

AMIGA DOS

10/90

10/90

ISSN 0937-2717
DMV-Verlag

Tuning:

Turbo-Karten im Test

Bewegte Bilder:

Alles zum Thema Grafik und Animation

AMOS!

Das Super-BASIC

Ab sofort:

*8 Seiten PD-Workshop zum
Heraustrennen und Sammeln*



Autorisierter Commodore-Fachhändler
Commodore Commercial Developer

- Entwicklung
- Produktion
- Hardware
- Software
- Service

HK-Computer
Ihr Amiga-Spezialist



*** HK-Professional ***
Top in Qualität, Funktion und Design

Professional Drive Diskettenlaufwerke

- 3 1/2" Laufwerk AMIGA 2000 intern DM 159,—
komplett mit Einbaubaukit und Anleitung
- 3 1/2" Laufwerk für alle AMIGAs extern DM 199,—
abschaltbar, Busdurchführung, AMIGAFarben
- 5 1/4" Laufwerk für alle AMIGAs extern DM 249,—
abschaltbar, Busdurchführung, 40/80 Tracks, AMIGAFarben

Professional RAM-Board IIC A500

- auf 1 MB DM 149,—
- jetzt noch leistungsfähiger!
- superschnelle Megabit-RAMs (4*514256)
- mit accugepufferter Uhr & Datum
- Writeprotect für die Uhr
- Accu abschaltbar
- Hard- und Softwaremäßig abschaltbar
- Superniedriger Stromverbrauch
- dto. Platine mit Uhr & Schalter ohne RAMs DM 89,—

Professional RAM-Board III A500

- auf 2,3 MB DM 498,—
- intern, incl. Gary-Adapter
- superschnelle Megabit-RAMs (16*511000)
- mit Uhr & Datum
- Hard- und Softwaremäßig abschaltbar
- dto. Platine mit Uhr & Schalter ohne RAMs DM 198,—

Professional RAM-Board IIIB A500

- auf 2,5 MB DM 498,—
- mit dem neuen Big Fat Agnus volle 2,5 MB!!
- intern, incl. Gary-Adapter
- superschnelle Megabit-RAMs (16*511000)
- mit Uhr & Datum
- Hard- und Softwaremäßig abschaltbar
- Hinweis: eine Lötstelle nötig
- dto. Platine mit Uhr & Schalter ohne RAMs DM 198,—

Professional RAM Board A2000

- 8 MB mit 2 MB bestückt, autokonfigurierend, einfach durch zusätzliche RAMs und Jumper umstecken aufrüstbar, keine neuen PALs erforderlich
- dto. Platine bestückt mit 4 MByte DM 1098,—
- dto. Platine bestückt mit 8 MByte DM 1698,—
- dto. Platine teilbestückt ohne RAMs DM 498,—
- RAM-Satz für 2 Megabyte DM 300,—

Professional MIDI für alle Amigas

- DM 139,—
- Superschnell
- Optokoppler mit 700 %! Kopplungsfaktor
- 1" In, 1" Thru, 3" Out
- Leitungstreiber an allen Ausgängen für lange Datenkabel
- Amigafarbenes Metallgehäuse
- abschaltbar, mit Betriebs-LED

Professional SCSI II

- Harddisk-Controller DM 498,—
- A2000 Slotkarte als Filecardträger ausgelegt
- voller 16bit-Datenbus
- vergoldete Kontakte
- durchgeführter SCSI-Bus für bis zu 8 SCSI-Devices
- Autoboot
- abschaltbar
- Komplette Filecards in verschiedenen Größen auf Anfrage

Disketten

	10 Stück	100 Stück
3 1/2" NoName 2 DD	DM 12,90	DM 119,—
3 1/2" Verbatim Verex 2DD	DM 18,90	DM 179,—
5 1/4" NoName 2S2D	DM 5,90	DM 57,—
5 1/4" Verbatim Verex	DM 9,90	DM 95,—

Größere Staffeln auf Anfrage

— NEU — NEU — NEU — NEU — NEU — NEU —

Maus & Joystick-Adapter DM 49,—

- jetzt AUTOMATISCH!, manuelles Umschalten überflüssig
- für gleichzeitigen Anschluß von Maus und Joystick
- Umschaltung erfolgt einfach durch Betätigen von Maus/Joystick
- alle Maussteuerleitungen elektronisch geschaltet, dadurch keine Spannungsspitzen

— NEU — NEU — NEU — NEU — NEU — NEU —

AMIGA-Bremse

- * der Highscore-Killer * DM 39,50
- intern für alle Amigas
- regelt die Geschwindigkeit stufenlos bis zum Stillstand
- ideal für schnelle Games und Bildschirmfotografie

Amiga-Bremse für A500 extern mit LED DM 69,—

- der Highscore-Killer jetzt auch extern!

Kick-ROM DM 49,—

- Kickstartumschaltplatine für zwei Original-ROMs

- Kick-ROM mit ROM DM 98,—
- OriginalROM 1.2 oder 1.3 DM 65,—

Kickstartumschaltplatine 3-fach DM 59,—

- für zwei OriginalROMs und eine Epromversion
- Umschaltplatine mit OriginalROM DM 108,—
- Brennservice incl. einem Epromsatz DM 79,—
(gegen Einsendung einer OriginalDiskette)

PowerFire Das Superding! DM 19,90

- Dauerfeuerinterface für Joystick und Maus
- optimale Impulsfolge für jedes Game einstellbar
- Dauerfeuer wird über Feuer- bzw. Maustaste aktiviert
- einfach zwischen Maus/Joystick und Rechner stecken
- abschaltbar

Drive-Expander DM 39,—

- bis zu drei Laufwerke direkt am Rechner anschließbar
- einstellbare Laufwerksnummer
- keine Kabellängenprobleme
- abschaltbar
- z. B. für externe Laufwerke ohne Busdurchführung
- bei Verwendung eines Boot-Selectors kann von jedem externen Laufwerk gebootet werden

BOOT-Selector elektronisch DM 49,—

- wahlweise Booten von allen Laufwerken

BOOT-Selector für Amigas DM 14,50

- wahlweise Booten von DF0: oder DF1: oder DF2: oder DF3: (bei Bestellung bitte angeben)

BTX Decoder mit FTZ (Drews) DM 199,—

Original Amiga-Maus DM 69,—

Reisware Maus für Amiga DM 89,—

Staubschutzhauben

- AMIGA 500 DM 16,50
- AMIGA 2000 Keyboard DM 16,50
- Monitor 14" DM 29,50
- Drucker 10" DM 24,50
- Drucker 15" DM 29,50

Die Staubschutzhauben sind aus Kunstleder mit weichem antistatischen Innenfutter

Software

- MEDUSA der Atari ST-Emulator DM 444,—
- RAM-Test Amiga DM 24,50
- zeigt defekte Speicherstellen grafisch an, 100 %
- Assembler jetzt auch für 32Bit-RAM, z. B. A2500/3000

- PACKIT DM 39,—
- Superschneller Cruncher nicht nur für Text und Grafik, sondern auch für Programme. Verschiedene Kompaktmodi, Auto- oder Loaderstart, räumt nicht nur Ihre Disketten, sondern auch Festplatten auf.

- XCOPY II DM 49,—
- XCOPY II mit Hardwarezusatz DM 69,—
- XCOPY professional DM 99,—
- Turboprint II DM 89,—
- Turboprint professional DM 188,—
- Quarterback (Festplatten-Backup) DM 119,—
- DPaint III DM 248,—
- Beckertext II DM 289,—
- GFA-Basic 3.5 DM 229,—

AMIGA Köln 90

Wir stellen aus
08.-11. November 1990
Halle 10, Stand 824

AMIGA-Computer

- Amiga 3000-16 MHz DM 1798,—
— 68030 CPU 16 MHz, 32 bit, 2 MB RAM, 40 MB SCSI-Harddisk
- Amiga 3000-25 MHz DM 898,—
— 68030 CPU 25 MHz, 32 bit, 2 MB RAM, 40 o. 100 MB SCSI-Harddisk
- Amiga 2500/30 DM 848,—
— 68030 CPU 16 MHz, 3 MB RAM, 40 MB Harddisk
- Amiga 2000 DM 898,—
- Amiga 500 DM 598,—
- Harddisk A590 20 MB für A500 DM 1098,—
- Colormonitor Commodore 1084 SP1 DM 1098,—
- Targa Multisync-Monitor TM 1480 DM 1098,—
— Lochmaske 0.28 mm, incl. A3000-Anschlußkabel

Festplatten

Kapazität	Speed	Filecard	A2000	A500
20MB/3 1/2"	35 ms	898,—	848,—	1048,—
30MB/5 1/4"	65 ms	—	898,—	1098,—
30MB/3 1/2"	35 ms	998,—	948,—	1148,—
40MB/5 1/4"	28 ms	—	1098,—	1298,—
50MB/3 1/2"	35 ms	1198,—	1148,—	1348,—
60MB/5 1/4"	28 ms	—	1298,—	1498,—

- Alle unsere Festplatten werden mit Autoboot-Hard- & Software ausgeliefert.
- AUTOBOOTMODUL für A2000 DM 119,—
- AUTOBOOTMODUL für A500 DM 149,—
- AUTOBOOT-KARTE für A2090-Controller DM 119,—
- TURBO-CHIPSATZ für A2090A-Controller DM 149,—
- Alle zum Nachrüsten, incl. Software und Anleitung

- Festplatten-Controller
- OMTI 5520B für MFM-Platten (20/40MB) auf Anfrage
- OMTI 5528B für RLL-Platten (30/50/60MB) auf Anfrage

Festplatten-Interface DM 99,—

- Die Adapterplatine paßt den PC-BUS eines Festplatten-Controllers an den AMIGA-BUS an. (Bitte Rechner typ angeben)

Autoboot-Set MFM

- OMTI 5520B, Autoboot-Modul, Festplatten-Interface, Kabelsatz
- für Amiga 2000 (interne Slotkarte) auf Anfrage
- für Amiga 500 (extern mit Gehäuse) auf Anfrage

Autoboot-Set RLL

- OMTI 5528B, Autoboot-Modul, Festplatten-Interface, Kabelsatz
- für Amiga 2000 (interne Slotkarte) auf Anfrage
- für Amiga 500 (extern mit Gehäuse) auf Anfrage

Autoboot-Filecardträger

- Harddisk-Trägerplatine mit DM 299,—
- integriertem Autoboot-Modul,
- Controlleradapter, Autoboot-Software

Hinweis: Alle unsere externen Geräte haben — soweit erforderlich — keine FTZ-Zulassung, wenn nicht gesondert angegeben. Ein Betrieb im Bereich der Deutschen Bundespost ist verboten.

Haben Sie Hard- oder Software für den Amiga entwickelt?
Wir bieten Ihnen eine großzügige Umsatzprovision und eine ehrliche Abrechnung.
► Sprechen Sie uns an ◀

HK-Computer

F. Hansmann & Th. Küpper GbR
Bonner Straße 37 · 5000 Köln 1

Telefon: 0221/31 1606 · Telefax: 0221/32 11 66 · BTX: * HK #

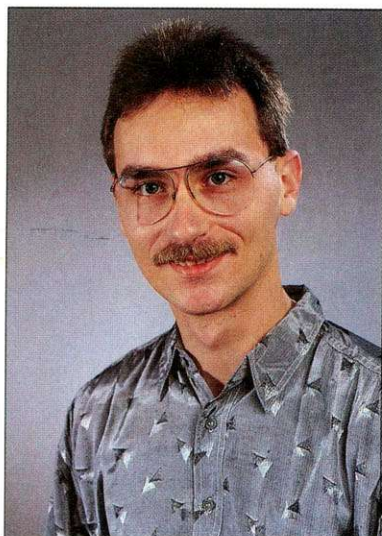
Geschäftszeiten: Mo.-Fr. 10.00 - 13.30 u. 14.30 - 18.30

Sa. 10.00 - 14.00 Uhr

Stadtsparkasse Köln: BLZ 37050198, Kto. 6342133

Nachnahme-Versand innerhalb Deutschlands per UPS oder Post DM 10,—. Nachnahme Ausland per Post DM 20,—. Großgeräte nach Gewicht.

Fordern Sie unser kostenloses Info an!



Feed back

Die Computerwelt befindet sich in einem stetigen Wandel – und ebenso die Medien, die über diese Entwicklungen berichten. Ganz augenscheinlich werden Sie dies feststellen, wenn Sie diese Ausgabe von AMIGA DOS einmal genauer unter die Lupe nehmen. An dieser Stelle erst einmal unser herzlicher Dank für Ihre Beteiligung an unserer Leserbefragung. Durch Ihre Zuschriften haben wir wertvolle Informationen erhalten, wie wir die AMIGA DOS für Sie so gestalten können, daß Sie möglichst genau Ihren Wünschen entspricht. Wir haben keine Mühen gescheut, diese Ideen in die Tat umzusetzen. So finden Sie ab sofort auf vielfachen Wunsch einen PD-Workshop zum Herausnehmen und Sammeln, der Hilfestellung zu der Vielfalt der PD-Programme gibt. Selbstverständlich finden Sie auch weiterhin in der AMIGA DOS, zusätzlich zu dem nun eingerichteten PD-Workshop, die PD-Werkzeugkiste und die Spieleshows, in der wir PD-Spiele genauer untersuchen. Denn kaum ein Software-Bereich bietet so viele Programme zu jeder denkbaren Anwendung wie die Public Domain. Und dies zu einem mehr als fairen Preis.

Viele Bereiche der AMIGA DOS haben ein neues Gesicht erhalten. Der Amiga ist nun einmal eine bunte und soundgewaltige Maschine wie kein anderer Computer. Neue Software für den Amiga nutzt diese Möglichkeiten immer konsequenter aus, eine Entwicklung, der auch wir auf jeden Fall gerecht werden wollen.

Nach wie vor steht natürlich Information über alle Neuheiten im Amiga-Bereich an oberster Stelle unserer Ziele. So werden wir Sie auch in Zukunft mit brandaktuellen Tests, News und allen zentralen Themen rund um den Amiga versorgen.

Enorme Grafik-Power ist eines dieser Themen, die den Amiga leistungsmäßig in die Nähe von professionellen Workstations rücken läßt.

Unser Titelthema Grafik und Animation hat diesen Themenbereich aufgegriffen, klärt grundsätzliche Fragen und zeigt auf, was Sie alles mit Ihrem Amiga realisieren können.

Unser Einkaufsführer gibt Ihnen schließlich die Tips, die Sie brauchen, um für Ihr Geld die größtmögliche Leistung zu erhalten.

Bleibt uns nur noch, Ihnen viel Spaß bei der Lektüre der neuen AMIGA DOS zu wünschen.

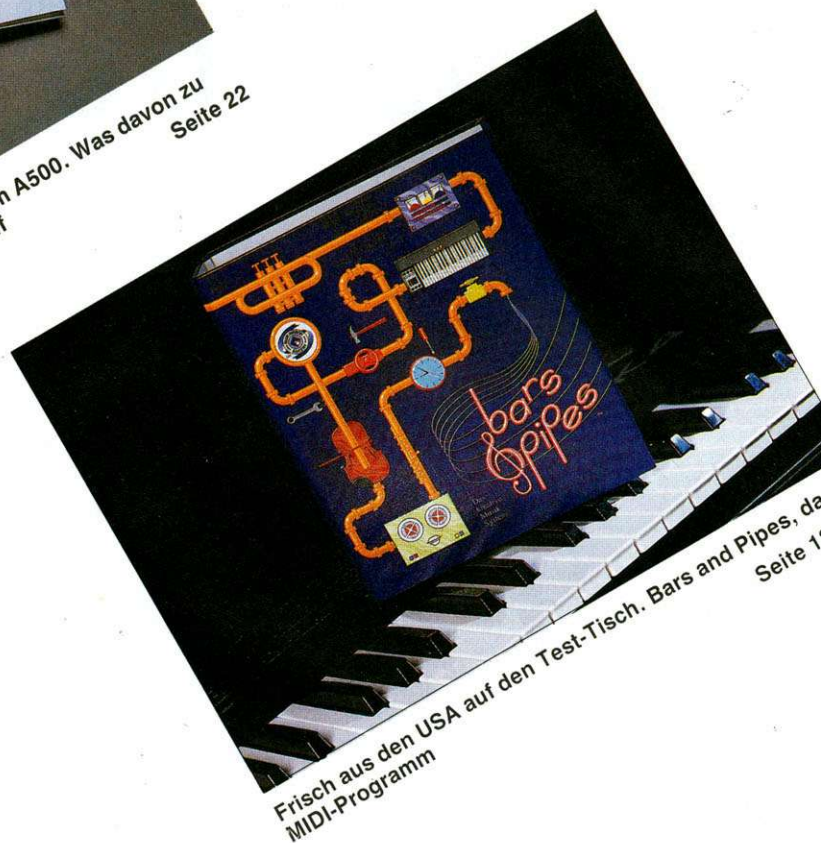
Herzlichst Ihr

Markus Matejka

Markus Matejka



MS-DOS endlich auf dem A500. Was davon zu halten ist, lesen Sie auf Seite 22



Frisch aus den USA auf den Test-Tisch. Bars and Pipes, das MIDI-Programm Seite 1

AMIGA DOS

AMIGA NEWS

Neues vom Amiga-Markt

6

SOFTWARE

MIDI in Farbe

Neuartiges MIDI-Programm im Test

12

Amos, der Gewaltige

Spieleprogrammierung leichtgemacht

16

Galerie auf Diskette

Kunstwerke für den Eigenbedarf

19

Drucken wie ein Profi

Turbo Print Professional im Test

20

HARDWARE

MS-DOS für den Kleinen

22

Vom Tisch in den Speicher

Der Sony-Scanner

24

Kraftpaket Vortex Athlet

25

Amiga mit Nachbrenner

68030-Prozessor-Karten im Vergleich

26

Tiefkühlkost auf Diskette

Wie Spiele eingefroren und verändert werden

29

TIPS & TRICKS

Gewußt wie

Tips für Einsteiger und Fortgeschrittene

44

Der Mauszeiger und die Zeit

54

Wie groß wird die Box?

Berechnungen für Lautsprecher-Volumen

62

TITEL

Faszination Grafik

Entstehung von animierten Grafiken

32

Grafik-Werkzeuge für jede Anwendung

Programmarten tiefer beleuchtet

40

Marktübersicht

Grafik-Soft- und Hardware

Wo gibt es was?

43

KURS

AmigaDOS leicht

und verständlich Alternativen zum CLI

50

Assembler ist keine Hexerei

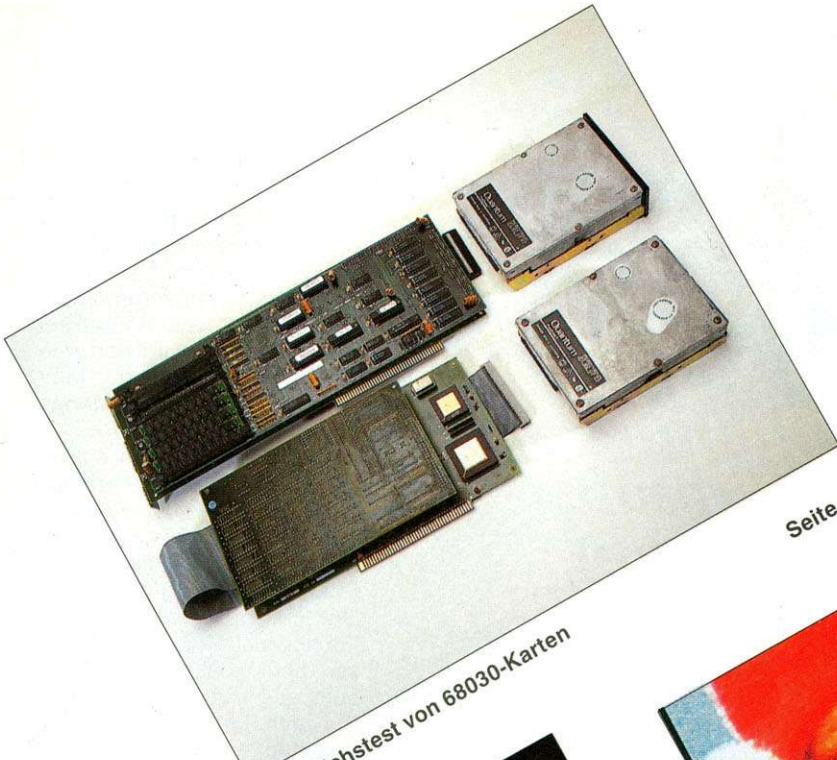
Codieren in Maschinensprache

56

Freie Fahrt für Modula2

Prozeduren und Variablen

72



Seite 26

Vergleichstest von 68030-Karten

Might & Magic II,
das neue Rollen-
und Fantasie-Spiel
auf dem Prüfstand
Seite 112



Was bietet der Amiga in Sachen Grafik und Animation? Wir
bieten Grundlagen und Marktübersicht
Seite 32

LISTING

Push it!!! 80
Das Geheimnis des Würfels

LISTING DES MONATS

Jingle Machine Teil 2 86
*Die Fortsetzung des
Super-Programms*

WERKSTATT

**Vielseitige Requester
für alle Fälle** 96
*Nutzen und Erstellen von
Fehler-Fenstern*

PUBLIC DOMAIN

PD-Workshop 63
*Zum Heraustrennen
und Sammeln*
SID-Workshop 64
Selector-Workshop 70

PD-Werkzeugkiste 106

PD-Spieleshow 108

SPIELE

Spieletests
Might & Magic II 112
Domination 113
Robot Commander 113
Damocles 120
Inspektor Griffu 121
Qatbol 121
Neuromancer 122
Rotox 123
Kickoff 2 124
Flimbo's Quest 124
Imperium 125
Dynasty Wars 126
Harley Davidson 126
AMIGA-DOS-Spieletips 114
Helpline

**Demnächst auf
Ihrem Computer** 128
Preview

RUBRIKEN

Editorial 3
AMIGA-DOS-Tip 83
Leserbriefe 94
Bücher 104
Wettbewerb 111
Impressum 129
Inserentenverzeichnis 129
Vorschau 130

AMIGA Köln '90

**Wir stellen aus
09.-11. November 1990
Halle 12, Stand 307**

Nachdem im Oktober 1989, was die Resonanz auf die erste europäische Amiga-Messe angeht, selbst die kühnsten Erwartungen übertroffen wurden, wird das Spektakel bald wieder Aufsehen erregen. Das Amiga-Ereignis '90 nimmt seinen Lauf...

Köln, 10. November 1989, 9.00 Uhr. Menschenmassen drängen zum Eingang der Halle 6 des Messegeländes. Ursprünglich war die Öffnung der Messe für 11.00 Uhr vorgesehen, doch schon bald sahen sich die Organisatoren gezwungen, die Hallentore früher zu öffnen. Dies war der Auftakt des größten Amiga-Ereignisses des letzten Jahres. 35000 Besucher sollte die Amiga '89 in den darauffolgenden zwei Tagen noch erleben (und fast in ein Chaos stürzen) – mehr als doppelt so viele, wie erwartet wurden.

Ganz anders wird es jedoch dieses Jahr vonstatten gehen. Die AMIGA DOS möchte Sie schon jetzt über die wichtigsten Fakten unterrichten, damit Sie zu den informiertesten Besuchern zählen. Rechtzeitig zur Amiga '90 wird dann in Ausgabe 11/90 ein ausführlicher Vorbericht erscheinen, der Ihnen zeigt, welche Aussteller mit Neuigkeiten auftrumpfen und wo Sie diese aufsuchen können. Für all diejenigen, die zu Hause bleiben mußten, aber trotzdem nicht auf Informationen aus erster Hand verzichten wollen, werden in Heft 12/90 Interviews, Neuheiten, Vorstellungen und

eine allgemeine Berichterstattung erscheinen.

Die Organisatoren der Amiga '90 haben sich dieses Jahr für jeden Ansturm gewappnet. Mit 25000 m² in zwei Hallen sowie bis zu 7 Meter breiten Gängen müßten etwa 140000 Besucher kommen, damit es so voll würde, wie im Jahre 1989. In der Zeit vom 9. bis 11. November finden Sie im Obergeschoß in Halle 10 des Kölner Messegeländes alles, was den Profi-Anwender interessiert. Mehr als 73 Aussteller werden in dieser Halle vertreten sein und das Beste zeigen, was zur Zeit auf dem Weltmarkt für den Amiga erhältlich ist. Doch auch das Obergeschoß der Halle 12 kann mit seinen 43 zusätzlichen Ausstellern einiges bieten. Dort wird ein Großteil der Verlage, Spieleproduzenten und Unterhaltungsanbieter Neues zeigen. Insgesamt also über 116 Aussteller – und täglich werden es mehr. Im Gegensatz zu den 74 Ausstellern im Oktober '89 sicherlich eine immense, aber berechtigte Steigerung.

Doch nicht nur Neuigkeiten werden Sie Anfang November bestaunen können. Wie die beiden letzten Amiga-Ausstellungen in Köln und Basel auch, ist diese Messe eben-

falls eine Verkaufsmesse. So manches Schnäppchen kann hier jeder machen, der genügend Geld mit auf die Messe nimmt und den günstigsten Zeitpunkt erwischt.

Hard- und Software können Sie ebenso zu speziellen Messepreisen erstehen, wie es möglich ist, mit Verlagen Kontakt aufzunehmen und vielleicht den einen oder anderen Auftrag an Land zu ziehen. Wahrscheinlich wird Sie sogar eine große Projektionswand an der Decke von Halle 12 über die aktuellsten Angebote beider Ausstellungsgebäude informieren und so die Suche nach bestimmten Produkten weiter vereinfachen.

Interessant sind sicherlich für alle engagierten Anwender auch die Masterclasses. Das sind Vorträge von internationalen Amiga-Spezialisten über ein bestimmtes Thema. Sie werden mit einem eigenen Eintrittspreis belegt, der eventuell benötigtes Material bein-

haltet und von Thema zu Thema variiert.

Kostenlos hingegen sind die Podiumsdiskussionen, die an allen drei Tagen stattfinden. Über Raubkopiererunwesen, PD-Software und Viren-Programmierung wird dort von Fachleuten debattiert. Man kann dort sicher eine Menge Anregungen erhalten.

Die Eintrittskarte kann entweder direkt auf dem Messegelände oder im Vorverkauf bezogen werden. Die Preise liegen bei 10,- DM für Schüler und Studenten sowie 15,- DM für Erwachsene. Möchten Sie den Fachbesuchertag am 8.11. nutzen, der für das normale Publikum gesperrt ist, so wenden Sie sich bitte am Veranstaltungstag mit einem entsprechenden Nachweis an eines der zahlreichen Kartenhäuschen.

Im Vorverkauf sind die Karten zu bekommen, wenn Sie bis zum 10. Oktober eine Bestellung mit Verrechnungsscheck über den erforderlichen Betrag an AMI-Shows, Zugspitzstr. 2a, 8011 Vaterstetten schicken. Später oder ohne Scheck eingehende Bestellungen können nicht beachtet werden; das Datum des Poststempels zählt! Möchten oder können Sie als Besucher nicht selbst für die An- und Abreise oder die Unterkunft sorgen, so wenden Sie sich an das Reiseunternehmen Giller, das anlässlich dieser Messe Spezialangebote im Programm hat. Informationen zu diesem Service erhalten Sie telefonisch unter 089/6132002, Bestellungen können per Fax unter der Nummer 089/6135019 abgegeben werden.

Und nun viel Spaß bei der Erwartung des frohen Ereignisses, welches die Welt des Amiga in einer solchen Gesamtheit darstellen wird, wie das selbst auf amerikanischen Messen unbekannt ist ... Rechtzeitig zur Messe finden Sie in der AMIGA DOS dann alle Angaben über Neuheiten, Standnummern, Besonderheiten und so weiter.

Natürlich finden Sie auch den DMV-Verlag und die Redaktion der Amiga DOS auf der Messe. Und wo finden Sie uns? Wenn Sie sich in Halle 12 begeben, dann brauchen Sie nur noch die Standnummer 307 zu suchen.

(Ottmar Röhrig/cd)

AMIGA-DOS-Kurzinfo

Name: Amiga '90

Organisation: AMI-Shows, Zugspitzstr. 2a, 8011 Vaterstetten

Ort: Messegelände Köln, Halle 10 und 12, jeweils im Obergeschoß

Zeit: 9., 10. und 11. November 1990

Fachbesucher: 8. November 1990

Eintritt: Schüler/Studenten 10,- DM, Erwachsene: 15,- DM

Organisierte Unterkunft und An-/Abreise: Giller Reisen, Tel.: 089/6132002

Vorverkauf: Bis 10.10. beim Organisator

Dr. T's drängt auf den Amiga-Markt

Nachdem bereits einige Programme des amerikanischen Musik-Software-Herstellers Dr. T's auf dem Amiga erhältlich sind, kann nach der neuesten Veröffentlichung davon ausgegangen werden, daß diese Firma die Amiga-Szene bewußt als Marktpotential erkannt hat. Mit »Tiger Cub« ist nämlich nun eine geringfügig abgespeckte Version der unbestritten bekanntesten Musiksoftware »Tiger« für den ATARI ST auch als Umsetzung für den Amiga erschienen. »Tiger Cub« ist eine sogenannte Sequencer-Software, die die Ansteuerung von MIDI-Instrumenten im weitesten Sinne unterstützt. So wird eine 12-Spur-Bandmaschine simuliert, die Musik auf MIDI-Geräten abspielen oder auch von ihnen aufnehmen kann. Auf einem Synthesizer gespielte Stücke können in den Amiga eingelesen und dort weiterbearbeitet werden. Im Gegensatz zur ATARI-ST-Version macht die Amiga-Version vollen Gebrauch von den leistungsfähigen Sound-Chips im Computer und kann somit begrenzt auch ohne MIDI-Instrument eingesetzt werden. In einem der drei Hauptbildschirme werden die verschiedenen Spuren des »Tonbandgerätes« fast ausschließlich mit Gadgets bedient, die denen eines normalen Recorders gleichen. Spuren können in ihrer Geschwindigkeit verändert, gemischt, stumm geschaltet, gefiltert, transponiert und sogar mit mehreren MIDI-Kanälen belegt werden. Diese Informationen werden im Standard-MIDI-Format abgelegt, das kompatibel zu den entspre-

chenden Formaten anderer Computer, insbesondere denen des ATARI ST, ist.

Im Grafikeditor, dem zweiten Hauptbildschirm, werden die eigentlichen Notenwerte bearbeitet. Anfangs etwas ungewohnt ist die Darstellung der Tonhöhe durch einen horizontalen Strich, der Tondauer durch die Länge dieses Striches und der Lautstärke durch einen zugeordneten vertikalen Strich. Doch hat man sich sehr schnell an diese Darstellungsweise gewöhnt und kommt danach gut mit ihr klar. Das Besondere an diesem Editor ist die Möglichkeit, Veränderungen an Tönen, Tempi, Velocity, Quantisierung und vielem mehr auch vorzunehmen, während das Stück abgespielt wird. So hat man jederzeit eine sofortige Kontrolle seiner Aktionen.

Der dritte Hauptbildschirm des Programms ist das Modul »Quickscore«. Es ermöglicht den Ausdruck der MIDI-Daten in normaler Notenschrift. Dabei sind Notenschlüssel, Taktzahlen pro Zeile, Anzahl der Notensysteme und auch Transponierungen für die jeweiligen Instrumente möglich. Die Qualität des Ausdruckes, der auf allen Preferences-Druckern vorgenommen werden kann, ist hoch.

Das gesamte Programmpaket ist für 199,- DM erhältlich. Ein Preis, der der Leistung mehr als angemessen erscheint.

**Info: HS & Y
Classen-Kappelmann-Str. 24
5000 Köln 41
Tel.: 0221/404078
Fax.: 0221/402365**

Sculpt 4D erweitert

Mit einem erweiterten und in deutsch übersetzten Handbuch erscheint dieser Tage das Raytracingprogramm »Sculpt 4D« auf dem Markt. Innerhalb der Software selbst wurde jedoch nichts geändert, Menüs und Benutzerführung sind weiterhin in Englisch. Dabei wurde das Handbuch der amerikanischen Version nicht nur 1:1 übersetzt, sondern auch leicht verständliche Passagen gekürzt und tiefergrei-

fende Aspekte eingehender beschrieben. »Sculpt 4D« wird ab sofort für 998,- DM inklusive beider Handbücher ausgeliefert. Das deutsche Handbuch ist für Interessenten zu einem Preis von 69,- DM auch separat lieferbar.

**Info:
HS & Y
Classen-Kappelmann-Str. 24
5000 Köln 41
Tel.: 0221/404078
Fax: 0221/402365**

In eigener Sache

Aus aktuellen Gründen mußte der Reflections-Kurs auf die nächste Ausgabe verschoben werden. Wir hoffen auf ihr Verständnis.

Ihre Amiga DOS-Redaktion

Tools für Bars&Pipes

Kaum ist das MIDI-Programm »Bars&Pipes« in Deutschland vorhanden, schon gibt es »Power Tools« eine »Werkzeugkiste«, die die Leistungsgrenze nach oben vervielfacht.

Die Erweiterung kann an jede »Pipeline« angehängt werden, ganz wie es benötigt wird. Als Beispiel können Keyboard Splitter, Event-Filter, Randomizer, Transposer, Harmony Generator, Echo, Note-Flipper und viele weitere Hilfsmittel genannt werden. Sollte trotz Vielfalt der Module nicht das richtige dabei sein, so können mit einem Macro-Tool eigene Module kreiert werden.

Ein weiteres Paket »MusicBox A«, das vollkommen in Deutsch gehalten ist, besteht aus 17 Tools. Diese erlauben unter anderem Notenvorschläge, MIDI-Meter, Event-Konverter, Eingangs- und Ausgangskontrollen von Daten, Programmwechsel von Pipelines, Entfernen von »hängengebliebenen Noten«, Senden von ARexx-Scripts, Komprimieren von MIDI-Controller-Daten und eine Einstellung der Farben von Bars&Pipes.

Wer jetzt das Programm »Bars&Pipes« erwirbt, erhält das Internal Sound Kit in voll-

ständiger deutscher Fassung als festen Bestandteil des Grundprogramms. Diese Erweiterung ermöglicht die Benutzung ohne MIDI-Keyboard. Vorteil: Die Eingaben der Noten können mit der Maus oder der Tastatur erfolgen.

Des weiteren werden Sounds und Instrumente auf zwei Disketten mitgeliefert.

Das letzte Paket das angeboten wird, nennt sich »Rules for Tools« und wendet sich an die C-Programmierer.

Die beiliegende Diskette beinhaltet den Source-Code von vielen Beispielen und ein 100seitiges Programmierhandbuch. Leider ist dieses Tool nicht wie die anderen Tools in der deutschen Sprache abgefaßt.

Die Preise für die Module betragen 98,- DM/89,- sfr.

Info:

**Für Deutschland:
DTN-Werbung & EDV
Poststr. 25
D-6200 Wiesbaden
Telefon: 06121/502059**

**Für die Schweiz:
Microtron Computerprodukte
Postfach 69
CH-2542 Pieterlen**



Ein mit »Sculpt 4D« konstruiertes Animationsmodell

Messe in Köln

Vom 25. bis zum 30. Oktober findet in Köln die ORGATEC '90 statt. Auf der internationalen Büromesse stellen mehr als 2000 Anbieter aus 30 Ländern auf einer Fläche über 250000 m² ihre Produkte aus. Die Ausstellungszeiten beginnen um 9 Uhr und enden um 18 Uhr. Die Tageskarte kostet im Vorverkauf 20,- DM und im Kassenvorverkauf 30,- DM. Wem ein Tag nicht ausreicht, der kann sich die Dauerkarte im Vorverkauf für 40,- DM sichern. Wer sich die Karte an der Kasse kaufen will, der muß ganze 60,- DM hinblättern. Auch der Daten- und Medien-Verlag wird mit seinen Zeit-

schriften auf der ORGATEC vertreten sein. Sie finden uns in Halle 2.1, Gang F, auf Stand 42.



