
DREWS	75
FSE	89
FIRST-PUBLIC-SHOP	61
FISCHER	24
HEIM	19, 30, 71
HARMS	61
IDEE-SOFT	15
IM	48
IRSEE-SOFT	89
KUPKE	132
MAXON	13, 45, 68, 81, 98, 107
OSSOWSKI	48
PRINT TECHNIK	51
PBC	51
POWERSOFT	75
PUBLIC-DOMAIN-CENTER	61
RAINBOW DATA	39
RHEIN-MAIN-SOFT	24
SOFTWARE 2000	23
STALTER COMPUTER	64
TRÖPS + HIERL	89
ÜPC	75
VIDEO-LOFT	24

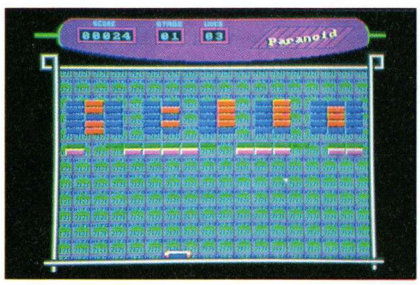
KICKSTART PU

Hallo PD-Fans,

diesmal liegt der Schwerpunkt der neuen PD-Disketten bei den Anwenderprogrammen - und was dabei geboten wird, kann sich sehen lassen. Sehr komfortabel und mit vielen Extras sind das Musikprogramm WIZARD OF SOUND, die Girokontenverwaltung GIROMAN und die Schießsportverwaltung NOSCOM. Weiterhin gibt es eine gute Textverarbeitung und ein Malprogramm. Zu den meisten Programmen wird auch der Quellcode mitgeliefert, so daß man gegebenenfalls Verbesserungen und Erweiterungen an den Programmen vornehmen kann. Dies ist ein Service, den man den Autoren nicht hoch genug anrechnen kann, denn bei der Anwendung hat man fast immer einen Änderungswunsch oder man will nur einzelne Routinen anschauen, um zu sehen, wie sie realisiert wurden.

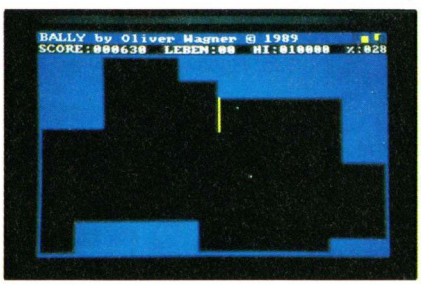
DIE NEUHEITEN

KICK 141: SPIELE



PARANOID: eine Breakout-Variante mit High-Score-Liste und Level-Editor. Mit dem Leveleditor kann man sich auf einfache Weise beliebige Levels gestalten. Auf der Diskette sind bereits 24 Levels vorhanden. Mit gutem Sound. Autor: Felix Wente

KICK 142: SPIELE

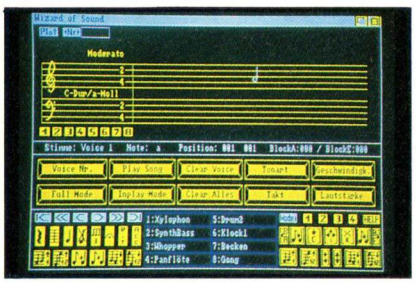


BALLY: Das Spielprinzip ist sicherlich bekannt - 75% der Fläche müssen umrandet und damit ausgefüllt werden, nur darf man sich nicht von den herumschwirrenden Punkten erwischen lassen. Sehr schnell, Joystick erforderlich! Autor: Oliver Wagner

GAMBLER: Ein allgemeines Spielsystem, das Strategiespiele wie Dame, Vier gewinnt, Tic-Tac-Toe u.ä. spielen kann. Nachdem im interaktiven Dialog die Regeln eingegeben wurden, kann man mit dem Computer das Spiel aufnehmen. Dieser lernt sogar während des Spiels dazu. Mit diesem Programm errang Guido Appenzeller einen Preis bei der Regionalausscheidung von "Jugend forscht" 1988/89.

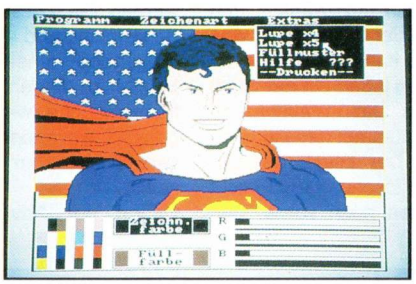
KICK 143: MUSIK

WIZARD OF SOUND: ein Musikprogramm zum Erstellen von Musikstücken nach Notenvorlage oder zumKomponieren. Einige Funktionen im Überblick: bis



zu 8 Instrumente gleichzeitig, 4 Stimmen, verarbeitet gesamplete Instrumente im IFF-Format (8SVX / 5 Oktaven), Inpay-Modus (Spielen mit Begleitmusik), Spezialfunktionen (Crescendo, Accelerando, Ritardando, Fermate, Triole usw.), autom. Taktreorganisation, Ausdruck der Notation, Player-Programm (läuft im Hintergrund), multitaskingfähig, komplette Maussteuerung, Help-Funktion, deutschsprachige Anleitung. Autor: Jörg Schließer

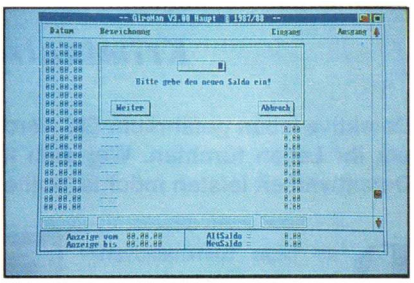
KICK 144: GRAFIK



AMIGAPAINT V1.5: ein deutsches Malprogramm mit vielen Optionen (Freihand, Punkte, Linie, Rechteck, Block, Fläche, Kreis, Füllen, Radiergummi, Text, Spraydose, Füllmuster und Lupe). Es arbeitet im Lores-Modus mit 320x200x5 und 32 Farben. Auf der Diskette ist das Listing in AmigaBASIC (etwa 1240 Programmzeilen) enthalten, so das man leicht Änderungen und Erweiterungen an dem Programm vornehmen kann. Somit kann man sich nach und nach auf dem Gerüst dieses Programms ein leistungsfähiges Malprogramm zusammenbauen, daß ganz individuelle Funktionen enthalten kann.

KICK 145: ANWENDUNGEN

GIROMAN V3.00: Ein sehr leistungsfähiges Girokondtoverwaltungsprogramm, das vom Autor schon seit 1980 auf einem PET eingesetzt wird. Das Programm ist sehr komfortabel und einfach zu bedienen. Zu den wich-



tigsten Funktionen zählen: Einträge suchen, Kontostand eines betimmten Tages abfragen, monatliche Festposten (Daueraufträge usw.) erledigen, Ausdruck mit Saldoangabe und manuelle Buchungen (hinzufügen, ändern, löschen). Mit Anleitung, komplett in Deutsch, Source in Aztec C. Autor: Sönke Netzband

KICK 146: ANWENDUNGEN

NOSCOM-MATCH 1.1: Hierbei handelt es sich um ein Auswertungsprogramm für Wettkämpfe des offiziellen nationalen Schießsports. Es ist sehr komfortabel und fast gänzlich mit der Maus zu bedienen. Es erleichtert Darstellung, Auswertung und Ausdruck der Ergebnislisten. Eine Version 1.2 ist bereits in Vorbereitung und wird vom Autor angeboten. Komplett in Deutsch, mit ausführlicher Dokumentation; benötigt 1MB RAM. Autor: Norbert Süßdorf

KICK 147: ANWENDUNGEN

M.S.TEXT: Eine leistungsfähige Textverarbeitung mit den wichtigsten Funktionen. Einfache Bedienung über Menüs oder Tastensequenzen, Blockfunktionen (markieren, kopieren, löschen, laden, speichern, drucken), eine Dateiauswahlbox und die schnelle Scrollgeschwindigkeit zeichnen sind die wichtigsten Merkmale. Quellcode in AZTEC C, komplett in Deutsch. Autor: Martin Steppler, Fulda

KICK 148: MODULA (AMOK 13)

COMPCONTROL: erleichtert die Bedienung von Editor, Compiler und Linker. **ROWS**: generischer Datentyp, während der Laufzeit dimensionierbare Arrays. **WITH**: ein neuer CLI-Befehl. **FLIPMOUSE**: invertiert die Richtung der Mausbewegung. **WBSHADOW**: Schatten für die Workbench. **WBPIC**: bringt ein Bild auf die Workbench.