**INTERNATIONALE
BÜROMESSE KÖLN
25.-30. Okt. 1990**

Wir stellen aus

HALLE 2.1

GANG F, STAND 42

Nachfolger des Digi-Splitt

RGB-Splitter sind Geräte, die ein Video-Farbsignal in seine Grundbestandteile Rot, Grün und Blau aufteilen und so eine Digitalisierung mit sogenannten Slow-Scan-Digitizern ermöglicht. Diese erfassen ein Bild in drei Durchgängen, in denen jeweils einer der Farbauszüge in den Computer übertragen wird. Der Nachfolger des bekannten RGB-Splitters Digi-Splitt heißt nun Digi-Gen.

Dieses Gerät ist in einem Pultgehäuse untergebracht und wird durch zahlreiche andere Funktionen der Bezeichnung RGB-Splitter eigentlich gar nicht mehr gerecht. Das Gerät kann wesentlich mehr und ist daher auch als Nachfolger des analogen Multiprozessors der gleichen Firma zu sehen. Ebenfalls in analoger Technik aufgebaut, verfügt das Gerät über die Möglichkeit, einen Digitizer vom Typ DigiView 4.0 oder DeluxeView 4.1 einzubauen. Eine getrennte Stromversorgung des Digitizers sorgt dafür, daß weniger Störungen auftreten. Über Schalter und daran angeschlossene Relais kann zwischen Digitizer und Drucker umgeschaltet werden. Letzterer muß dazu auch an das Digi-Gen angeschlossen sein und spart einem das dauernde Umstecken.

Digi-Gen beinhaltet jedoch, wie der Name schon sagt, auch ein Genlock. Diese mit einem Testbildgenerator aus-

gerüstete Baugruppe sorgt dafür, daß Sie Computerbilder mit denen eines Videorecorders oder einer -kamera mischen können. Dabei stehen verschiedene Überblendeffekte und Stanzverfahren zur Verfügung. Wischeffekte können dabei entweder manuell oder in einer vorher definierten Zeit durchgeführt werden. Der integrierte Videocolorprozessor sorgt für die Umwandlung der gängigsten Signale RGB, S-VHS, Hi8 und FBAS des Videobereichs. Neben den Standardbuchsen zum direkten Anschluß entsprechender Recorder bietet Digi-Gen die Möglichkeit der vollen Korrektur von Farbsättigung, Kontrast, Helligkeit sowie der Farbanteile.

Digi-Gen kann somit (und durch die eigene Stromversorgung) auch ohne den Computer als Videoprozessor eingesetzt werden, wenn beispielsweise mit zwei Recordern kopiert werden soll.

Das Gerät ist ab sofort zu einem Einführungspreis von 998,- DM erhältlich, die späteren Marktpreise liegen bei 1298,- DM (ohne Digitizer) und 1698,- DM (inklusive Digitizer).

**Info: PBC,
Peter Biet Computer
Dietershausener Str. 28
6409 Dipperz
Tel.: 06657/8606
Fax: 06657/8605**

Transputer für Amigas

Transputer sind Computer oder Einsteckkarten, die eine unglaubliche Datenmenge in kürzester Zeit verarbeiten können.

Was noch vor kurzer Zeit als unmöglich galt, ist bald schon Wirklichkeit. Als Beispiel: Ein Transputer erreicht eine weit- aus höhere Rechenleistung als ein Amiga mit 68030-Pro-

zessor und einer Taktfrequenz von 50 Mhz.!

Die Transputerkarten wird es für den Amiga 500, Amiga 2000 und dem Amiga 3000 geben.

**Info:
X-Pert Computer Services
GmbH, Weiherwiese 27
D-6270 Idstein/Ts.
Telefon: 06126/3056 (8809)**

Neue Grafikkarten

In der Entwicklung sind neue Grafikkarten, die eine völlig neue Dimension der Grafikdarstellung bieten sollen.

Vorgesehen sind Auflösungen bis zu 1600x1280 Punkte. Die Anzahl der Farben richtet sich je nach Auflösung von 2 bis 4 aus 16 Millionen Farben. Solch eine Karte ist nur durch die al-

lerneuesten Custom-Chips möglich. Die Einsteckkarten sind für den Einbau im Amiga 2000/A2500 und dem Amiga 3000 vorgesehen.

**Info:
X-Pert Computer Services
GmbH, Weiherwiese 27
D-6270 Idstein/Ts.
Telefon: 06126/3056 (8809)**

Lernprogramme für Kinder

Die meiste Software, die auf dem Markt feilgeboten wird, ist nicht für kleinere Kinder geeignet. Um diesem Mißstand abzuweichen, werden von der Software-Firma Database mehrere Produkte angeboten, die Kindern auf spielerische Weise Dinge des alltäglichen Lebens näherbringen.

Die Programme werden für verschiedene Altersstufen angeboten: bis fünf Jahre, für Fünf- bis Siebenjährige und über sieben Jahre. Grafiker, Erzieher und Programmierer haben die Programme so ge-

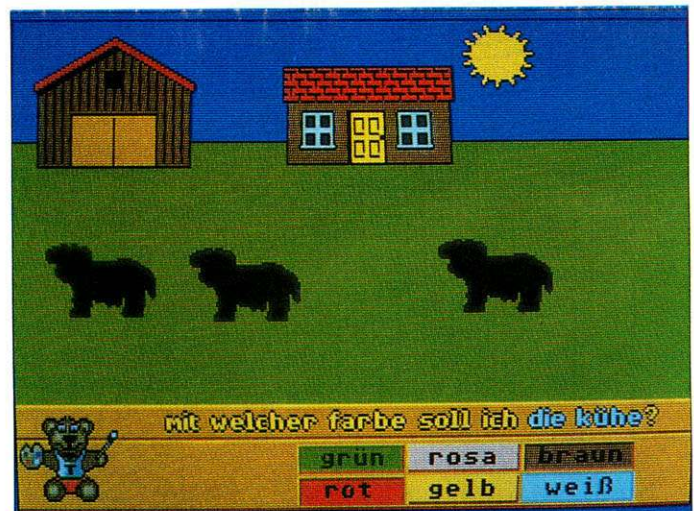
staltet, daß die Phantasie angeregt und schnell ein Lernerfolg erzielt wird.

Themenbereiche sind der Umgang mit Zahlen, Wörtern und die Zuordnung von Bildern.

Bilder und Objekte werden kunstvoll mit Musik und objekt-orientierter Sprache unterstützt.

Die Titel lauten: Fun School 1-3.

**Info:
United Software
Hauptstr. 70
4835 Rietberg**



Welche Farbe hat eine Kuh? Diese Frage wird speziell an Kinder gerichtet

DER SENSATIONELLE DURCHBRUCH...

Amiga ACTION REPLAY ^{✓ 1.5}

NUR
189DM
ZUZUEGLICH
VERSANDKOSTEN

MEHR POWER UND UTILITIES ALS
JE ZUVOR. DAS AMIGA ACTION
REPLAY IST DA!

DIES IST EINE AUSWAHL DER UNGLAUBLICHEN MOEGlichkeiten UND FUNKTIONEN:

**AMIGA ACTION REPLAY EINFACH IN DEN ERWEITERUNGSSTECKPLATZ IHRES AMIGA 500/1000
STECKEN UND SCHON KOENNEN SIE FAST JEDES PROGRAMM FREEZEN.**

● ABSPEICHERUNG DES COMPUTERSPEICHERS AUF DISKETTE

Durch ein spezielles Pack - Verfahren ist es moeglich bis zu drei Programme auf eine Diskette abzuspeichern (ten Programm vierfach schneller wieder einzuladen. (Voellig unabhagig von der Cartridge!))

● EINZIGARTIGER TRAINERMAKER FUER UNENDLICHE LEBEN

Der Trainermaker ist ein Hilfsmittel, mit dem Sie in Ihren Spielen unendlich viele Leben erstellen koennen. Dies war bisher ein sehr schwieriges Unterfangen. Sehr einfach im Gebrauch!

● SPRITEEDITOR

Ein vollstaendiger Spriteeditor macht es moeglich, komplette Sprites anzusehen und zu editieren.

● VIRUS DETECTOR

Umfassende Viruserkennung. Er erkennt alle bekante Viren.

● SPEICHERT BILDER UND AUF DIE DISKETTE

Bilder und Musik werden auf Diskette abgespeichert. Die im IFF - Format abgespeicherten Bilder und Musikstuecke koennen mit den meisten Musik - und Grafikprogrammen verarbeitet werden.

● ZEITLUPE

Jetzt koennen Sie Ihre Programme langsamer laufen lassen. Die Geschwindigkeit ist einfach einzustellen. Ideal als Hilfe bei schwierigen Programmteilen.

● NUESTARTEN VON PROGRAMMEN

Einfach eine Taste druecken und schon laeuft das Programm weiter, wo Sie aufgehoeht haben.

● COMPUTER STATUSANZEIGE

Nach Druecken einer Taste erhalten Sie einen Ueberblick ueber den momentanen Zustand Ihres Computers (Fast RAM, Chip RAM, RAM - Disk, Floppy Status, usw.).

● NEUE FUNKTIONEN V 1.5

Mit der neuen Version Mempeeker kann man den ganzen Computerspeicher auf Bilder durchsuchen und mit vielen extra Funktionen das Bild beeinflussen. Gefreeze Programme koennen jetzt auch auf RAM-Disk abgespeichert werden. RAM-Tester. Automatischer Sprung ins Freezermenu bei illegalem Opcode, so dass man z.B. mit dem naechsten Assembler-Befehl fortfahren kann. Ausgabe von wichtigen Prozessor Exeptions, sowie von TRAP Vektoren.

UPDATE SERVICE

Nach Einsendung Ihrer alten Version, bringen wir es auf den neuesten Stand.

Kosten DM 40,- + Versand.

PLUS DEM UNWAHRSCHEINLICH STARKEN MASCHINENSPRACHE-FREEZER/MONITOR

MEHR MOEGlichkeiten ALS JE ZUVOR!

- Kompletter M68000 Assembler/Disassembler
- Voller Bildschirmditor
- Laden/Speichern von Bloecken
- Schreibe String in den Speicher
- Springe zu einer bestimmten Adresse
- Zeige RAM als Text
- Zeige eingefrorenes Bild
- Spiele residentes Sample
- Zeige und Editiere CPU Register und Flags
- Rechner
- Hilfe Kommando
- Volle Suchmoeglichkeiten
- Der einzigartige Custom Chip Editor erlaubt es Ihnen alle Chipegister anzusehen und zu modifizieren, sogar "Write Only" Register
- Anmerkungen
- Diskettenhandlung - Zeite aktuellen Track, Disketten - Synchronisation, usw.
- Dynamische Breakpoint Behandlung Zeige Speicher als HEX, ASCII, Assembler, Dezimal
- Copper Assembler/Disassembler

**DER STATUS DES EINGEFRORENEN PROGRAMMES INKLUSIVE ALLER REGISTER STEHT UNVERAENDERT
IM SPEICHER IHRES COMPUTERS - WICHTIG FUER DEN DEBUGGER**

WIE BESTELLEN SIE IHR **ACTION REPLAY...**
TEL. - **02822 45589/45923**

ALLE BESTELLUNGEN, AUCH IN DIE DDR, IN 48 STUNDEN LIEFERBAR

EUROSYSTEMS
Hühnerstr. 11, 4240 Emmerich,
Deutschland
Telefax 0031/8380/32146, Tag- & Nacht-Bestellservice

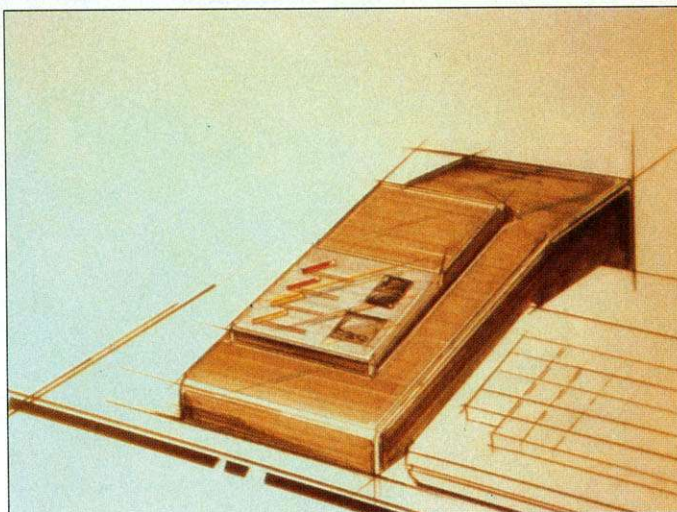
auch erhältlich bei allen ALLKAUF-SB-Warenhäusern und Foto-Fachgeschäften und allen CONRAD-ELECTRONIC-Fillialen sowie bei unseren Fachhändlern

Distributor für BERLIN: **Mükra Datentechnik**, Schöneberger Str. 5,
1000 Berlin 42, Tel.: 030/7529150/60
für ÖSTERREICH: **Computing Zechbauer**, Schulgasse 63, 1180 Wien,
Tel.: 0222/4085256
Rechner-Ring, Grazer Str. 90, 8605 Karpfenberg, Tel.: 03862/24950
für die SCHWEIZ: **Swiss Soft AG**, Obergasse 23, CH-2502 Biel,
Tel.: 032/231833
für HOLLAND: **Eurosystems NL**, Postbus 179, 6710 BD Ede,
Tel.: 085/516565

A500 mit Rucksack

Letzte Meldung von der Firma bsc büroautomation: Auch Besitzern des "kleinen" Amiga stehen demnächst alle Möglichkeiten des A2000 zur Verfügung. Die Technik der Universal 500 Box steht fest, für das Design existierte zum Redaktionsschluß erst eine Studie.

Das Gehäuse wird an den Expansionbus angesteckt und bietet Platz für 2 bis 8 MByte Speichererweiterung und zwei Zorro-Karten in voller Baulänge. Die Anwendungsmöglichkeiten sind zahlreich. A.L.F.2- und A.L.F.3-SCSI-Controller können ebenso untergebracht werden wie jede A2000-Erweiterung. Wer also irgendwann



Eine Design-Studie der A500-Erweiterungsbox

die Anschaffung eines A2000/3000 plant, kann jetzt schon in die Zukunft investieren und Teile kaufen, die später weiterverwendet werden können. Der Preis wird bei zirka 750 DM liegen.

In letzter Minute! Neue Mitteilung von der Firma bsc: das Erweiterungsgehäuse wird jetzt »Multi Mega Box« heißen und einige Änderungen erfahren. Der Preis bleibt der gleiche.

Info:
bsc büroautomation AG
Lerchenstraße 5, Aufgang II
8000 München 50
Tel. 089/3084152
Fax 089/3071714

Universelles CLI-Tool

Für Einsteiger, die sich nicht mit dem umfangreichen Befehlssatz des CLI herumschlagen wollen, bietet der Markt & Technik-Verlag einen AmigaDOS-Manager an.

Mit der mausorientierten Oberfläche lassen sich alle AmigaDOS-Funktion aufrufen, wie Abspielen von IFF- und Sonix-Sounds, Darstellen von Bildern mit Color-Cycling, Editor, Text-Display in Hex und ASCII, schnelles File- und Disk-Kopierprogramm, Formatieren

von Disketten, File Recover, das gelöschte Programme wieder ins Computerleben ruft. Wer Platz, Zeit und Geld sparen will, der bekommt noch zusätzlich eine Routine zum Packen und Entpacken von Dateien dazu.

Alles, was Sie brauchen, ist ein Amiga mit 512 kByte und 69,- DM für das Programm.

Info:
Markt&Technik
Hans-Pinsel-Str. 2
8013 Haar

Neue parallele Schnittstelle

Von der Firma BSC gibt es eine Multi I/O-Karte, die mit vier Eingabe-/Ausgabe-Ports ausgerüstet ist, wobei beim Einsatz von mehreren Karten bis zu 22 Ports angesteuert werden können. Diese Karte dient zum Anschluß der unterschiedlichsten Peripheriegeräte und unterstützt alle Standard-Baudraten inklusive MIDI 31250 (Musical-Interface) auf den seriellen Schnittstellen. Und erstmalig zwei parallele Schnittstellen für Drucker, Digitizer, Digi View Gold und so weiter.

Die herkömmliche mechanische Umschaltung entfällt

durch ein Treiberprogramm, das die Ausgabe auf alle Schnittstellen gleichzeitig ermöglicht.

Durch den halblangen Aufbau der Karte bietet sich noch genügend Platz für eine File-Card oder ähnliches.

Der Preis von 578,- DM schließt zwei Kabelsets, Treiber-Software, ein ausführliches deutsches Handbuch und eine technische Dokumentation ein.

Info:
BSC Büroautomation AG
Lerchenstr. 5
8000 München 50
Telefon: 089/3084152

Amiga-World in Wien

Vom 27. bis zum 30. September geht im Wiener Messepalast der Amiga ab.

Dort werden die allerneuesten Soft- und Hardware-Produkte ausgestellt, wie ein 1.52-Megabyte-Laufwerk, das unter AmigaDOS formatiert werden kann, Raytracing-Programme, die wesentlich schneller sind als »Sculpt 4D« oder »Turbo-Silver«. Auch die Video- und Musikfans kommen nicht zu kurz, ihnen werden die neuesten Sampler und Framegrabber geboten. Ge-

plant sind auch Musik- und Videoshow. Also für jeden ist etwas dabei.

Die Öffnungszeiten: 9 bis 18 Uhr.

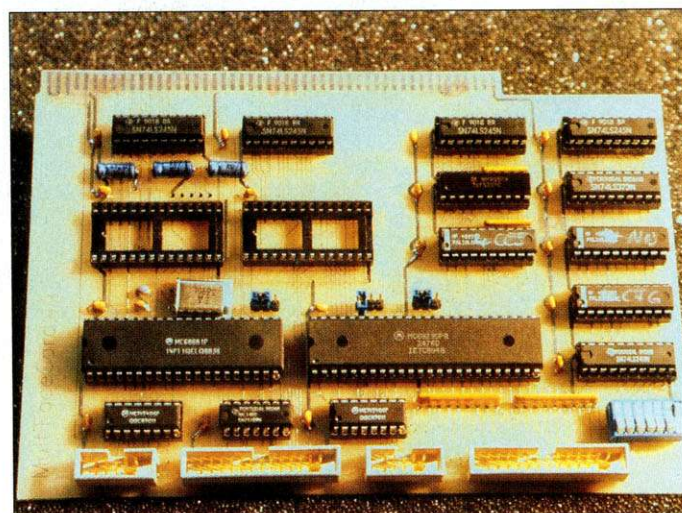
Die Tageskarte kostet 100,- ÖS, für Jugendliche bis 14 Jahre: 60,- ÖS

Info:
ECI Expoconsult International
Veranstaltungs- und Beratungs-Gesellschaft mbH
Messeplatz 1
A-1071 Wien
Tel.: 0222/931524-229

Korrektur

Die Diskette zum bei Markt & Technik erschienen Buch »Amiga-Hardware-Tuning« wurde überarbeitet. Senden Sie die Original-Diskette zusammen mit einem an Sie adressierten

und frankierten Rückumschlag (1,70 DM) an Uwe Gerlach, Brühlstr. 23, D-6440 Bebra 1. Bei Sendungen aus dem Ausland sind zwei internationale Antwortscheine beizulegen.



Halblange Schnittstelle für Peripheriegeräte

Auf geht's zur

AMIGA

KÖLN

90

**Unter der
Schirmherrschaft
von Commodore
und dem
Amiga Magazin**

09.11. - 11.11.90

**Köln Messe,
Halle 10 und 12**

(08.11.90 Fachbesuchertag)

Öffnungszeiten:

**08.11.90/Fachbesuchertag:
10:00 - 18:00 Uhr**

**09.11. - 11.11.90:
9:00 - 18:00 Uhr**

**Tageskarte 09. 11. - 11. 11. 90
Schüler/Studenten: DM 10.-
Erwachsene: DM 15.-**

**Fachbesucherkarte: DM 35.-
(Gültigkeit: alle vier Messetage)**

**Pro Messe erwarten Sie
über 100 Aussteller und
30000 AMIGA-Fans!**

**Kommen Sie vorbei,
wir freuen uns auf Sie!
Das Reiseunternehmen
Giller Reisen wird Ihnen bei
der An-, Abreise und Unter-
kunftssuche in Köln
gerne behilflich sein:**

Giller Reisen GmbH
Frau Gruber
Tel.: 089-6132002
Fax: 089-6135019

Die Messe wird ein Hit, das weiß ich. Deshalb bestelle ich
schon jetzt zu Vorabpreisen ☐ Tageskarten à DM 10 und schlage
dem Veranstalter damit ein Schnippchen, weil's sonst teurer wäre.
Ein Scheck liegt meiner Bestellung bei.
Bitte einsenden an: AMI Shows Europe, Zugspitzstraße 2a, D-80711 Vaterstetten

Name _____
Straße _____
Ort _____
Einsendeschluß: 10. 10. 1990
(Datum d. Poststempels)

MIDI in Farbe

Bisher war der MIDI-Anwender auf dem Amiga auf "spartanische" Benutzeroberflächen angewiesen, wenn er mit professionellen Sequenzern arbeitete. Daß es auch anders geht, beweist »Bars & Pipes«.

MIDI ist die Abkürzung für "Musical Instruments Digital Interface" und bedeutet übertragen digitale Schnittstelle für Musikinstrumente. Durch den MIDI-Standard ist es möglich, verschiedene Musikgeräte unterschiedlicher Hersteller miteinander kommunizieren zu lassen. Jedes Instrument, das seine Töne auf digitale Weise erzeugt, kann in ein MIDI-System integriert werden. Ein ebenfalls mit dem System verbundener Computer kann die Daten von allen angeschlossenen Geräten lesen und bearbeiten, sofern die entsprechende Software vorhanden ist. Ein Programm, das die Notenwerte der einzelnen Geräte mitschneidet, wird Sequenzer genannt.

Dem MIDI-Anwender war es bisher versagt, komfortable und übersichtliche Programmoberflächen zu bedienen, wie sie von Programmen wie »Soundtracker« oder »Quartet« her bekannt sind. Meist gestaltete sich die Benutzeroberfläche sehr schlicht, und

es waren kaum mehr als vier Farben zu erwarten. »Bars & Pipes« von Blue Ribbon Bakery geht nicht nur hier neue, unkonventionelle Wege. Dieser MIDI-Sequenzer präsentiert sich mit einem Hauptmenü, das dem Bedienungspult einer Tonbandmaschine ähnelt und vollen Überblick über die "Bandspuren" gestattet. Doch neben den optischen Novitäten sind auch in der Musikprogrammierung, Verwaltung und Modifizierung einige Punkte hinzugekommen, die dem MIDI-Musiker die Arbeit wesentlich erleichtern.

Objektorientiertes Komponieren

»Bars & Pipes« setzt sich aus fünf Hauptkomponenten zusammen. Zunächst ist dort die »Pipeline«. Alle MIDI-Daten müssen diese Pipelines durchlaufen, bevor sie an das MIDI-Gerät (Synthesizer, Drumcomputer und so weiter) gesendet oder von diesen empfangen werden. Jeder Pipeline kann ein spezieller Kanal zugeord-



net werden, so daß mehrere MIDI-Geräte oder ein multitimbraler (mehrstimmiger) Synthesizer angesteuert werden können. Um MIDI-Daten zu beeinflussen, bevor sie an den entsprechenden Kanal gesendet werden, stehen die sogenannten Tools (Werkzeuge) zur Verfügung, die die zweite Komponente von »Bars & Pipes« darstellen. Diese Tools können nach Belieben mit Hilfe der Maus in jeder einzelnen Pipeline eingesetzt werden. Die ausgehenden MIDI-Daten werden je nach Art des Tools verändert. So gibt es zum Beispiel ein Quantisierungs-Tool, das die Noten auf genaue Taktpositionen ausrichtet, oder ein Harmonie-Tool, das harmonische Obertöne passend zur Note generiert.

Werkzeuge selbst erstellen

Die Tools sind wohl die überzeugendste Eigenschaft von »Bars & Pipes«. War der Anwender bisher auf die im Sequenzer fest installierten Werkzeuge angewiesen (Quantisierung, Echo oder ähnliches), so können bei »Bars & Pipes« sämtliche benötigten Modifizierer selbst entworfen werden. Dazu steht die Toolbox zu Verfügung, die es erlaubt, auch "exotische" Werkzeuge aus den bereits vorhandenen zu kreieren. Solche, aus bereits vorhandenen Tools zusammengesetzte Werkzeuge, werden als "Makro-Tools" bezeichnet. Die Anzahl der Tools und Makro-Tools richtet sich lediglich nach dem Platz auf Ihrer Diskette beziehungsweise Harddisk. Im Lie-

ferumfang von »Bars & Pipes« befinden sich einige dieser Tools. Accessories (Zubehör), die dritte Komponente, ähneln den Tools in vielen Punkten. Auch sie sind lediglich durch den Disketten- oder Festplattenspeicher begrenzt. Im Gegensatz zu den Tools können die Accessories nicht selbst editiert werden. Sie stellen einzelne Hilfsprogramme dar, die von Blue Ribbon Bakery ständig erweitert und auf Zusatzdisketten erworben werden können. Die zwei mitgelieferten Accessories heißen »Muffy« (speichert Bars-&-Pipes-Songs im MIDI-File-Format) und »Follow the leader« (druckt Akkorde und Texte).

Sequenzer – das Tonband im Amiga

Der Sequenzer zeichnet im Aufnahme-Modus jede eingehende MIDI-Information (MIDI-Event) auf. Die Notenauflösung beträgt 192 Clocks (Zeiteinheiten) pro Viertelnote. Eine ganze Takt besteht also aus 768 möglichen Notenpositionen.

»Bars & Pipes« ist in der Lage, SMPTE-Codes zu senden und zu überwachen. SMPTE-Codes werden benötigt, um Mehrspur-Tonbandmaschinen zeitgenau mit MIDI-Geräten laufen zu lassen (Overdubbing). SMPTE-Code kann aber auch für andere Anwendungen genutzt werden. In Diskotheken oder bei Live-Konzerten wird gelegentlich SMPTE-Code benutzt, um die Lichtanlagen zu steuern. Dazu wird der SMPTE-Code auf eine spezielle Spur

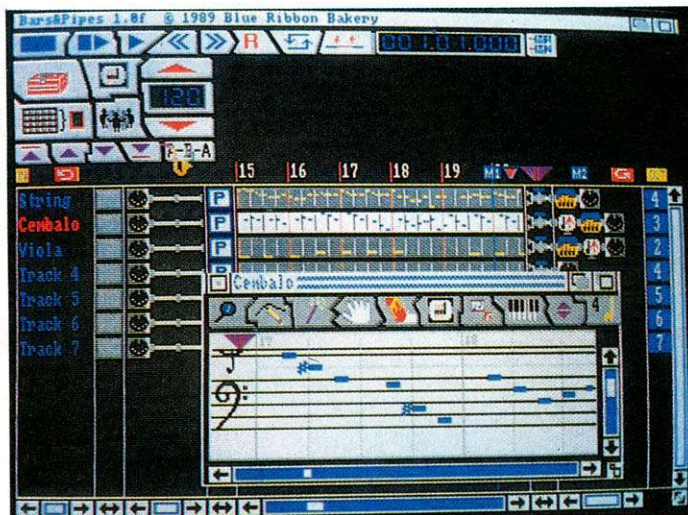


Bild 1. Komfortables Editieren im Trackmenü



InterComputing Deutschland Inc.

Schönebecker Str. 55-57
Telefon 0202/89155

5600 Wuppertal-2
Telefax 0202/89304

USA: InterComputing, Inc.

2112 Sandy Lane
Dallas, TX 75220

Telefon: 214-556-9666
Telefax: 214-556-2336

Frankreich: InterComputing France

34, Avenue des Champs Elysees
75008 Paris

Phone: (1) 42821603
FAX: (1) 42806649

MEMORY

8-UP Karte-2MB	699.-DM
8-UP Karte-4MB	999.-DM
8-UP Karte-6MB	1399.-DM
8-UP Karte-8MB	1799.-DM
Starboard II für A1000	
mit 512K	499.-DM
mit 1MB	649.-DM
M501 Erweiterung mit 512K/ batteriegepufferter Uhr	179.-DM

VIDEO

Digi View 4.0	329.-DM
Flicker Fixer (PAL)	1199.-DM
Vid Tech Scanlock	1999.-DM

LAUFWERKE/ FESTPLATTEN

Internal 3.5 laufwerke für A2000	199.-DM
Cutting Edge MAC laufwerke für AMAX	449.-DM
Quantum Hardcard 40MB	1299.-DM
Quantum Hardcard 80MB	1999.-DM
HardFrame SCSI Kontroller	499.-DM
GVP SCSI Kontroller mit raum für 8MB Speicher	499.-DM
Supra Wordsync Karte	399.-DM



Supra Corporation

GVP 68030 KARTEN

GVP 68030 - 28 68882 / 4MB	4799.-DM
GVP 68030 - 33 68882 / 4MB	6399.-DM
GVP 68030 - 50 68882 / 4MB	8999.-DM

SOFTWARE

3D Professional	745.-DM
Turbo Silver	249.-DM
Disney's Animation Studio	299.-DM
Saxon Publisher	649.-DM
Scene Generator	79.-DM
Pro Write 3.0	299.-DM
Deluxe Paint III	249.-DM
Lattice 5.0	499.-DM

AMIGA 500

Supra 40MB Quantum	1599.-DM
Supra 80MB Quantum	2199.-DM

AMIGA 1000

Supra 40MB Quantum	1699.-DM
Supra 80MB Quantum	2299.-DM

ICD ADRAM 540 mit 4MB

Ersetzt die reguläre 512K Erweiterung in Ihrem AMIGA 500!
999.-DM

Händler- und Auslandsanfragen erwünscht!

Bestellungen bitte schriftlich oder telefonisch an eines unserer Büros. Lieferung erfolgt gegen Vorkasse zzgl. DM 5.- Versandkosten oder per Nachnahme zzgl. DM 10.- Versandkosten. Kompletter Katalog gegen DM 3.50 in Briefmarken.

*Sie können diese Produkte auch
bei den folgenden Händlern beziehen:*

BERLIN:

Goldvision • 030 / 88 33 505
Prolinea • 030 / 26 18 387

WUPPERTAL: Hifi-Studio 9 GmbH • 0202 / 55 50 53

eines "normalen" Tonbandes oder Kassettenrecorders geschrieben und ausgelesen. Der Sequenzer führt alle Anweisungen korrekt aus, jedoch ist ein kontinuierlicher Loop-Modus (ständige Wiederholung von Tracks, zum Beispiel für Schlagzeug) nicht möglich. Ein oder mehrere kurze Tracks können höchstens achtmal wiederholt werden. Dieser Punkt ist aber auch das einzige Manko von »Bars & Pipes«.

Aufgezeichnete MIDI-Informationen können im Editor verändert, entfernt, "geklebt" oder kopiert werden. Hier können die MIDI-Events auch genauer "unter die Lupe" genommen werden. Wenn einer der Tracks (Spuren) des Sequenzers angeklickt wird, öffnet sich ein Bildschirmfenster, das vollen Zugriff auf alle Daten dieses Tracks erlaubt. Diejenigen Benutzer, die des Notenlesens mächtig sind, können die MIDI-Events in einer der Notenschrift ähnlichen Schreibweise editieren. Doch auch an "Uneingeweihte" wurde bei »Bars & Pipes« gedacht: Die Daten sind gleichzeitig nach einem Prinzip aufgezeichnet, das als "Pianostreifen" eines automatischen Klaviers bezeichnet werden kann. Hierbei zeigt sich der Vorsatz von Blue Ribbon Bakery, »Bars & Pipes« sowohl für professionelle Anwender als auch für interessierte Einsteiger zu programmieren.

Einfache Installation

Eine herausragende Eigenschaft von »Bars & Pipes« soll besondere Erwähnung finden: Das Programm ist in keiner Form kopiergeschützt. Sicherheitskopien oder die Installation auf Festplatte stellen also keine Probleme dar. Dank dem umfangreichen und sehr gut verständlichen Handbuch gestaltet sich diese Installation selbst für unerfahrene Amiga-Benutzer denkbar einfach. Durch den Symbolcharakter der einzelnen Bedienungselemente des Programms wird ein hoher Lerneffekt erzeugt, der dem Anwender ein ständiges Nachschlagen im Handbuch erspart, da er sich durch die logischen Symbole schnell an die Funktion des entsprechenden Schalters erinnert.

Durch das offene Prinzip von »Bars & Pipes« werden Erweiterungen des Programms ohne Schwierigkeiten möglich. Blue Ribbon Bakery bietet bereits ein "Internal Sound Kit" an, das es ermöglicht, »Bars & Pipes« ohne MIDI-Geräte zu betreiben und auf die internen Amiga-Sounds (Samples) zuzugreifen. Auch neue Tools und Accessories sind bereits erhältlich.

Erweiterungen ohne Ende

Daneben bietet Blue Ribbon Bakery auch Bücher an, die es dem erfahrenen Anwender erlauben, noch nicht vorhandene Tools und Accessories in der Sprache C zu programmieren. Auch MIDI-Anwender, die der englischen Sprache nicht unbedingt mächtig sind, können aufatmen: »Bars & Pipes« ist in einer Version erhältlich, in der sowohl Programm als auch Handbuch in Deutsch gehalten wurden. »Bars & Pipes« ist ein MIDI-Sequenzer und Editor, dessen Benutzeroberfläche als beispielhaft bezeichnet werden kann. Beim Test konnten keine Fehler festgestellt werden, alle MIDI-Daten wurden korrekt bearbeitet, und auch im Multitasking-Betrieb kam es zu keinen Funktionsstörungen. Einziges Manko ist die Größe des Programms: »Bars & Pipes« benötigt mindestens 1 MByte Rechner-Speicher.

Kleines MIDI-Glossar

Wie in den meisten Bereichen von Computeranwendungen existiert auch bei MIDI ein Fachjargon, mit dem der MIDI-Einsteiger wenig anfangen kann. Aus diesem Grund haben wir ein kleines MIDI-Glossar zusammengestellt, das die wichtigsten Fachbegriffe erläutert.

MIDI – Abkürzung für "Musical Instrument Digital Interface", eine genormte Schnittstelle, die Noten oder SysEx-Daten verschiedener Geräte überträgt.

SysEx – Abkürzung für "System Exclusive Daten". Verschiedene Synthesizer haben verschiedene Klangparameter, die einen Sound beeinflussen können. Da diese Daten gerä-

tespezifisch sind, werden sie "systemexklusiv" genannt.

Track – Ein Sequenzer zeichnet eingehende MIDI-Daten in einem Track auf, der in etwa mit einem Stück Tonband vergleichbar ist. Die einzelnen Tracks sind nach Kanalnummern getrennt und können kombiniert und aneinandergelangen werden.

MIDI-Event – Jede Form von MIDI-Daten, die nicht systemexklusiv sind, werden als Events (zu deutsch: Ereignis) bezeichnet. Dazu gehören zum Beispiel die »Note on/off«-Befehle, »Pitchbend« (Tonhöhenänderung) oder »Program Change« (umschalten auf eine andere Klangeinstellung).

Aftertouch – Die meisten neueren Synthesizer können erkennen, ob nach dem Anschlag einer Taste "nachgedrückt" wird. In den Klangparametern kann ein bestimmter Effekt (zum Beispiel Pitchbend) auf diesen Aftertouch gelegt werden.

Velocity – Bezeichnet die Geschwindigkeit des Tastenanschlags (Attack Velocity) oder des Loslassens der Taste (Release Velocity). Anschlagsdynamische Synthesizer können diesen Wert erkennen und entsprechende Effekte auf den Klang legen (zum Beispiel Lautstärke oder Hüllkurven).

Sequenzer – Der Sequenzer ist ein Programm oder eine dafür vorgesehene Hardware, die MIDI-Daten aufzeichnet und Manipulationen erlaubt. So können zum Beispiel Notenwerte oder Tracks editiert und verändert werden. Der Sequenzer kann diese MIDI-Daten wieder zum MIDI-Gerät zu-

rücksenden. Somit ist es möglich, jedes Instrument eines Musikstücks einzeln in den Sequenzer zu spielen und anschließend zu kombinieren.

Loop – ist die ständige Wiederholung eines oder mehrerer Tracks. So kann durch Aufnahme eines Schlagzeug-Tracks durch »loopen« eine kontinuierliche Wiederholung des Schlagzeugs über das ganze Musikstück erreicht werden.