KICK 149: MODULA (AMOK 14)

ARP: M2AMIGA-Implementation der arp.library, die das BCPL-DOS durch leistungsfähigere Befehle ersetzt. **TIMERSUPPORT**: erleichtert die Benutzung des Timer-Deviced

KICK 150: UTILITIES

TASKDRIVE: ermöglicht ein bequemes Ansehen und Ändern der Abarbeitungsprioritäten jedes Prozesses/Tasks im System. Eigener Screen, Source in Lattice C. Autor: Rudolf Gerlinger (Österreich)

IMAGCALC: ein Taschenrechner, der mit komplexen Zahlen arbeitet und deshalb z.B. die Wurzel einer negativen Zahl in Imaginär- und Realteil zerlegt. Autor: Peter Schaefer

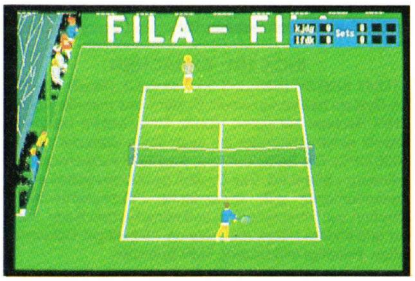
CODE: ein Chiffrier- und Dechiffrier-Programm mit Paßwortschutz. Autor: K.J. Wolf

SHC (Screen-Hard-Copy): Ein sehr schnelles (Assembler) Druckprogramm für eine Bitplane eines Screens. Für STAR LC-10, NL-10, EPSON LX-800 und Kompatible. Mit Source. Autor: Karsten Kalweit

BLIC DOMAIN

KICK 131-140

KICK 131: SPIELE

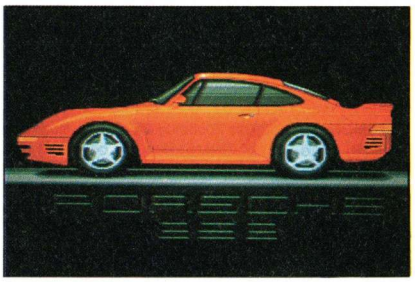


TENNIS: Tennisspiel für zwei Spieler mit Trainingsmodus, Grund-linien- und Netzspiel. Die Schlagstärke und Richtung können mit dem Joystick bestimmt werden. Erfordert 1MB Speicher. *Autor: Christian Dreke.*

KICK 132: SOUNDS

TGB SOUND DISK NO.1: Fünf Musikstücke (erstellt mit Soundtracker) werden vor dem Hintergrund von HAM-Bildern abgespielt. *Komponist: Thomas-Georg Brüggemann.*

KICK 133: GRAFIK-SHOW



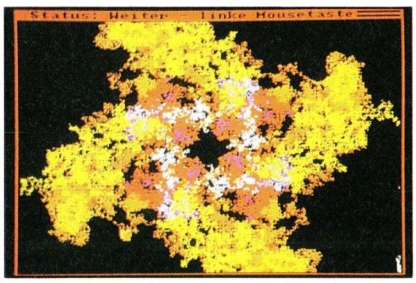
CRAZY-MOTOR-SHOW: gut gezeichnete Sportwagen und eigene Konstruktionen mit Angabe der technische Daten und interessanten Hintergrundinformationen. Mit digitalisierter Musik. 1MB Speicher erforderlich. *Autofan: I.K.*

KICK 134: BILDER



Viele gute Grafiken (SexyRobot, Roland SH-101, Sailboat, Max, Ferrari GTO, Dali, ST), Cartoons (Batman, Garfield) und digitalisierte Bilder (Supercollage von Marilyn, Mona Lisa, Alien).

KICK 135: GRAFIK



MANDEL 2.1: Mandelbrotgenerator für repititive Muster. Source in AmigaBASIC. *Autor: Michael M.*

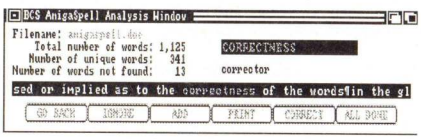
Machen Sie mit!

Möchten auch Sie selbstgeschriebene Programme der Allgemeinheit zur Verfügung stellen, so schicken Sie sie uns einfach zu (bitte mit Dokumentation auf Diskette!).

Als kleine Anerkennung können Sie sich dafür 5 Disketten aus unserem PD-Service auswählen.

MAXON-Computer GmbH
KICKSTART-Redaktion
PD-Einsendung
Industriestraße 26
6236 Eschborn

KICK 136: ANWENDUNGEN



AMIGASPELL: Wörterbuch (Wortschatz: 9085 englische Wörter) zum Korrigieren von (ASCII)Texten. Falsche bzw. nicht vorhandene Wörter können ausgedruckt, im Text korrigiert (der Satzzusammenhang wird angezeigt!) oder übernommen werden. Schnell (Assembler) und speichereffektiv. Es kann leicht (wenn auch mit Arbeit) z.B. ein deutsches oder hessisches Wörterbuch erstellt werden.

NAG: der absolute, blinkende, schreiende, singende Terminkalender, damit Sie nie wieder einen Geburtstag, eine Fete oder den täglichen Feierabend vergessen. Sehr komfortable Eingabe, viele Möglichkeiten, immer griffbereit.

HP11: Emulation eines HP11C-Calculators mit allen Finessen.

KICK 137: SPRACHEN

Für alle Programmiersprachenbegeisterten ist hier eine kleine Fundgrube. **SCHEME**: ein Dialekt ähnlich Lisp, **PILOT**: eine Umsetzung dieser Sprache für den AMIGA, **ADA:** Syntaxkontrolle für ADA, mit YACC-Source, **PERL**: Interpreter

KICK 138: UTILITIES

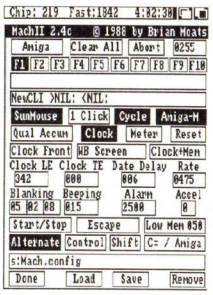
Auf dieser und der KICKPD 139 befinden sich die Sourcecodes der meisten Programme von Matt Dillion, der sicherlich jedem PD-Anwender bekannt ist. Im einzelnen sind dies: CONFIG V1.00, CLOCK V1.00, DME V1.31, DMOUSE V1.10, BACKUP V2.01, SUPLIB, LIBREF, DRES V1, DASM V2.11, FTOHEX, FILES V1.2, SHELL V2.10, FINDIT V1.00, LIBS V1.00, SCAT, ADDCR, REMCR, CMP.

KICK 139: UTILITIES

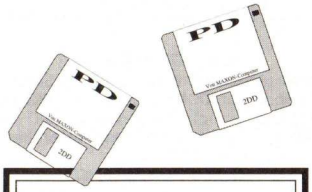
Hier sind die restlichen Sources der Programmsammlung von Matt Dillion (siehe KICK 138).