(tb)

AMIGA DOS Blitz ★ licht

Name: Bars & Pipes
Funktion: MIDI-Sequenzer
Hersteller: Blue Ribbon Bakery Inc.
Vertrieb: Atlantis Software
Dunantstraße 54, 5030 Hürth
Tel.: 02233/41081
Fax: 02233/46266
DTM Werbung & EDV, Poststraße 25, D-6200 Wiesbaden, Tel: 06121-502059
Preis: 498,- DM

Positiv:

- offenes Konzept
- vorbildliches Handbuch
- übersichtliche Bedienung
- leicht erlernbar

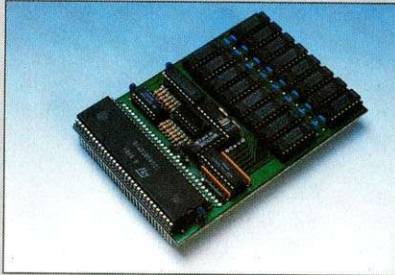
Negativ:

- großer Speicherbedarf (1 MByte)
- kein echter Loop-Modus



Bild 2. Makro-Tools selbst entwerfen

ROSSMÖLLER...



A 8MB/1000 598 DM

8 MByte Ramkarte; mit 2 MByte bestückt
kinderleicht einzubauen, nicht löten – nur einstecken
abschaltbar, ohne Waitstates



3 1/2" Disk-Drive extern 239 DM

5 1/4" Disk-Drive extern, 40/80 Tracks 279 DM

Unsere Floppy-Stationen sind ausschließlich mit Qualitäts-
laufwerken der Firmen NEC und TEAC ausgestattet. Alle
Laufwerke sind abschaltbar; der Bus ist durchgeführt
zum Anschluß weiterer Drives.

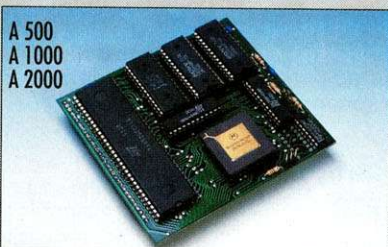
Filecards für AMIGA® 2000 und externe Festplatten für AMIGA® 500
mit Metallgehäuse und eigenem Netzteil. Selbstverständlich sind
unsere Festplatten autobootend, ALF-formatiert und ALF-installiert!

Amiga 500/1000		Amiga 2000	
20 MB	998 DM	20 MB 40ms	898 DM
30 MB	1089 DM	30 MB 40ms	998 DM
40 MB 28ms	1298 DM	49 MB 40ms	1398 DM
50 MB 40ms	1398 DM	40 MB 19ms SCSI	1498 DM
60 MB 28ms	1698 DM	60 MB 28ms SCSI	1598 DM
A1000 – Aufpreis 120 DM		83 MB 28ms SCSI	1798 DM



386-si Power Board 1298 DM

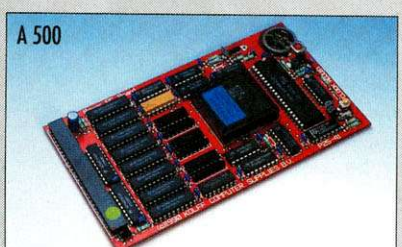
verwandelt Ihre XT-Karte ruckzuck in einen 386-SX
16 MHz Taktfrequenz und 16 KByte CACHE-Speicher
für höchste Geschwindigkeit: macht die XT-Karte
bis zu 12x schneller (NORTON SI-Faktor 11,5)!
Steckplatz für 387-SX-Coprozessor



TORNADO 498 DM

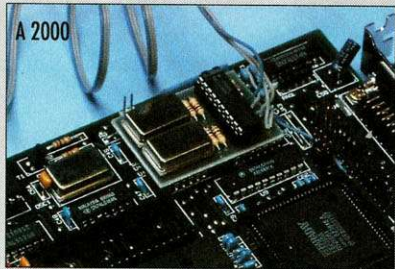
Jetzt geht für Ihren AMIGA® 500/1000/2000
die Post ab:

TORNADO macht ihn doppelt so schnell!
Und mit Arithmetik-Prozessor (optional) ist
Ihr AMIGA® sogar bis zu 20x schneller!



Power PC-Board 798 DM

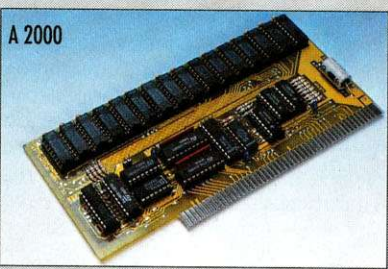
Der Geheimtipp: Ihr AMIGA® 500 IBM-kompatibel!
Alle AMIGA® Ein- und Ausgänge werden unterstützt:
Maus, Joystick, int.+ext. (auch 5,25 Zoll) Laufwerke,
par./ser. Schnittstelle, 1 MByte Ram und Uhr auch für
den AMIGA® 500 ansprechbar, NORTON SI-Faktor 3,3



TURBO-XT 199 DM

macht die Commodore XT-Karte doppelt so schnell
(XT-Karte nicht im Lieferumfang)

XT-RAM 768 k 298 DM
erweitert Ihre XT-Karte ON BOARD auf 768 KByte!



A 8MB/2000 598 DM

8 MByte Ramkarte, 2 MByte bestückt;
abschaltbar autokonfigurierend, 0 Waitstates,
4 MBit-Technologie.

Jetzt auch 68020/30-Kompatibel (A 2630)

Für AMIGA® 500/1000/2000

BTX-Interface für MULTITERM-Software	138,00 DM
BTX-Interface für COMMODORE-Software	138,00 DM
VESUV Eprom-Programmiergerät für die Eproms 2716-27512, 27513, 27011	199,00 DM
"HAPPY" 3/89 Gesamturteil "SEHR GUT"	
DigiSmooth Grafiktablett f. A500/1000	748,00 DM
Auflösung 1000 Pixel/Zoll; f. A2000	698,00 DM
Medusa, Atari ST-Emulator f. A500/2000	498,00 DM
Soundsampler, Profi-Modell	89,00 DM
MIDI-Interface für höchste Ansprüche	94,00 DM
Lightpen für Sculpt 3D/4D, DPaint etc.	79,95 DM
3-fach Kickstart-Umschaltplatine	59,95 DM
TRACK-DISPLAY 2000 für 2 HD's + 2 FD's	168,00 DM

...WAS SONST!

Innovativ, ausgereift, für preisbewußte Anwender – das ist Roßmüller Computer Technik. HiTec-Qualität made in
Germany (1 Jahr Garantie), die mit jedem Geldbeutel kompatibel ist. Mit perfektem Service: So stehen Ihnen an
der Telefon-Hotline von Montag bis Freitag, 16-17 Uhr, die Entwickler unserer AMIGA-Produkte Rede und
Antwort. So nehmen wir Ihre telefonischen Bestellungen an 365 Tagen im Jahr; rund um die Uhr unter der
Nummer 0203/5195130, persönlich entgegen! Na, überzeugt? Dann fordern Sie unseren Gratiskatalog an!

ROSSMÖLLER
COMPUTER TECHNIK

Roßmüller Handshake GmbH • Neuer Markt 21 • 5309 Meckenheim • Tel. 02225/2061-62-63

AMOS, der Gewaltige

Wer tolle Spiele programmieren wollte, jedoch außer BASIC keine Sprache beherrschte, stand bislang auf dem Trockenen – bis AMOS kam, der Action-Interpreter.

Falls Sie gerade mal schlecht drauf sind, und sich zusätzlich noch einen Tiefschlag verpassen wollen, dann versuchen Sie doch mal in AmigaBASIC ein IFF-Bild zu laden!

Das war eben kein Scherz. Meine ersten Programmierversuche auf dem damals frisch-erworbenen Amigä mit dem mitgelieferten BASIC blieben bei genau diesem Punkt hängen. Im Handbuch stand so gut wie gar nichts darüber. Weiterführende BASIC-Literatur erklärte mir dafür, daß ich auf "Bodies" und "Chunk-Flags" zu achten hätte, wobei das IFF-Ladeprogramm die Länge einer mittelgroßen Datenbank erreichen würde und daß in Assembler sowieso alles besser geht. Na danke!

Wird einem soviel Unentwirrbares beschert, ist das schon ein Düsseldorf Uraltbier wert. Um der Traurigkeit und der programmlosen Zukunft zu entgehen, wendet man sich schweren Herzens anderen

Programmiersprachen zu, mit der Folge, daß der benötigte Seelentrost wächst (wobei nicht gesagt wird, daß Assembler- oder C-Programmierer frustrierte Typen sein müssen). Hätte es denn nicht einfacher sein können?

AMOS kam, sah und siegte

AMOS hatte auf dem Atari ST einen Vorgänger, das »STOS«, das für Furore sorgte. Beim Amiga blieb ich angesichts des mitgelieferten BASIC skeptisch, bis dann die Packung auf meinem Schreibtisch lag, mit der Bitte, es ausführlich zu testen. AMOS als die Spieleprogrammiersprache schlechthin war erst mal nicht denkbar. Doch probieren geht bekanntlich über studieren, also wurde erst mal nachgeschaut, was das Handbuch so versprach.

AMOS ist eine Interpretersprache, die auf einem eigenen

Editor erstellt wird. Beim ersten Test erlebte ich gleich die erste Überraschung; der Editor erinnerte mich an meine C64-Zeiten. Aufgebaut wie ein normaler Texteditor läßt sich jedes Zeichen beliebig erreichen und verändern. Viele integrierte Menüpunkte, die mit der Maus aufrufbar sind, geben dem Ganzen tatsächlich einen professionellen Rahmen. In einer Statuszeile werden Textzeile und -spalte, Textspeicher, Chipmem und Fastmem sowie der Programmname angezeigt. Speicheranpassung ist bei AMOS kein Problem, jeder zusätzliche RAM-Bereich kann mit dem AMOS-Programm belegt werden. Irgendwie kommt einem jetzt schon der Gedanke, daß AmigaBASIC eigentlich so hätte aussehen sollen. Die im Editor zu sehenden Funktionen reichen auf jeden Fall aus, um einen auf den Rest neugierig zu machen.

Der Editor enthält wie auch AmigaBASIC Blockoperationen; den Unterschied findet man aber in der Geschwindigkeit: Während AmigaBASIC bei »CUT« und »PASTE« zwischendrin zweiwöchige Weltreisen erlaubt (Das war eine glatte Übertreibung!), benimmt sich der Editor wie ein Computerprogramm. Per Maus werden die Blöcke markiert und per Menü verschoben, entfernt, versteckt, oder was auch immer. »Suchen und Ersetzen« ist für AMOS ebenfalls kein Problem, wie Groß- und Kleinschreibung. Wer Editoren wie den »Cygnus Ed« kennt, weiß so etwas zu schätzen. AMOS kann seine Pro-

gramme als Quelltext oder ASCII abspeichern; nachzuladende Programmteile (mit »MERGE«) müssen auch unter AMOS als ASCII abgespeichert werden. Wenn man eine Bewertung abgeben sollte, dann würde AMOS im Gegensatz zum Platzhirsch AmigaBASIC jetzt schon nach Punkten siegen. Aber noch ist kein Programm geschrieben, die Skepsis bleibt noch ein bißchen.

Von »ABS« bis »ZOOM« ist alles da

Grundsätzlich ist AMOS ein BASIC-Interpreter wie AmigaBASIC oder GFA-BASIC auch; wichtige Unterprogramme wie

```
FOR Zeile=1 TO 20
LOCATE 1,Zeile PRINT "Hallo
Welt"
REM (Würg)
NEXT Zeile
```

lassen sich ohne ins Handbuch zu schauen leicht realisieren. Die Standardbefehle beherrscht AMOS ohne weiteres, Konvertierungen von anderen BASIC-Interpretern sollten selbst Neulingen ohne größere Probleme gelingen. An Variablen verarbeitet AMOS einmal Vier-Byte-Integerzahlen im Wertebereich von -147483647 bis +147483648, Realzahlen mit 4 Byte Länge im Wertebereich von $1 \cdot 10^{-14}$ bis $1 \cdot 10^{15}$ sowie Stringvariable. Alle Variablennamen können bis zu 255 Zeichen lang sein, wer also Variablen von der Art »Dies_ist_meine_liebste_Variable_die_ich_gerne_einsetze\$« mag, dem

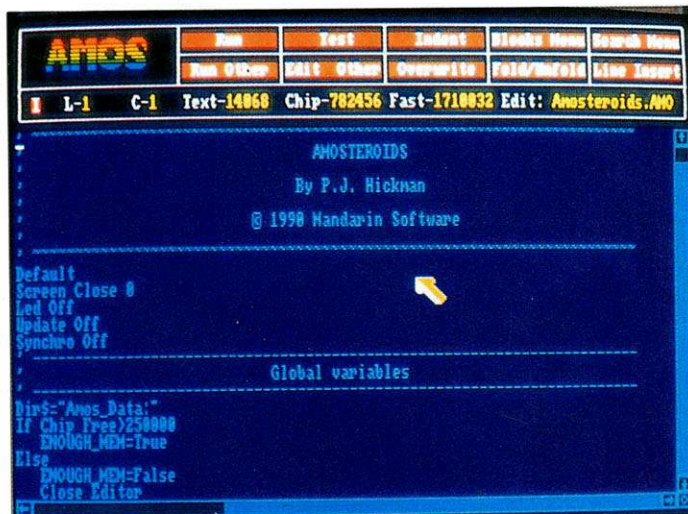


Bild 1. Der AMOS-Editor verdient seinen Namen zu Recht



Bild 2. Eines der Beispielprogramme von AMOS

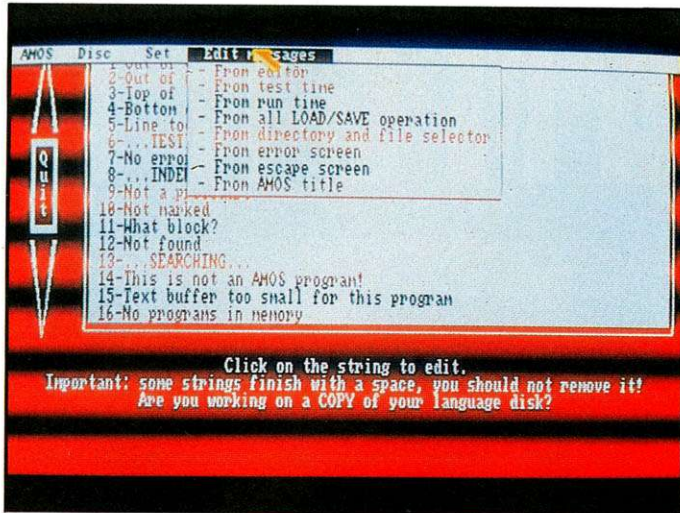


Bild 3. Das Konfigurationsprogramm von AMOS

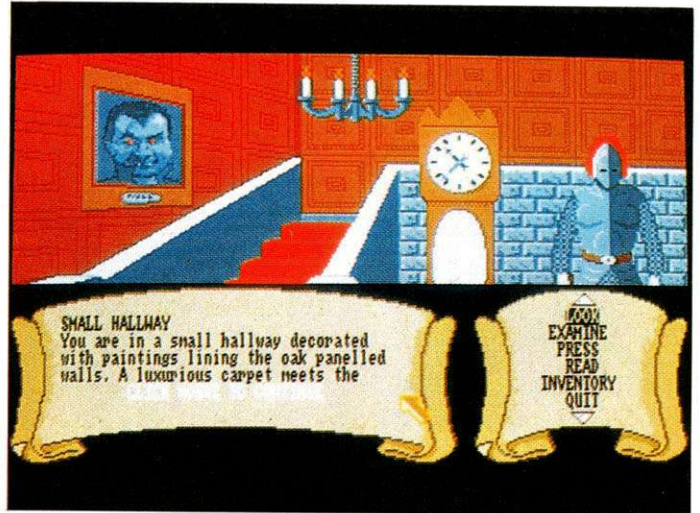


Bild 4. Ein mit AMOS erstelltes Adventure

sei es gestattet. Werte für Variablen oder Konstanten können dezimal, hexadezimal oder dual (binär) sein, AMOS-BASIC läßt diese drei Möglichkeiten zu.

Logische Verknüpfungen können in Befehlsfolgen vom Typ »IF...THEN...ELSE...ENDIF« stattfinden, aber auch als pure Bitberechnung eingesetzt werden. Zeilen wie

NOT(z)=x AND y

sind also möglich. Auch zwei- oder dreidimensionale Arrays werden von AMOS-BASIC verwaltet. Man sieht also, AMOS ist im Grunde ein BASIC-Dialekt wie andere auch – wenn da nicht die speziellen Befehle wären.

AMOS zielt auf die Spieleprogrammierer, und daher sind gerade der Grafik, den Blitter Objects (BOBs) und den Sprites, aber auch dem Sound mächtige Befehle reserviert, die teils einen, teils mehrere Parameter benötigen. Sich das Hirn verrenken wie bei manch anderer Programmiersprache, braucht man bei AMOS allerdings nicht; die Parameter haben eine "natürliche" Form, den bekannten Zeiger auf den Zeiger, der auf den Zeiger zeigt, sucht man hier vergebens.

Fangen wir mit den Farben an, hier geht der Fortschritt allerdings schon los: »INK«, »PAPER«, »PEN«, »COLOUR«, »PALETTE« und die Funktion »x=COLOUR«, die die gerade aktuelle COLOUR-Farbe wiedergibt, bringen die Pixel farbig auf den Bildschirm. Text im Grafikbildschirm läßt sich durch »GR LOCATE« und

»TEXT x,y,(STRING)« genau positionieren und darstellen. Befehle zum Zeichnen von Linien, Flächen, Boxen gehören ebenfalls dazu, wie das Auslesen der Farbe einzelner gesetzter Punkte. Die Maus- und Joystick-Positionen werden mittels spezieller Funktionen ausgelesen, Maus- oder Joystick-Tasten ebenfalls. Ach ja, gehen wir zurück zum Anfang; das Laden einer fertigen IFF-Grafik kennt man vom Amiga-BASIC bisher als (programmier-) zeitraubende Aktion; bei AMOS dagegen reicht ein kleiner Einzeiler:

LOAD IFF "Grafikname"

Es lassen sich zum Beispiel Zonen einrichten, die bei Erreichen durch den Joystick oder die Maus abfragbar sind. Wer bis jetzt Probleme hatte, Spielsteine mittels Maus zu bewegen, der kann sie mit AMOS vergessen:

RESERVE ZONE 1
SET ZONE 1,0,0 TO 10,10

Diese beiden Befehle erledigen die Speicherplatzreservierung und das Anlegen einer Zone im Bereich der Koordinatenpunkte x1,y1 und x2,y2. So einfach geht das.

Die einzelnen Befehle gehen nach kurzer Zeit schnell von der Hand, BASIC ist eben immer noch die am schnellsten erlernbare Sprache. Was allerdings verwundert, ist die Geschwindigkeit, in der der Interpreter die Programme abarbeitet. Um eine Warteschleife einzurichten, verfiel der Autor dieser Zeilen in die Unart, eine For-Next-Schleife mit dem Wert 2000 einzurichten (»FOR zeit=1 TO 2000:NEXT zeit«).

Von AmigaBASIC ist man ja gewohnt, daß man in der Zwischenzeit den Hund Gassi führen kann, aber diese Schleife dauerte in AMOS noch nicht einmal eine Sekunde. Auch die Grafikbefehle werden blitzartig ausgeführt. Woher AMOS diese Geschwindigkeit nimmt? Nun, es ist in Assembler programmiert, aber das kann eigentlich nicht alles sein. Wahrscheinlicher ist, daß die BASIC-Befehle ohne unnötigen Ballast abgearbeitet werden.

AMOS-BASIC ist BASIC zum Liebhaben

Wer AMOS erwirbt, hat bestimmt keinen schlechten Handel gemacht. Wer außerdem bisher vor der Spieleprogrammierung kapitulierte, hat besonderen Grund einmal in AMOS reinzuschneppen. Das AMOS-BASIC ist hervorragend geeignet, eigene Spielvorstellungen in die (Computer-) Realität umzusetzen. Und die erstellten Programme brauchen sich nicht hinter denen anderer Sprachen zu verstecken, wie die verschiedenen Beispiele zeigen. Zum AMOS-BASIC wird es später auch noch einen Compiler geben, so daß die Programme in kompilierter Form lauffähig auf Diskette gebracht werden können (auf die dann vorhandene Geschwindigkeit darf man jetzt schon gespannt sein!). Wer AmigaBASIC kennt, aber nicht mag, sollte erst recht AMOS ins Auge fassen, hier findet er alles, was er bei AmigaBASIC

entbehren muß. Zu bemängeln gibt es eigentlich nichts, nur, daß Zeilen des Grafikbildschirms ab und zu im Editor auftauchen, dort allerdings keine Auswirkung haben.

(jb)

AMIGA DOS Blitz ★ licht

Name: AMOS the Creator
Hersteller: Mandarin
Funktion: BASIC-Interpreter
Quelle: GTI GmbH,
Zimmersmühlenweg 73,
6370 Oberursel
Preis: 129,- DM

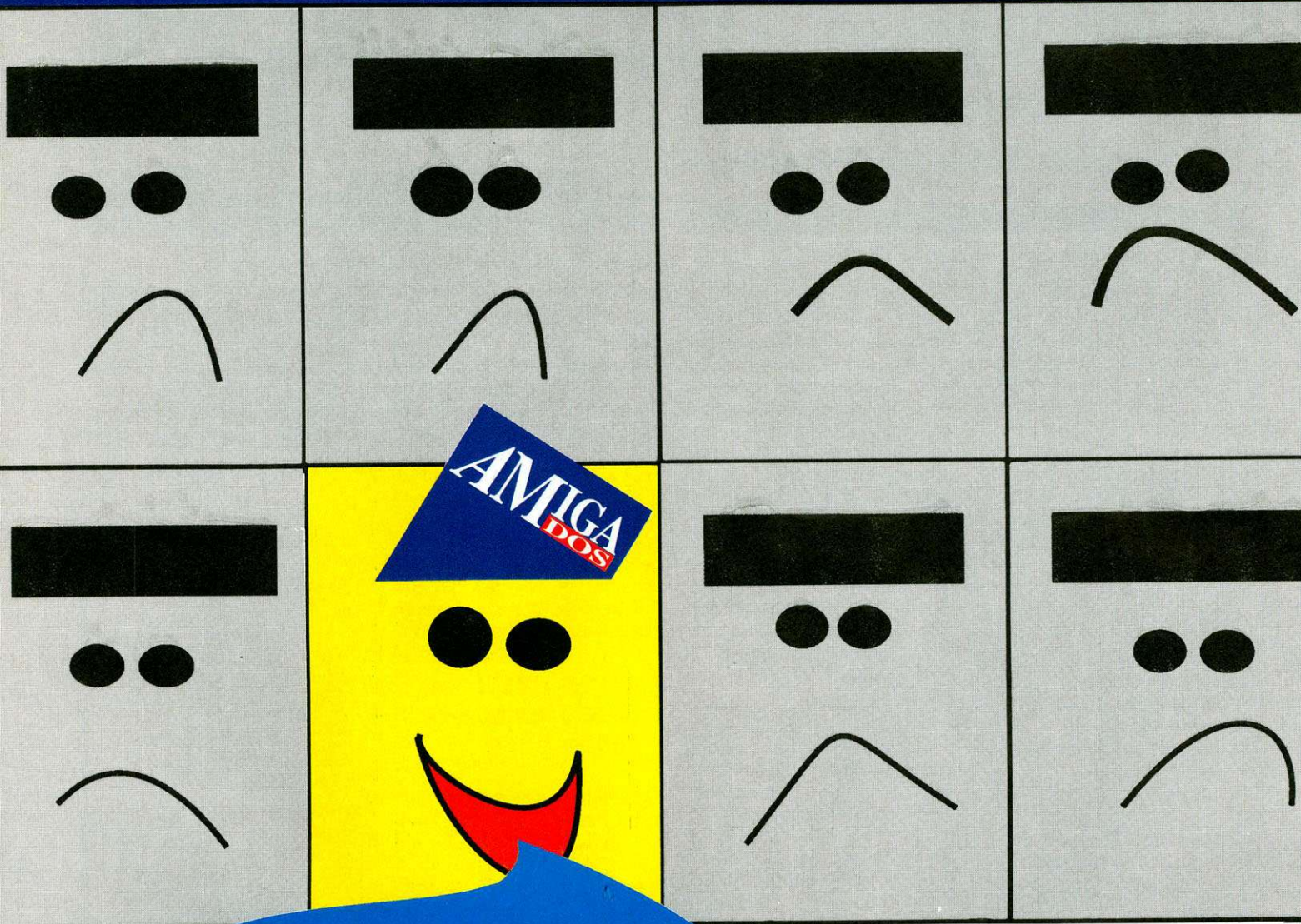
Positiv:

- sehr komfortabler Editor
- normaler BASIC-Sprachschatz implementiert
- mächtige Grafik- und Sound-Befehle (betrifft auch Sprites und BOBs)
- erzeugt schnellen BASIC-Code
- sehr schnell erlernbar
- IFF-Grafiken ohne umständliche Programmierung einbindbar
- enthält komplexe Copper- und Blitterbefehle
- Libraries nutzbar
- gut gegen Abstürze gefeit
- Syntaxprüfung

Negativ:

- Dokumentation nur ausreichend, spezielle Befehle nur unzureichend beschrieben
- Fehler bei der Copper-Programmierung werden nicht abgefangen

ABONNEMENT



**INFORMATIONEN AUS
ERSTER HAND**

**Ein Abonnement ist
praktisch und bequem.**

Widerrufsrecht

Jeder Abonnent hat das Recht, seine Bestellung innerhalb einer Woche beim DMV-Verlag, Postfach 250, 3440 Eschwege, schriftlich zu widerrufen. Die rechtzeitige Absendung des Widerrufs Schreibens genügt zur Fristwahrung.

**AMIGA
Köln 90**
Wir stellen aus
09.-11. November 1990
Halle 12, Stand 307

**AMIGA DOS
kostet im Abonnement:**

Im Inland bzw. West-Berlin:

6 Ausg. = 35,- DM

12 Ausg. = 70,- DM

Im europäischen Ausland:

6 Ausg. = 50,- DM

12 Ausg. = 100,- DM

Im außereuropäischen Ausland:

6 Ausg. = 60,- DM

12 Ausg. = 120,- DM

Bitte benutzen Sie die Bestellkarte.

DMV-Verlag • Postfach 250 • 3440 Eschwege

DMV
Daten- und
Medienverlag

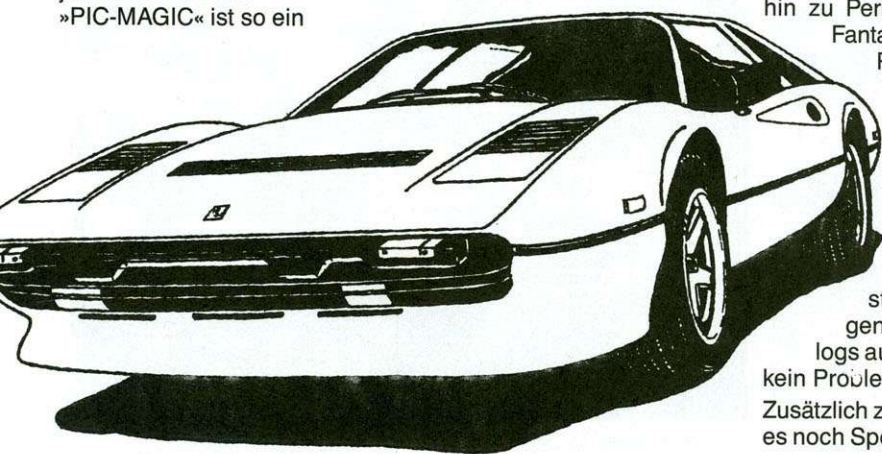
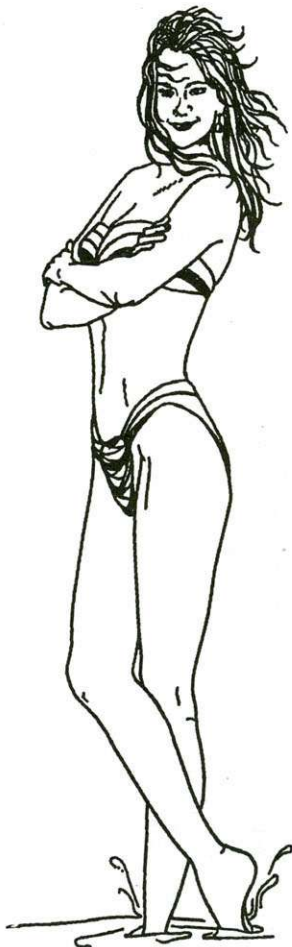
Galerie auf Diskette

Was tun, wenn die eigene Zeichenkunst bei der Gestaltung von DTP-Seiten versagt? Schauen Sie doch mal bei »PIC-MAGIC« nach...

Wir haben schon einige Desktop-Publishing-Programme für den Amiga vorgestellt und Ihnen gezeigt, was mit diesen Programmen von der gestalterischen Seite her alles möglich ist. Nur ist nicht jeder ein kleiner van Gogh, die eigenen Zeichenkünste enden meistens beim Zeichnen von Familienmitgliedern, die sich selbst nicht mehr wiedererkennen. Ohne Zeichnungen, ob witzig oder originell, zum Thema passend oder nur schmückend, ist eine Layout-Seite so trocken wie zehn Tage altes Brot.

In diese Lücke passen seit einiger Zeit sogenannte Clip-Art-Disketten, randvoll mit kleinen Zeichnungen rund um sämtliche Themen, die sich anbieten. Schon in der Public Domain finden sich diverse Sortimente mit teils guten, teils mittelmäßigen Ergebnissen. Wer hier sichergehen will, Zeichnungen und Grafiken für sein Geld zu bekommen, die den eigenen Qualitätsanforderungen entsprechen, der muß sich schon bei professionellen Clip-Art-Paketen umschauen, wenn er nicht "Kanten- und Ecken-Klecksel" in sein Projekt einsetzen will.

»PIC-MAGIC« ist so ein



professionelles Clip-Art-Paket. Auf insgesamt zehn Disketten befinden sich die verschiedensten Zeichnungen in teils hervorragender Qualität.

Für jeden Geschmack etwas

Um die Disketten gebrauchen zu können, reicht ein Amiga mit 512 kByte aus. Alle Grafiken sind im IFF-Format vorhanden und können von den handelsüblichen Grafikprogrammen übernommen werden. Sie erstrecken sich aber über den kompletten Grafikbildschirm mit der größten Auflösung und sind somit zum Beispiel unter »Deluxe Paint« nicht komplett auf dem Bildschirm sichtbar. Diese hohe Auflösung ermöglicht allerdings genaueste Zeichnung, über die Qualität der Grafiken können Sie sich anhand der abgedruckten Beispiele ein Bild machen. Sie wurde mit einem HP-Laserjet-kompatiblen Laserdrucker mit einer Verkleinerung von zirka 1:4 ausgedruckt.

Was findet man nun auf den Disketten? Angefangen von verschiedenen Pflanzen- und Tierzeichnungen geht es weiter über Sport und (teils martialisches) Fahr- und Flugzeuge bis hin zu Personenportraits und Fantasy-Produkten. Zum

Paket wird ein Katalog ausgeliefert, in dem sämtliche Grafiken mit den entsprechenden Diskettennummern verzeichnet sind. Bestimmte Zeichnungen anhand des Katalogs auszusuchen, das ist kein Problem.

Zusätzlich zum Hauptpaket gibt es noch Spezialpakete für Fan-

tasy-Grafiken (zwei Stück), ein »Weddings Package one« für die, die's schon sind oder mit Hilfe des Amigas werden wollen, ein »Business-Package« und noch andere. Diesen Paketen liegen zwar die Disketten bei, nicht jedoch ein Katalog wie beim Hauptpaket, sondern nur eine Aufstellung.

Clip-Art und Layout-Programm passen zusammen

Die Clip-Art-Disketten von "Joe's First Company Inc.", so der Name des Herstellers, bieten eine recht gute Mischung verschiedener Grafiken zu verschiedenen Themen in einer teils sehr guten Qualität. Die Grafiken sind teilweise als EPS-Grafiken (Encapsulated PostScript) erhältlich und somit direkt in die entsprechenden Layout-Seiten von Desktop-Publishing-Programmen einsetzbar. Der Preis erscheint mit zirka 200 DM ein bißchen hoch angesetzt, hier verhindert nur die gebotene Qualität aufkommenden Ärger über den Preis. Die Pic-Magic-Disketten sind angesichts des Preises eigentlich damit nur dem ernsthaften Anwender von DTP- oder Präsentationsprogrammen zu empfehlen, obwohl Sie eigentlich sehenswert sind.

(jb)

AMIGA DOS

Blitz★licht

Name: PIC MAGIC

Hersteller: Joe's First Company Inc.

Funktion: Clip-Art-Grafik-Disketten

Quelle: Pulsar, Erlanger Str. 8-10, 5000 Köln 91, Tel.: 0221/873359 (873657 od. 874321) Atlantis, Postfach 1141, D-5030 Hürth, Tel.: 02233/41081

Preis: zirka 170 bis 200 DM

Positiv:

– teilweise sehr gute Qualität

Negativ:

– hoher Preis

Das Programm »TurboPrint Professional« ist keine Erweiterung des »TurboPrint II«, sondern als eigenes Programm gedacht, welches einfach mehr kann als der »kleine Bruder«. »TurboPrint II« wird weiterhin verkauft, natürlich wird aber auch ein Upgrade von »TurboPrint II« auf »TurboPrint Professional« angeboten. Da »TurboPrint II« bereits in der Ausgabe zwei vorgestellt wurde, gehen wir hier nicht mehr auf die Funktionen ein, die dort schon vorhanden sind.

Was ist »TurboPrint Professional« eigentlich? Das Programm ist ein Ersatz für die Druckerschnittstelle. Die Workbench-Druckertreiber sind alle recht langsam und nicht sehr leistungsfähig. Ist »TurboPrint Professional« installiert, so wird die im Programm eingebaute, softwaremäßige Schnittstelle benutzt und nicht die der Workbench. Somit übernimmt das Programm die Funktionen, die eigentlich die Workbench-Druckertreiber sonst übernehmen. Diese werden durch die neuen Treiber ersetzt, die sehr viel leistungsfähiger sind und dem Anwender mehr Beeinflussungen des Ausdrucks gestatten. Der Ausdruck kann mit Hilfe des Programnteils »TurboPrefs« beeinflusst werden. Hier befinden sich auf vier Menüseiten vielfältige Einflußmöglichkeiten. Im Menü »Drucker« muß zuerst der passende Druckertreiber ausgewählt werden. Dabei kann der Anwender zwischen verschiedenen Druckqualitäten wählen. Bei den meisten Matrixdruckern kommt es vor,

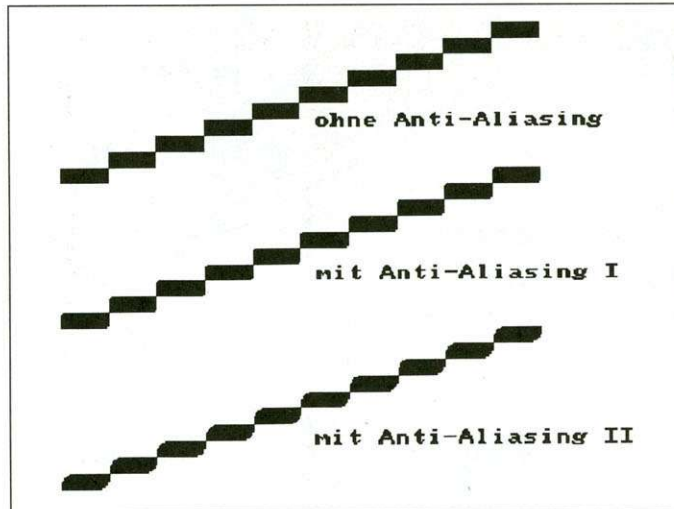


Bild 1. Auswirkung der Funktionen von »Anti-Aliasing«

Andreas Polk

Drucken wie ein Profi

IrseeSoft ist bekannt für gute Druck-Utilities, angefangen bei »TurboPrint« bis zur Nachfolgerversion »TurboPrint II«. Seit kurzem ergänzt »TurboPrint Professional« dieses Angebot.

daß das Bild streifig wird. Das kann daran liegen, daß der Zeilenvorschub nicht exakt arbeitet. Ist dies der Fall, kann mit dem Gadget der Zeilenabstand erhöht und verringert werden. Könnte so der Effekt nicht unterbunden werden, hat man noch die Möglichkeit,

mit Hilfe des »2/3-Druckes« ein besseres Ergebnis zu erzielen. Dabei wird nur mit 2/3 der Nadeln gedruckt, was erstens stromsparend ist und zweitens (wichtiger!) bei manchen Druckern zum besseren Ergebnis führt. Das beste Ergebnis erreicht man mit dem Halb-

zeilendruck. Hierbei werden die Zeilen immer unvollständig gedruckt, dafür wird der Zeilenvorschub aber verringert. Der Ausdruck dauert dann zwar länger, das Ergebnis ist aber sehr gut. Bei Schwarzweiß-Druckern müssen Farben in Graustufen umgewandelt werden, was durch Rasterung geschieht. »TurboPrint Professional« bietet dem Anwender nun insgesamt 14 Muster. Somit hat er immer genug Wahlmöglichkeiten und sollte immer ein Muster finden, das ihm gefällt. Im zweiten Menü mit dem Namen »Preferences« können einige Grundeinstellungen vorgenommen werden. So kann hier das Format und die Art des Ausdrucks (schwarzweiß, grau, Farbe) eingestellt werden. Hinzugekommen ist eine neue Anti-Aliasing-Funktion. Wird sie aktiviert, so werden auch nicht-diagonale Linien geglättet. In Bild 1 sehen Sie die Auswirkung der beiden Anti-Aliasing-Funktionen.