KICK 140: UTILITIES

RAMCOPY: Kopierprogramm für ein Laufwerk und mind. 1MB Speicher
DISKSALV: rettet gelöschte Dateien
SPUDCLOCK: Uhrzeitansage



MACHII: Maustool, Hotkeys, Popcli usw.
MULTICALC: RPN-Taschenrechner
MRPRINT: Druckprogramm für Dateien, mit vielen Optionen
AFTTERM: Terminalprogramm
MFIX: patcht das Kopierprogramm MARAUDER und steigert damit die Geschwindigkeit erheblich



ANRUF GENÜGT
06196 / 481811

KICK 130-94

KICK 130: ANWENDUNGEN

SUPERLIGA: Fußballergebnisverwaltung mit 577 Mannschaften aus 23 Ländern und 34 Ligen. Über 10000 Ergebnisse können gespeichert werden. Das Eingeben gestaltet sich sehr einfach. Aus den bisherigen Ergebnissen kann eine Prognose (Regressinsanalyse) für kommende Spiel und die Meisterschaft erstellt werden. Autor: Rolf Morlock

KICK 129: SOUNDS

Neue Kompositionen von Andreas Starr aus Gronau, erstellt mit Sonix. Effektvolle Stereostücke mit Player.

KICK 128: SOUNDS

Scores, Drumkits und Soundfiles für Dynamic Drums von New Wave Software.

KICK 127: GRAFIK

Mandelbrot-Show II: dies ist der zweite Teil der Reise durch die Welt der Mandelbrotgrafiken. Autor: R. Beck

KICK 126: GRAFIK

Mandelbrot-Show I: eine sehr informative Reise durch die Welt der Mandelbrotgrafiken. Jedes Bild wird ausführlich erläutert und das jeweils darauffolgende ist eine Vergrößerung eines Ausschnitts davon. Mit Mandelbrotgenerator MSE. Autor: R. Beck

KICK 125: GRAFIK

SLIDESHOW: Komfortables Diashow-Programm für DPaint-Bilder (320x200). Alle Bilder des aktuellen Direktories werden nacheinander auf Mausklick angezeigt. **DARK**: Copper-Demo, die viele Elemente eines Ballerspiels enthält. C-Source. **GLOBE**: Eine kleine drehende Weltkugel. PopUp-Menu, Source in C. **ICONS**: verschiedenste, animierte Icons (Disk, Drawer, Tool, Project, Garbage). Sehr originell und lustig!

KICK 124: ANWENDUNG

STARCHART: umfangreiches und interessantes Sternenprogramm mit unendlich vielen Einstellungen. Mit der Maus können Informationen über die Sterne abgerufen werden. **AIRFOIL**: ein einfacher Windkanalsimulator, verschiedene Einstellungen.

KICK 123: UTILITIES

Neue Utilities, die aus den aktuellen Einsendungen zusammengestellt werden. Hier gibt es immer ein riesiges Angebot, denn der AMIGA läßt viele Möglichkeiten für Hilfsprogramme.

KICK 122: MODULA (AMOK 12)

Auch auf dieser AMOK-Diskette gibt es wieder jede Menge Routinen und Programme: Modul zum Ansteuern des Fischertechnik-Interfaces, ein schnelles und sicherers HD-Backup, Disassembler für M2Amiga-Object-Files, ArraySort, Janus, Timer, StripText, ConvLibSym, MakeInLine, Diff, Tips zum M2-Compiler

KICK 121: MODULA (AMOK 11)

R.o.M.: ein komfortables Mathematikprogramm und Funktionsplotter.**TrackDiskSupport**: erleichtert Programmierung der TrackDisk. **M2Druka**: Druckutility für ASCII-Dateien. **Beep**: erzeugt akustisches Warnsignal.

KICK 120: UTILITIES

PCopy, HexCalc, Dux5 (CLI-Utility), SDBackup, FlamKey (Computerschloß), VirusX, SCT (Farbeinstellung von WB), Surveyor, HPMan (totale Kontrolle über den Laser)

KICK 119: AMOK #10

FlightDemo u. 3D-Demo (Echtzeitdemonstrationen bewegter räumlicher Grafik - super), FileRequester, SuperLists1.3, Speech (dt. Sprachausgabe), Feigenbaum.

KICK 118: AMOK #9

GraphicSupport (Makros für Graphics-Primitives/Intuition), PrintIt2.0 (Hardcopyroutine für Epson), M2Test (Dokumentation über Fehler bei ARRAYs mit Index BOOLEAN), TextFont (Laden v. Zeichensätzen), Sounds und Graphics.

KICK 117: AMOK #8

MemSystem (Speicherverwaltung), PrinterSupport (unterstützt Printer.device), IDCMP (Hilfsroutinen), Blitter (direkte Blitterprogrammierung), SpaceDemo (3-D Demo), IFF8SVXLoad (Laden/Abspielen gesampelter Sounds).

KICK 116: AMOK #7

Reminder (Wecker für Programmierer), WarpText (schnelle Textausgaberoutine), MuchMore (Softsroll), M2Test (Hinweise auf Fehler in den Bibliotheksmodulen), SoundTask, AVL-Bäume, Listen, Queues, Stack.

KICK 115: SPIELE

CHEES-BOXES ist das allseits bekannte "Käsekästchen"-Spiel, das man jedoch nicht mit dem folgenden verwechseln sollte. 2-4 Spieler, mausgesteuert, AMIGABasic. Autor: J.D. Mallander. **VIER GEWINNT**: zwei Spieler, mausgeteuert, AMIGABasic. Autor: Björn Kessels. **COSMIC**: Strategiespiel. Source in C. Autor: Carl Edman. **SOLITAIRE** löst beliebige Stellungen des bekannten Spiels. Source in C. Autor: Thomas Lagally **GRID IT**: spannendes Rennen für zwei Spieler. Benötigt Joystick, Source in C. Autor: Jörg Tuttas **LABYRINTH II**: Textadventure in Stil der INFOCOM-Spiele. Englisch, C.

KICK 114: BILDER-SHOW

Eine Grafik-Show mit Mandelbrotbildern und Colorcycling. Dadurch wird die Wirkung der eindrucksvollen Bilder hervorgehoben. Autor: Hermann Dörries, Wildeshausen

KICK 113: SOUNDS

Auf dieser Diskette ist der zweite Teil der KICKSTART SOUND DEMO III mit den Stereo-MIXES: Return to Defender, Dark Castle Mix, Chicago-Tune, Disco-Mix und Crazy Tune. Wichtig: Die Diskette läuft nur mit KICK 112!

KICK 112: SOUNDS

Zum dritten Mal haben unsere Soundspezialisten TOB & TWP zugeschlagen. Das Ergebnis ist die KICKSTART SOUND DEMO III mit einer Menge hervorragender Sounds auf zwei Disketten (siehe KICK 113). Mit von der Partie sind in der Rubrik CHARTS: Relax, Jean Michelle Jarre, Every Breath You Take, Axel F., Push, Tell It To My Heart und Blue Monday. Alles HiFi-Stereo mit einer extralangen Spielzeit.

Aus Platzgründen können leider nicht mehr alle PD-Disketten aufgeführt werden. Sollten Sie keine ältere KICKSTART zur Hand haben, dann können Sie die Liste auch bei uns anfordern. Senden Sie dafür einen adressierten und frankierten Rückumschlag an folgende Adresse:

MAXON Computer GmbH
KICKSTART PD-Liste
Postfach 55 69
6236 Eschborn

KICK 111: BILDER

20 wunderschöne IFF-Grafiken von Frauenprofilen. Mit Diashow, sehenswert!

KICK 110: UTILITIES

MEMORYCLOCK: zeigt Uhrzeit, Datum, RAM- und Chip-Memory. **DATAMAKER, IFF-CONVERTER**: wandelt Dump-Format-Bilder in IFF-Format. Source zu allen Programmen in Assembler, Autor: Roger Fischlin.
SDBACKUP: Festplatten-Backupprogramm, das die komplette Platte in komprimierter Form auf Disketten abspeichert. Viele Optionen. **RUNBACK:** läßt Programme im Hintergrund laufen, damit sie keine CLI-Fenster benötigen.

KICK 109: SCHULE

VOK-BOY V1.0: Vokabeltrainer in AmigaBasic. Autor: Michael Hennemann
WORD: Fremdsprachenlernprogramm, das Begriffe und ganze Sätze abfragt. Mit umfangreichen Lektionen in Englisch und einigen in Französisch. Kann leicht erweitert werden. Verwaltet beliebige Sprachen, sehr komfortabel. Autor: Daniel Neukomm, Bern (Schweiz)

KICK 108: BILDER

Viele schöne Bilder zum Anschauen und Weiterbearbeiten.

KICK 107: ANIMATIONEN

Weitere Animationen von Tobias Richter: JET (Düsenjet-Landeanflug), RELIANT (ein Raumschiff im frontalen Anflug), NELSON (rotierendes Raumschiff). Die Animationen demonstrieren, wie eindrucksvoll die mit VideoScape erstellten Sequenzen sein können. Sehenswert!

KICK 106: ANIMATIONEN

Tobias Richter hat uns wieder einige seiner mit VideoScape 3D-Animationen geschickt: LOTUS (der bekannte Lotus in einer neuen Sequenz), WILKERSON, DURETT und RELIANT (Raumschiffe in schneller Rotation).

KICK 105: ANIMATIONEN

LLEWELLYN ist eine mit dem Sculpt 3D-Animator erstellte Filmsequenz mit musikalischer Untermalung. Eine schöne Demo dieses Programms von Michael Clinard.

KICK 104: ANWENDUNGEN

DIRECTORY MASTER V1.1: Dieses Programm bringt Ordnung in Ihre Diskettensammlung. Die Dateien jeder eingelegten Diskette werden automatisch in eine Liste übernommen. Diese kann dann auf komfortable und vielfältige Weise bearbeitet und verändert werden. Sehr viele Funktionen, einfache Bedienung, sehr leistungsfähig und schnell. Als Zugabe gibt es die jeweils aktuelle Liste des KICKSTART PUBLIC DOMAIN SERVICE.
MEMOPAD: ist ein echt nützliches Hilfsprogramm für vergeßliche AMIGA-Anwender. Hier können wichtige Termine und Daten festgehalten werden. Bindet man den Befehl CHECKMEMO in die Startup-Sequence ein, dann erinnert das Programm an diese Termine. Einfache Mausbedienung, sehr komfortabel und hilfreich.

KICK 103: GRAFIK

C-LIGHT: ein sehr einfach zu bedienendes, mausunterstütztes Raytracer-Programm. Die Einstellungen erfolgen über viele Regler und Schalter. **NEWJUGGLER**: ein Grafikprogramm, das den Eric Graham's JUGGLER innerhalb eines Bildes ablaufen läßt. Source in C; Autor: Peter Weiland, Wietze. **HAM'S**: einfaches Malprogramm für den HAM-Modus in AmigaBasic. Das Programm kann leicht geändert werden, da der Source-Code gut strukturiert ist. Autor: Henning Frommer, Vallendar

KICK 102: SPRACHEN

STONY BROOK PROLOG: auf dieser Diskette befinden sich die Source-Codes (Lattice C 4.0 und Prolog) des PD-Prolog-Systems von Diskette 101.

KICK 101: SPRACHEN

STONY BROOK PROLOG Version 2.3.2: Dieses PD-Prolog-System basiert auf einem System, das ursprünglich für UNIX-Rechner konzipiert wurde. Das System beinhaltet sowohl einen Interpreter als auch einen Compiler (bis zu 3000 LIPS). Der Sprachumfang entspricht weitgehend dem Edinburgh-Standard. Das System besitzt zusätzlich einige Besonderheiten z.B. dynamische Prädikateinbindung, Mischen von interpretiertem und compiliertem Kode, Makros und andere Hilfen. Inklusive ca. 100seitiger Dokumentation. Benötigt mindestens 1MB Speicher.

KICK 100: JUBEL-DISK

Auf dieser Diskette sind Programme zusammengefaßt, von den wir denken, daß sie für einen AMIGA-Besitzer unverzichtbar sind. Die Palette reicht von Kopierprogrammen, CLI-Hilfen bis zu Spielen. Lassen Sie sich überraschen - es wird sich lohnen!

KICK 99: SPIELE

BrainWork: Puzzle. *Autor: Timmy.* **Amiga-Wurm**: Tron-Variante; AmigaBasic. *Autor: Christoph.* **May Valley of the Aztec's**: ähnlich "Kaiser"; ABasic. *Autor: Matthias Hakuba*
Bauernskat: Source in AmigaBasic. *Autor: Oliver Peter*
Breakout: 99 Level mit Editor; Source in C. *Autor: Peter Händel.* **Kniffel**: bekanntes Würfelsspiel für 4 Personen oder gegen Computer; sehr schöne Grafik. Mausbedienung. *Autor: Michael Teistler.* **PACCIE**: Pacman-Variante in Basic.

KICK 98: SUPER-SOUNDS

KICKSTART SOUND DEMO II: sicherlich eine der besten Sounddemos, die es derzeit auf Disk gibt. 18 programmierte Musikstücke, aktiongeladener Sound, Stereo, sehr lange Spielzeit, kurze Ladezeit durch gepacktes Format, Sound-Slideshow. *Komponist: Tom Beuke*

KICK 97: UTILITY

TASKMON V0.4: zeigt die Zustände aller Tasks; Source in C. *Autor: P. Erpenbeck.* **FilterSwitch 2.0**: damit läßt sich der Tiefpaßfilter per Alt-Taste an- und ausschalten. **DMASwitch 2.0**: Bildschirmabschalter; Source in C. *Autor: Andreas Jung.* **MouseTool** (Fenster hervorholen, vergrößern, verkleinern, PALPatch (verändert Intui.-Lib., vergrößert alle Screens und Windows auf PAL-Größe), ChangeTaskPri, CLI4WB (CLI-Programme von der WB starten); alle Prg. mit Source (Profimat-Assembler). *Autor: Richard Englert, Fürth.* **RossiDress V1.0**: benutzerfreundliche Adressenverwaltung; AMIGABasic. *Autor: Jürgen Rogg.* **Milben**: kleiner Screenhack; Source in C. *Autor: Guido Appenzeller*

KICK 96: UTILITY

TurboBackup: komfortables und schnelles Kopierprogramm; kopiert auf bis zu 3 Laufwerke mit Verify.
MRBackup2.1: Harddisk-Backupprogramm, sichert auch überlange Dateien.
DMouse: Bildschirmschoner, Mausabschalter, Mausbeschleuniger, Fensteraktivator, Funktionstastenbelegung.
ARC: Programm zum Packen und Entpacken von Dateien. Hiermit können Programme platzsparend abgelegt werden (siehe PD 94/95).

KICK 95: ANWENDUNGEN

AnalytiCALC: professionelles Tabellenkalkulationsprogramm (Spreadsheet), das auch schon für MS-DOS, VAX und PDP-11 existiert. Das Programm arbeitet im Interlace-Modus und benötigt deshalb 1MB Speicher. Sehr leistungsfähig (18.000 * 18.000 Zellen usw.).
Wichtig: Die Dateien der Disketten 94 & 95 sind in komprimierter Form abgelegt. Sie benötigen das Programm ARC (PD 96), um sie wieder zu entpacken.