Auch die Traktorbreite kann eingestellt werden. So werden nun ebenfalls DIN-A3-Drucker unterstützt. Durch Anwählen von PAR2 wird das »parallel.device« umgangen und durch ein eigenes Device ersetzt. Dies bringt nicht nur Geschwindigkeitsvorteile, sondern spart auch Speicherplatz. Auf der dritten Menüseite können Sie alle »Special-Funktionen« einstellen. Neben den schon bekannten »Hardcopy« und »NoFastMem« kann hier auch der Posterdruck aktiviert werden. »TurboPrint Professional« zerlegt dann das auszudruckende Bild in mehrere ein-

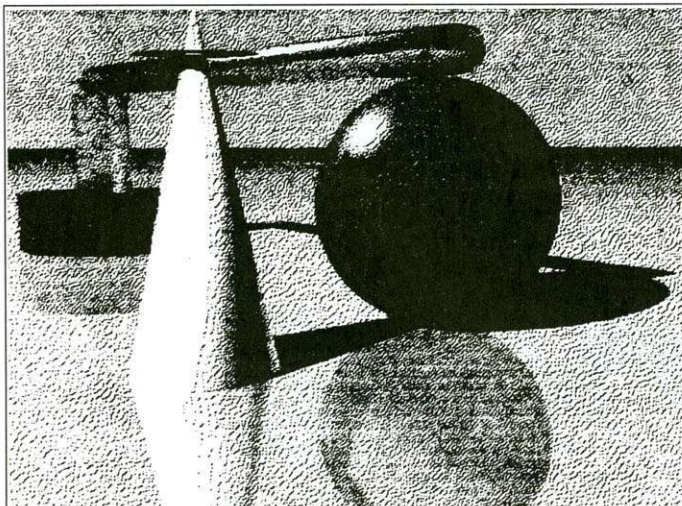


Bild 2. Ein Ausdruck mit 360*360 dpi und Halbzeilendruck

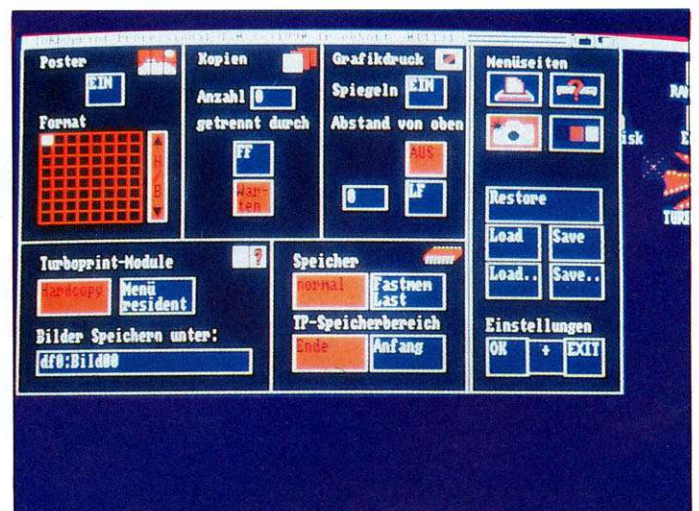


Bild 3. Das Menü »Special«

stellbare Stücke und druckt diese dann aus. Die einzelnen Teilbilder können später dann zu einem Poster zusammengeklebt werden. Schön ist, daß das Ausdrucken eines Teilbildes wiederholt werden kann, wenn etwas nicht geklappt hat. So muß nicht der gesamte Ausdruck neu gestartet werden, was doch sehr zeit- und kostenaufwendig wäre. Neu ist auch, daß ein Bild automatisch mehrmals hintereinander ausgedruckt werden kann. Die Bilder werden dann wahlweise durch ein Form-Feed getrennt, oder direkt untereinander gedruckt. Weiterhin kann eine Funktion aktiviert werden, die bewirkt, daß vor jedem Druck eine einzustellende Anzahl an Line-Feeds ausgeführt werden. So werden Grafiken nie direkt untereinander gedruckt. Möchten Sie gespiegelte Ausdrücke anfertigen, so können Sie dazu den Spiegeldruck aktivieren. Das ist dann sinnvoll, wenn Sie Drucke anfertigen, die hinterher auf T-Shirts aufgebügelt werden. Das vierte Menü namens »Farbumsetzung« ist das wichtigste. Hier kann der Anwender die Ausdruckqualität manuell beeinflussen. So findet der Anwender eines Schwarzweiß-Druckers einen Regler für die Gamma-Korrektur. Sie bewirkt, daß das Bild entweder heller oder dunkler wird. Mit den vier Reglern rechts daneben können Kontrast und Farbwahl bestimmt werden. So ist es mit diesen Reglern sogar möglich, ein Bild invertiert auszu- drucken. Natürlich kann dies aber auch als Extra-Menü- punkt aktiviert werden. Besitzer von Farbdruckern haben

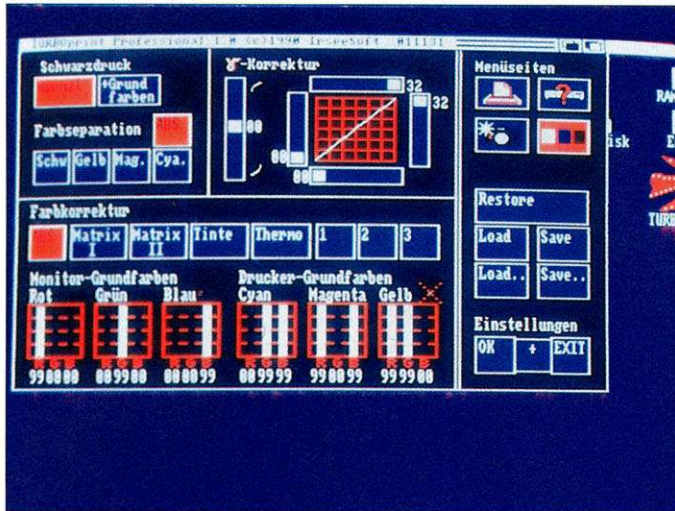


Bild 4. Das Menü »Druckeranpassung«

hier noch mehr Möglichkeiten. So können nämlich mit den vielen Reglern in der unteren Bildschirmhälfte die Druckfarben an die Monitorfarben oder umgekehrt anpassen. Die Einstellungen sind zwar recht umfangreich, dank der beiden mitgelieferten Demobilder (eins für Schwarzweiß- und eins für Farbdrucker) artet dies allerdings nicht in Arbeit aus. Insgesamt gibt es schon vier Voreinstellungen, die zuerst einmal ausprobiert werden sollten. Drei eigene Einstellungen sind speicherbar. So muß man die Regler nicht immer wieder neu einstellen. Beim Starten wird immer eine Grundeinstellung geladen. Diese kann geändert und wieder abgespeichert werden. Beim nächsten Mal wird Sie dann automatisch aktiviert. Auch andere Einstellungen können abgespeichert und später per Hand geladen werden.

Jeder, der Grafiken ausdrucken muß, mit Textverarbeitungen arbeitet oder die Texte als Grafik zu Papier bringen will (zum Beispiel »Excellence«), wird mit »TurboPrint Professional« wesentlich effektiver arbeiten.

Wer braucht »TurboPrint Professional«?

Grafiken wurden bisher noch nie so gut zu Papier gebracht wie mit »TurboPrint Professional«. Allerdings erreichen Schriften immer noch nicht das Aussehen, welches gute Schriften haben sollten. Hier muß man entweder auf die teure PostScript-Ausgabe zurückgreifen, oder DTP-Programme nutzen. Durch »TurboPrint Professional« werden die Schriften zwar erheblich besser, sie erreichen aber noch lange nicht die bestmögliche Qualität. Schade ist

nur, daß »TurboPrint Professional« beim Start immer nach der Originaldiskette fragt. Erst gegen Einsendung der Registrierkarte und einer Gebühr von 10,- DM erhält man ein Programm mit Passwort. Die Originaldiskette wird dann nicht mehr gebraucht und kann sicher verwahrt werden. Ein Hinweis noch: Wie bereits eingangs erwähnt, wurden hier nur die Neuheiten von »TurboPrint Professional« beschrieben. Solche Dinge wie die eingebauten Hardcopy-Funktionen oder der Aufruf des Menüs durch einen »Hot-key« besitzt auch schon »TurboPrint II«.

»TurboPrint Professional« kann eben alles, was der »kleine Bruder« auch kann – und sehr viel mehr.

(jb)

AMIGA DOS Blitz ★ licht

Name: TurboPrint Professional
Preis: 189,- DM (10,- DM für die Passwort-Version)
Anbieter: IrseeSoft,
Grüntenstr. 6, 8951 Irsee,
Tel.: 08341/74327

Positiv:
– sehr flexibel mit allen Druckern
– schnell
– viele Zusatzfunktionen (Hardcopy, »LastMemLast«...)
– gutes Handbuch

Negativ:
– hoher Preis
– Kopierschutz

Eine Bitte an unsere Abonnenten

Vermerken Sie bei Schriftverkehr und Zahlungen neben der vollständigen Anschrift stets Ihre Abo-Nummer.

Sie vermeiden damit unnötige Verzögerungen bei der Bearbeitung Ihres Abonnements.

Vielen Dank

Ihre DMV-Versandabteilung



1180 Wien, Schulgasse 63

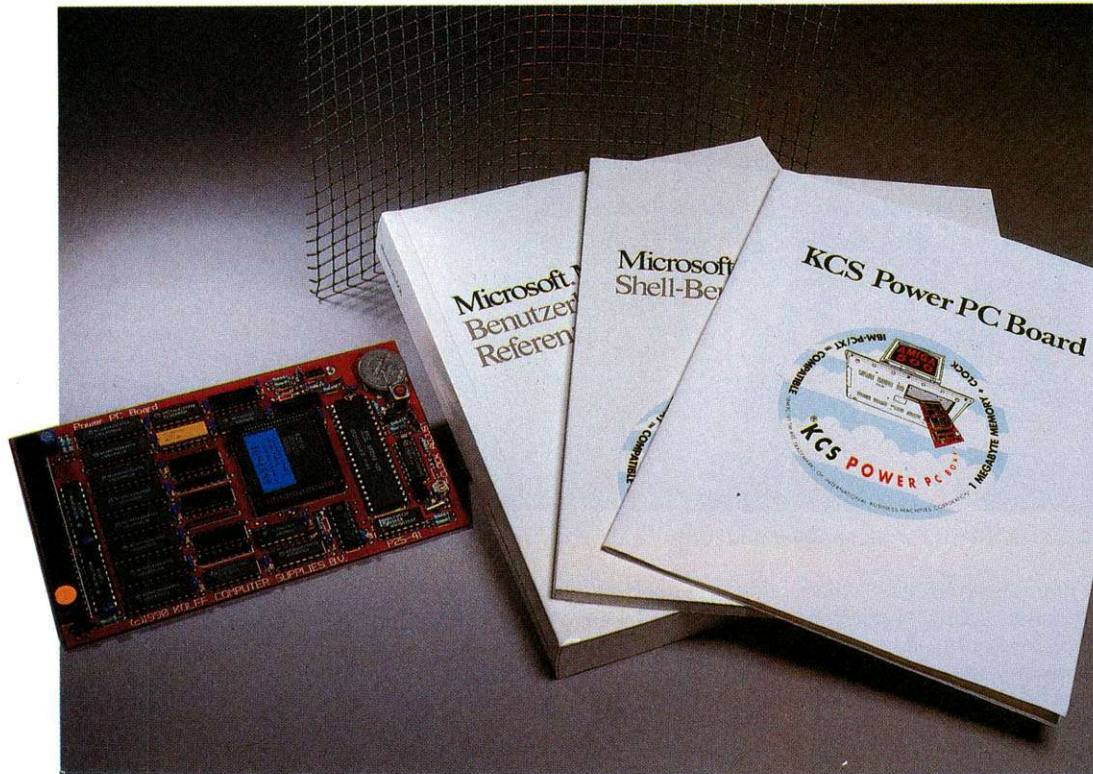
Tel: (0 222) 408 52 56

Telefax: (0 222) 408 99 78

Postversand - Teilzahlung - Leasing

Amiga 500 Speichererweiterung 512 KB + Uhr	öS 1.490,- (DM 213,-)
Amiga 5.25 Zoll Laufwerk, Bus/Ein- u. Ausschalter, 40/80 Tr.	öS 1.990,- (DM 285,-)
Amiga 2000, 2 MB Memory	öS 4.990,- (DM 713,-)
Amiga 2000 XT-Karte inkl. Laufwerk	öS 5.990,- (DM 856,-)
Amiga 2000 XT-Turbo mit 8 MHz	öS 1.590,- (DM 227,-)
80 MB SCSI Harddisk / 24 ms, für A 500 komplett	öS 15.990,- (DM 2285,-)
Amiga Ersatzmaus	öS 590,- (DM 85,-)
Control-Center 500 (Top-Styling!)	öS 1.490,- (DM 213,-)
Digi View 4.0 Gold	öS 2.690,- (DM 385,-)
Genlock Pal Ver 1.3	öS 4.990,- (DM 713,-)

Eurosystems (Midi-Manager, Pro Sampler, Syncro Express, Handyscanner) lagernd
GVP (45 MB 28 ms, 40 MB 19 ms, 80 MB 19 ms, 68030 Karte,...) lagernd



Robert Marz

MS-DOS auf dem Kleinen

Bislang war die Emulation eines MS-DOS-Rechners nur den "großen" Amigas mittels XT- beziehungsweise AT-Brückenkarte vorbehalten.

Eine XT-Emulation auf dem Amiga 500? Ich stand der Karte zunächst sehr skeptisch gegenüber. Aber schon der Lieferumfang überzeugte mich vom Gegenteil. Neben der Karte und der dazugehörigen Software wird das komplette MS-DOS 4.01 samt Handbuch und Shell mitgeliefert.

Auch der Aufbau der Karte überzeugt. Sie ist sehr kompakt und beherbergt einen NEC-Processor des Typs "V30", der zum Intel 8086 voll kompatibel ist, insgesamt 1 MByte RAM und eine batteriegepufferte Uhr. Die Karte wird wie eine Speichererweiterung in den Port auf der Unterseite des Amiga gesteckt, und als solche fungiert sie auch im normalen Betrieb. Obwohl von dem einem MByte nur 512 kByte als Erweiterung genutzt werden können, bleibt der restliche Speicher nicht ungenutzt. Er steht nämlich als

RAM-Disk jederzeit zur Verfügung. Mit der eingebauten RAM-Karte und der Uhr gab es während der gesamten Testphase keinerlei Probleme.

Der Amiga 500 als MS-DOS-Computer

Hält man beim Booten der Startdiskette den linken Mausknopf gedrückt, kommt man in ein Menü, das dem Benutzer erlaubt, die Befehle zum Installieren der RAM-Disk und der Uhr in der »Startup-Sequence« seiner Boot-Diskette zu integrieren. Des weiteren kann man hier das Programm »PC-Preferences« starten, mit dem die Karte konfiguriert wird. Aber dazu später mehr. Will man die Karte in ihrer Hauptfunktion nutzen, nämlich als XT-Emulator, muß dieser Teil zunächst aktiviert werden. Dies geschieht durch Booten mit der Startdiskette. Schon bald bekommt man

dann den im MS-DOS-Bereich üblichen Memory-Check zu sehen, und ihr Amiga ist ein vollwertiger XT mit 704 kByte nutzbarem Schreib-/Lesespeicher.

Bei den MS-DOS-Rechnern ist, im Gegensatz zum Amiga, keine fest eingebaute Maximalauflösung für den Bildschirm vorgegeben. Die grafischen Möglichkeiten werden hier, wie vieles andere auch, durch eingesetzte Erweiterungskarten bestimmt.

Die Grafikkarten

Trotz seiner Fähigkeiten ist der Amiga allerdings nur in der Lage, die beiden kleinsten Karten zu emulieren. Dies sind die Hercules- oder MCGA-Karte mit 720 x 348 Punkten bei zwei Farben und die CGA-Karte mit 320 x 200 Pixel bei acht Farben beziehungsweise 640 x 200 Pixel bei ebenfalls zwei Farben. Hier stößt der Amiga an

seine Grenzen, aber auch in diesen Grafikmodi läßt sich fast jede Software starten. Was bietet nun das PC-Board, beziehungsweise wie nutzt das Board die Fähigkeiten des Amiga?

An Laufwerken werden die Drives »DF0:« und, wenn vorhanden, ein Zweitlaufwerk unterstützt, die dann auf der PC-Seite mit den Namen "A" und "B" titulierte werden. Der Druckerport heißt ab sofort "LPT1" und versieht wie gewohnt seinen Dienst. Die beiden Maus- beziehungsweise Joystickports bekommen die Namen COM1 und COM2 und werden als serielle Schnittstellen benutzt, bieten allerdings auch weiterhin Maus und Joystick Platz, wobei allerdings darauf geachtet werden muß, daß bei der Entwicklung der IBM-kompatiblen Computer die Maus nicht von vornherein als Eingabemedium vorgesehen war, wie das beim Amiga der Fall ist. Es gibt aber zwei weit verbreitete Mausarten: die Microsoft- und die "Mouse Systems"-Maus, die von der Karte emuliert wird. Wollen Sie Ihre Maus also in gewohnter Art benutzen, muß ein Maus-treiber installiert werden.

Mit der Tastatur gab es allerdings ein paar Probleme. Eine PC-Tastatur ist ein wenig anders aufgebaut als die Amiga-Tastatur. Man kann dort nämlich mittels der Numlock-Taste den Zahlenblock entweder als Zehner-Tastaturblock oder als spezielle Cursor-Steuerungstasten, wie [Insert], [Delete], [PicUp], [PicDown], [Home] oder [End] benutzen. Des weiteren gibt es noch die Tasten [Scroll], [Lock], [Pause/Unterbrechen] und [PrintScreen]. Die Tasten, die ein "Lock" im Namen tragen, verfügen normalerweise über eine LED, die ihren Zustand anzeigt. Da diese LEDs beim Amiga nicht existieren, wird ihr Zustand als blaue Box auf dem Monitor gekennzeichnet.

Eine wichtige Taste fehlt ganz, nämlich [Pause/ Unterbrechen]. Den Backslash ([\]), im DOS-Betrieb eines der wichtigsten Zeichen, kann man zumindest im Betrieb mit der deutschen Tastaturbelegung nicht direkt ansprechen. Hier hilft nur die Direkteingabe des ASCII-Codes (für den Backslash ist er 92) mittels des Zehnerblocks und der Taste [Alter-

nate]. Aber das sind Neben-sächlichkeiten, über die man hinwegsehen kann.

All diese Einstellungen und einiges mehr können von der PC-Seite aus mittels eines Menüs, das über die Tasten [Alternate] und [Help] aufgerufen wird, und von der Amiga-Seite mittels des oben erwähnten Menüs verändert und in einer Preferences-Datei abgespeichert werden.

Im MS-DOS-Betrieb kann auf verschiedene Weisen ein Reset ausgelöst werden. Zum einen kann mittels [Control]-[Amiga]-[Amiga] der gewohnte Total-Reset und somit der Austieg aus dem Emulator-Betrieb vollzogen werden. Mittels [Control]-[Alternate]-[Delete] kann der PC-Dreifingergruß durchgeführt werden, der in Verbindung mit der Taste [F10] zum totalen Reset der Karte führt. Bei den beiden letzten Möglichkeiten bleibt man allerdings in der PC-Ebene.

Die Software zur PC-Karte

Kommen wir zur Software. Das mitgelieferte Microsoft DOS 4.01 läuft einschließlich sämtlicher Hilfsprogramme einwandfrei. Der Diskettenzugriff ist schnell und braucht sich von der Geschwindigkeit her nicht hinter einem XT zu verstecken. Zum Lieferumfang gehören »GW-Basic«, ein einfacher BASIC-Dialekt, »Edlin«, ein Texteditor nicht unähnlich »Ed«, dem Editor des Amiga-DOS, und »Debug«, einen Maschinensprache-Debugger. Nachdem also die Pflicht ohne

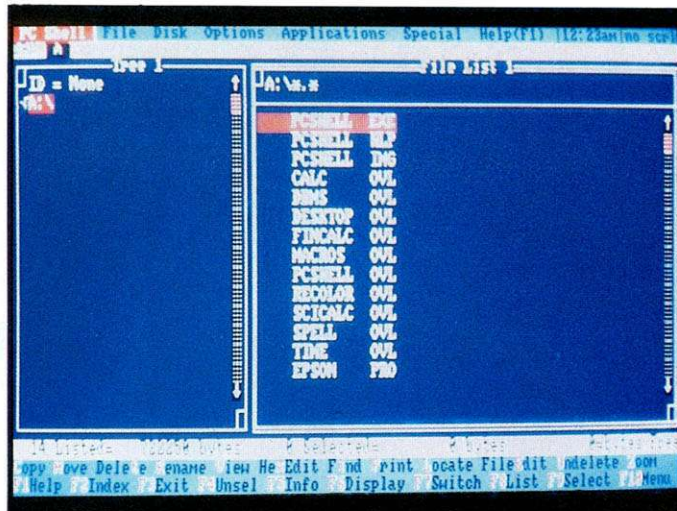


Bild 1. Die »PC-Tools« stellten ihr Können unter Beweis

Absturz durchlaufen wurde, haben wir uns der Kür, sprich anderer Software zugewendet. So sehr wir es auch versuchten, wir haben keinen Absturz hervorrufen können, an dem der Emulator Schuld war.

Eines der Haupteinsatzgebiete von IBM-Kompatiblen ist das Schreiben von Texten. So ist es ganz natürlich, daß einer der Testkandidaten eine Textverarbeitung war. »ConText Pro« arbeitet nach dem WYSIWYG-Prinzip (»What You See Is What You Get«) und verrichtete seinen Dienst ohne Fehl und Tadel. Die nächsten Programme, die getestet wurden, waren die »PC-Tools«, einem Tool-Paket mit Diskettenmonitor und Datenträgerorganisation. Auch hier traten keine Probleme auf. Ebenso mußten Spiele zum Test erhalten. Die Entscheidung viel auf »Horus«, ein Jump'n'Run-Spiel, das auch für den Amiga erhältlich

ist. Die getesteten Programme liefen alle ohne Probleme, solange sie über eine Option zum Ansteuern der verwendbaren Grafikkarten verfügten. Einige Programme, wie zum Beispiel der »Flugsimulator IV« der Firma Micro Soft benötigen vor dem Start eine Initialisierung des Grafik-Bildschirms durch den Befehl »MODE«.

Zum Schluß das Ergebnis

Erfreulich ist die hohe Kompatibilität und Geschwindigkeit. Es ist allerdings nicht mit der Brückenkarte des Amiga 2000 zu vergleichen, mit der man zwischen MS- und Amiga-DOS kommunizieren kann, da es ein reiner Emulator ist.

Erfreulich ist der Service der Firma KCS, die die Software ständig weiterentwickelt und

die Updates kostenlos zur Verfügung stellt. Wer ein Modem besitzt, kann sich auch jederzeit die neueste Version aus einer von über 150 Mailboxen besorgen. So soll mit der zweiten Version, die uns bereits kurz vor Redaktionsschluß erreichte, eine Festplatten-Unterstützung möglich sein.

Die Batterie hat übrigens eine Lebensdauer von 8 bis 10 Jahren, damit erübrigt sich der Einsatz von Akkus.

Wegen ihrer hohen Kompatibilität, sowohl im MS-DOS- als auch im Amiga-Betrieb, und vor allem wegen des sehr niedrigen Preises ist die Karte zu empfehlen.

(jb)

AMIGA DOS Blitz ★ licht

Name: KCS Power PC Board
Hersteller: KCS Computer Supplies BV
Quelle: Generalvertrieb Deutschland Fa. Roßmüller
Preis: 510,- DM

Positiv:

- 1 MByte Speicher
- Hardware-Uhr
- komplettes MS-DOS 4.01 beiliegend
- Update-Service unter anderem über Mailboxen

Negativ:

- Hardware-Uhr nur batteriebetrieben.

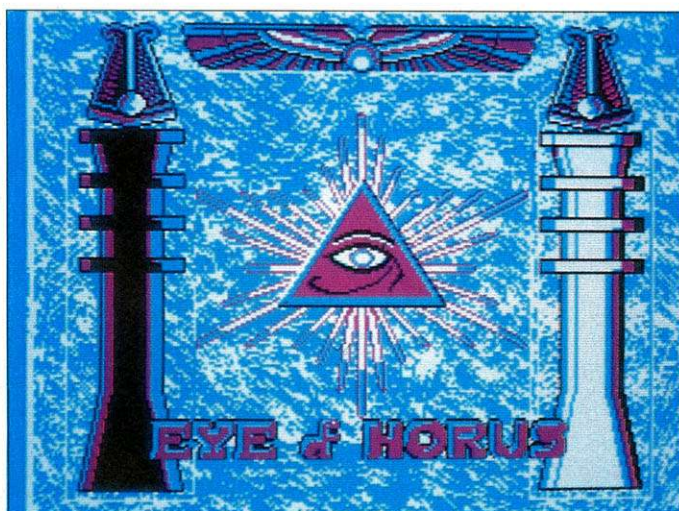


Bild 2. Auch Spiele sind mit der Karte möglich

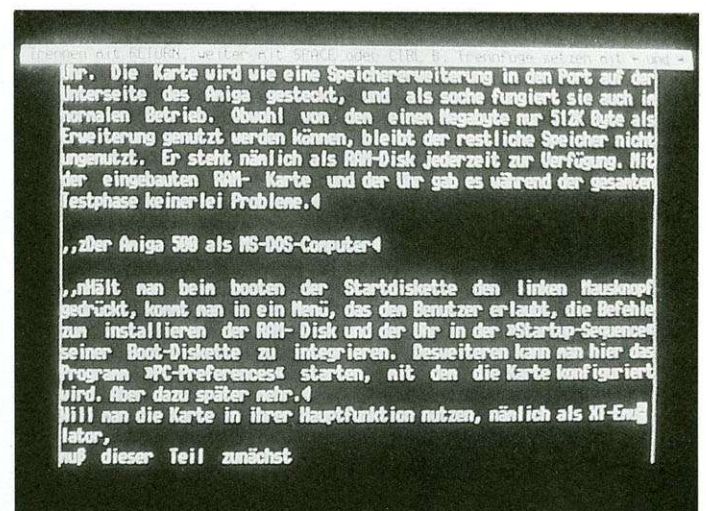


Bild 3. Die Textverarbeitung »ConText PRO PC«

Vom Tisch in den Speicher

Nicht alle Scanner müssen mit der Hand über die Vorlage gezogen werden, manche machen das ganz automatisch...

Scanner gibt es in den verschiedensten Ausführungen; haben wir Ihnen in Heft 7/90 Hand-Scanner vorgestellt, so ist diesmal ein Tisch-Scanner an der Reihe, der keinerlei Handbedienung benötigt, sondern fast alles alleine erledigt.

Leise und präzise – der Sharp JX-100

Der Scanner JX-100 sieht auf den ersten Blick gar nicht so aus, wie man sich einen Scanner vorstellt; rechteckig, mit einem großen Sichtfenster versehen, unterscheidet er sich gewaltig von den Handgeräten. Die Bedienung ist jedoch sehr viel leichter. Vorlagen werden unter das Gerät gelegt, und zwar in die eingezeichnete Randbegrenzung des Fensters. Das Scan-Prinzip ist ähnlich wie bei einem Kopierer; der eigentliche Scanner wird im Gehäuse über die Vorlage gezogen und schickt das Bild zeilenweise zum Rechner. Dabei werden je nach Auflösung mehrere Durchläufe benötigt. Als Lichtquelle benutzt der Scanner eine Weißlicht-Leuchtstoffröhre, die mit der Scan-Elektronik in einem Führungsschlitten sitzt, der wiederum, angetrieben durch einen Schrittmotor, entlang einer Führung innerhalb des Gehäuses läuft. Die Scan-Vorlagen können bis zu einer Größe von 160mm x 100mm unter den Scanner gelegt werden.

Der JX-100 ist ein Farb-Scanner, das heißt, daß die Vorlagen in mehreren Auflösungen gescannt werden können, darunter Low-Res-, High-Res- und HAM-Format. Nach Anwählen eines Formates im Scan-Pro-

gramm beginnt der Scanner mit seiner Aufgabe, der Schlitten fährt an die Anfangsstellung und beginnt fast geräuschlos mit seiner Arbeit. Während man bei den Hand-Scannern das Tempo weitgehend selbst beeinflussen kann, muß man beim JX-100 etwas Zeit mitbringen, der Scan-Vorgang bei der bestmöglichen Auflösung beträgt etwas mehr als fünf Minuten. Die Daten werden über die serielle Schnittstelle des Amiga empfangen, wohl der Hauptgrund für die geringe Geschwindigkeit des Scan-Vorgangs. Die Qualität der erzeugten Grafiken ist jedoch sehr hoch, was vor allem der ruhigen Führung des Scanners und der glatten Einbettung der Vorlage zu verdanken ist. Der Scanner bekommt seine nötige Versorgungsspannung über ein externes Steckernetzteil.

Zwei Programme gehören zum Software-Paket

Durch die Benutzung der seriellen Schnittstelle ist der Scanner auch an anderen Computern einsetzbar, vorausgesetzt die entsprechende Software ist vorhanden. Da der Anschluß an einen neunpoligen Sub-D-Stecker vorgesehen ist, der ja bei vielen MS-DOS-Rechnern die serielle Schnittstelle repräsentiert, liegt dem Scanner noch ein Adapterkabel von neun- auf 25poligen Anschluß bei.

Die Software, die dem Scanner beiliegt, entpuppt sich als sehr komfortabel. Da ist zuerst einmal das eigentliche Scan-Programm »Scanlab100«. Es enthält alle Funktionen, um die Bilder in den verschiedenen



Der Scanner Sharp JX-100

Auflösungen einzulesen. Im ersten Teil des Programms werden die Bilddaten in der größten Auflösung verarbeitet, während der zweite Teil den hochauflösenden Modus gilt. Die Bilder stehen komplett im Speicher, es ist allerdings nur ein Ausschnitt auf dem Monitor sichtbar. Das Bildformat ist über »Schiebereglern« einstellbar, dadurch lassen sich die Längen- oder Breitenmaße so einrichten, daß das Bild nicht auseinandergezogen wird. Der Bildkontrast kann ebenfalls geregelt werden, so daß dunkle Stellen aufgehellt werden können. Die Zusammenarbeit zwischen Scanner und Scan-Software klappt vorzüglich, die erstellten Bilder können sofort im IFF-Format oder als komprimierte Scan-Daten abgespeichert werden.

»The_Art_Department« ist ein zusätzliches Programm auf der Diskette, mit dem die gescannten Bilder weiterbearbeitet werden können. Hier ist zum Beispiel eine Funktion enthalten, die die Bilder aussehen läßt, als ob sie aus einer Bleistiftträdierung entstanden wären. Die Bilder können wieder von diversen anderen Weiterverarbeitungsprogrammen (DTP / Grafik / Textverarbeitung mit Bilder-Funktion) genutzt werden und von dort aus auf Papier gelangen. Der Scanner und seine Software sind eine gut gelungene Ein-

heit, alles funktioniert reibungslos und sehr zufriedenstellend.

Ein Negativpunkt bleibt jedoch: Die Scan-Fenster bestehen aus Plexiglas; jeder noch so kleine Stupser kann ihnen einen Kratzer verpassen, der sich beim Scannen unliebsam bemerkbar macht. Hier hilft nur eine stabile Schaumgummiunterlage, wenn man lange etwas vom Gerät haben will. Der Preis des Scanners rechtfertigt den Einsatz allerdings nur zu professionellen Zwecken.

(jb)

AMIGA DOS Blitz★licht

Name: Farb-Scanner Sharp JX-100 + Scanlab100 + »The Art Department«
Quelle: CompuStore GmbH, Fritz-Reuter-Str. 6, 6000 Frankfurt/Main, Tel.: 069-567399
Preis: zirka 2500 DM

Positiv:

- gut durchdachte Software
- genaues Fixieren der Vorlage ermöglicht gute Wiedergabequalität
- leichte Installation

Negativ:

- Oberfläche des Scanners zerkratzt leicht (Plexiglas)

Filecards nennt man Harddisks, die intern in Form einer Steckkarte in den Rechner eingebaut werden. Die Vortex Athlet ist eine solche Filecard für den Amiga 2000. Versprochen werden Leistungsdaten von bis zu 25 ms Zugriffszeit. Wir haben uns die Platte zu einem ausführlichen Test vorgenommen.

Zuerst erfolgt der Einbau in den Amiga 2000. Überraschenderweise paßt die Filecard so exakt, daß kaum Druck und Hebelwirkungen auf die Platine ausgeübt werden müssen. Die Hardware selbst ist solide gebaut, die Schaltungen sind übersichtlich aufgebaut und sehr sauber verlötet. Die Verkabelung ist gut, so daß es keine überstehenden Flachbandkabel gibt, die beim Einbau störend im Weg sind. Die Platte ist so dimensioniert, daß weitere Steckkarten im Amiga 2000 nicht berührt werden.

Nach dem Einbau und dem Schließen des Amiga-Gehäuses (irgendwo muß ja der Monitor draufstehen) erfolgt die Installation der Athlet. Nach dem Booten der mitgelieferten Installationsdiskette geben wir »HDSETUP« ein. Wir wählen den ersten Menüpunkt, der die Platte Low-Level-formatieren soll. Nach Eingabe einiger Parameter wird die Platte formatiert.

Nicht ganz fehlerfrei

Die Formatierung selbst läuft ohne Probleme ab, ebenso die Initialisierung, bei der einige wichtige Dateien wie Treiber und Systemparameter auf die

Kraftpaket Vortex Athlet

Die neue Festplatte von Vortex trägt den Namen "Athlet". Ist diese Platte wirklich so "sportlich"?

Platte geschrieben werden. Probleme gibt es bei der Suche nach Fehlern. Es ist uns nicht gelungen, trotz mehrmaligen Startens der Fehlersuche keinen Programmabsturz zu erhalten. Irgendwann hängt sich das Programm auf und sucht einfach nicht mehr weiter. Schließlich gaben wir entnervt auf, ignorierten alle potentiellen Fehler und begannen mit der Partitionierung der Platte in drei logische Laufwerke: »DH0:«, »DH1:« und »DH2:«. Hier wurden wieder keinerlei Fehler festgestellt, alles lief glatt. Wir konnten also mit dem eigentlichen Test beginnen.