KICK 94: ANWENDUNGEN

RIM-5 ist ein relationales Datenbanksystem, für Verwaltung großer Datenmengen. Integrierte Programmiersprache. Help-Funktion. Incl. 80-seitiger Anleitung und Source in Absoft-Fortran .

KICK 93-68

PD 93: MODULA-2 (AMOK #6)

SoftScroll (zum Scrollen einer BitMap), MathLib (Bibliothek für Vektoren und Matrizen), IFFSupport (Laden und Speichern von ILBM-Bildern), ListAll (erweitert LIST für Subdirectories.

PD 92: MODULA-2 (AMOK #5)

Graphics (IFF-Bilder mit Mausklick), PrinterSupport, Print (TYPE TO PRT:), MemSystem (intelligenter Heap-Ersatz), Superlist1.2 (für Scrollbar-Requester), PrintIt! (Hardcopyprogramm), IconSupport (zum Erzeugen von Icons)

PD 91: MODULA-2 (AMOK #4)

CHECKER ist im Rahmen einer Studienarbeit im Fach Technisches Design entstanden. Thema war die "Erarbeitung neuer Möglichkeiten der Informationsdarbietung in Kraftfahrzeugen unter Zuhilfenahme eines microcomputergestützten Grafikbildschirms".

PD 90: UTILITIES

PERFECTSOUND V2.1: Ein umfangreicher Soundsampler mit vielen Features (z.B. Stereo samplen, Stereo erzeugen, Instrument erzeugen, Bereiche manipulieren, umkehren, löschen uvm.). Einfache Mausbedienung (siehe auch Bericht in KICKSTART 10.88).
CROSSREF: Dieses Programm erzeugt eine Aufstellung aller Variablen und Befehle mit Zeilenverweis. Unverzichtbar für Programmierer; verwendbar für alle ASCII-Dateien. M2-Modula. *Autor: Jörg Gutzke, Mönchengladbach*
DISKLAB: ein Programm zum Drucken von Disketten-Labels auf 9 Nadel-Druckern (Anpassung an andere möglich!). Mausgesteuerte Auswahl von Direktoryeinträgen, Grafikeinbindung, sehr komfortabel. Source in AMIGABasic. *Autor: Manfred Illi, Ludwigshafen.*
BOOTBLOCK-CHECKER: automatisches Testen des Bootblocks, akustischer Alarm, ASCII-Ausgabe u.a.
Autor: Matthias Kühn, Weinheim

PD 89: DIGI-SOUNDS

The Sound of Music (Volume 3): weitere digitalisierte Musikstücke von *Harald Schneider*, mit Player. Hitparadensounds für eigene Programme und Vorspänne.

PD 88: MUSIK

Weitere Musikstücke von *Andreas Starr*, die man nicht verpassen sollte. Stereo, mit Player.

PD 87: MUSIK

Sehr schöne Musikkompositionen schickte uns *Andreas Starr* aus Gronau. Er hat sie mit Sonix erstellt. 5 Stücke (u.a. Miami Vice-Variation) in echtem Stereo mit einer Spielzeit von fast 20 Minuten. Die Musik wird im Hintergrund abgespielt. Lassen Sie sich dieses Erlebnis nicht entgehen.

PD 86: SPIELE

CYCLES 1.0: Eine sehr spannende Tron-Variante, bei der man sein Fahrzeug um Gegner und Quadrate herumlenken muß. In höheren Leveln wird die Geschwindigkeit größer, und immer mehr Quadrate erschweren die Suche nach einem freien Weg.

ESCAPE FROM JOVI: Wer wagt mit dem Raumschiff die Flucht durch die engen Gänge von Jovi? Source in Assembler. *Autor: Oliver Wagner, Sprockhövel*
ZAUBERFLÄCHEN: Erfordert gute Taktik, die verschiedenen Farbfelder zu sortieren. Kein Spiel für Leute, die gleich verzweifeln. Source in AMIGABasic. *Autor: Heiko Jahn*

PD 85: SPIELE

Eines der besten und spannendsten Strategiespiele ist **RISK V3.0**. Bis zu 5 Spieler müssen versuchen, die Herrschaft über die Welt zu erlangen. Dabei entscheidet nicht nur das Glück, sondern es kommt auch auf geschicktes Verhandeln an. Besonders wenn viele mitspielen, ist es schnell um einen geschehen, bzw. man wird nie sein Ziel erreichen. Das Spiel hat eine sehr schöne Grafik, ist sehr einfach mit der Maus zu bedienen und hat eine Vielzahl von Optionen, die den Spielablauf erleichtern oder erschweren können. Ein ungeheurer Spaß für lustige Spielerrunden. *Autoren: Michael Berling & Hartmut Stein (Bernstein Zirkel Softworks)*

PD 84: BILDER

Eine Bildershow mit fantastisch digitalisierten Bildern, Sound und Effekten. Die sehr realistischen und kunstvollen Aufnahmen zeigen was man, mit der entsprechenden Ausstattung, auf dem AMIGA machen kann. Erstellt mit dem DVS-2000 System von PBC Computerdesign.

PD 83: MODULA-2 (AMOK #3)

IFF-Loader in Assembler, Konverter für IFF-Brushes, Rekorder für Maus- und Tastenaktionen, Tool für GELs und anderes mehr.

PD 82: MODULA-2 (AMOK #2)

Die zweite Diskette des Modula-Klubs AMOK mit Routinen zu folgenden Themen: IFF-Loader, Halfbrite- und HAM-Demo, Hilfe zur Intuitionprogrammierung, Requesterroutinen.

PD 81: MODULA-2 (AMOK #1)

7 begeisterte Modula-Fans aus dem Raum Stuttgart haben sich zum Amiga Modula Klub (AMOK) zusammengeschlossen. Ihre Erfahrungen machen sie jedem Interessierten über die AMOK-Disketten zugänglich, auf denen sich hilfreiche Routinen in Form von Modulen befinden. Zu jedem Modul gibt es eine ausführliche Anleitung und ein Beispielprogramm. Auf der ersten Diskette befinden sich Programme wie z.B. Sprite- und Bob-Handhabung, Joystick-Routine, Interruptsteuerung und vieles mehr.

PD 80: UTILITIES

BOOTBLOCK CHAMPION testet den Bootblock auf Viren und erlaubt es, Assemblerprogramme darauf zu schreiben. Außerdem kann der Bootblock gesichert werden.
Autor: Roger Fischlin

VIRUSTEST und **VIRUSCHECK** sind Virustestprogramme, die in die Startup-Sequence eingebunden werden können. *Autor: Roger Fischlin*

DFC ist ein einfach zu bedienendes, aber leistungsfähiges Kopierprogramm. Formatieren, Verify und Kopieren auf bis zu 4 Disketten sind die Optionen des Programms.

PD 79: SCHULE

PROCALC 1.1 ist eine komfortable Taschenrechnersimulation eines HP11G. Die Bedienung solcher Rechner ist einfach, jedoch anfangs gewöhnungsbedürftig (UPN). Der große Leistungsumfang wird in einer ausführlichen deutschen Anleitung erklärt. *Autor: Götz Müller*

GRAPH-MASTER V3: ein sehr umfangreiches Programm zum Darstellen von Funktionen. *Autor: Rolf Beck*

3-D FUNKTIONEN ist ein komfortabler und schneller 3D-Plotter. Source in AMIGABasic.