Standardtest

Zeit ist eine Konstante. Eine Sekunde ist immer eine Sekunde lang. Bei einigen Festplatten scheint die Zeit jedoch relativ zu sein. Für die Vortex Athlet ist eine Sekunde ziemlich lang, denn sie schafft es, in dieser Sekunde immerhin 730 kByte an Daten im »Fast-File-System« zu übertragen. Somit ist die Herstellerangabe

einer mittleren Zugriffszeit von 25 ms durchaus gerechtfertigt. Ermittelt wurde dieser Wert mit dem PD-Programm »Disk-speed«, das bei allen Festplattentests in der Redaktion herangezogen wird. Somit ist gewährleistet, daß alle Werte vergleichbar bleiben.

Was eine Festplatte wirklich zu leisten vermag, erkennt der Benutzer erst, wenn er mit der Platte »arbeitet«, das heißt, diese im täglichen Betrieb benutzt. Hier erst erweisen sich die wahren Vorzüge der Vortex Athlet.

Die Praxis

Betriebsgeräusche sind kaum wahrzunehmen, auch das »Schnarren«, das verursacht wird, wenn die Schreib-/Leseköpfe auf eine neue Spur gefahren werden, hält sich in Grenzen. Erfolgt eine Zeit lang kein Zugriff auf die Platte, fahren die Köpfe in eine Park-Position (Autopark-Funktion). Die Autopark-Funktion verhindert, daß bei einer eventuell auftretenden Erschütterung die Schreib-/Leseköpfe die Ober-

fläche der Platte berühren, was meist fatale Folgen hat. Die Platte läuft bei Zugriff nach dem Parken recht flott an und bleibt dabei doch relativ ruhig. Ausfälle, wie zum Beispiel Schreib-/Lese-Fehler, konnten nicht festgestellt werden, die Platte schreibt und liest zuverlässig, trotz der mißlungenen Fehlererkennung des Installationsprogramms.

Neben der uns vorliegenden 40-MByte-Version gibt es diese Platte noch mit 90, 130 oder sogar 180 MByte. Auf der Filecard sind bereits Steckplätze für SIMs (Speicherbausteine), mit denen sich ein zusätzlicher Rechner-Speicher von bis zu 4 MByte aufbauen läßt. Die passenden Speicherkarten können zusammen mit der Filecard bestellt oder aber nachträglich erworben werden.

Ausbaufähig

Die Vortex Athlet ist eine zuverlässige Filecard, die ihren Preis mit Sicherheit wert ist. Die Möglichkeit, die Platte platzsparend mit zusätzlichem Rechner-Speicher auszurüsten, und das dafür bereits existierende Lieferangebot machen die Athlet zu einer erschwinglichen Harddisk, die bei Bedarf jederzeit erweitert werden kann. Als einziges Manko muß die Installations-Software angesehen werden, die wohl noch verbesserungswürdig ist.

(tb)

AMIGA DOS Blitz ★ licht

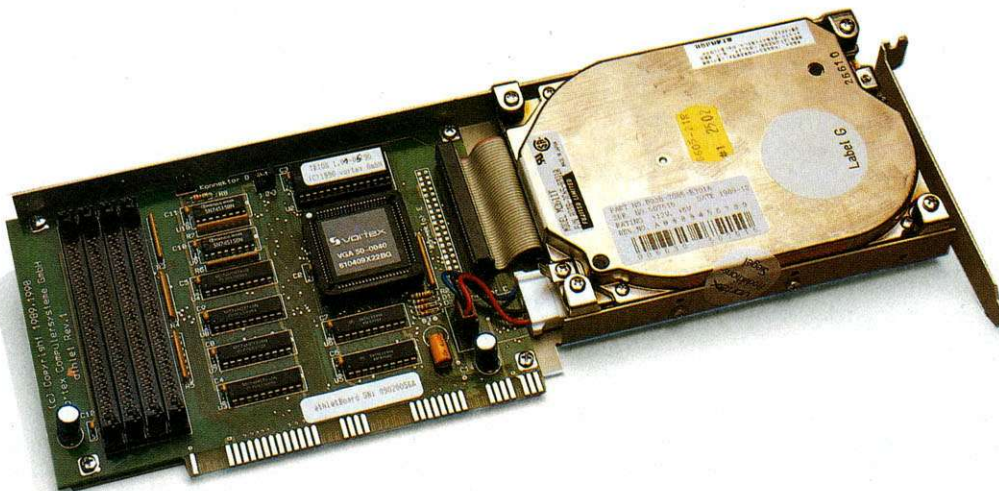
Name: Vortex Athlet
Quelle: Vortex Computersysteme
Falterstr. 51-53
7101 Flein
Tel.: 07131/50880
Fax: 07131/55063
Anbieter: Vortex
Funktion: Festplatte
Preis: zirka 1300 DM

Positiv:

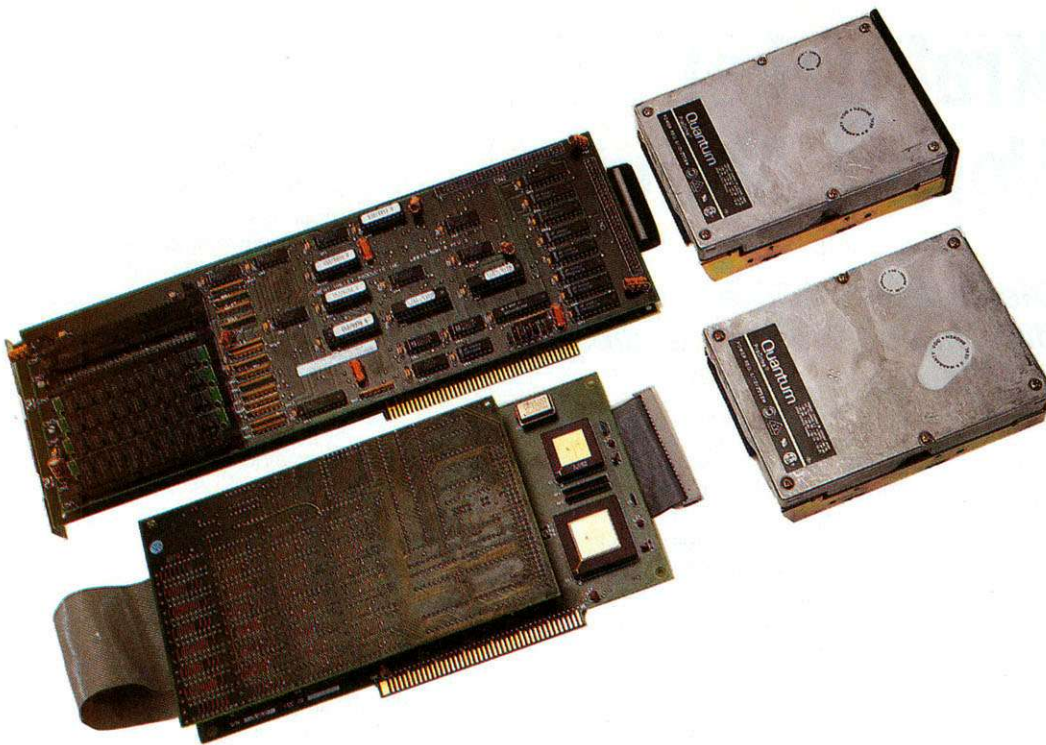
- leichter Einbau
- solide Verarbeitung
- zuverlässiger Betrieb
- verständliches Handbuch

Negativ:

- Installations-Software fehlerhaft



Die Filecard Vortex Athlet



Amiga mit Nachbrenner

Der Einsatz einer 68030-Karte macht den Amiga schnell. Wie schnell, wollten wir anhand zweier angebotener Karten wissen.

Die 68020-Boards, vor wenigen Monaten noch das Nonplusultra, werden mehr und mehr von den 68030-Boards verdrängt. Grund genug, zwei davon einmal unter die Lupe zu nehmen.

Der Amiga ist mit einer offenen System-Architektur (OSA) gebaut worden; er kann also mit den Bedürfnissen und mit der Zeit wachsen. Diese innovative Technik wird durch eine Vielzahl von Erweiterungs-Slots unterstützt.

Die ZorroII-Slots sind für Festplatten, RAM-Erweiterungen oder anderes vorgesehen. Dazu besitzt er noch mehrere PC-Einschübe und schließlich den Prozessor-Slot direkt neben dem Netzteil. Dort können Erweiterungskarten eingesteckt werden, auf denen sich Prozessoren befinden, die den Rechner beschleunigen.

Ronin und GVP stellen solche Boards her. Beide gibt es in verschiedenen Ausführungen,

wobei sich die verschiedenen Versionen in den Taktraten des Prozessors und des Coprozessors unterscheiden. Hier stehen Taktraten von bis zu 50 MHz zur Verfügung. Diese Versionen sind aber noch recht teuer und müssen auch genau auf die Hardware abgestimmt sein, da diese Boards sehr zeitkritisch sind. Getestet wurden die 25-MHz-Versionen mit dem Coprozessor 68882, der ebenfalls mit 25 MHz getaktet war.

Ausschlaggebend für die Performance eines Boards ist auch, welche Art von Speicher zur Verfügung steht. Richtig schnell werden die Boards erst mit 32-Bit-Speicher, da der Prozessor beim Zugriff auf das RAM hier keine Wartezeiten, die sogenannten Wartezyklen einlegen muß. Beide Boards waren mit 4 MByte 32-Bit-RAM ausgestattet. Die RAM-Erweiterung wird als Huckepackplatte auf das Board aufgesetzt.

Somit wird ein Erweiterungs-Slot gespart. Das ist jedoch nicht alles, beide haben ebenfalls einen Festplatten-Controller implementiert. Beim GVP-Board ist dies ein AT-Controller, der bis zu zwei Festplatten gleichzeitig ansprechen kann, beim Ronin-Board versieht ein SCSI-Controller seine Dienste. Dieser Controller kann bis zu sieben Geräte gleichzeitig ansprechen. Es wurden jeweils Komplettpakete getestet. Zum Einsatz kam beim GVP-Board eine 80-MByte-Quantumplatte, beim Ronin-Board eine 40-MByte-Platte, ebenfalls von Quantum.

Die Installation eines 68030-Boards

Die Installation geht recht einfach vonstatten. Beide Karten werden in den Prozessor-Slot gesteckt, und anschließend

muß die Installationsdiskette gebootet werden. Das GVP-Board ist autokonfigurierend, was bedeutet, daß das Board direkt nach dem Anschalten des Rechners oder nach einem Reset erkannt wird. Beim Ronin-Board muß zuerst das Programm »HurricaneConfig« aufgerufen werden, welches das Board und den Speicher in das System einbindet. Die meiste Zeit nimmt das Einrichten der Harddisk in Anspruch. Diese Arbeit wird bei GVP von dem Programm »ATHDInstall« übernommen. Andere Controller, die sich bereits im System befinden, können auch weiterhin benutzt werden. Hat man alle vom Programm geforderten Angaben gemacht, wird die Platte formatiert und, falls gewünscht, in die entsprechenden Partitionen unterteilt und das GVP-Board ist einsatzbereit.

Das Ronin-Board läßt sich ebenfalls einfach installieren, wenn man weiß, daß eine eventuell vorhandene Festplatte für die PC-Karte entfernt werden muß. Ansonsten verläuft die Installation ähnlich einfach. Leider war die Gerätenummer der Festplatte nirgendwo vermerkt, so daß hier erst ein Anruf bei IM, dem deutschen Distributeur von Ronin, weiterhalf.

Rechenintensive Programme, wie beispielsweise Raytracing- und DTP-Anwendungen oder Konvertiererroutinen, die viele Berechnungen durchführen müssen, profitieren besonders von den Turboboards. Die sieben Praxistests wurden nach diesen Kriterien ausgewählt. Die ersten beiden Tests befaßten sich mit dem Thema Raytracing. Im ersten Test wurde ein Bild mit »TurboSilver3.0« in den Formaten »HiRes-Interlace« und »Overscan« berechnet. Hier kam die Floating-Point-Version zum Einsatz, die den 68882 direkt unterstützt. Auch beim zweiten Test mit »Sculpt-4D«, ebenfalls in der FP-Version, wurde ein Bild in den beiden eben genannten Formaten berechnet. Der dritte Test befaßte sich mit dem Thema DTP. Mit »PageSetter« wurde ein 99 Seiten umfassendes Dokument erstellt und ein über 700 kByte langer Text importiert. Um den Geschwindigkeitsvergleich in bezug auf Konvertierungsprogramme anstellen zu können, wurden im

vierten Test eine Bitmapgrafik mit »VectorTraceV1.1« in eine Vektorgrafik konvertiert. Im fünften und sechsten Test kam »Beckertext« zum Zuge. Test 5 befaßte sich nicht mit der Performance des Prozessors, sondern mit der der Festplatte. Hier wurde ein 270 Seiten langer Text als ASCII-Datei abgespeichert. Dieser Text umfaßt ein ganzes Buch und ist mehr als 700 kByte lang. Im sechsten Test wurde dieser Text einmal ganz im Blocksatz durchformatiert. Der siebte Test bestand darin, die Geschwindigkeitssteigerung beim Compilieren, Linken und Optimieren eines Modula2-Sourcecodes zu messen. Tabelle 1 zeigt die Zeiten, die die verschiedenen Konfigurationen für diese Tests benötigten. Tabelle 3 zeigt die benötigten Zeiten und Tabelle 4 die Geschwindigkeitsfaktoren gegenüber einem normalen Amiga 2000 mit Festplatte an.

Hier nur einige Ergänzungen zu den Tests: Alle Tests wurde bei aktiviertem Daten- und Befehls-Cache durchgeführt. Die Kickstart wurde ebenfalls ins 32-Bit-RAM kopiert, wodurch sich noch einmal eine Geschwindigkeitssteigerung erzielen läßt. Die Ergebnisse sind verschieden. Die größten Vorteile erhält man beim Raytracing und beim Konvertieren der Grafik. Hier ist der »Turbo-

Test	Programm	GVP	Ronin	Amiga2000
Test 1:	TurboSilver	0:40:18h	0:57:31h	5:33:28h
Test 2:	Sculpt-4D	0:29:50h	0:35:42h	7:04:43h
Test 3:	PageSetter	0:01:14h	0:01:50h	0:02:14h
Test 4:	VectorTrace	0:00:22h	0:00:30h	0:05:38h
Test 5:	Beckertext (Speichern)	0:00:26h	0:00:36h	0:02:00h
Test 6:	Beckertext (Formatieren)	0:26:35h	0:15:20h	1:37:39h
Test 7:	ModulaII (Compilieren)	4,51 sek	6,41 sek	13,14 sek
	(Linken)	3,55 sek	5,15 sek	8,37 sek
	(Optimieren)	7,25 sek	9,84 sek	21,39 sek

Test	Programm	GVP	Ronin	Amiga2000
Test 1:	TurboSilver	8,27	5,80	1
Test 2:	Sculpt-4D	14,24	11,90	1
Test 3:	PageSetter	1,81	1,22	1
Test 4:	VectorTrace	15,36	11,27	1
Test 5:	Beckertext (Speichern)	4,62	3,33	1
Test 6:	Beckertext (Formatieren)	3,67	6,37	1
Test 7:	ModulaII (Compilieren)	2,91	2,05	1
	(Linken)	2,36	1,63	1
	(Optimieren)	2,94	2,16	1

Abb. 1: Die Geschwindigkeitsvergleiche...

Amiga« bis zu 15mal schneller, während er bei den anderen Tests eine durchschnittliche Geschwindigkeitssteigerung zwischen 300 und 500 Prozent erreichte.

Doch nicht nur bei rechenintensiven Programmen war ein merklicher Geschwindigkeitszuwachs zu verzeichnen. Das gesamte Handling mit dem Rechner geht wesentlich schneller vonstatten. Die Wartezeiten beim Booten sind nicht mehr so lange wie gewohnt, »WordPerfect« braucht nicht mehr so lange zum Editieren eines Textes und im CLI erscheint nach der Eingabe von »dir« das Inhaltsverzeichnis fast ohne Verzögerung. Das Arbeiten mit dem Amiga macht diesbezüglich wesentlich mehr Spaß und geht viel lockerer von der Hand. Um den Test noch abzurunden, wurden zwei Benchmark-Tests durchgeführt. Außerdem verwendete ich den »CPU SpeedtestV2.0« von Ronin, der sich ebenfalls im Lieferumfang des Boards befindet. Die Ergebnisse können Sie in Tabelle 3 und 4 nachlesen. Zum »Savagetest« ist anzumerken, daß hier für den Coprozessor Extra-Versionen zur Verfügung stehen, die diesen unterstüt-

zen. Somit läßt sich die Steigerung um den Faktor 60 erklären. Allerdings zeigt er auch, welche Faktoren mit gut programmierter Software erreicht werden können.

Die Festplatte im Kreise des Systems

Natürlich sind auch die Geschwindigkeiten der Festplatten interessant. Hier sind Praxistest am aussagekräftigsten. Hierzu sollten Sie Test 5 und 7 aus Tabelle 1 betrachten. Der Vollständigkeit halber haben wir zwei Speedtests, die alle die Platte unter AmigaDOS testen, ausgeführt. Die Ergebnisse können mitunter variieren. Schauen Sie hier in Tabelle 5 und in den Bildern nach. Alle Tests wurden mit fast vollen 40-MByte-Partitionen durchgeführt. In der Praxis ist der Controller des GVP-Boards etwas schneller als der des Ronin-Boards. Bei der GVP-Platte mußte man sich manchmal über Hänger der Platte beschweren. Hierbei sausen dann die Köpfe mehrmals um einen Track. Dadurch wird die Arbeit natürlich empfindlich verzögert.

Mit dem GVP-Board wird eine Diskette ausgeliefert, auf der sich neben Installationsprogrammen noch zwei Programme zum Testen des Speichers befinden. Des weiteren befinden sich noch ein paar Texte auf der Diskette, die Tips zur Programmierung des 68030 geben. Das Programm »Set-CPU« befindet sich im Verzeichnis »C« der Diskette. Die-

AMIGA DOS Blitz ★ licht

Name: GVP A3001 Upgrade Kit
Anbieter: DTM, Poststr. 25,
6200 Wiesbaden,
Tel.: 06121/502059
Preis: 1998,- DM

Positiv:

- einfache Installation
- schnell
- verträgt sich gut mit weiterer Hardware
- unterstützt auch andere Controller
- Autoboot

Negativ:

- manchmal »hängt« die Platte
- lange Karte, kein Platz für interne Geräte wie MIDI-Interface
- kein Schalter

AMIGA DOS Blitz ★ licht

Name: Ronin 68030 Board
Anbieter: Intelligent Memory,
Wächtersbacherstr. 89, 6000
Frankfurt, Tel.: 069/410071-72
Preis: ca. 2000 DM

Positiv:

- einfache Installation
- viele Utilities werden mitgeliefert
- abschaltbares Autoboot

Negativ:

- bootet oft im NTSC-Modus
- nicht autokonfigurierend
- nicht so schnell wie das GVP-Board
- kein Schalter

Tabelle 3

Test	Programm	GVP	Ronin	Amiga2000
Test 1:	CPU SpeedV2.0	---	----	-----
Test 2:	Sieve	5,08sek	5,42sek	46,36sek
Test 3:	Savage	0,18sek	0,18sek	10,22sek

Tabelle 4

Test	Programm	GVP	Ronin	Amiga2000
Test 1:	CPU SpeedV2.0	11,50	11,04	0,98
Test 2:	Sieve	9,13	8,55	1
Test 3:	Savage	56,78	56,78	1

Tabelle 5

DiskPerf2-Operation	GVP	Ronin	Amiga2000
Create/Close	18 files/sek	11 files/sek	13 files/sek
ExNext	256 entries	119 entries	105 entries
Lock/Examine/Unlock	156 files/sek	59 files/sek	31 files/sek
Delete	59 files/sek	20 files/sek	17 files/sek
Seek/Read	222 files/sek	111 files/sek	73 files/sek
Read (buf=131972 Bytes)	601248 Byte	339565 Byte	137681 Byte
Write (buf=131972 Bytes)	420103 Byte	219184 Byte	111267 Byte

Abb. 2: ...und die Performance-Tests

ses Programm ist geradezu lebenswichtig, da hiermit die Cache-Speicher direkt beeinflusst werden können und die Kickstart ins schnelle RAM geladen werden kann, was ebenfalls zu Geschwindigkeitsvorteilen führt. Leider ist die Karte eine lange Karte, was bedeutet, daß sie bis zum Gehäuse-rücken des Rechners reicht. Somit lassen sich keine internen Erweiterungen, wie bei-

spielsweise ein Midi-Interface, einbauen. Beim Lieferumfang ist Ronin etwas großzügiger. Neben den Installationsprogrammen befinden sich noch einige zusätzliche Programme auf der Diskette. So findet man hier Benchmarks, das Programm »CPU SpeedtestV2.0« (siehe Tabelle 3 und 4) und andere PD-Programme, wie beispielsweise ein Programm zum Er-

stellen von Mandelbrotgrafiken. Auch das Programm »SetCPU« befindet sich mit Anleitung auf der Diskette. Nachteilig ist, daß der Rechner sehr oft im NTSC-Modus bootet. Erst wenn der Rechner etwas warmgeworden ist, wird immer im PAL-Modus gebootet. Beim Einbau der RAM-Erweiterung und der Festplatte stürzte der Rechner ab. Er lief aber einwandfrei mit einer PC-

Platte und der PC-Karte. Auch sonstige Erweiterungen störten nicht. Schön ist auch die Möglichkeit, bei Inkompatibilitäten die Möglichkeit des Auto-bootens per Knopfdruck abzuschalten. Schade ist bei beiden Boards, daß Sie nicht abschaltbar sind. Auf beiden Platinen befinden sich Jumper, mit denen dies erledigt werden kann. Einen Schalter muß sich der Anwender selbst einbauen.

Lohnt sich die Anschaffung eines 68030-Boards?

Der Amiga wird mit einer der beiden Karten wesentlich schneller. Die gesamte Arbeit macht einfach mehr Spaß, da nicht mehr so lange auf den Rechner gewartet werden muß. Natürlich kann sich nicht jedermann solch eine Erweiterung leisten. Sie sind also nur für diejenigen interessant, für die Zeit auch Geld ist, also in erster Linie professionelle Anwender. Welche von den Karten man sich zulegt, bleibt jedem selbst überlassen. Dies hängt sicherlich auch vom Preis ab. Da dieser je nach Version schwankt, können feste Angaben nur bei IM beziehungsweise DTM direkt erfragt werden; mehrere tausend Mark sind für so ein komplettes Produkt aber in jedem Fall zu veranschlagen.

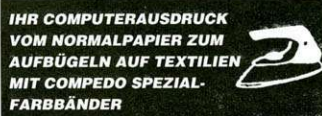
(jb)



SPESIALFARBÄNDER

Für 90% aller Matrixdrucker in Rot, Gelb, Blau und Schwarz, oder als 4-Colorband für Farbdrucker erhältlich! z. B.:

CITIZEN SWIFT	34,90	STARLC 10	33,90
EPSON LQ 500/800	35,90	STAR LC 10 4-COLOR	46,90
EPSON LX 80/90	31,90	STAR LC 24 - 10	36,80
EPSON LQ 2550 4-COLOR	49,90	STAR NL 10 / NB 24 - 10	35,90
COMM. MPS 802	36,90	NEC P2200	37,90
COMM. MPS 803	36,80	NEC P6 + / P7 +	39,90
COMM. MPS 1500 4-COLOR	49,00	NEC P6 + / P7 + 4-COLOR	59,90
COMM. MPS 1224 4-COLOR	47,80	NEC P2 / P6	37,50
SEIKOSHA SP	35,90	NEC P2 / P6 4-COLOR	59,90
PRÄSIDENT 63 XX	29,90	PANASONIC KXP 10 80/90	36,90
OKI ML 390	36,70	PANASONIC KXP 1124	38,90
OKI ML 292 4-COLOR	59,90	APPLE IMAGEWRITER	36,90



Normalmarkenfarbbänder
- auch in Sonderfarben -
zu Superpreisen! z. B.:

CITIZEN SWIFT	9,10	STAR LC 10	7,80
EPSON LQ 500/800	10,90	STAR LC 10 4-COLOR	15,70
EPSON LX 80/90	8,50	STAR LC 24 - 10	11,30
EPSON LQ 2550 4-COLOR	24,50	STAR NL 10 / NB 24 - 10	9,10
COMM. MPS 802	9,20	NEC P2200	12,00
COMM. MPS 803	9,30	NEC P6 + / P7 +	12,70
COMM. MPS 1500 4-COLOR	18,95	NEC P6 + / P7 + 4-COLOR	28,40
COMM. MPS 1224 4-COLOR	18,50	NEC P2 / P6	10,20
SEIKOSHA SP	12,10	NEC P2 / P6 4-COLOR	28,40
PRÄSIDENT 63 XX	7,90	PANASONIC KXP 10 80/90	10,70
OKI ML 390	10,40	PANASONIC KXP 1124	11,70
OKI ML 292 4-COLOR	29,20	APPLE IMAGEWRITER	8,90

Generalvertrieb Bayern
UNI COMP
HANDELSGESELLSCHAFT MBH

Postfach 11 10 8221 St. Georgen
Tel: 0 86 69 / 3 66 93 Fax: 0 86 69 / 1 26 00



Postfach 13 52 5860 Iserlohn
Tel: 0 23 71 / 4 10 71 - 74 Fax: 0 23 71 / 4 10 75

Persönlicher Bestellservice von 9.00 - 21.00 Uhr, auch Sa. u. So. Händlerkonditionen auf Anfrage!
Versandpauschale DM 6,- Nachnahme o. Vorkasse (Ausland).

Jetzt auch auf Keramik,
Glas, Alu, Metall u. a.
Werkstoffe aufdrucken!

- Anwendung
- Gegenstand lackieren
 - Transfer-Ausdruck mit Klebeband aufkleben
 - 15 min. einbrennen (z.B. im Backofen)
 - Ausdruck entfernen - Fertig!

Die Entscheidung
für das Creative

- Bügeln auf T-Shirts, Jacken, Regenschirme, Kissen etc.
- waschecht - ideal für Werbung
- Lebensdauer wie normales Markenfarbband



Lackset...17.90

(Speziallack, Pinsel, hitzefestes Klebeband und Abroller)

Weiteres Zubehör für den Transferdruck: T-Shirts, Kissenbezüge, Filzposter, Kalender und Puzzles zum bedrucken, auf Anfrage.

Tiefkühlkost auf Diskette

Ist Ihr Lieblingsprogramm kopiergeschützt? Bekommen Sie keine Sicherheitskopie? Dann versuchen Sie es einmal mit »Action Replay«!

Werkzeuge, die Programme im aktuellen Zustand "einfrieren" und auf Diskette sichern, werden "Freezer" genannt. »Amiga Action Replay« stellt ein solches Werkzeug dar. Es stehen jedoch noch viel mehr Funktionen zur Verfügung, die dem Amiga-Anwender das Leben leicht machen sollen.

»Action Replay« wird in Form eines Hardware-Moduls ausgeliefert, das am Expansionsport des Amiga 500 angeschlossen wird.

Stillstand auf Knopfdruck

Bevor Sie sich daran machen, ein Programm zu "freeze", sollten Sie eine Diskette formatieren, auf der die Programme abgespeichert werden. Ist die Diskette fertig formatiert, kann sie mit einem Hilfsprogramm der mitgelieferten Tooldisk installiert werden und ist danach bootfähig. Wird diese Diskette gebootet, erscheint ein Auswahlmenü, in dem alle vorhan-

denen Programme angezeigt und mit Mausklick gestartet werden können. Hier zeigte sich im Test, daß noch längst nicht alle Programme "gefroren" werden können.

Programmstart auch ohne Karte

Die Programme, die erfolgreich abgespeichert wurden, starten genau an der "gefrorenen" Position. Zum Starten ist die Anwesenheit von »Action Replay« nicht nötig; von der bootfähigen Diskette laufen die Programme auch ohne Modul.

»Action Replay« läßt sich hervorragend als "Trainer" einsetzen. Dazu stehen drei Möglichkeiten offen: der "Freezer", die Bremse und der Monitor. Mit dem "Freezer" kann der Spielstand vor "kritischen" Situationen einfach abgespeichert werden. Danach heißt es üben, laden, üben ..., bis das gewünschte Ziel erreicht ist. Wem das Spiel zu schnell ist,



Bild 1. Kompakte Hardware...

kann die Bremse des »Action Replay« einsetzen. Dazu wird ein kleiner Schalter, der auf dem Gehäuse des Moduls angebracht ist, umgelegt und mittels eines Drehreglers die Geschwindigkeit des Prozessors vom Stillstand bis Normaltempo geregelt.

Die dritte Möglichkeit ist der leistungsfähige Monitor/Disassembler. Dieses Hilfsmittel kann dazu verwendet werden, die Anzahl der "Leben" zu verändern. Angenommen, Sie haben noch drei Raumschiffe zur Verfügung, brauchen aber viel mehr, um das Spiel zu beenden. Drücken Sie einfach die "Freeze"-Taste des Moduls, und Sie befinden sich im Hauptmenü von »Action Replay«. Starten Sie den Monitor und geben Sie »TS !3« ein. Jetzt werden vom Monitor alle Adressen ausgegeben, in denen eine "3" steht. Diese Adressen sollte man sich merken und weiterspielen. Hat sich die Anzahl der "Leben" weiter verringert, wird dieser Vorgang wiederholt, bis einige Adressen regelmäßig erscheinen. Diese Adressen enthalten höchstwahrscheinlich den Wert der Bildschirmleben und können geändert werden. Wenn alles funktioniert hat, haben Sie eine ganze Menge "Leben" mehr.

Neben den Möglichkeiten, Spiele und andere Programme zu manipulieren, wartet »Action Replay« mit einigen nützlichen Werkzeugen auf. Sämtliche Prozessor-Register und Speicherwerte können editiert werden, und sogar ein Assembler/Disassembler steht zur Verfügung. Bilder und Sound-

daten können ebenfalls aus laufenden Programmen herausgezogen und abgespeichert werden.

Leistungsfähige Werkzeuge

Ein einfacher Sprite-Editor, der im Monitor integriert ist, macht die "Werkzeugkiste" komplett. »Amiga Action Replay« ist eine Hardware-Erweiterung, die nicht in allen Fällen, aber doch den meisten für "Spielefreaks" und Programmierer ein überaus nützliches Hilfsmittel darstellt. Debugging (die Fehlersuche in laufenden Programmen) von Selbstentwicklungen wird sehr vereinfacht, und es sind einige Operationen möglich, die mit einem Software-Monitor nicht bewerkstelligt werden können.

(tb)

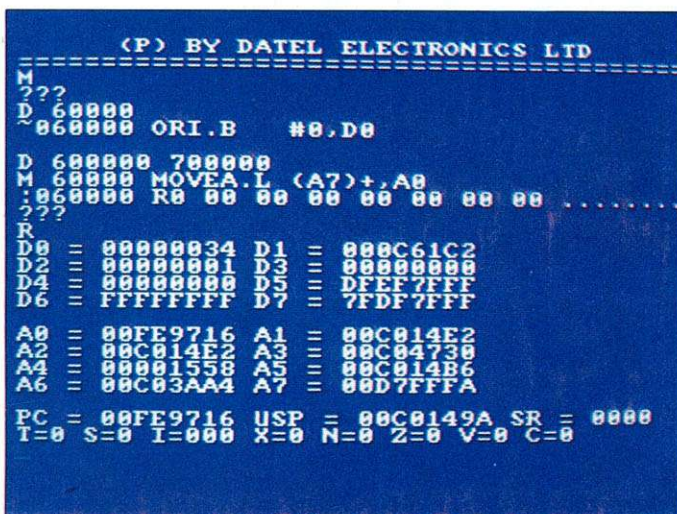


Bild 2. ...und schlichtes Menü

AMIGA DOS Blitz ★ licht

Name: Amiga Action Replay
Funktion: Freezer und Bremse
Preis: 189,- DM
Quelle: Eurosystems,
Hühnerstraße 11, 4240 Emmerich, Tel: 02822/45589

Positiv:

- funktioneller Monitor
- einfache Bedienung
- robuster Aufbau

Negativ:

- nur bedingt einsatzfähig
- unverträglich mit einigen Programmen

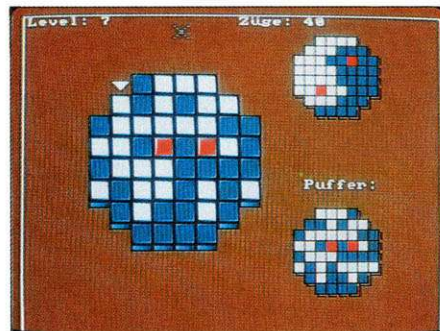
Alle Programme sofort nutzen

Auch in diesem Fall hilft Ihnen die Databox von Amiga DOS aus der Patsche: Neben den Quelltexten im ASCII-Format finden Sie jeweils auch das fertige, lauffähige Programm. Sie brauchen es nur von der Databox-Diskette aus zu starten.

DATA

**Keinen Ärger mehr mit
Tippfehlern**

Zermartern Sie sich nicht den Kopf, bis Sie die falsche Zahl im Datafeld gefunden haben. – Alle Dateien auf der Databox zur Amiga DOS sind vom Autor und der Redaktion auf Fehlerfreiheit geprüft und im dazugehörigen System “probegelaufen”.



Dem Würfel auf der Spur. Lösen Sie das Geheimnis der Farben

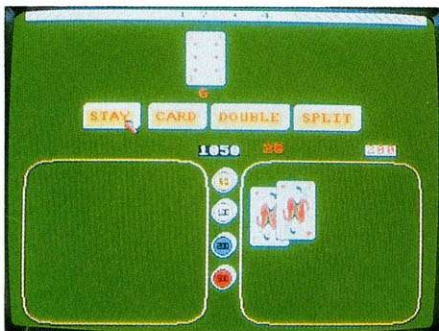


**Spielen Sie Discjockey
mit der »Jingle-
Machine«**

Alle Listings und Programme auf Diskette –
Computer einschalten – Diskette einlegen –
los geht's



BOX



Setzen Sie Ihr Glück auf eine Karte. Mit »17+4« können Sie gegen Computer oder Partner spielen

24,- DM

Wenn Sie über den DMV-Bestellservice bestellen, gilt folgendes:

Inland:		Ausland:	
Einzelpreis	24,- DM	Einzelpreis	24,- DM
zzgl. Versandkosten	4,- DM	zzgl. Versandkosten	6,- DM
Endpreis	28,- DM	Endpreis	30,- DM

Zahlungsweise:

Am einfachsten per Vorkasse (Verrechnungsscheck) oder als Nachnahme zuzüglich der Nachnahmegebühr. (Bei Lieferungen in das Ausland ist Nachnahme nicht möglich.)

Bitte benutzen Sie die Bestellkarte.

DMV-Verlag · Postfach 250 · 3440 Eschwege



Basic

C

Modula-2

Assembler

Lauffähiges Programm

Inhalt der Databox AMIGA DOS 10/90

X	X	Wer eine Lautsprecherbox bauen will, der muß für die Volumenberechnung sorgen. Ein Amiga-Basic-Programm macht's möglich.
X	X	»Pushit« fördert das logische Denken.
X X	X	Unser Requester-Kurs bringt Ihnen die verschiedenen Aufrufe zu unterschiedlichen Programmiersprachen näher. Beispiele in »C«, »AmigaBasic« und »Assembler«.
X	X	Wie Prozeduren aufgebaut sind, können Sie den Beispiel-Listings aus unserem Modula2-Kurs entnehmen.
X X	X	Der Mauszeiger als Informationsträger. Wichtige Daten wie Uhrzeit und Speicherplatz auf einen Blick.
X X	X	Die Jingle-Machine läßt Ihren Amiga swingen.
X	X	Kleine Programme zu unserem Assembler-Kurs.

Und zusätzlich auf der Diskette:

»17+4«, ein grafisches Glücksspiel, das in GFA-Basic geschrieben wurde und für ein bis zwei Personen geeignet ist.

Faszination

Grafik

Der Amiga verfügt über eine faszinierende Grafik. Möchten Sie wissen, wie die Bilder auf den Bildschirm gelangen? Genaueres erfahren Sie in unserem Artikel.





den Sie feststellen, daß jedes Bild aus einzelnen Punkten besteht. Die Anzahl der darstellbaren Punkte nennt man Auflösung.

Je tiefer, desto bunter

fikrogramm ist »Deluxe Paint III«.

Bei der **Vektorgrafik** wird das Bild aus Koordinaten berechnet. Diese Koordinaten stellen Punkte in einem (fiktiven) Raum dar, die durch Linien verbunden sind. Durch diese Punkte-Linien-Verbindungen (Vektoren) wird ein Körper definiert. Damit ein Programm aber einen Körper "sichtbar" machen kann, müssen die physikalischen Eigenschaften des Körpers (Farben, Material, Lichtbrechung und so weiter) dem Programm ebenfalls bekannt sein. Die Darstellung geschieht dann aber wiederum in Pixeln. Bei der Vektorgrafik kann nun noch zwischen 2D- und 3D-Grafik unterschieden werden. 2D-Vektorprogramme sind zum Beispiel die meisten CAD-Programme. Die bekanntesten Vertreter der 3D-Programme sind »Videospace 3D« und alle Raytracing-Programme.

Ungeachtet der Art der Bilderzeugung (Pixel oder Vektor) muß das Bild irgendwie auf dem Monitor dargestellt werden. Wenn Sie einmal sehr nahe an Ihren eingeschalteten Bildschirm herangehen, wer-

Der Amiga ist in der Lage, verschiedene Bildschirmauflösungen darzustellen. Die geringste Auflösungsstufe besteht bei einem PAL-Amiga aus 320 mal 256 (= 81920) Bildschirmpunkten und wird als Low-Resolution bezeichnet. Im Low-Resolution-Modus ist der Amiga in der Lage, bis zu 4096 verschiedene Farben zu benutzen. Der 4096-Farben-Modus wird HAM-Modus (hold-and-modify, zu deutsch: halten und verändern) genannt. Ohne den HAM-Modus stehen bis zu 32 Farben zu Verfügung.

Die nächste Auflösungsstufe heißt High-Resolution, in einigen Programmen auch Medium-Resolution genannt, und bringt bei einem PAL-Amiga 640 mal 256 (= 163820) Punkte auf den Monitor. In diesem Zustand können jedoch maximal 16 Farben verwendet werden. Der HAM-Modus arbeitet nicht mit High-Resolution zusammen, obwohl es einem Hersteller eines Digitizers bereits gelungen ist, dies zu umgehen.

Eine ganz besondere Art der Darstellung ist das Interlace-Verfahren (Zeilensprung). Das Interlace-Verfahren ist bereits

berüchtigt, da es zu einem mitunter sehr störenden Bildschirmflimmern kommt. Hier wird zwischen jede Zeile eines Bildes eine zweite Zeile eingeblendet, so daß sich die vertikale Auflösung verdoppelt. Im Low-Resolution-Modus sind also 320 mal 512 (= 163840) Bildschirmpunkte erreichbar, und das sowohl im 32-Farben- als auch im HAM-Modus. Die Kombination von High-Resolution und Interlace bringt es nun auf ganze 640 mal 512 (= 327680) Punkte, dies allerdings ebenfalls nur mit 16 Farben. Die Anzahl der maximal darstellbaren Farben richtet sich also nach der Grundauf Auflösung (Low-Resolution oder High-Resolution).

Die Daten des Bildes, das dargestellt werden soll, müssen ja irgendwo im Speicher stehen. Aber wieviel Speicherplatz verbraucht nun so ein Bild? Um den Speicherplatz zu errechnen, müssen wir erst einmal herausfinden, wie denn nun die ganzen Farben zustande kommen.

Der Speicher begrenzt die Farben

Der Amiga stellt seine Grafiken in sogenannten Bitplanes dar. Diese Bitplanes sind Speicher-Landkarten, und eine Menge von Bitplanes wird als Bitmap bezeichnet. Die Farbe eines Bildschirmpunktes ergibt sich daraus, auf welchen Bitplanes ein Bit gesetzt wurde. Die Anzahl der Bitplanes gibt darüber Aufschluß, wie viele Farben darstellbar sind. Die Zahl 2 wird mit der Anzahl potenziert und ergibt die Menge der verfügbaren Farben. Ein Beispiel: Ein Bildschirm hat eine Bitmap, die aus 3 Bitplanes besteht. Also lautet die Rechnung $2^3 = 8$. Es sind also 8 Farben verfügbar. Die größte An-

Wer den Amiga zum ersten Mal in Aktion sieht, ist neben dem Sound auch von der verblüffenden Grafik fasziniert. Vielleicht war eben diese auch einer der wichtigsten Gründe für einige unserer Leser, sich diesen Rechner zuzulegen. Wer sich jedoch näher mit dieser Materie auseinandersetzt und nicht nur Bilder malen, sondern diese auch in eigenen Programmen einsetzen will, kommt um einige Grundlagen nicht herum.

Zunächst kann man die Art der Grafikerzeugung in zwei Bereiche einteilen. **Pixelgrafik** ist zweidimensional und besteht aus einzelnen Bildschirmpunkten, die das fertige Bild ergeben. Hier sind von Amiga keine größeren Berechnungen nötig. Ein typisches Pixelgra-

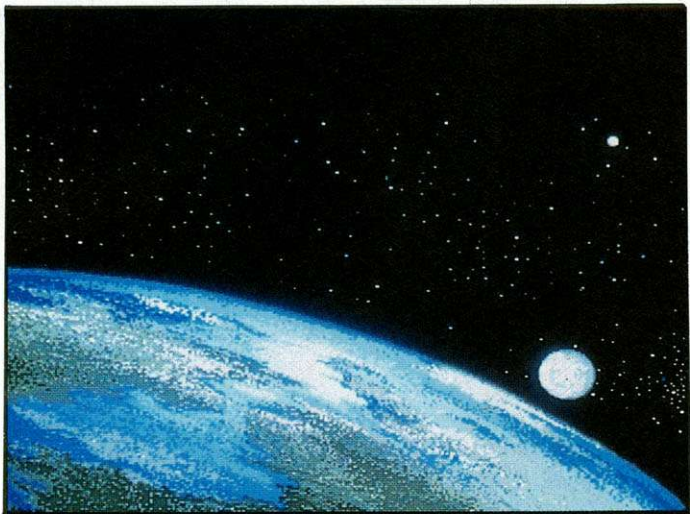


Bild 1. Pixelpracht in 32 Farben...

zahl an Bitplanes beträgt 5 und ergibt 32 Farben. Wenn Sie eine sechste Bitplane einrichten, befinden Sie sich bereits im HAM-Modus. Diese sechste Bitplane dient dazu, die anderen 5 Bitplanes zu verändern, um so auf 4096 Farben zu gelangen. Ein ganz besonderer Darstellungsmodus ist noch der Extra-Halfbright-Modus. Hier dient die sechste Bitplane ebenfalls zur Modifikation der restlichen 5 Bitplanes, allerdings werden hier zu den üblichen 32 Farben noch einmal dieselben 32 Farben zur Verfügung gestellt, die nur die halbe Leuchtintensität haben.

Wir haben jetzt alle Faktoren zusammen, um den Speicherplatz zu ermitteln, den ein Bild verbraucht. Die Formel dafür ist recht einfach: (vertikale Auflösung mal horizontale Auflösung) mal Bitplanes geteilt durch 8. Bei einem Low-Resolution-Bild mit 32 Farben lautet die Rechnung $((320 \times 256) \times 5) / 8 = 51200$ Byte beziehungsweise 50 kByte.

Für die Ewigkeit – Bilder auf Diskette

Einen Haken hat die Sache jedoch. Die Hardware des Amiga ist nur in der Lage, auf Grafikdaten zuzugreifen, die im Chip-Mem liegen. Als Chip-Mem wird der Speicherbereich bezeichnet, der direkt vom Grafikprozessor (AGNUS) adressiert werden kann. Bei Rechnern, die noch den alten Fat AGNUS besitzen, sind dies die unteren 512 kByte, der neuere Big AGNUS kann bis zu 1

MByte Chip-Mem adressieren. Es ist also nicht möglich, trotz vorhandenen Speichers beliebig viele Bilder auf verschiedenen Screens darzustellen.

Wenn Sie sich einmal vom CLI aus mit dem List-Befehl die Längen diverser Bilder ansehen, werden Sie feststellen, daß kaum ein LowRes-32-Farben-Bild an die berechneten 50 kByte herankommt. Woran liegt denn das nun wieder? Die Lösung heißt IFF und bedeutet Interchange File Format, das heißt ein zwischen verschiedenen Programmen austauschbares Dateiformat. IFF-Dateien gibt es inzwischen für Musik, Texte, Animationen und eben auch für Grafiken. Wir werden uns hier ausschließlich mit dem IFF-Format für Pixelgrafik beschäftigen, das IFF-ILBM-Format genannt wird.

IFF-ILBM-Dateien setzen sich aus sogenannten Chunks zusammen, das heißt auf

Auflösung	Punkte	Farben	Speicher
Low-Resolution	320x256	32	50 kByte
Low-Resolution + Interlace	320x512	32	100 kByte
High-Resolution	640x256	16	80 kByte
High-Resolution + Interlace	640x512	16	160 kByte
Hold-and-modify	320x256	4096	60 kByte
Hold-and-modify + Interlace	320x512	4096	120 kByte
Extra-Halfbright	320x256	64	64 kByte

Tabelle 1. Auflösungsstufen des PAL-Amiga

deutsch "Klumpen" oder "Brocken". Diese Klumpen müssen eine bestimmte Form haben, um als solche erkannt zu werden.

IFF – die Lösung zum Bilderaustausch

Eine IFF-ILBM-Datei besteht, wie alle Dateien auch, aus Bytes. Mehrere dieser Bytes bilden einen Chunk, und mehrere dieser Chunks bilden die IFF-ILBM-Datei.

Der erste Chunk jeder IFF-Datei heißt »FORM« und besteht aus dem Wort »FORM« (4 Byte) und der Länge der nachfolgenden Daten (ebenfalls 4 Byte). Anhand dieser 8 Bytes erkennt ein lesendes Programm schon einmal, ob es sich überhaupt um eine IFF-Datei handelt. Verständlicherweise kann ein Grafikprogramm (selten) etwas mit einem IFF-Musikstück anfangen. Zu diesem Zweck wird nach dem FORM-Chunk das 4 Byte lange Wort »ILBM« geschrieben. ILBM steht für Interleave Bitmap, was nichts weiter heißt als zusammenhängende Bitmap. Hier gibt es übrigens noch ein altes For-

mat, welches sich ACBM nennt und von dem "antiken" Programm »Graphicraft« benutzt wird.

Nachdem nun erkannt wurde, daß ein ILBM-Bild vorliegt, wird der BMHD-Chunk gelesen. Dieser »BitMapHeader« besteht (wie sollte es anders sein) aus dem Wort »BMHD«. Nun folgen aber die ersten bildspezifischen Daten. Zuerst werden in je 2 Byte die Breite und die Höhe des Bildes gelesen. Anschließend folgt die horizontale und vertikale Position des Bildes in je 2 Byte. Dies ist wichtig, falls es sich um einen Bildausschnitt handelt. Nach den Positions-Bytes folgt die Tiefe des Bildes, also die Anzahl der Bitplanes, in einem Byte sowie die Masken und das Kompressions-Byte. Die Masken teilen dem Programm mit, welche Bildpunkte einer Grafik eventuell überschrieben werden dürfen und welche nicht. Das Kompressions-Byte gibt Aufschluß darüber, ob ein Bild gepackt abgespeichert wurde, und sein Wert kann entweder null (nicht gepackt) oder eins (gepackt) betragen. Mit dem Prinzip des Packens werden wir uns etwas später befassen.

Bits und Bytes – ohne sie geht es nicht!

Wenn Sie die Anzahl der Bytes mitgezählt haben, werden Sie feststellen, daß alle Daten bisher gerade Byte-Zahlen verwendet haben. Dies ist aber nach den letzten 3 Byte nicht mehr der Fall. Damit die ganze IFF-Datei wieder in die Reihe kommt, wird nun ein Füll-Byte eingefügt, und schon stimmt es wieder. Das lesende Programm wird dieses Füll-Byte einfach ignorieren.

Im Anschluß an dieses Füll-Byte folgt die Nummer der transparenten Farbe mit eben-

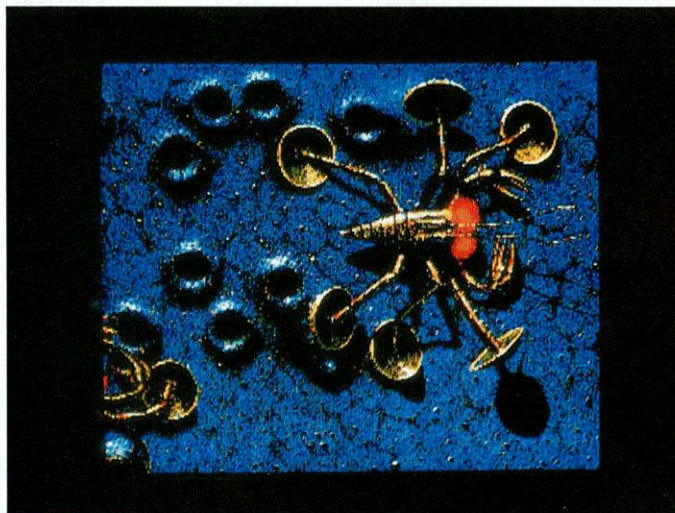


Bild 2. ...oder im HAM-Modus

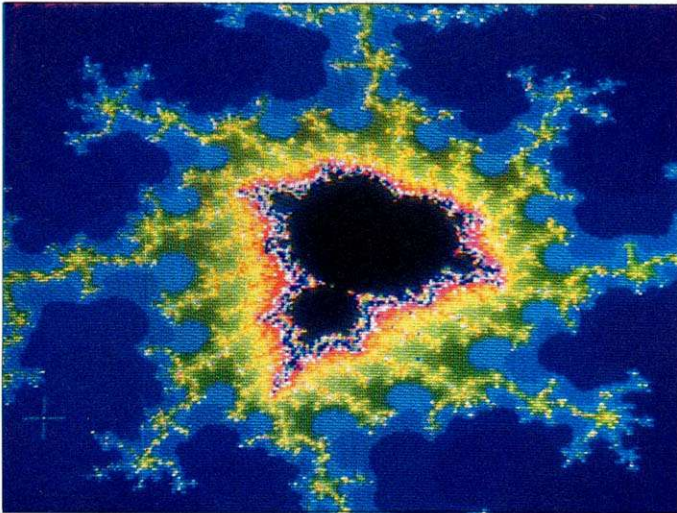


Bild 3. Das Apfelmännchen aus dem Chaos

falls 2 Byte. In den meisten Fällen ist dies die Hintergrundfarbe und hat den Wert null.

Ganz besondere Daten stellen die vier nun folgenden Bytes dar. Sie nennen sich »AspectX« und »AspectY«. In ihnen wird das Seitenverhältnis der Grafik angegeben. Dies ist wichtig, wenn eine Grafik in eine andere Auflösung übertragen (zum Beispiel von Low-Resolution nach High-Resolution/Interlace) oder gar auf einen anderen Rechner portiert werden soll.

Die letzten 8 Byte des BMHD-Chunks beinhalten die Breite und Höhe des Screens (Bildschirms), auf dem die Grafik dargestellt werden soll.

War das etwa zuviel? Schließlich müssen ja Sonderregelungen in einem Format, das zum allgemeinen Datenaustausch herangezogen wird, auch Beachtung finden. Sollten Sie selbst einmal ein Programm schreiben, das ILBM-Dateien liest, können Sie getrost unwichtige Chunks »überlesen«. Sie sollten aber dann wissen, welche Länge der jeweilige Chunk hat.

Vielleicht haben Sie sich schon gewundert, warum das Grafikprogramm immer die korrekten 32 Farben (bei einem Low-Resolution-Bild) verwendet, obwohl doch insgesamt 4096 Farben zur Auswahl stehen. Die Werte dieser Farbregister sind im CMAP-Chunk abgespeichert und setzen sich aus je einem Byte für den Rot-, Grün- und Blauanteil einer Farbe zusammen. Da jedes dieser Bytes eine Zahl zwischen 0 und 15 beinhaltet, also 16 verschiedene Werte möglich sind,

ist auch das Rätsel der 4096 Farben gelöst: 16 mal 16 mal 16 ist gleich 4096! Und eben diese Werte stehen für jede Farbnummer im CMAP-Chunk. Hierbei kann es vorkommen, daß eine ungerade Anzahl Bytes vorliegt. Weil ja alles schön gerade sein muß, wird auch hier notfalls ein Füll-Byte eingefügt.

Einfache Animation durch Color-Cycling

Sicher haben Sie schon einmal Bilder gesehen, in denen einfache Bewegungen stattfinden: Regen fällt vom Himmel, ein Fluß fließt, oder ein Blitz durchbricht den Himmel. Diese kleinen Animationen werden durch Color-Cycling ermöglicht. Dazu werden bestimmte Farbnummern einfach durchkopiert, das heißt, alle Farben rutschen eine Stelle

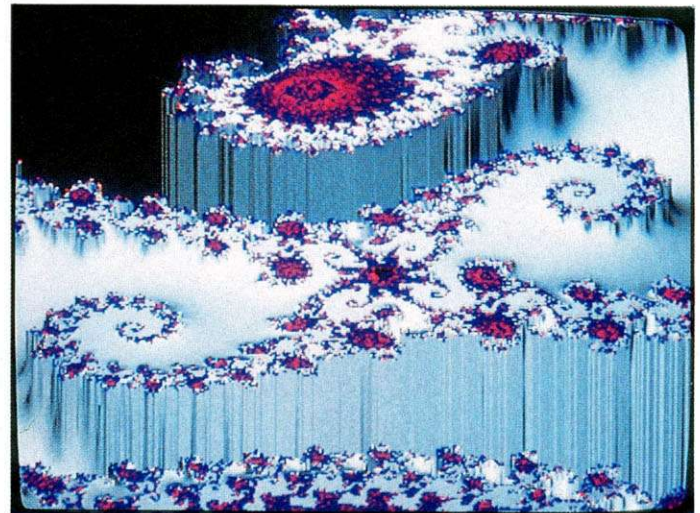


Bild 4. Fraktale Landschaft

nach vorn, und die erste Farbe wird hinten wieder angefügt. Welche Farbnummern dies im einzelnen sind, wird im CRNG-Chunk (ColorRaNGe) festgelegt. Seltsamerweise stehen eventuelle Füll-Bytes hier am Anfang des Chunks, gefolgt von der Geschwindigkeit (2 Byte), dem Aktivitätsmodus (eingeschaltet oder nicht; 2 Byte) und den Werten für die niedrigste und die höchste Farbnummer (je 1 Byte).

Analog zum CRNG-Chunk gibt es noch den CCRT-Chunk (ColorCycle Range and Timing). Der CCRT-Chunk gehört wieder einmal zu Commodores Graphicraft, der CRNG-Chunk zu Electronic Arts Deluxe Paint. Trotzdem soll dies nicht unerwähnt bleiben. Der CCRT-Chunk besteht aus der Richtung (vorwärts oder rückwärts zählend, 2 Byte), den Werten für die erste und letzte Farbnummer (je 1 Byte), die Zeit in Sekunden und Mikrosekunden

(je 4 Byte) und dem obligatorischen Füllwert (2 Byte).

Um Langeweile vorzubeugen und den Rahmen dieses Artikels nicht zu sprengen, werden wir auf »exotische« Chunks wie GRAB (für »Hotspots«, zum Beispiel Mauszeiger), DEST (zur Belegung von Bitplanes) und SPRT (Sprite-Chunk) nicht näher eingehen. Um einen wichtigen Chunk kommen wir jedoch nicht herum: der CAMG-Chunk. CAMG heißt nichts anderes als »Commodore AMiGa computer« und enthält den sogenannten View-Mode. Das hört sich neu an, ist es aber nicht. View-Mode ist nichts anderes als die Aussage, ob es sich um ein Low- oder High-Resolution-Bild handelt, ob Interlace eingeschaltet ist oder aber der HAM-Modus aktiviert werden muß. Der CAMG-Chunk ist ganze 4 Byte lang.

Alles hat ein Ende...

Jetzt ist es endlich soweit! Es folgt der BODY-Chunk, der nun endlich die Bilddaten, den »Körper« des Bildes, enthält. Und das ist es schließlich, was wir sehen wollen. Die Daten des Bildes werden in Zeilen abgespeichert. Bei einem Low-Resolution-Bild mit 4 Farben werden zuerst die Daten der ersten Zeile der ersten Bitplane und dann die erste Zeile der zweiten Bitplane gespeichert. Darauf folgt die zweite Zeile der ersten Bitplane, die zweite Zeile der zweiten Bitplane, und so weiter. Daher kommt auch der Name des ILBM-Chunks: zusammenhän-



Bild 5. Eine Vektor-Szene



Bild 6. Fotorealismus mit Raytracing

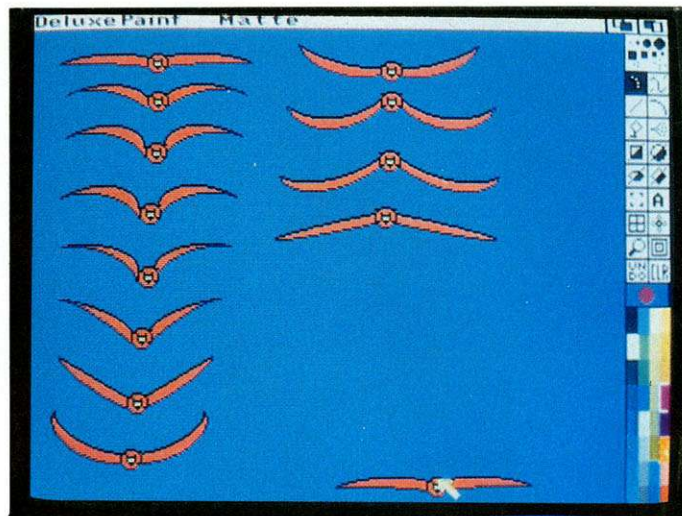


Bild 7. Pixelanimation verlangt Arbeit

gende Bitplanes! Diese Daten des BODY-Chunks sind es auch, die gepackt werden.

Gepackte Daten sparen Diskettenplatz

Ob die Daten im Body-Chunk gepackt vorliegen, wurde bereits im Kompressions-Byte des BMHD-Chunks festgestellt. Sind die Daten nicht gepackt, können sie einfach in den Speicher eingelesen werden. Bei gepackten Daten gibt es jedoch etwas an Arbeit.

Stellen Sie sich vor, auf einem schwarzen Hintergrund ist ein kleines Motiv dargestellt. Jedes Byte der Bitmap, auf der "nichts" zu sehen ist, also der Hintergrund, beträgt null. Beim Packen sieht das Programm ganz einfach nach, wie viele Nullen aufeinanderfolgen. Diese Anzahl merkt es sich und fügt sie an ein Byte an, dessen erstes Bit gesetzt wird. Wenn in einem Byte das erste Bit eingeschaltet ist, beträgt der Wert des Bytes

mindestens 128. Da aber nur die Werte 0 bis 15 für ein Farbregister erlaubt sind (wir erwähnten dies bereits), weiß das Programm, daß nun gepackte Daten folgen. Aus der Anzahl ermittelt es, wieviele Werte es ursprünglich waren, und aus dem folgenden Byte entnimmt es, welcher Wert dies ist. So kann aus einer Reihe von zum Beispiel 50 Null-Bytes eine 3 Byte lange gepackte Reihe erzeugt werden: Kenn-Byte, Anzahl-Byte und Wert-Byte.

Und damit ist unser kleiner IFF-Lehrgang endgültig zu Ende.

Wo kommen denn die kleinen Bilder her?

Mit an Sicherheit grenzender Wahrscheinlichkeit wird auch bei Ihnen der Wunsch erwachen, nicht nur Bilder mehr oder weniger bekannter Künstler zu betrachten, sondern selbst einmal kreativ zu werden. Außer dem Talent, das Sie

selbst mitbringen müssen, stehen Ihnen mehrere Möglichkeiten der Bilderzeugung offen.

Die einfachste und zugleich schwierigste ist das Malen von Pixelgrafik. Es befinden sich etliche Programme auf dem Markt, die Sie hier hilfreich unterstützen können. Der bekannteste Vertreter der Pixelmalprogramme ist »Deluxe Paint«, das inzwischen bereits in der dritten Version vorliegt. In diesem Programm und allen seinen Artverwandten bearbeiten Sie den Bildschirm wie eine Leinwand. Sie können verschieden große "Pinsel" benutzen oder sogar eigene Pinsel definieren, indem Sie Bildteile ausschneiden. Außerdem stehen viele Funktionen zur Erzeugung geometrischer Figuren zur Verfügung. So können Sie auf einfache Weise Linien, Kreise, Rechtecke und ausgefüllte Polygone (mehrsichtige Flächen) erschaffen. Zur Bildbearbeitung stehen selbstverständlich noch viele Funktionen mehr zur Auswahl. So können Teile des Bildes

ausgeschnitten oder kopiert, gedreht und gespiegelt, verbogen oder verzerrt werden. Alle Funktionen sind mit der Maus bedienbar, außer beim Platzieren von Text in einer Grafik. Hier müssen Sie noch zur Tastatur greifen.

Die fertigen (oder unvollendeten) Bilder können selbstverständlich auf Diskette gesichert und wieder eingeladen werden. Was ja auch einleuchtend ist, sonst wäre ja die ganze Pixelpracht nach einem Reset dahin. Und dank des IFF-ILBM-Formats können diese Bilder dann auch bei anderen Malprogrammen oder sogar einigen Textverarbeitungen weiterverwendet werden.

Apfelmann und Lichtstrahl – Bilder berechnen

Wer sich an die "Handarbeit" der Bilderstellung nicht herantraut, weil er sich nicht musisch inspiriert fühlt, kann seine Bilder auch vom Amiga berechnen lassen. Hier reicht es



Bild 8. 3D-Animationssequenz

natürlich nicht, dem Amiga ein kerniges "Mal mir mal ein Schaf!" entgegenzuschleudern. Je nach Programm ist mehr oder weniger Vorarbeit nötig.

Am einfachsten zu bedienen sind die meisten "Apfelmännchen"-Programme. Die Bezeichnung "Apfelmännchen" rührt von einem bestimmten grafischen Muster her, das immer wieder in diesen Bildern auftaucht. Korrekt heißen diese Bilder "Mandelbrotsche Mengen" und wurden von einem Mathematiker namens Mandelbrot beschrieben.

Neben diesen "Apfelmännchen" gibt es auch noch Julia-Mengen, graphale Muster und vieles mehr. Alle diese Programme sind Randprodukte der Chaosforschung und arbeiten mit sogenannten Fraktalzahlen. Dies sind Zahlen, die nach bestimmten Algorithmen (Rechenvorschriften) errechnet wurden und in eine Bitmap kopiert werden. Die sich daraus ergebenden Grafiken sind selbstähnlich (weshalb das "Apfelmännchen" immer wieder auftaucht), was laut neuesten wissenschaftlichen Erkenntnissen eine Eigenschaft des Chaos ist.

Alle Fraktalprogramme verlangen vom Benutzer bestimmte Werte, die dem Algorithmus zugänglich sind. Dies können zum Beispiel Werte wie Rechentiefe, horizontale und vertikale Ausdehnung und ähnliches sein. Einige Programme erlauben es auch, mit der Maus einen Bildausschnitt zu definieren und diesen neu berechnen und vergrößern zu lassen oder sogar in dreidimensionale Darstellung umzuwandeln.

Fraktalgrafikprogramme sind häufig in der Public Domain zu finden, es sind aber auch einige kommerzielle Produkte auf dem Markt. Sehen Sie sich doch einmal ein paar Fraktalbilder an. Sie werden merken, wie faszinierend Chaos sein kann.

Flächen und Körper – es wird räumlich

Wesentlich mehr Arbeit verlangen 3D-Grafikprogramme vom Benutzer. Hier ist es nicht damit getan, mal ein paar Werte einzugeben und einige Sekun-

den bis Minuten auf das Bild zu warten. Ein 3D-Programm möchte alles haarklein wissen, und die Aufgabe, alle Werte einzugeben, liegt bei Ihnen. Glücklicherweise verfügen die meisten Programme über einen Editor, in dem die Körper und Flächen nicht mit Zahlenkoordinaten eingegeben werden müssen, sondern sichtbar dargestellt werden.

Einige Animationsprogramme, zum Beispiel »Videoscape 3D«, sind in der Lage, ein IFF-ILBM-Bild zu berechnen und abzuspeichern. Die Stärke dieser Programme liegt aber, wie der Name schon sagt, in der Animation. Mit Animationen werden wir uns später noch beschäftigen, zunächst interessieren uns nur die Programme, die aus dreidimensionalen Szenen zweidimensionale Bilder berechnen.

3D-Programme arbeiten meist mit "Szenen". Eine Szene ist die relative Platzierung von Körpern in einem fiktiven Raum. Das mag sich kompliziert anfühlen, ist aber nicht ganz so schlimm. Stellen Sie sich ganz einfach vor, eine Kugel liegt in einem leeren Zimmer. Welche Daten können Sie ermitteln? Zuerst ist es möglich, die Position der Kugel im Raum festzustellen, indem Sie mit einem Maßband den Abstand der Kugel von allen Wänden messen. Bei der Gelegenheit können Sie auch gleich Höhe und Breite sämtlicher Wände ermitteln und einen eventuellen Neigungswinkel der einen oder anderen Wand feststellen (so etwas soll es ja geben). Jetzt brauchen Sie noch die Position einer vorhandenen Lichtquelle (ohne Licht sieht man bekanntlich wenig) sowie die Helligkeit des Lichts. Zuletzt sehen Sie noch nach, welchen Durchmesser die Kugel hat. Die Werte der Szene sind nun komplett und könnten einem 3D-Editor übergeben werden.

Bei der Programmbedienung sieht das natürlich etwas anders aus. Sämtliche Körper und Räume existieren zunächst nur in Ihrer Phantasie oder als Koordinatenwerte im Programm. Diese Koordinaten, also Punkte im Raum, werden Vektoren genannt. Daher rührt auch der Name "Vektorgrafik".

Vektorgrafik sieht ein wenig langweilig aus, weil sie meist

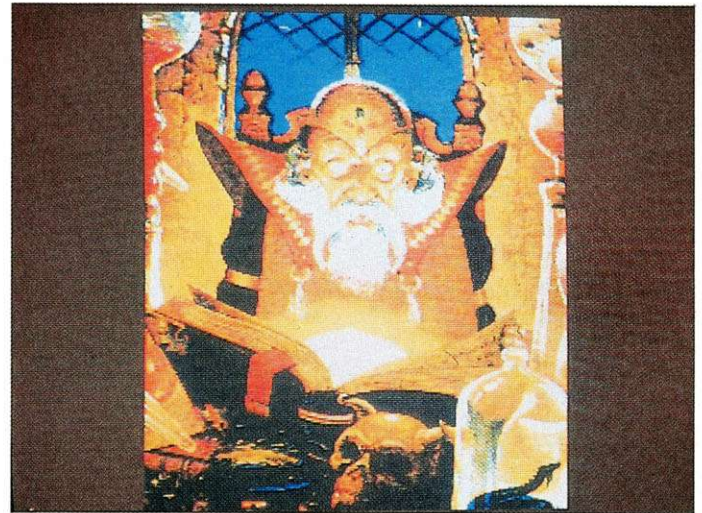


Bild 9. Digitalisierte Pixelgrafik

nur aus Linien besteht. Deshalb erlauben es einige Vektorprogramme, den erstellten Körpern Materialeigenschaften zu verleihen. Unbedingt nötig ist dies sogar bei den "Ray-Tracern". Raytracing bedeutet Strahlenverfolgung. Hier ist nicht etwa eine Umweltschutzorganisation gemeint, sondern es handelt sich bei den Strahlen um Licht. Natürlich nicht um "gewöhnliches" Licht, sondern um fiktives Licht, das vom Computer berechnet wird.

Und sie bewegt sich doch – die Animation

Greifen wir das Beispiel der Kugel im Raum noch einmal auf. Nehmen wir einmal an, Sie möchten die Wände blau und weiß haben. Den Boden des Raums belegen Sie mit einem Schachbrettmuster, die Kugel selbst soll verchromt sein. Diese Angaben teilen Sie dem Raytracing-Programm mit und starten den Berechnungsvorgang. Eine fiktive Kamera simuliert das Auge des Betrachters und fängt alle Lichtstrahlen auf, die von irgendwelchen Objekten reflektiert oder gebrochen werden. Jetzt ist es an der Zeit, sich eine Tasse Kaffee zu brühen oder gleich ins Bett zu gehen. Denn die Berechnungszeiten der meisten Raytracer betragen zwischen einer und 12 Stunden.

Wenn das Raytracing-Bild fertig berechnet ist, können Sie die Pixelgrafik im IFF-ILBM-Format abspeichern und

Ihrer Sammlung zufügen. Raytracing-Bilder sehen meist sehr fotorealistisch aus, vor allem im HAM-Modus, und sind eine der reizvollsten Grafiken auf dem Amiga.

Bisher haben wir nur eine Art der Animation kennengelernt: das Color-Cycling. Es geht aber noch viel besser.

Die komplizierteste Form der Animation ist die Pixelanimation. Das Prinzip ähnelt dem bekannten Zeichentrickfilm: Jedes Bild (beziehungsweise der Teil des Bildes, der bewegt wird) wird gesondert gezeichnet und das ganze Produkt zu einer Animationssequenz zusammengebunden. Hier gibt es ebenfalls einige Programme, die den Animator bei seiner Arbeit unterstützen. Animator ist hier keinesfalls abwertend zu verstehen, genauso wenig würde man eine weibliche Animatuerin als Anmierdame bezeichnen.

Durch Zeit und Raum – die 3D-Animation

Bekannte Vertreter von Pixelanimationsprogrammen sind »Deluxe Video«, »Fantavision« oder »Pageflipper«. Mit solchen Programmen sind auch Raytracing-Animationen möglich, allerdings muß die Pixelgrafik vorher von einem Raytracer berechnet werden, bevor sie einem Pixelanimationsprogramm zur Verarbeitung übergeben werden kann. Bei einer Animationssequenz von 50 Bildern können Sie sich selbst ausrechnen, wie lange die Berechnung der Bilder dauert. Etwas bequemer geht



es mit einem kombinierten Raytracing- und Animationsprogramm wie zum Beispiel »Sculpt 3D/4D«. Trotzdem müssen Sie mit einem erheblichen Zeitaufwand rechnen.

Schnelle Computer-Animationen sind Ihnen bestimmt aus dem Fernsehen bekannt. Die Kamera schwenkt in ein Stadion und dreht eine Runde über die Aschenbahn, oder in einer Automobilwerbung entsteht aus Vektoren (wir wissen ja jetzt, was das ist) ein Nobelschlitten, der anschließend über eine viel zu glatte Straße donnert.

Solche Animationen werden meist mit sehr teuren Spezialrechnern erstellt. Aber der Amiga ist mit der entsprechenden Software in der Lage, ver-

gleichbare Effekte zu erschaffen.

Damit die Berechnung von Körpern nicht allzu lange dauert, verzichten die meisten 3D-Animationsprogramme auf Materialeingaben für die Körper. Auch werden die Flächen nicht wie beim Raytracing berechnet, sondern das Programm »sieht« nur nach, ob das imaginäre Licht im rechten Winkel, im stumpfen Winkel oder gar nicht auf eine Fläche trifft. Die entsprechende Fläche wird dann komplett mehr oder weniger hell gefärbt. Das einzige, was der Benutzer wählen kann, ist die Grundfarbe des entsprechenden Körpers.

Was bewegt sich denn eigentlich bei einer Animation? Hier

gibt es zwei Möglichkeiten. Entweder die Körper ändern ihre Position, oder die Kamera (der Betrachter der Szene) bewegt sich oder aber beides. Ist eine Szene erst einmal fertig konstruiert, können Sie die Objekt- und Kamerabewegungen in »Scripts« festlegen. Scripts sind Anweisungsfolgen, die vom Animationsprogramm ausgeführt werden. Sie brauchen also nicht wie beim Raytracing jede Position der Kamera und Objekte gesondert festzulegen, sondern teilen dem Programm die Bewegungsphasen mit. Ein Script könnte sich zum Beispiel folgendermaßen aufbauen:

- Kamera 0,0,0
- Objekt 0,0,0
- Kamera auf Y-Achse im Kreis bewegen in 6°-Schritten, bis 360° erreicht sind.