PD 78: GRAFIK-SHOW

FRAKTAL-MOVIE ist ein Film aus 200 s/w-Fraktalgrafiken, die hintereinander gezeigt werden (ca. 8 Bilder/s). Dabei wird aus einem Fraktal langsam herausgezoomt, bis man das nächste erkennen kann usw.. Benötigt für den kompletten Film 1MB Speicher! *Autor: Jürgen Weineit*

PD 77: SPIELE

KART ist ein Autorennspiel, bei dem zwei Spieler ihren jeweiligen Wagen über 6 verschiedene Rennstrecken lenken müssen. Höchst-, Lenkgeschwindigkeit und die Rundenanzahl können vorgewählt werden. Gute Grafik und Sounds; erfordert zwei Joysticks. Source in C. *Autor: Axel Ilenburg*
BUNDESLIGA ist ein Spiel, bei dem eine Fußballmannschaft zur Meisterschaft geführt werden soll. Geschicktes Kaufen und Verkaufen und die Aufstellung des Teams sind die wichtigsten Maßnahmen für den Erfolg. Source in AMIGABasic. *Autor: Ingo Hirsch*

PD 76: SPIELE

KAMPF UM ERIADOR V2.0 ist ein Fantasy-Strategiespiel für zwei Personen. Im Jahre 6735 wogt der Kampf zwischen Gut und Böse im Land Eglador. Auf einem großen Feld stehen sich die Legionen der beiden Seiten gegenüber. Mit strategischem Geschick müssen Sie versuchen, den gegnerischen Magier zu erreichen, der sich am anderen Ende des Spielfeldes in seiner Festung befindet. Gute Grafik und Sounds; benötigt 1MB Speicher!
Autor: Ralf Böwing, Roland Hartz

PD 75: SUPER-SOUNDS

KICKSTART SOUND DEMO: Nicht digitalisierte Sounds, sondern programmierte Musikstücke schickt uns Tom Beuke, damit wir "Abstand nehmen von den schnöden Digi-Sounds (kann doch jeder, oder?)". 17 actiongeladene Musikstücke; Länge 12-95kB. *Komponisten: TOB & TWP*

PD 74: DIGI-SOUNDS

The Sound of Musik (Volume 1): Digitalisiert Musikstücke zur Verwendung in Vorspännen, Demos und eigenen Programmen. Sehr gute Qualtität; mit Player; 11 Samples; Länge 30-100kB. *Sampler: Harald Schneider*

PD 73: SPIELE

LIGHT CYCLE 3D: ist eine sehr gut gemachte TRON-Variante, bei der zwei Fahrer versuchen, dem Gegenspieler den Weg abzuschneiden. Die 3D-Grafik ist dabei ein wirkliches Erlebnis. Da jeder Fahrer eine undurchdringbare Bahn hinterläßt wird das Spielfeld schnell eng. Mit vielen Anzeigeinstrumenten und Anleitung. *Autor: Tobias Richter*
ARTILLERIE-DUELL: in einer zerklüfteten Berglandschaft sehen sich zwei Kanonen gegenüber. Jeder der zwei Spieler muß nun den Abschußwinkel und die Stärke einstellen, mit der er seine Kugel abfeuern will. *Autor: Peter Kliem*
TREFFER: eine Kniffel-Variante, die komplett mit der Maus gesteuert wird. Bis zu vier Spieler, mit Zwischenstand, Hi-Score und Statistik. *Autor: Michael Jänecke*

PD 72: ANIMATIONEN

Für alle **STAR TREK**-Fans hat Tobias Richter zwei Animationen mit VideoScape 3D erstellt. Die erste zeigt ein Klingonenschiff, die zweite die U.S.S. ENTERPRISE.

PD 71: UTILITIES

PRTDRVGEN 2: der Druckertreibergenerator zum Anpassen der exotischsten Typen an den AMIGA. Sehr einfache Benutzerführung mit ausführlicher (englischer) Erläuterung aller Optionen (siehe Bericht in KICKSTART 8/9 88).
VIRUS X: läuft im Hintergrund und prüft den Bootsektor jeder eingelegten Diskette auf Viren und mögliche Veränderungen. Mit Source.
Overscan: ein Patch für die Intuition-Library, mit der amerikanische Programme das PAL-Format ausnutzen könnten, ohne daß sie modifiziert werden müssen.
HP: ein RPN-Taschenrechner mit vielen Funktionen (BIN, OCT, DEC, HEX, SIN, COS, TAN, ATN, LOG, LN, LG2, ATN, y^x ...) und 32 Registern. Mit Source in C

PD 70: SPIELE

WHEEL: Bei diesem Spiel muß ein Ausdruck erraten werden. Diese Glücksrad-Simulation ist grafisch gut aufgemacht und komplett mausgesteuert. Da die zu erratenden Begriffe in englisch sind, ist dieses Spiel gut zum Erlernen dieser Sprache geeignet. **MASTERMIND**: Kombinationsspiel, bei dem eine Farbkombination erraten werden muß. Mausgesteuert! **RISTINOLLA**: Eine spielstarke GO-Muko-Variante. Mausgesteuert und sehr schnell! **CLUE**: das bekannte Brettspiel für pfiffige Detektive in einer ansprechenden Computerversion. Verschiedene Räume müssen untersucht, Personen befragt und die Ereignisse kombiniert werden um den Mörder herauszufinden. Die gute Grafik und die einfache Mausbedienung machen eine Anleitung weitgehend unnötig.

PD 69: GRAFIK-DEMOS

AMUC_DEMO: Ein Bild mit 200 x 2000 Bildpunkten scrollt vertikal über den Bildschirm. Ein sehenswertes Erlebnis.
HBHILL: Dies ist Kevin Sullivans Beitrag zum BADGE KILLER DEMO CONTEST. Die Demo nutzt den Extra-Half-Brite-Modus (64 Farben) des AMIGA.
HAMmmm: Eine mit Musik unterlegte Linendemo von Phil Burk. Inclusive Source-Code in JForth. **STARS**: Ein interessanter Flug durch einen Sternenhimmel.

PD 68: GRAFIK-DEMOS

WBLANDER: versuchen Sie das Raumschiff sicher auf einem Workbench-Fenster zu landen. Auch dieses Programm hat am BADGE KILLER DEMO CONTEST teilgenommen
DRUNKENMOUSE: wenn Sie nicht betrunken sind, dann liegt es sicherlich an diesem Programm, daß Sie Schwierigkeiten haben, etwas anzuklicken.
MACHINE: eine hervorragende Demo von Allen Hastings-erstellt mit VidoeScape 3D.

VERSANDBEDINGUNGEN:

Um einen schnellen und problemlosen Versand zu gewährleisten, beachten Sie bitte folgende Punkte:

- Bestellungen per Nachnahme oder Vorrauskasse
- Für jede Diskette ergibt sich ein Unkostenbeitrag von DM 8.-
- Pro Sendung kommt ein Versandkostenbetrag (für Porto und Verpackung) von DM 5.- (Ausland DM 10.-) hinzu.

Bei einer Bestellung von 5 oder mehr Disketten entfällt der Versandkostenbeitrag!

- Bei Nachnahme zuzüglich 3.70 DM

Anschrift:

MAXON Computer GmbH
KICKSTART PD
Postfach 55 69
6236 Eschborn

Die Diskettenbestellung kann auch telefonisch erfolgen. Der Versand erfolgt dann per Nachnahme.

Tel.: 06196 / 48 18 11
(Mo.-Fr. von 9^{00}-13^{00} u. 14^{00}-17^{00} Uhr)

VORSCHAU

CeBIT '89

Was gibt's Neues von COMMODORE?

Die diesjährige CeBIT in Hannover hat mit Sicherheit wieder einiges zu bieten. Nicht nur, daß wir selber mit einem eigenen Messestand vertreten sind, wir sind für Sie auch unterwegs und stürzen uns ins Messegewühl. Was es rund um den AMIGA zu berichten und zu sehen gibt, können Sie in der nächsten KICKSTART ausführlich nachlesen.