Kamera läuft!

In diesem (natürlich nur symbolischen) Script wird die Kamera einmal rund um das Objekt gefahren. Bei der angegebenen Schrittweite von 6° ergibt dies 60 Einzelbilder. Diese Einzelbilder werden vom Animationsprogramm errechnet und als Animationsdatei auf Diskette gespeichert. Je nach Programm und Komplexität der Szene dauert die Berechnung zwischen 10 Minuten und einer Stunde. Der Unterschied zur normalen Pixelanimation ist der, daß die Bilder hier nicht einfach hintereinander abgespeichert werden, sondern das Programm speichert nur die Unterschiede zwischen den einzelnen Bildern. So können Animationen mit einer hohen Anzahl Bilder platz- und speichersparend gesichert werden. Eine Animationssequenz aus 60 Einzelbildern im Low-Resolution-Format würde sonst ja 3 MByte belegen, und das wäre wohl doch etwas viel. Der bekannteste Vertreter dieser 3D-Animationsprogramme ist »Videoscape 3D«.

Eine weitere Methode, um Bilder im Amiga zu erzeugen, ist das Digitalisieren. Hier wird eine Videoquelle, zum Beispiel das Signal vom Videorecorder, einer Videokamera oder eines Scanners, in den Amiga eingespeist und in Pixelgrafik umgerechnet.

Neben dem Digitizer-Programm wird eine Hardware benötigt, die die analogen Videosignale in digitale Informationen umwandelt. Je schneller dies geschieht, desto besser. Beste Ergebnisse lassen sich mit einer Schwarzweiß-Videokamera erzielen, die im Elektronik-Versandhandel ab zirka 150 DM erhältlich ist. Das Bild wird in drei Farben digitalisiert, erst Rot, dann Grün und schließlich Blau. Die drei Bilder werden dann im Amiga zusammengemischt, und es entsteht das fertige Bild. Bei einer Schwarzweißkamera müssen die drei Farbanteile durch Farbfilter getrennt werden, die während des Digitalisierens vor das Objektiv der Kamera gehalten werden. Wenn Sie mit einer Farbkamera oder einem Videorecorder arbeiten, brauchen Sie noch zusätzliche Hardware: einen RGB-Splitter. Wie Sie sich sicher denken können, spaltet der RGB-Splitter das eingehende Bildsignal in Rot-, Grün- und Blauanteile auf, so daß es in drei Durchgängen eingelesen werden kann. Der Vorgang des Digitalisierens dauert meist weniger als eine halbe Minute, einige Digitizer arbeiten sogar in Echtzeit.

Vom Videoband in den Amiga – Digitalisieren

Die Software zum Digitalisieren ist meist in der Lage, die erstellten Bilder im IFF-ILBM-Format abzuspeichern, so daß diese mit einem Malprogramm nachbearbeitet, verfremdet oder sogar zu einer Pixelanimation zusammengesetzt werden können.

Digitizer sind ab zirka 400 DM erhältlich, ein Farbscanner kostet zirka 2000 DM.

An dieser Stelle möchten wir, auch um den Umfang dieses Artikels nicht zu sprengen, unseren Streifzug durch die Amiga-Grafikwelt beenden. Wir hoffen, Ihr Interesse geweckt zu haben, und wünschen Ihnen viel Erfolg bei Ihren eigenen Kunstwerken.

Im folgenden geben wir Ihnen eine Übersicht über die bekanntesten Grafik- und Animationsprogramme sowie Bezugsquellen für die benötigte Soft- und Hardware.

(tb)

Literaturhinweise

Kurse und Workshops in der AMIGA DOS:

Deluxe-Paint-III-Werkstatt
Heft 1/90 bis Heft 5/90

Butcher-Werkstatt
Heft 6/90 und Heft 7/90

Reflections-Werkstatt
Ab Heft 8/90

Pixelgrafik:

Name: Das große Deluxe Paint III Buch
Autoren: Mirko Langlotz und Nikola Vignjevic
Verlag: Data Becker
ISBN: 3-89011-369-9
Preis: 39,- DM

Name: Deluxe Paint III auf dem Amiga
Autoren: Stein, Welcherer & Partner
Verlag: Markt & Technik
ISBN: 3-89090-355-X
Preis: 49,- DM

Name: Professionelles Arbeiten mit Deluxe Paint II
Autor: Walter Friedhuber
Verlag: Gabriele Lechner
ISBN: 3-926858-01-X
Preis: 39,- DM

Pixelanimation:

Name: Animationen mit Deluxe Paint III
Autor: Walter Friedhuber
Verlag: Gabriele Lechner
ISBN: 3-926858-23-0
Preis: 69,- DM

Name: Amiga Trickstudio A Bookware
Autor: keine Angaben
Verlag: Markt & Technik
ISBN: 3-89090-886-1
Preis: 99,- DM

Name: Computermalschule Trickfilmzeichnen
Autor: Walter Friedhuber
Verlag: Gabriele Lechner
ISBN: 3-926858-07-9
Preis: 59,- DM

3D-Animation und Raytracing:

Name: Amiga 3D-Sprinter Bookware
Autor: Dr. Glaeser und T. Grohser
Verlag: Markt & Technik
ISBN: 3-89090-109-3
Preis: 98,- DM

Name: Das große Animationsbuch
Autoren: Andreas Polk und Peter Schulz
Verlag: Data Becker
ISBN: 3-89011-332-X
Preis: 39,- DM

Name: Turbo Silver 3D Workshop
Autor: Gabriele Lechner
Verlag: Gabriele Lechner
ISBN: 3-926858-12-5
Preis: 69,- DM

Fraktalgrafik:

Name: Fraktale Grafik auf dem Amiga
Autor: H. Knappe
Verlag: Markt & Technik
ISBN: 3-89090-600-1
Preis: 79,- DM

POWER PC-BOARD: BEI ROSSMÖLLER INCL. KNOW HOW-GARANTIE

Lange Zeit Top Secret, nun der Topseller: Das Power PC-Board, das den AMIGA® 500 IBM®-kompatibel macht. Kenner kaufen es für nur 798 DM bei ROSSMÖLLER. Denn bei ROSSMÖLLER ist eine Know How-Garantie gleich im Preis inclusive. Als erfahrener AMIGA®-Zubehörentwickler und renommierter Hardware-Preisträger bietet ROSSMÖLLER fundiertes Know How. Zum Beispiel an der Telefon-Hotline: Dort sind die Entwickler unserer AMIGA®-Produkte von Montag bis Freitag (16-17 Uhr) ganz Ohr für Ihre Fragen. ROSSMÖLLER – Know How- und Service-Qualität aus einer Hand. Kennen Sie eine bessere Adresse für den Kauf Ihres Power PC-Boards?



Der neue Katalog ist da!
– Jetzt anfordern!

Roßmüller Handshake GmbH · Neuer Markt 21 · 5309 Meckenheim · Tel. 02225/2061-62-63

Computer & Zubehör

ONLY ONE

Der Umbausatz, der Ihren AMIGA in die richtige Form bringt.

Der Umbausatz bietet Ihrem AMIGA 500 nicht nur ein neues Gehäuse mit abgesetzter Tastatur, wie bei einem A1000 oder A2000, sondern auch erheblich mehr Platz. So bietet er außer für eine Speichererweiterung unter anderem noch für ein zweites Laufwerk oder für eine 3,5" Festplatte genügend Platz. - Und das alles ohne Lötarbeiten. Auch für die Zukunft ist gesorgt, eine Erweiterungsbox mit vielen Features wird es ab ca. Dezember geben.

Machen Sie Ihren A500 zu einen PC

Mit dem **MW 500 System**
und dem **PC Power-Board**
Kombiangebot

DM 1098,--

Besuchen Sie uns auf der

AMIGA Köln 90



MW 500 System

Grundausstattung, komplett mit Kabelsatz und Tastaturgehäuse

DM 349,--

Warum eine Festplatte mit Gehäuse und nicht ein Gehäuse mit Festplatte ??

40 MB autobootend und MW 500 Gehäuse
Kombiangebot

DM 1399,--

3,5" ext. Laufwerk, durchg. Bus, absch.	DM 189,--
3,5" int. Laufwerk für A2000 u. A500	DM 159,--
5,25" ext. Laufwerk, durchg. Bus absch.	DM 249,--
512 kB Speichererweiterung absch.	DM 149,--
Reismouse die Mousealternative	DM 89,--

Miky Wenngatz
08105/24540

Kennwort AD 10/90
Jägerweg 31
8031 Gilching

Noch schneller, noch besser !

Sie bestellen schriftlich oder telefonisch zwischen 0-24 Uhr und die Ware verläßt innerhalb 24 Stunden unser Haus (sofern wir sie auf Lager haben).

Fragen - Informationen
unsere Hotline erreichen
Sie zwischen 10 u. 14 Uhr.

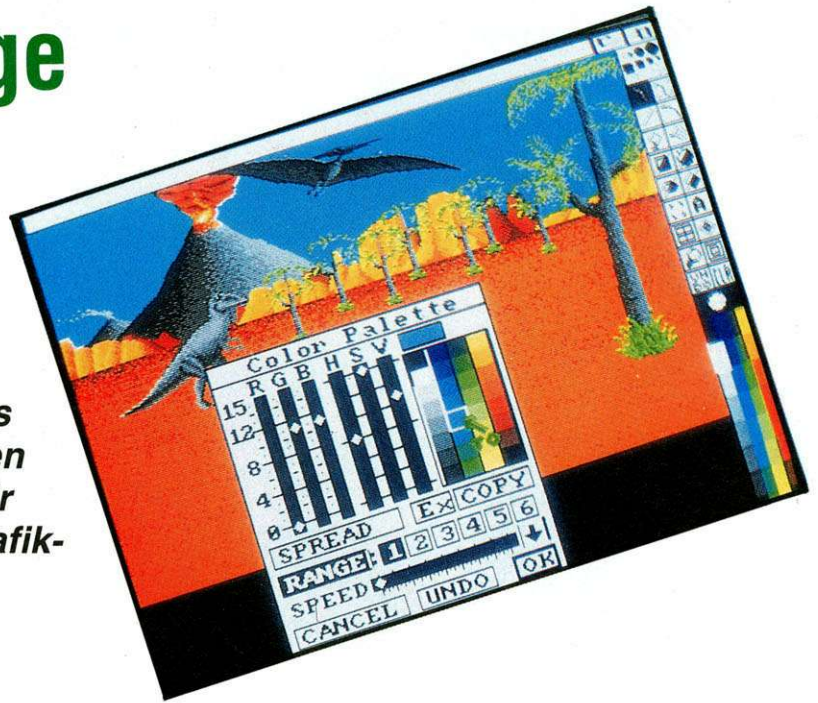
Lieferbedingungen:

Wir liefern ab Lager Gilching, solange der Vorrat reicht, per UPS Nachnahme zuzüglich Versandkosten. Bei Vorkasse frei Haus. DDR per Post-Nachnahme. Ausland per Vorkasse zuzüglich DM 40,- Versandkosten.

Fordern Sie unseren kostenlosen Katalog an.

Grafik-Werkzeuge für jede Anwendung

Wenn Sie die Grafikfähigkeiten des Amiga ausnutzen wollen, benötigen Sie entsprechendes Werkzeug. Wir stellen Ihnen die bekanntesten Grafikprogramme vor.



Die Flut an Grafikprogrammen für den Amiga ist inzwischen nahezu unüberschaubar geworden. Wir haben für Sie die gebräuchlichsten Programme ausgesucht, um Ihnen einen Überblick über die verschiedenen Anwendungsmöglichkeiten zu geben.

Deluxe Paint III – der Klassiker

Die Präsentation der ersten Version von »Deluxe Paint« riß die Zuschauer zu wahren Begeisterungstürmen hin. Nicht umsonst, denn die Möglichkeiten, die dieser Klassiker der Pixel-Malprogramme bot, stellte alles bisher Dagewesene in den Schatten. Inzwischen ist die dritte Version dieses Programms erschienen, und die Vielfalt der Anwendungen ist noch gesteigert worden.

»Deluxe Paint III« kann folgende Auflösungsstufen des Amiga darstellen: Low-Resolution, High-Resolution, Interlace und Extra-Halfbright. Der Over-scan-Modus (Ausschalten des Bildschirmrandes) ist nun ebenfalls möglich.

Zur Bildbearbeitung beziehungsweise -erstellung stehen reichhaltige Funktionen zur Verfügung. Diese Funktionen werden entweder auf dem Bildschirm in einer Werkzeugleiste angezeigt oder können über die Tastatur und die Pull-down-Menüs angewählt werden. Neben den verschieden

großen Pinseln, kann auch ein Bildschirmausschnitt (Brush) definiert werden. Mit diesen Brushes läßt sich jedoch noch viel mehr anstellen. So können diese gespiegelt, gedreht, verbogen oder vergrößert und verkleinert werden. Ein definierter Brush kann im IFF-ILBM-Format auf Diskette gespeichert und in anderen Bildern weiterverwendet werden.

Neben den Pinselfunktionen stehen einige Hilfsmittel zur Erzeugung geometrischer Figuren zur Verfügung. Linien, Rechtecke, Kreise oder Polygone (Vielecke) können ohne Aufwand gezeichnet werden.

»Deluxe Paint III« verfügt über einen Animationsteil, der nach dem Pageflipping-Prinzip funktioniert. Dabei werden verschiedene Bilder »durchgeblättert«, wodurch ein fließender Animationsablauf entsteht. Diese Animationen können auf Diskette oder Harddisk gespeichert werden.

Name: Deluxe Paint III
Pixel-Malprogramm
Hersteller: Electronic Arts
Vertrieb: Fachhandel
Preis: zirka 250 DM

»Deluxe Paint III« ist nicht kopiergeschützt; eine Installation auf Festplatte gestaltet sich problemlos. Wer jedoch über wenig Speicher verfügt (512 kByte), kann nicht alle Funktionen nutzen.

HAM – Malen mit 4096 Farben

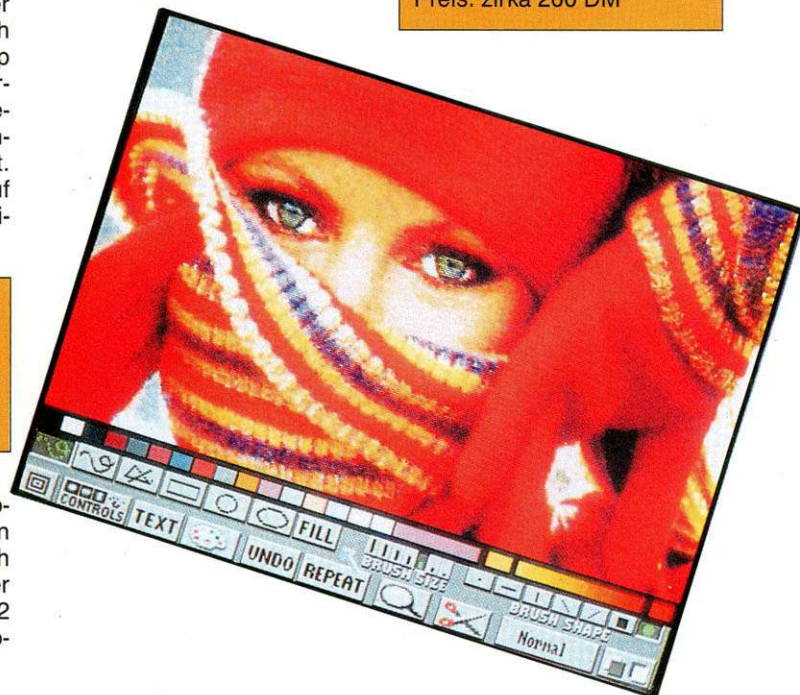
Den farbenreichen HAM-Modus (4096 Farben) des Amiga nutzt das Programm »DIGI-Paint 3«. Die Funktionen dieses Programms sind zum größten Teil identisch mit denen von »Deluxe Paint III«. Einige Besonderheiten sollen jedoch erwähnt werden.

Ein Color-Cycling (zyklisches Durchzählen der Farbre-gister) gibt es im HAM-Modus nicht, da ja alle möglichen Werte der Register bereits verwendet werden. Die Auflösungskombi-

nation High-Resolution plus Interlace ist auch im HAM-Modus nicht möglich, da die Hardware des Amiga dies nicht gestattet.

»DIGI-Paint 3« verfügt jedoch über sehr komfortable Werkzeuge, um HAM-Bilder (auch digitalisierte) zu bearbeiten. So gibt es zum Beispiel eine Funktion, die ein zu dunkles Bild aufhellt oder Pinsel beziehungsweise Textfonts transparent in das Bild einfügt.

Name: DIGI-Paint 3
HAM-Malprogramm
Hersteller: NewTek
Vertrieb: Fachhandel
Preis: zirka 200 DM



Sämtliche Bilder und Brushes können wie bei »Deluxe Paint III« auch im IFF-ILBM-Format abgespeichert werden.

Das Bild, das aus dem Chaos kam

Programme, die mit Hilfe von Algorithmen (Rechenvorschriften) aus Zahlen Bilder berechnen, gibt es in der Public Domain im Überfluß. Fraktalgeneratoren, die Landschaften berechnen, Mandelbrot-Programme und Julia-Mengen sind nur einige wenige Möglichkeiten, interessante Bilder auf dem Amiga zu erzeugen. Leider sind nur wenige dieser Programme in der Lage, die fertigen Produkte im IFF-ILBM-Format abzuspeichern. Man kann jedoch einen Screengrabber benutzen, der im Hintergrund als Task läuft und bei Druck auf eine bestimmte Tastenkombination den aktuellen Bildschirm im IFF-ILBM-Format abspeichert. Eine kurze Auflistung einiger Programme und der betreffenden PD-Disk soll Ihnen einen kleinen Überblick geben. Es empfiehlt sich jedoch, einmal die Anzeigen der AMIGA DOS zu durchforsten und sich von einem PD-Händler einen Katalog senden zu lassen.

Programm	PD-Disk
»Mandel«	Fish 4
»Mandelbrot«	Fish 5
»Grafik-Disk«	RW 6
»ATV3d«	RPD 49
»Chaos«	RPD 93
»Planet 1.0«	RPD 100
»Scenery 1.0«	Fish 155

Raytracing mit Reflections

»Reflections« ist, trotz des englischen Namens, ein Programm deutscher Herkunft. Dies hat den nicht zu verleugnenden Vorteil, daß auch das Handbuch in Deutsch gehalten wurde.

Der Editor des Programms ist leicht zu erlernen und logisch aufgebaut. Es dürfte auch dem Einsteiger nicht schwerfallen, Szenen zu editieren und berechnen zu lassen. Hier soll jedoch noch einmal auf den »wunden Punkt« aller Raytracing-Programme hingewiesen werden: die zuweilen enorme Rechenzeit.

»Reflections« gestattet es, Oberflächenbeschaffenheit (Textures) oder Materialien selbst zu erstellen und in den Szenen zu verwenden. Es können mehrere Lichtquellen definiert werden, auch innerhalb von Körpern (zum Beispiel einer Glühlampe). Die Bilder können in nahezu allen Auflösungen berechnet werden, beste Ergebnisse erzielt man jedoch mit Low-Resolution in 32 Farben oder im HAM-Modus. »Reflections« stellt auch die Möglichkeit zur Verfügung, ein »Testformat« zu generieren. Hier werden die Bilder wesentlich kleiner berechnet (zum Beispiel 64 mal 52 Punkte), so daß man sehen kann, ob die Szene gelungen ist. Dies spart beträchtliche Zeit, da die Berechnung meist weniger als eine halbe Stunde beansprucht.

Selbstverständlich werden die fertigen Bilder auf Wunsch im IFF-ILBM-Format abgespeichert.

Name: Reflections
Raytracing-Programm
Hersteller: Markt & Technik
Vertrieb: Fachhandel
Preis: 100,- DM

Pixel-Animation mit Deluxe Video

»Deluxe Video III« ist ein Programm, das neben der bekannten Pixel-Animation auch eine Pseudo-3D-Animation bietet. Hierbei werden die Animationsobjekte (meist Brushes) verkleinert oder gedreht, so daß sich ein räumlicher Effekt ergibt. Die Hauptanwendung von »Deluxe Video III« liegt je-

doch im Zeichentrickverfahren. In Scripts werden sämtliche Animationsphasen festgelegt, wobei zusätzlich die Möglichkeit besteht, an frei wählbaren Punkten des Films ein Geräusch oder Musik ertönen zu lassen.

Das Editieren eines Films findet an einem »Schnittpult« statt, das einen Zugriff auf alle beteiligten Grafiken und Sounds gestattet. Die Grafiken können mit »Deluxe Paint III« erstellt werden, es kommen jedoch auch alle anderen im IFF-ILBM-Format erstellten Animationsobjekte in Frage.

Fertig erstellte Filme werden auf Diskette oder Festplatte gespeichert und können mit einem mitgelieferten Abspielprogramm weitergegeben oder vorgeführt werden.

Name: Deluxe Video III
Pixel-Animationsprogramm
Hersteller: Electronic Arts
Vertrieb: Fachhandel
Preis: zirka 300 DM

Videoscape 3D bringt räumliche Bewegung

Wer dreidimensionale Animationen erstellen, aber die langen Rechenzeiten von Raytracing-Programmen vermeiden

möchte, der ist bei »Videoscape 3D« an der richtigen Adresse.

Die Szenen werden in einem komfortablen Editor erstellt, der von einigen separaten Hilfsprogrammen unterstützt wird. Zugunsten der Geschwindigkeit wurde auf eine spezielle Materialvergabe an Objekte verzichtet; es ist jedoch möglich, den Objekten eine bestimmte Farbe zuzuordnen.

Bewegen lassen sich Objekte auf zweierlei Weise. Zum einen kann die Position der Kamera festgelegt werden, zum anderen die Position der Objekte selbst.

Name: Videoscape 3D
3D-Animationsprogramm
Hersteller: Aegis
Vertrieb: Fachhandel
Preis: zirka 400 DM

Diese Positionsangaben können entweder von Hand eingegeben werden oder aber (um zum Beispiel die Kamera im Kreis um ein Objekt herumzubewegen) in einem Script. Es besteht außerdem die Möglichkeit, jedes einzelne Bild der Animation im IFF-ILBM-Format abzuspeichern und einem Pageflipping-Programm zu übergeben, das aus Einzelbildern eine Animation zusammenstellt.





Digi Tiger – der Digitizer

Wer Bilder vom Videoband oder einer Kamera in den Amiga einlesen möchte, benötigt einen Digitizer. Dieses Gerät, das meist mit der dazugehörigen Software ausgeliefert wird, wandelt eingehende (analoge) Signale in für den Rechner verständliche (digitale) Daten um.

Eins dieser Geräte ist der »Digi Tiger«. Die Software erlaubt das Digitalisieren von Bildern in allen möglichen Auflösungsstufen inklusive Overscan (ohne Bildschirmrand). Die Bilder können sowohl in Schwarzweiß (16 Graustufen), in Antik (vergilbt) und in Pseudo-Farben (Falschfarben) als auch in »echten« Farben (je nach Auflösung bis zu 4096 Farben) digitalisiert werden. Ein RGB-Splitter, der die Farben in ihre Rot-, Grün- und Blauanteile aufspaltet, ist bereits in der Hardware eingebaut. Mit all diesen Einstellungsmöglichkeiten ist eine Verfremdung »realer« Motive kein Problem mehr. Digitalisierte Bilder speichert »Digi Tiger« im IFF-ILBM-Format, so daß auch Animationen erstellt werden können.

Name: Digi Tiger Digitizer
Hersteller: Klaus D. Tute
Vertrieb: Klaus D. Tute
Mathildenstr. 12
3000 Hannover 91
Tel: 0511-629825
Preis: 698,- DM

3D Professional – Grafik vom Feinsten

Ein sehr umfangreiches Raytracing- und Animationsprogramm stellt »3D Professional« dar. Mit diesem Programm ist jede nur erdenkliche Manipulation von dreidimensionalen Objekten möglich. Zum Lieferumfang gehören zwei Programmversionen (für 68000- und 68020/881-Prozessoren), ein ausführliches Handbuch sowie ein VHS-Videoband, das dem Einsteiger die Bedienung des Programms »live« erklärt. Wir werden Ihnen in der nächsten Ausgabe einen ausführlichen Test präsentieren.

Name: 3D Professional
Raytracing-Animationsprogramm
Vertrieb: Fachhandel
Preis: zirka 950 DM

Video Page – Titel für Ihre Videofilme

»Video Page« dient dazu, eigene Videofilme mit einem Titelvorspann wie bei einer professionellen Produktion zu verse-

hen. Dazu werden diverse Fonts (Zeichensätze) zur Verfügung gestellt, die auch außergewöhnliche Zeichen enthalten. Die Schrift kann in verschiedenen Modi über den Bildschirm bewegt werden (selbstverständlich im Overscan- und PAL-Modus), wobei der Text sehr weich gescrollt wird. Es lassen sich außerdem verschiedene Effekte einsetzen, zum Beispiel Ein- und Ausblenden, Buchstabieren, Verwischen, Zooming und ähnliche Effekte, die aus dem Fernsehen hinreichend bekannt sind. »Video Page« benötigt mindestens 1 MByte Rechner Speicher sowie ein Genlock-Interface, welches das Signal zum Videorekorder überträgt.

Name: Video Page
Titelgenerator
Hersteller: Heinrichson
Schneider & Young
Vertrieb: Fachhandel
Preis: 199,- DM

Comic Setter – Superman aus Pixeln

Etwas aus dem Rahmen der »gewöhnlichen« Grafikprogramme fällt der »Comic Setter«. Dieses Programm dient

dazu, Comics zu erstellen und diese zu Papier beziehungsweise auf den Bildschirm zu bringen. Zur Erstellung der Comics werden vielerlei Hilfsmittel zur Verfügung gestellt: Sprechblasen, Bildumrandungen, diverse Zeichenfunktionen und eine Auswahl bereits vorgefertigter Hintergrundgrafiken und Comicfiguren.

Name: Comic Setter
Comic-Zeichenprogramm
Hersteller: Gold Disk
Vertrieb: Fachhandel
Preis: zirka 100 DM

PAL-Genlock für Videoprofis

Genlocks werden benötigt, wenn zwei Videosignale (zum Beispiel vom Videorekorder und vom Amiga) gleichzeitig auf einem Bildschirm dargestellt werden sollen. Wer mit dem Amiga selbsterstellte Trickfilme und Animationen auf ein Videoband aufnehmen möchte, benötigt ein Genlock, um das Signal des Amiga für den Videorekorder »verständlich« zu machen. Dabei können auch Bilder von einem zweiten Rekorder mit dem Amiga-Bild gemischt werden. So können Sie zum Beispiel Ihr Trickfilmmännchen vor einem realen Hintergrund agieren lassen.

Das PAL-Genlock verfügt über einen eingebauten RGB-Splitter, einen PAL-Modulator und einen Video-Kopierschutzdecoder. Es ist für alle Amiga-Modelle erhältlich.

Name: PAL-Genlock
Video-Genlock
Hersteller: Electronic Design
Vertrieb: Electronic Design
Detmoldstraße 2
8000 München 45
Tel: 089-3515018
Preis: 598,- DM

(tb)



Marktübersicht Grafik-Soft- und Hardware

Das universelle Grafikprogramm, das Zeichnen in allen Modi und Digitalisieren inklusive Animationen in Raytracing erlaubt, gibt es noch nicht.

Um Ihnen zu helfen, die richtigen Werkzeuge für Ihre Grafikanwendung zu finden, haben wir eine Marktübersicht zusammengestellt. Damit diese Marktübersicht auch das bleibt, was sie darstellen soll, haben wir die einzelnen Programme beziehungsweise Geräte nach Einsatzgebieten getrennt. So können Sie schnell das für Ihren Zweck passende Programm herausfinden. Alle Hard- und Software sollte, sofern nicht anders angegeben, im gut sortierten Fachhandel erhältlich sein. Um jedoch Mißverständnisse auszuschließen, wird der Hersteller auf jeden Fall angegeben.

Pixel-Malprogramme

In diese Kategorie fallen alle Programme, mit denen Sie in Pixelgrafik malen können, auch im HAM-Modus.

- Deluxe Paint III, Electronic Arts, zirka 250 DM, vielseitig, alle Auflösungen inklusive Overscan, kein HAM-Modus, Pixel-Animationen möglich
- DIGI-Paint 3, NewTek, zirka 200 DM, alle Auflösungen, HAM-Modus, besondere Effekte
- Photon Paint 2.0, Micro Illusions, zirka 250 DM, alle Auflösungen, HAM-Modus
- Deluxe Photolab, Electronic Arts, zirka 200 DM, viele Funktionen zur Nachbearbeitung digitalisierter Bilder

3D- und CAD-Programme

Hier finden Sie Programme, mit denen Sie technische Zeichnungen oder dreidimensionale Körper in Vektorgrafik konstruieren können.

- IntroCAD, PPS, zirka 150 DM, preiswert, für einfache Aufgaben geeignet
- NewIO (Standard), Alphasatron, zirka 550 DM, Platinen-Layoutprogramm
- Draw Plus, Aegis, zirka 500 DM, leistungsfähiges CAD-Programm, zweidimensional
- Professional Draw, Gold Disk, zirka 250 DM, zweidimensionales CAD-Programm
- Design 3D, Gold Disk, zirka 190 DM, 3D-CAD-Programm

Raytracing-Programme

Die hier aufgeführten Programme erlauben das Darstellen von Körpern und Szenen nach dem Raytracing-Verfahren (Lichtstrahl-Verfolgung).

- Reflections, Markt & Technik, zirka 100 DM, leistungsfähiger Raytracer, leichte Bedienung

- Turbo Silver 3.0, Impulse, zirka 400 DM, Raytracer und Animationsprogramm
- Sculpt/Animate-4D, Byte by Byte, zirka 750 DM, sehr guter Raytracer inklusive Animation
- Sculpt/Animate-4D Junior, Byte by Byte, zirka 300 DM, wie Sculpt/Animate-4D, einfachere Version

Pixel-Animationsprogramme

Diese Programme ermöglichen Animationen nach dem "Pageflipping-Prinzip" (Zeichentrickfilm).

- Deluxe Video III, Electronic Arts, zirka 300 DM, einfach zu bedienendes Animationsprogramm
- Fantavision, Broderbund, zirka 100 DM, preiswertes und leistungsfähiges Animationsprogramm
- Lights! Camera! Action!, Aegis(Oxxi), zirka 120 DM, Animationen und Geräusche/Musik
- Page Flipper Plus F/X, Mindware, zirka 300 DM, schnelles Pageflipping
- Zoetrope, Antic(GFA), zirka 200 DM, leistungsfähiges Pageflipping-Programm
- The Director, The Right Answer Group, zirka 100 DM, preiswertes und leistungsfähiges Programm

3D-Animationsprogramme

Mit diesen Programmen können Sie dreidimensionale Szenen erstellen und animieren. Ausgenommen von diesen Programmen sind die Raytracing-Animationsprogramme, die Sie bei den Raytracing-Programmen finden.

- Caligari, Octree, zirka 500 DM, umfangreiches 3D-Animationsprogramm
- Caligari Professional, Octree, zirka 4000 DM, wie Caligari, aber mehr Funktionen
- Design 3D, Gold Disk, zirka 250 DM, leistungsfähiger Editor und Animator
- Videoscape 3D, Aegis, zirka 400 DM, leicht bedienbar und leistungsfähig
- Forms in Flight II, Centaur Software, zirka 160 DM, preiswertes Animationsprogramm

Digitizer-Hard- und Software

Hier finden Sie alles, was Sie zum Digitalisieren (Einlesen von Videobildern) benötigen. Da diese Artikel seltener im Fachhandel angeboten werden, haben wir die Hersteller gesondert angegeben.

- Digi Tiger, zirka 700 DM
Klaus D. Tute
Mathildenstraße 12
3000 Hannover 91
Tel: 0511-629825
leicht bedienbar, eingebauter RGB-Splitter, Echtzeit-Effekte, HAM und Overscan
- Digi View 4.0, zirka 400 DM
DTM
Poststraße 25
6200 Wiesbaden
Tel: 06121-502050
HAM-Modus auch in High-Resolution mit Interlace
- Deluxe View 4.1, zirka 400 DM
Hagenau Computer
Alter Uentroper Weg 181
4700 Hamm 1
Tel: 02381-880077
- Vidichrome Amiga, zirka 500 DM
PR8-Soft
Otto-Hahn-Straße 10
8702 Estenfeld
Tel: 09305-8211

(tb)

Gewußt wie!

Weiter geht es mit allerhand Wissenswerten rund um unseren Lieblingscomputer.

Unmanipulierbare Fenster

Wenn Sie Ihren selbsterstellten Spielen und Actiongames einen professionellen Touch geben wollen, kommen Sie nicht darum herum, die Fenster-Gadgets, Titelleiste und den Bildschirmrahmen zu entfernen. (Oder können Sie sich bei Spielen wie »R-Type« oder »BarbarianII« vorstellen, daß eine Titelleiste erscheint, wenn man den rechten Mausknopf drückt?) Dies ist in eigenen Programmen sehr leicht (in BASIC allerdings relativ kompliziert) zu erreichen. Dafür verantwortlich sind die Window-Flags in der Window-Struktur. Tragen Sie dort für Ihre eigenen Windows die Zahl \$00011800 ein, und schon ist das Problem gelöst. (Für den Fensternamen tragen Sie bitte eine \$0 ein!)

Die Behauptung (AMIGA DOS 7/90, Seite 86), das Problem wäre durch das Setzen des »GimmeZeroZero-Bits« behoben, trifft leider nicht zu! Das GZZ-Bit verlegt den Nullpunkt des Windows lediglich an den unteren Rand der Titelleiste und trennt diese und das Fenster selber in zwei unterschiedliche Speicherbereiche. Höherer Speicherverbrauch und viel(!) langsames Arbeitstempo sind der Nachteil!

Nun zur Aufschlüsselung der Zahl \$00011800: Verantwortlich sind die Bits 11 und 16. Zusätzlich habe ich noch Bit 12 gesetzt, welches lediglich eine sofortige Aktivierung des Windows bewirkt. Bit 11 (BORDERLESS=\$0800) läßt die Schirmumrandung verschwinden. Bit 16 (RMBTRAP=\$00010000) verhindert ein Erscheinen der Titelleiste. Ist es gesetzt, so kann der rechte Mausknopf durch die Routine »EXEC_GetMsg« abgefragt und für den eigenen Bedarf programmiert werden.

Ein so aufgebautes Fenster kann durch den Anwender nicht vergrößert, verkleinert, verschoben oder in den Hintergrund gebracht werden. Also der optimale Hintergrund für ein deftiges Actiongame!

(Frank Abbing/jb)

Tastatur neu belegt

Turbo-ED-Utilities, die die Tastatur des Amiga komplett neu belegen können, gibt es inzwischen wie Sand am Meer. Mit ihnen kann man zum Beispiel die Funktionstasten mit wichtigen und häufigen Systembefehlen (wie »cd df0:«, »dir«, »run« »ed«, etc.) belegen, um sich damit eine Menge an Tipparbeit zu ersparen.

Doch damit sind die Anwendungsmöglichkeiten solcher Utilities noch lange nicht erschöpft. Der Systemeditor »Ed« besitzt sehr vielfältige Funktionen, die jedoch total umständlich und unkomfortabel zu bedienen sind, so daß man bald nach einem besseren Systemeditor Ausschau hält. Doch diesem Dilemma kann Abhilfe geschaffen werden, indem man die gebräuchlichsten Ed-Befehlskolonnen direkt erreichbar auf einprägsame Tasten (und Tastenkombinationen) legt. Durch diesen Trick kann man sich die Ed-Befehle auf der Tastatur besser merken und auch viel schneller ausführen. Diese Hilfe ist für mich eine wesentliche Erleichterung bei der regelmäßigen Arbeit mit dem Ed. Es entfällt in den meisten Fällen das frustrierende, endlose Nachschlagen nach bestimmten Codes im Handbuch. Durch diese neue Tastaturbelegung wird aus dem lahmen »Schnecken-Ed« ein schnittiger »Turbo-Ed«.