AMIGA-Hardware programmierung

Im nächsten Heft starten wir einen neuen Kurs, der sich der Hardwareprogrammierung widmet. Wollten Sie nicht immer wissen, wie man den Copper programmiert, um die vielfältigsten Farbspielereien zu erhalten, oder wie Sie dem Soundchip des AMIGA die Töne entlocken können? Ab dem nächsten Heft erfahren Sie mehr.

AREXX

Für AMIGA-Insider mit Sicherheit ein Begriff - AREXX. AREXX stellt die leistungsfähige Sprache REXX auf dem AMIGA zur Verfügung. Was sie auf dem AMIGA leistet, wie man sie am besten nutzen kann und vieles mehr erfahren Sie in einem ausführlichen Test in der nächsten Ausgabe.

Änderungen vorbehalten

Ab 14. April an Ihrem Kiosk.

Impressum

KICKSTART

Chefredakteur:
Uwe Bärtels (Chefredakteur)(UB)
Markus Nerding (Stellvertreter) (MN)

Redaktion:
Andreas Krämer (AK)
Harald Schneider (HS)
Marcelo Merino (MM)
Harald Egel (HE)
Christian Keller (CHK)

Herausgeber:
MAXON Computer GmbH
Industriestraße 26
Postfach 5569
6236 Eschborn
Tel.: 06196/481811
FAX: 06196/41137

Redaktionelle Mitarbeiter:
Gerald Carda (GC)
Jobst Hermeier (JH)
Carsten Borgmeier (CBO)
Martin Silbernagl (MS)
Sven Stillich (SV)
Florian Du Bois (FDB)
Meinhard Ulrich (MU)

Public Relations:
Claus Peter Lippert

Auslandskorrespondenz:
D. Dela Fuente (ddf/GB)
L. Hennely (USA)

Verlag:
Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt 13
Tel.:06151/56057
FAX:06151/55689
06151/56059

Verlagsleitung:
Hans-Jörg Heim

Anzeigenverkauf:
Kyriakulla Margaritis
Uwe Heim (Ltg.)

Anzeigenpreise:
nach Preisliste Nr.3, gültig ab 1.1.88

Fotografie, Titelbild:
Klaus Ohlenschläger Foto & Design

Illustrationen:
Romana

Produktion:
Karl-Heinz Hoffmann

Druck:
Ferling Druck W.Frotscher GmbH, Darmstadt

Bezugsmöglichkeit:
Zeitschriftenhandel, Kauf- und Warenhäuser, Commodore-Fachhändler oder direkt beim Verlag.
KICKSTART erscheint 11 mal im Jahr

Einzelpreis: DM 7,- ,ÖS 56,- SFr 7,-
Jahresabonnement Inland: DM 70,-
Europ. Ausland DM 90,-
Luftpost DM 120,-

Alle in KICKSTART erscheinenden Beiträge sind urheberrechtlich geschützt. Reproduktionen gleich welcher Art, ob Übersetzung, Nachdruck, Vervielfältigung oder Erfassung in Datenverarbeitungsanlagen, sind nur mit schriftlicher Genehmigung des Herausgebers und des Verlags erlaubt. Programmlistings, Bauanleitungen und Manuskripte werden von der Redaktion gerne entgegengenommen. Sie müssen frei von Rechten Dritter sein. Mit ihrer Einsendung gibt der Verfasser die Zustimmung zum Abdruck und der Vervielfältigung. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.

Sämtliche Veröffentlichungen in KICKSTART erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes, auch werden Warennamen ohne Gewährleistung einer freien Verwendung benutzt.

Für Fehler in Text, in Schaltbildern, Aufbauskizzen, Stücklisten, usw., die zum Nichtfunktionieren oder evtl. zum Schadhaftwerden von Bauelementen führen, wird keine Haftung übernommen.

(c) Copyright Heim Verlag

Profilaufwerk 3,5"

Metallgehäuse ● einstellbare Lauwerknummer mit Displayanzeige ● digitale Trackanzeige ● Write Protect am Laufwerk schaltbar ● abschaltbar ● durchgeschleifter Bus
1 Jahr Garantie
Super ALCOMPreis **329,-**

Laufwerk 5,25"

40/80 Track ● Laufwerksbus durchgeschleift ● abschaltbar ● einstellbare Adressen ● MS- DOS- kompatibel ● mit Diskchange

Super ALCOMPreis	**298,-**
HD 1,6 MB (umschaltbar)	**318,-**
Amigafarbene Blende	**+10,-**
Write Protect Schalter	**+15,-**

Gemischtes Doppel 3,5/5,25"

einzeln ein-/abschaltbar ● einstellbare Laufwerksnummern mit Anzeige ● durchgeschleifter Bus ● bei 5,25" 40/80 Tracks umschaltbar ● Metallgehäuse ● 1 Jahr Garantie
Super ALCOMPreis **548,-**

500er Speichererweiterung

Für 512k zusätzliches RAM ● alle RAM-s gesockelt ● selbstkonfigurierend ● abschaltbar ● Uhrenschaltung auf Platine mit Akku- bzw. Batteriepufferung nachrüstbar

Komplett mit 512k	**Preis auf Anfrage**
Superpreis mit Uhr	**Preis auf Anfrage**
Bauteilesatz für Uhr ohne Akku	**24,-**
Leerplatine mit Stecker	***39,-**

*mit Schaltplan und Bestückungsliste

Laufwerkanschlußkabel

Zum Anschluß von Laufwerken an alle Amigas ● mit Ansteuerelektronik

Für 3,5" Laufwerk	**39,-**
Für 5,25" Laufwerk	**49,-**

Steckplatzerweiterung 3-fach für Laufwerke

Jeder Steckplatz abschaltbar und einstellbare Laufwerksnummer ● Steckplatzerweiterung direkt am Amigagehäuse ● Dadurch keine Kabellängenprobleme
Anschlußfertig zum Super ALCOMPreis **49,-**

Soundsampler

Für alle Amiga's mit Software ● Type bei Bestellung bitte angeben ● 8-Bit Datenbreite ● Betrieb am Parallelport (Druckerport) ● Mit Vorverstärker für Micro-Anschluß (Cinch-Buchsen) ● Musik- und Sprachdigitalisierung möglich ● Arbeitet mit fast allen Digitizer-Programmen ● Formschönes Gehäuse
Super ALCOMPreis **79,-**

Sampler Studio

● Professionelles Sampler-Programm ● 4-Kanal-Technik ● speichern auf 4 Disketten hintereinander möglich ● alle gängigen Formate (IFF, Data, Future) ● Echtzeitdisplay mit Zoomfunktion ● viele Verfremdungsmöglichkeiten ● Echo, Hall, Reverse **69,-**

Paket: Sampler + Software **129,-**

MIDI-Interface

4 Kanäle einschließlich 1 Thru ● Optische Datenanzeige ● Formschönes Gehäuse
Wahnsinnspreis von nur **89,-**

ausgereifte Ingenieurleistung ● 14 Tage Umtauschrecht ● fast alle IC's gesockelt ● nur professionelle Leiterplatten ● Bauteile namhafter Hersteller ● mit Bedienungsanleitung







3,5" Laufwerk

Für alle Amiga's ● einstellbare Gerätenummer ● abschaltbar ● Metallgehäuse ● superflach ● 1 Zoll (2,54cm) ● durchgeschleifter Bus ● TEAC Laufwerk
1 Jahr Garantie

komplett anschlußfertig	**239,-**
Amigafarbene Blende	**+10,-**

Basislaufwerke

1 Jahr Garantie

TEAC FD 135 FN 3,5" 1 MB superslimline 1,6 MB Diskchange	**218,-**
Amigafarbene Blende	**+10,-**
3,5" Gehäuse	**25,-**
5,25" Gehäuse	**25,-**
Gehäuse für "Gemischtes Doppel"	**65,-**

Bootselector **19,90**

Amiga Eprommer

- Für A 500/1000
- Expansionsportanschluß
- Für EPROM's 2764-27011 (8K-128K) Alle A-Typen und CMOS-Typen
- Funktionen:
  LEERTEST LADEN VON DISK
  VERGLEICHEN SPEICHERN AUS DISK
  AUSLESEN HEXDUMP
  BRENNEN
- vier Programmieralgorithmen 50mS/Byte - Superschnell 64K-1,5 min
- Programm zum Generieren und Brennen von Kickstarts direkt von Diskette oder aus ROM
- Mit Software + Gehäuse **225,-**

Meß- und Steuerinterface

- 8 ADC-Kanäle 0-2,55V in 0,01V Stufe
- 1 DAC-Kanäle 0-2,55V in 0,01V Stufe Genauigkeit- 1,5 LSB
- 8 frei programmierbare TTL-I/O Kanäle
- Mit Gehäuse, Anschlüße auf Schraubklemmen
- interne Referenzspannung
- Expansionsanschluß
- Einfache Programmierung in Basic möglich Multitasking tauglich
- incl. DEMO-Software auf 3,5" Diskette

239,-

Trackanzeige

Für DF0-DF3 einstellbar ● für alle Laufwerke (3,5"/5,25") ● Laufwerkbus durchgeschleift ● mit Gehäuse
Super ALCOMPreis **69,-**

Einführungsangebot Amiga-Harddisks

komplett anschlußfertig	
Platte 20 MB A 2000	**798,-**
30 MB A 2000	**898,-**
40 MB A 2000	**1 098,-**
65 MB A 2000	**1 348,-**
Platte A 500/A 1000	
20 MB	**898,-**
30 MB	**998,-**
40 MB	**1 248,-**
65 MB	**1 498,-**
für den Selbstbau	
HD-Interface A 2000	**198,-**
HD-Interface A 500/A 1000	**249,-**

Vokabeltrainer

2500 englisch-deutsche Vokabeln incl. Hilfssatz ● Merkfunktion ● komfortabler Editor zur Vokabelverwaltung ● Wörterbuch zum Dateiendurchsuchen **59,-**



Selbstbootende Harddisk für Amiga ohne PC-Karte!

Die Amiga-Festplatte von ALCOMP:
● Selbstbootend wie "Card" oder "Rad"! ● Als Einbau-Festplatte für den "Amiga 2000" ● Als Externe Einheit für den "Amiga 500" und 1000 mit Gehäuse, eigenem Netzteil und Erweiterungsanschluß ● Erhältlich mit 20, 30, 40 und 65 Megabyte ● Kopiert 1 Megabyte in unter 4 Sekunden ● Speichert schneller als "1.2-Ramdisk" ● Läuft mit "FastFileSystem" ● Einfach einstecken, Formatieren, "Mountlist" und "Startup-Sequence" ändern und los geht's!
Entwickler: Stephan und Stefan
Für den Selbstbau: Harddisk-Interface incl. Steuersoftware ● Anschluß mit Slot für Omti-Controller

Frischer Wind auf dem Hardwaresektor

ALCOMP
COMPUTERHARDWARE

Kickstartumschaltung

Bauen Sie die anderen Kickstart-Versionen in Ihren Amiga 500 ● Einfacher Einbau ohne Löten ● für Original-Kickstart-ROM und 2 zusätzliche Versionen auf EPROM ● EPROM-Programmierservice auf Anfrage

SuperALCOMPreis	**59.-**
Kickstartversion auf EPROM's	**120.-**

Userport + Experimentierkarte für Expansionport

Mit Lochraster und 2 x 6522 Ports

Leer	**59.-**
komplett aufgebaut	**89.-**

Wir suchen ständig Hardware-Entwicklungen. Wir garantieren gute Umsatzprovisionen und ehrliche Abrechnung

kostenloses Info anfordern!!!

Bestellung und Versand

ALCOMP GmbH
Glescher Weg 22
5012 Bedburg
Tel. 0 22 72/20 93

Nachnahmeversand NN-Spesen 7.50 DM b. Vorkasse 3.- DM. Auslandsbestellungen: Nachnahmeversand NN-Spesen 10.- DM b. Vorkasse 5.-DM. Wir liefern Ihnen auf Ihre Rechnung und Gefahr zu den Verkaufs- und Lieferbedingungen des Elektronikgewerbes.
Postgiroamt Köln
(BLZ 370 100 50) 275 54-509

Entwurf + Druck PLUM, Düren-Oberzier, TEL. 02428/711



HD 3000

externes Festplattensystem für alle Amiga
überragende Leistungmerkmale wie Datentransferrate bis **327K/sec** ● NEC Qualitätsplatten ● förmschönes Gehäuse als Monitorunterbau ● eigenes Netzteil ● Lüfter ● sehr komfortabeles Format-Menue ● defekte Zylinder werden erkannt, gesperrt und erstetzt, so erhalten Sie auch bei Harderrors immer die volle Plattenkapazität und brauchen nicht nach Fehlerrate Ihre Partitionen zu wählen

20 Megabyte	**998.-**
30 Megabyte	**1098.-**
40 Megabyte	**1498.-**
60 Megabyte	**1698.-**



Golem Memory Station

externes Festplattensystem für alle Amiga
technisch mit den gleichen überragenden Leistungsmerkmalen wie HD 3000 ● zusätzlich kann eine weitere Festplatte oder ein 5,25 Zoll Laufwerk und ein 3,5 Zoll Laufwerk eingebaut werden ● Elektronik mit zwei Trackdisplays für die Laufwerke ist bereits vorhanden ● Im Harddisk Interface ist ein 2 Megabyte dynamik Ram Controller integriert
Nutzen Sie die Möglichkeit sehr preiswert Ram und Floppies nachzurüsten

20MB	**1398.-**
30MB	**1498.-**
40MB	**1898.-**
60MB	**2098.-**
Nachrüstsätze zur Memory Station	
2MB Rámbank	**1098.-**
3,5 Zoll Laufwerk NEC incl. Kabelsatz	**200.-**
5,25 Zoll Laufwerk NEC incl. Kabelsatz	**220.-**

Sie finden uns auch auf der 12.

Hobby-tronic

12. - 16. April 1989
Halle 6, Stand 6054

(02 31)
81 83 25-27



Eprommer

Eprombrenner für alle gängigen Eproms bis zum Megabit-Eprom leistungsfähiger integrierter Monitor ● komfortabeles Menue sehr schnell z.B. 27512 (64KB) in 15 Sekunden

Golem Eprommer incl. Brennsoftware u. Monitor **249.-**

GOLEM
HARDWARE

AKTUELL

8MB für A2000

2MB bestückte Einsteckkarte ● 8MB gesockelt ● autokonfigurierend **1398.-**

Viruskiller V2.0

Das Superprogramm erkennt jeden gängigen Virus und beseitigt ihn, prüft Ihren Speicher auf Viren, schützt Ihre Disk vor erneutem Virusbefall, preiswerter Apdate Service
Einführungspreis nur **49.-**

Netzteile 150W

PC-Schaltnetzteil incl. Lüfter überall einsetzbar z.B. für Ihre Selbstbaufestplatte
zum Sensationspreis **99.-**

Amiga Action Spiel

P.O.W. oder Capone	**99.-**
Lightgun Anschluß am Joystickport	**129.-**
Powerpack Gun + wahlw. ein Spiel	**199.-**

Kupke Computertechnik · Burgweg 52a · D-4600 Dortmund 1 · ☎ 02 31/81 83 25-27