Nachfolgend einmal meine Tastaturbelegung des »Ed« als Anregung, die, wie schon erwähnt, durch diverse Utilities leicht nachvollzogen werden kann. Da viele Codes nur als Hex-Code eingegeben werden können, gebe ich gleich die jeweiligen Hex-Codes für die jeweilige Taste an:

TASTE	HEX-CODE	Ed-Funktion
SHIFT+Keypad 0	9b,55,0d (=CR)	Zeile einfügen
SHIFT+Keypad .	1b,64,0d	Zeile löschen
SHIFT+Keypad 7	1b,62,73,0d	Blockanfang mark.
SHIFT+Keypad 1	1b,62,65,0d	Blockende markieren
ENTER	01,9b,41,14,9b,42	Autoindent
SHIFT+Pfeil auf	04	Textanfang
ALT+ Pfeil auf	1b,74,0d	Scroll down
SHIFT+Pfeil ab	15	Textende
ALT+ Pfeil ab	1b,62,0d	Scroll up
SHIFT+Pfeil rechts	1d	Anf./Ende der Zeile
ALT+ Pfeil rechts	14	Nächstes Wort
SHIFT+Pfeil links	1d	Anf./Ende der Zeile
ALT+ Pfeil links	12	Letztes Wort
CONTROL+F1	1b,73,61,0d	Textpuff. abspeich.
CONTROL+F2	1b,78,0d	Textpuff. abspeich. und Editor-Ende
CONTROL+F10	1b,71,0d	Editor verlassen
HELP	1b,75,0d	Undo-Funktion

Natürlich kann jeder Ed-Befehl auch auf die herkömmliche Art und Weise immer noch ausgeführt werden.

Durch eine gescheite neue Tastaturbelegung kann man sehr viele Programme aufmöbeln. Hier hat man fast unbegrenzte Möglichkeiten. So kann man zum Beispiel eine eigene Tastaturbelegung für Adventures entwerfen, durch die man ein Adventure bequem auf Tastendruck durchspielen kann. Die wichtigsten Schlüsselbefehle werden einfach auf die Tastatur gelegt (wie zum Beispiel <go north>, <open door> etc), oder man ruft ein wichtiges System-Utility auf, während ein anderes Programm gerade läuft und man sich nicht im CLI befindet. So kann man zum Beispiel den Aufruf »run monitor« auf eine Taste legen, und man hat immer die Möglichkeit, dieses Utility aufzurufen. Sicherlich gibt es in dieser Hinsicht noch vielfältige Möglichkeiten, den Amiga besser auszunutzen. Deshalb warte ich auch schon gespannt auf weitere Vorschläge und Ideen von anderen Lesern.

Fehler im GFA-BASIC!

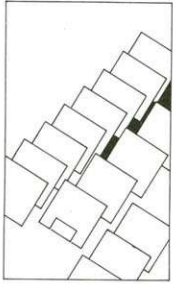
Neulich habe ich in GFA-BASIC einen Fehler entdeckt: Wenn man mit IDCMP-Flags arbeitet und eine Message eintrifft, soll (angeblich) die Adresse des Windows in »MENU(10)« stehen. Bei meiner Version jedenfalls steht diese Adresse jedoch in »MENU(9)«!

»SADD« in GFA-BASIC

GFA-BASIC kennt den Befehl »SADD« nicht. Das ist von Nachteil, wenn man mit Libraries arbeitet. Die Amiga-BASIC-Zeile »CALL Text(rp&,SADD(text\$),LEN(text\$))« versteht GFA-BASIC nicht. Hier hilft allerdings die Anweisung »LPEEK(*text\$): ~Text(rp%,LPEEK(*text\$),LEN(text\$))«

Betrifft Unterroutinen und Funktionen

Unterroutinen können in GFA-BASIC zur besseren Übersicht gefaltet werden, Funktionen leider nicht. Wenn man nun eine Routine vierzigmal aufruft und das Ergebnis immer in einer anderen Variablen benötigt, hilft zwar eine Funktion, aber bei langen Quelltexten wird es unübersichtlich. Es gibt aber eine Lösung, nämlich beim Procedure-Aufruf die Adresse der Variablen, in die man das Ergebnis wünscht, anzugeben. Die Adresse erhält man durch ein Sternchen vor der Variablen. Ein Beispiel: »GOSUB test(*x&)«. In der Unterroutine »Test«



Tausche Tip gegen DATABOX

Wir machen Ihnen wieder unser Angebot: Sie schicken einen oder mehrere kurze Tips an uns; bei Gefallen bekommen Sie von uns im Austausch die DATABOX des Heftes zugeschickt, in dem Ihr Tip erscheint.

Dazu gibt es aber jetzt noch etwas mehr:

Die drei besten Tips werden mit je 100,- DM belohnt.

Die Redaktion der AMIGA DOS sucht ab sofort unter allen Einsendern bis zu drei **Supertips pro Monat** aus, die außer der DATABOX noch je einhundert deutsche Märker (100,- DM) bekommen.

Dabei sollten Sie noch folgendes beachten:

- Schreiben Sie Ihren Tip als ASCII-Datei auf Ihre Diskette, möglichst ohne eigensinnige Verschnörkelungen, da alle Texte zur weiteren Verarbeitung auf Personal-Computer transferiert werden müssen. Sonder- und Steuerzeichen haben dabei meist seltsame Auswirkungen.
- Schreiben Sie auf den Brief sowie auf einen kleinen Aufkleber auf der Diskette Ihre lückenlose Anschrift und einen kleinen Hinweis, was auf der Diskette zu finden ist.

Dann können Sie sicher sein, daß die DATABOX Sie erreicht. Schicken Sie das Ganze an den

DMV-Verlag
Redaktion AMIGA DOS
Kennwort "Tausche Tip gegen Databox"
Postfach 250
D-3440 Eschwege

kann nun das Ergebnis, das zurückgegeben werden soll, einfach in diese Variable geschrieben werden, zum Beispiel:

```
1 PROCEDURE test(i%) ! Adresse von "x%" (s.o.) nach "i%" (Immer Integer!)
2 *i%=5             ! Variable "x%" (s.o.) den Wert 5 zuweisen.
3 RETURN            ! ...und zurück.
```

Jetzt kann man das Programm auch mit »GOSUB test("d")« aufrufen - in der Variablen "d" wird der Wert 5 stehen!

LET it be...

Der Syntax-Check von GFA kann manchmal nützlich sein, aber auch oft nerven. So setzt der Interpreter dem Programmierer, der darauf besteht, einen String namens "dir\$" ins Programm einzubauen, bei der Zeile »dir\$=a\$« doch glatt »DIR "\$=a\$« vor. Wenn man eine Prozedur namens "load" mit »load(a,b,c)« aufruft, erhält man zwar auch »LOAD "(a,b,c)«, aber hier kann man sich mit GOSUB abhelfen. Was macht man nun mit Variablen oder Strings? Man setzt einfach ein kurzes »LET« davor. »LET dir\$=a\$« nimmt der Interpreter an.

HERMES LoGG

NEU

Version 1.0 / 115 K © 1990 A. G. Kartsatos HERMES
Mit 10seitiger deutscher Anleitung

LoGG das neue, universelle Tool als Ersatz für das CLI. Als übersichtliches Programm, vollkommen mit der Maus gesteuert, bringt LoGG Komfort bei der Arbeit auf DOS-Ebene.

- LoGG hat **48 Funktionen**, die gebräuchlichsten **CLI-Befehle**, im Programm eingebaut. Zudem können vom Benutzer bis zu **48 Macros** eingegeben werden, um Programme zu starten oder Befehle auszuführen.
- LoGG kann **packen** und **entpacken**, mit den Befehlen ARC, LHARC, PAK, ZOO und WARP.
- LoGG können Sie nach Ihren Wünschen die eigenen **Menü-Farben** geben und Ihre vorhandenen **Laufwerke** selber im Programm einbinden.
- LoGG **spielt** IFF-Musik und **zeigt** IFF-Bilder.
- LoGG hat einen eigenen **Format-** und **Diskcopy-**Befehl.
- LoGG **erstellt** einen Katalog der Harddisk, speichert ihn oder schickt die Daten zum Drucker.
- LoGG hat einen sehr schnellen **File Reader** mit einigen zusätzlichen **Menü-Funktionen** und gutem Scrolling.
- LoGG **säubert** Ihre ASCII Files von Binary-Zeichen.
- LoGG **sucht** Files nach Namen oder nur nach einem String ab und zeigt, wo sie zu finden sind.
- LoGG **zeigt** Icons in jeder Größe und in 2 bis zu 16 Farben.
- LoGG eignet sich besonders für den **Einsteiger**, weil es logisch und leicht verständlich aufgebaut ist.
- LoGG für den **Profi**, weil es die Arbeit schneller und leichter macht als im CLI!

HERMES IconLab

Version 1.3 / 164 K © 1988-1990 A. G. Kartsatos HERMES
Mit deutscher Anleitung und Übungsbeispiel

Räumen Sie auf mit den langweiligen Schubladen der Workbench, geben Sie Ihren Programmen eigene, selbst kreierte Icons! Mit IconLab ist dies möglich. Das Tool, das alles beherrscht, wenn es um Icons geht, ganz einfach mit der Maus zu bedienen.

IconLab macht Disk-, Drawer-, Tools- und Project-Icons. Entwerfen Sie mit dem **IconLab** Malprogramm neue Icons oder bearbeiten Sie bereits bestehende Icons in jeder Größe und in 2 bis zu 16 Farben. **IconLab** wandelt Brushes in Icons oder Icons in Brushes und kann Brushes verkleinern, damit diese als Icons verwendet werden können. **IconLab** vereinigt zwei Einzel-Icons zu einem Doppel-Icon oder trennt Doppel-Icons. **IconLab** hat eine Text-Option, mit der jeder Font verwendet und beliebig platziert werden kann. Einfacher und sicherer als mit **IconLab** gehts nicht !!

Auf der Programm-Disk finden Sie noch zwei weitere HERMES-Programme: HERMIT, einen Screen Saver und COLORLAB, um Farben nach HEX-Zahlen zu bestimmen.

Ich bestelle:

- ☐ **LoGG** zum Preis von 35.- DM
- ☐ **IconLab** zum Preis von 25.- DM

In bar oder mit Check, plus 5.- DM Versandkosten.

Bei Nachnahme 10.- DM Versandkosten. Bitte Ihre Adresse nicht vergessen!

HERMES Software
Postfach 5
CH-6345 Neuheim/ZUG



Die rote Disk

Voreinsteller ohne »Prefs«

Möchte man schnell mal die Farben, den Druckertreiber oder die Grafikdicke ändern, muß man erst das Programm »Preferences« laden. Schneller geht das mit folgendem Trick: Irgendein Project- oder ein selbsterstelltes Icon mit dem Icon-Editor in das Verzeichnis »Prefs« kopieren. Das Icon wird jetzt einmal mit der Maus angeklickt und in der Menüleiste »Info« aufgerufen. In das Feld »Default Tool« wird jetzt »C:\IconX« eingegeben, dann wird das Fenster mit »Save« verlassen. Jetzt muß das CLI oder die Shell geöffnet und eingegeben werden: »ED Prefs/Name der Datei«. Hier kann ein beliebiger Name eingesetzt werden. Wurde »ED« geladen, gibt man jetzt ein:

```
C:\SetPrefs Devs:"Name der Datei".conf
```

Dann wird der Editor mit den Tasten [Esc] und [X] verlassen. Jetzt muß man noch Preferences laden und die gewünschten Einstellungen vornehmen, zum Beispiel die Schriftfarbe oder am Mauszeiger. Sind alle Einstellungen erledigt, wird »Preferences« mit < Save > verlassen. Die weitere Vorgehensweise sieht dann so aus:

CLI oder die Shell öffnen und im Verzeichnis »Devs« die Systemkonfiguration mit »Rename« umbenennen in »Name der Datei.conf«. Wenn man jetzt das Icon zur Datei zweimal anklickt, werden alle Einstellungen, die in der Konfigurationsdatei gespeichert sind, geladen. So kann man sich verschiedene Einstellungen zusammenstellen, ohne jedesmal »Preferences« zu laden. Verschiedene Druckertreiber, Mauszeiger, Hinter- und Vordergrundfarben sind so mit einem Schlag geladen.

Zum Schluß kopiert man sich noch die original Systemkonfiguration auf seine Startdiskette und gibt im »ED« ein:

```
C:\SetPrefs Devs:System-Configuration
```

Dann wird noch ein Icon erstellt (Typ Project, wie oben beschrieben) mit dem Namen »Normal«. Beim Klick auf dieses Icon wird dann die original Systemkonfiguration geladen. Es ist natürlich auch möglich, »SetPrefs« von der Startup-Sequence aufzurufen, um zum Beispiel beim Booten der Diskette verschiedene Farben für Hintergrund und Schrift anzuzeigen und am Schluß auf die Originaleinstellung zurückzuschalten.

Ein Beispiel für die Startup-Sequence:

```
1 C:\SetPatch >NIL: r
2 C:\SetPrefs Devs/Schwarz.conf ; (je nach gespeicherter Einstellung in Schwarz.conf).
3 Setmap d
4 Echo "Datum Uhrzeit..."
5 SetClock load
6 FF >NIL: -O
7 Resident CLI L:Shell-Seg System pure add
8 Dir Ram:
9 Makedir Ram:T
10 Assign T: Ram:T
11 Makedir Ram:Env
```

```
12 Assign Env: Ram:Env
13 Makedir Ram:Clipboards
14 Assign Clips: Ram:Clipboards
15 C:\SetPrefs Devs/System-Configuration ; (wieder auf normale Einstellung zurück).
16 LoadWB -debug
17 EndCLI >NIL:
```

Das Programm »SetPrefs« gibt es auf vielen PD-Disketten (Fred Fish, RPD) und sollte in das Verzeichnis »C« der Startdiskette kopiert werden.

(Marcel Würzburg/jb)

Immer Ärger mit den deutschen Zeichen

Viele amerikanische PD-Programme akzeptieren die Eingabe von "ä, Ä, ö, Ö, ü, Ü und ß" nicht, und man muß dafür "ae" oder die anderen Kombinationen eingeben. Diese Kombinationen lassen sich mit dem Zeileneditor EDIT aus dem Verzeichnis »C« der Workbench wieder in die deutschen Buchstaben umwandeln. Wir benötigen dazu den Befehl »GE« (global exchange – überall ersetzen) vom »edit«. Wir erstellen uns eine Datei mit einem beliebigen Namen, zum Beispiel "Umwandeln", die wie folgt aussieht:

```
ge /\ae/ä/
ge /\Ae/Ä/
```

Erklärung: »ge« ist klar, daß ist der Befehl; "/\ae/" ist die erste Zeichenkette mit "Kenntlichmacher". "Kenntlichmacher" ist das Zeichen Backslash (" \ "). Es wird später noch deutlicher, warum wir den Backslash voranstellen.

"ä/" ist die zweite Zeichenkette. In den Buchstaben "ä" wollen wir die erste Zeichenkette "\ae" umwandeln. Nachdem wir uns diese Umwandlungsdatei erstellt und abgespeichert haben, wollen wir das Ganze ausprobieren. Nicht vergessen, auch für ö, Ö, ü, Ü, ß den entsprechenden Ersatz einzutragen. Wir geben zum Beispiel in einen Editor ein paar Wörter ein. Beispiele: W\oerter, \aerger, Sch\ueler. Dann speichern wir das Ganze als Testdatei ab. Jetzt rufen wir den Zeileneditor EDIT auf:

```
EDIT Test to Testdatei with Umwandeln
```

Wenn wir uns jetzt die Testdatei ansehen, erkennen wir, daß dort anstelle von "W\oerter" das Wort "Wörter" steht.

Der Editor EDIT ist auf eine Zeilenbreite von 120 Zeichen eingestellt. Reicht das nicht aus, gibt es eine Option, mit der sich das verändern läßt. In unseren Beispiel würde das so aussehen:

```
»EDIT Test to Testdatei with Umwandeln opt w200«
```

Der ganze Vorgang läßt sich auch noch durch eine Batch-Datei automatisieren. Da aber in der AMIGA DOS 8/90 ein ausführlicher Artikel über Batch-Dateien war, will ich meinen Tip hier beenden.

(Franz Zimmermann/jb)

Die Supertips

Mehr Speicher mit »MoreMemory«

Welcher AMIGA-User stand noch nie vor dem Problem, zu wenig RAM-Speicher zu besitzen? Deshalb habe ich meinen Amiga 500 auf 1 MB auferüstet. Seit ich nun stolzer Besitzer einer Harddisk bin, verliere ich jedoch zusätzliches RAM für den Treiber. Also machte ich mich auf die Suche nach wertvollen Bytes. Die bekannte Lösung, 21 kByte zu gewinnen, ist das Abschalten einer Bitplane. Meine Lösung nun sucht und entfernt alle Tasks mit dem Namen »track-disk.device«. Dies bringt pro Laufwerk einen Speicher-gewinn von 17 kByte! Allerdings können die Laufwerke erst

nach einem Reset wieder angesprochen werden – für den Harddisk-Benutzer wird dies aber nicht weiter tragisch sein, da er seine wichtigen Daten auf der Festplatte gespeichert hat und mit Disketten nicht mehr so oft hantiert.

Ein zusätzlicher Effekt des Entfernens der Tasks ist, daß das oft lästige "Ticken" der Laufwerke aufhört. Somit kann ich aus meinem Amiga mit zwei Laufwerken durch den Befehl »add21k« und der kurzen Assembler-Routine 55 kByte RAM gewinnen – fast ein C-64! Es ist zu empfehlen, das Programm ins Verzeichnis »C« der Harddisk zu kopieren und es in der Startup-Sequence aufzurufen. Das Listing wurde übrigens mit dem SEKA-Assembler erstellt, zum Abtippen

und Assemblieren beachten Sie bitte die Weisungen aus dem Handbuch.

```

move.l 4.w,a6
loop:
    lea taskname,a1
    jsr -294(a6);FindTask
    tst.l d0;Task gefunden?
    beq.s end;nein -> Ende
    move.l d0,a1
    jsr -288(a6);RemTask
    bra.s loop;naechsten Task suchen
end:
    moveq #0,d0
    rts
taskname:
    dc.b "trackdisk.device",0
    
```

(Jürg Anderegg/jb)

Zufallsgenerator in C

Ein Zufallsgenerator ist in vielen Programmierprojekten unumgänglich. Aus diesem Grund habe ich einen neuen Zufallsgenerator in C realisiert. Er basiert auf dem Prinzip der Verknüpfung von sich laufend ändernden Hardware-Registern.

Das beiliegende Listing beinhaltet die Funktion »random ()« und das Hauptprogramm »main ()«. Der Rückgabewert der Funktion »random ()« ist eine "float-zahl" zwischen 0 und 1. Dieser Wert kann leicht konvertiert werden. So simuliert »main ()« sechs Würfeldurchgänge, die ausgegeben werden.

In der Funktion »random ()« wird der Inhalt der Adressen \$dff006 und \$dff007 verarbeitet. Diese Adressen stellen die aktuellen Strahlenpositionen des Elektronenstrahles des Bildschirms dar. Als Rückgabewert erhält man eine Zufallszahl zwischen 0 und 1.

```

/* Zufallsgenerator in C *
 * (C) by Thorsten Becker *
 * 0 < x < 1 = random () *
 * Compiler Options: *
 * cc random.c *
 * ln random.o -lm -lc */
main ()
{
    float random ();
    int i;
    for (i=0;i<6;i++)
        printf ("%d\t",(int)(6*random()+1));
}
float random ()
{
    unsigned char *pos=0xdff006;
    return ((*pos+*(pos+1))/482.0);
}
    
```

(Thorsten Becker/jb)

Adventure-Programmierer aufgepaßt!

Wer kennt das nicht: Ein tolles selbstgeschriebenes Spiel hat keine spezielle Save-Routine für den Spielstand. Jeder beliebige Amiga-Besitzer könnte nun das File zum Beispiel mit »MEMacs« laden, irgendwelche Daten verändern und dadurch das Spiel eventuell vereinfachen. Adventure-Programmierer, wehrt euch! Jetzt ist Schluß damit. Die Lösung sollte möglichst schnell sein, also wurde ein entsprechendes Programm mit GFA-BASIC 3.041 geschrieben. Das Utility ist selbst bei Datenmassen sehr schnell. In meinem Programm speichert es 3589 Variablen und Strings in noch nicht einmal zehn Sekunden! Alle Amiga-BASIC-Besitzer müßten eigentlich auf diese Routine verzichten, da

Festplatten

in 16 Bit Technologie
mit Boil3 exklusiv bei

FSE von **Quantum**

für A500/1000 :

AutoBoot, 2 Jahre Garantie,
BOIL3, 750² KB/S, nach Diskperf 620 KB/S
19¹ ms, sehr leise und zuverlässig
42 MB 1278.- 84 MB 1798.- 210³ MB 2998.-

Die Preialternative :

66 MB 1278.-

28¹ ms, 450² KByte/Sek, 1 Jahr Garantie, 8 Bit, OMTI -
Controller, AutoBoot, BOIL3, durchgeführter Bus, NEC

SCSI für A2000

Flexibler SCSI - Controller mit Anschlußmöglichkeiten
für Festplatten, Wechselplatten, Streamer, Laserdrucker
usw. Leistungsdaten wie bei 16 Bit Interface; siehe oben.

Filecards Quantum - Festplatten mit 2 J. Garantie :
42 MB 1178.- 105 MB 1798.- 210³ MB 2898.-

Wechselplatte SyQuest SQ555, inkl. Medium :
44 MB, 20¹ ms 1678.- Medium SQ 400, 44 MB 198.-

1) Herstellerangabe 2) Datentransferrate auf dem Bus 3) Bei Liefer-
schwierigkeiten von Quantum, können auch Rodime HDDs geliefert werden.

Ram - Erweiterungen

Für AMIGA 500 :	Für AMIGA 2000 :
512 KB, Uhr 128.-	2 MB, max. 8 MB 628.-
512 KB, max. 2 MB 428.-	4 MB, max. 8 MB 898.-

Diskettenlaufwerke

Vollkompatibel, anschlußfertig, abschaltbar, **TEAC** -
Markenlaufwerke, Bei 5.25" Stationen 40/80 Track
schaltbar. **1 Jahr Garantie, Mit durchgeführtem Bus :**

3.5": 198.- 5.25": 228.-

AMIGA
Köln 90

Wir stellen aus
09. - 11. November 1990
Halle 10, Stand 623-724

Frank Strauß Elektro-
nik

Schmiedstraße 11
6750 Kaiserslautern
Tel.: (0631) 67096 - 98
Fax: 60697

Lieferung erfolgt ab Lager mit
UPS oder Post per Nachnahme.

es dort mehrere Minuten dauert (vom Laden ganz zu schweigen), falls nicht jemand einen besseren Vorschlag hat.

Die Save-Routine:

Aufruf: GOSUB write(var, str\$)
var : Beliebige Variable
str\$: Beliebiger String

Hier das Programm:

```
1 PROCEDURE write(i%, i$)
2   write_main_loop:
3   IF i% < > -12345678
4     x$=STR$(i%)
5     i%=-12345678
6   ELSE IF i% < > "-!"
7     x$=i$
8     i$="-!"
9   ELSE
10    x$=""
11  ENDIF
12  IF x$ < > ""
13    write_loop:
14    i!=FALSE
15    l!=LEN(x$)
16    IF l! < 3
17      x$=x$+STRING$(3-l!, "$")
18    ELSE IF l! > 3
19      i!=TRUE
20      x1$=x$
21      x$=LEFT$(x$, 3)
22    ENDIF
23    b$=MID$(x$, 1, 1)
24    asc!=ASC(b$)
25    ADD asc!, 20
26    full$=CHR$(asc!)
27    b$=MID$(x$, 2, 1)
28    asc!=ASC(b$)
29    ADD asc!, 21
30    full$=full$+CHR$(asc!)
31    b$=MID$(x$, 3, 1)
32    asc!=ASC(b$)
33    ADD asc!, 22
34    full$=full$+CHR$(asc!)
35    IF i!
36      full$=full$+"1"
37    ELSE
38      full$=full$+"0"
39    ENDIF
40    PRINT #1, full$;
41    IF i!
42      x$=RIGHT$(x1$, LEN(x1$)-3)
43      GOTO write_loop
44    ENDIF
45    GOTO write_main_loop
46  ENDIF
47 RETURN
```

Ein Beispiel: »GOSUB write(TRUE, "Hallo")«
»GOSUB write(ii, "")«

Wenn Sie nur einen String speichern wollen, geben Sie »GOSUB write(-12345678, str\$)« an, wollen Sie nur eine Variable speichern, »GOSUB write(var, "-!)"«. "-12345678" zeigt dem Programm, daß es keine Variable speichern soll, "-!" daß es keinen String speichern soll. Bei »ADD asc!, x« können Sie für "x" eine beliebige Zahl einsetzen, die es Ihnen erlaubt, die Daten nach Ihren Wünschen zu codieren. Diese Zahl errechnet sich wie folgt:

Kleinsten ASCII-Wert der zu speichernden Daten plus "x" muß größer als 32 sein.

Größter ASCII-Wert der zu speichernden Daten plus "x" darf 255 nicht überschreiten.

Beispiel: »GOSUB write(13, "AMIGADOS" + CHR\$(13))«

Das Zeichen mit kleinstem ASCII-Wert ist CHR\$(13), auch Carriage Return (13), das mit dem größten das "m" (109). Es gilt: x = >20 (33-13) und x <= 146 (255-109).

Die Load-Routine:

Das Laden der Daten geht einfacher. Zuerst das Listing:

```
1 PROCEDURE read(i%)
2   full$=""
3   read_loop:
4   a$=INPUT$(4, #1)
5   b$=MID$(a$, 1, 1)
6   asc!=ASC(b$)
7   SUB asc!, 20
8   char$=CHR$(asc!)
9   full$=full$+char$
10  b$=MID$(a$, 2, 1)
11  asc!=ASC(b$)
12  SUB asc!, 21
13  char$=CHR$(asc!)
14  full$=full$+char$
15  b$=MID$(a$, 3, 1)
16  asc!=ASC(b$)
17  SUB asc!, 22
18  char$=CHR$(asc!)
19  full$=full$+char$
20  IF RIGHT$(a$, 1) = "1"
21    GOTO read_loop
22  ENDIF
23  WHILE RIGHT$(full$, 1) = "$"
24    full$=LEFT$(full$, LEN(full$)-1)
25  WEND
26  IF TYPE(i%)=1
27    *i%=full$
28  ELSE
29    *i%=VAL(full$)
30  ENDIF
31 RETURN
```

Hier müssen Sie die Zeilen »SUB asc!, x« den Variablen "x" aus Listing 1 anpassen.

Aufruf:

»read(Adresse der Variable/String)«

Beispiel:

```
»read(*i!)«
»read(*ha$)«
»read(*o&)«
```

Was geladen wurde, finden Sie nun automatisch im angegebenen String oder der Variablen wieder. Aber nicht das Sternchen vergessen! Wenn Sie es also mit »read(*xyz)« aufrufen, befinden sich die Daten automatisch in der Variablen "xyz"! Ein Service, den man nicht unterschätzen sollte. Bei Arrays wird es allerdings kritischer. Hier ist der Aufruf »read(*i#)«! "i#" hat Dummy-Funktion, »xyz # (4) = i#«! Ab nach xyz # (4) da aus irgendeinem Grund »read(V:xyz # (4))« nicht funktioniert.

Für beide Routinen müssen Sie die Datei mit Kanalnummer 1 von Hand öffnen und schließen.

Einschränkung: Das Zeichen "\$" darf in Strings nicht vorkommen!

Jetzt sollte es jedem möglich sein, seine Daten optimal zu kodieren. Probieren Sie mal:

```
1 OPEN "o", #1, "df0:TEST"
2 d!=TRUE
3 GOSUB write(d!, "AmigaDOS")
4 GOSUB write(-12345678, "presents:")
5 GOSUB write(FALSE, "-!")
6 CLOSE #1
7 OPEN "i", #1, "df0:TEST"
8 read(*d!)
9 read(*a$)
10 read(*b$)
11 read(*e!)
12 PRINT EOF(#1)
13 PRINT d!, a$, b$, e!
14 CLOSE #1
```

Sie erhalten: -1

-1 AmigaDOS presents: 0

Grafikkarte Highgraph V

mehr Auflösung zu einem guten Preis !!!

technische Daten zur Grafikkarte:

- * max. Auflösung von 832 x 620 Punkten
- * keine schwarzen Zeilen im Non-Interlaced Modus
- * volle 4096 Farben darstellbar - läuft auch im HAM-Modus
- * 50 Hz Ausgabefrequenz (Vollbild)
- * 31.25 kHz Horizontal-Ablenkfrequenz
- * 768 kByte dynamischer Ram
- * 9 pol. Sub-D Ausgang für RGB Analogsignal
- * RGB Digitalausgang auf der Platine über Steckerleiste

Preis: 898,- DM

Speichererweiterung CA2000.01 für Amiga 2000

- * 2 Megabyte: 698,- DM
- * 4 Megabyte: 1048,- DM
- * 6 Megabyte: 1398,- DM
- * 8 Megabyte: 1748,- DM

- abschaltbar autokonfigurierend
- 0 Wait-State
- industriell gefertigt
- hochwertige Präzisionssockel
- Steckerkontakte vergoldet

Die Karte wird grundsätzlich mit den PAL-Sätzen für 2/4 und 6/8 Megabyte ausgeliefert!

Quantum 3,5 Zoll Festplatten:

Model	Kapazität	Preis	
ProDrive 40S	42 MB	998,- DM	
ProDrive 80S	84 MB	1548,- DM	Preise für größere Kapazitäten auf Anfrage.

Die Festplatten sind mit einem **SCSI-Interface** ausgerüstet.

Quantum Festplatte 42 MB 19 msec mit SCSI-Controller: 1698,- DM

Quantum Festplatte 84 MB 19 msec mit SCSI-Controller: 2248,- DM

Einbaurahmen für 3 1/2 Zoll Festplatten im 5 1/4 Zoll Schacht des Amiga 2000: 20,- DM

Disketten:

ECC 3.5 Zoll 1.0 MB 2DD Double Sided:
10er Pack 19,90 DM

Lieferung ab Lager!
Händleranfragen erwünscht!

Höchste Herstellungsqualität sichert Ihnen eine lange Lebensdauer!

RALF JOCHHEIM COMPUTER TUNING

Osnabrücker Straße 96 - 4802 Halle - Tel.: 02823/1275 - Fax: 02823/1350

Den Abschied erleichtern uns in erster Linie einige neue Features von AmigaDOS 1.3, aber auch der übrige Software-Markt. Doch vor diesem großen Schritt wollen wir uns nochmals genauer mit den Umgebungsvariablen und dem Befehl »EVAL« beschäftigen, die wir in der letzten Folge schon kurz angeschnitten hatten. Umgebungsvariablen sind eigentlich ein sehr nützlicher Bestandteil eines Betriebssystems – aber nur dann, wenn sie wirklich praktikabel gestaltet sind, und das ist bei AmigaDOS leider nicht der Fall. Hier herrscht das Flair des Provisoriums vor, das die Arbeit recht umständlich macht...

Der Wahrheit über dieser Umgebungsvariablen kommt man recht schnell auf die Spur. Legen wir zu Testzwecken mal eine Variable an, zum Beispiel mit

```
SETENV Test "Das ist es"
```

Schauen wir uns nun die RAM-Disk etwas genauer an, werden wir dort ein recht "verdächtiges" Verzeichnis namens »ENV:« finden. Dieses wiederum enthält eine Datei namens "Test", und wenn wir die mit

```
TYPE ENV:Test
```

anzeigen, sehen wir genau jenes, was wir vorher eingegeben haben. Diese sogenannten Variablen sind also nichts anderes als in einem besonderen Verzeichnis abgelegte Dateien. Die neuen Befehle »SETENV« und »GETENV« machen also nichts anderes, als solche Dateien anzulegen und anzuzeigen. Keine Sensation also, mit »ECHO« und »TYPE« kommt man genausoweit. Zwar ist vorgesehen, für diese Variablen in einer späteren Version von AmigaDOS eine bessere Handhabe einzurichten, aber bis dahin muß man mit dieser Lösung leben. Und dieses Leben ist nicht unbedingt das angenehmste.

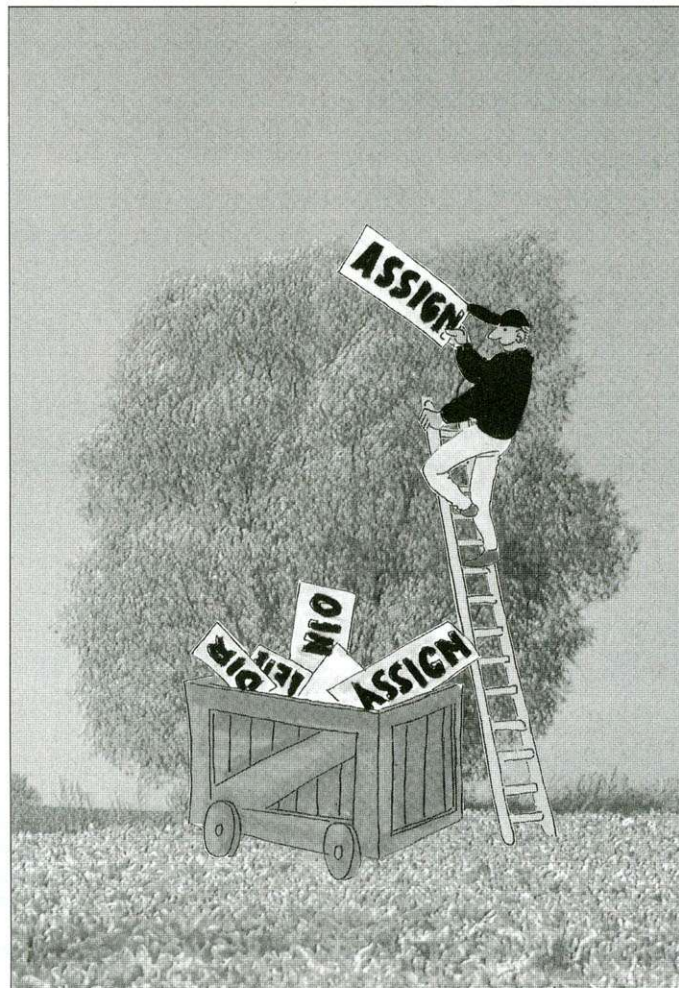
Schaut man sich mal "ordentliche" Vertreter dieser Variablen an, zum Beispiel die von MS-DOS, merkt man sehr schnell, daß es sich bei AmigaDOS eben um keine echten Variablen handelt, sondern um ganz normale Dateiumleitung. Also ein alter Hut in neuem Gewand – oder ein klassischer Fall von Etikettenschwindel. Für die praktischen Konse-

Michael Anton

Das AmigaDOS – leicht und verständlich!

Im Verlauf dieses Kurses haben wir uns mit dem CLI vertraut gemacht. Allerdings gibt es auch Alternativen zum CLI.

Kursfahrplan
Teil 1: Einführung in den CLI
Teil 2: Was kann AmigaDOS
Teil 3: Stapeldateien
Teil 4: Neu dabei bei 1.3
Teil 5: Alternativen zum CLI
Teil 6: Multitasking



quenzen vergleichen wir mal ein Beispiel in MS-DOS und AmigaDOS, bei der der Inhalt einer Variable als Parameter an ein Programm übergeben werden soll. Unter MS-DOS sieht das Ganze wie folgt aus:

```
SET Variable=Inhalt
PROGRAMM %Variable%
```

Dies führt im Endeffekt zur Sequenz

```
PROGRAMM Inhalt
```

Unter AmigaDOS muß statt dessen umgeleitet werden:

```
SETENV Variable Inhalt
PROGRAMM <ENV:Variable
```

Von der Theorie, daß der Inhalt der Variablen automatisch eingesetzt wird, bleibt in der Praxis nicht mehr viel übrig, lediglich »IF« erkennt solche Variablen auch ohne Umleitung, wenn sie als »\$Variable« in einem Script auftauchen. Statt dessen kann es in der Praxis zu erheblichen Problemen kommen, da die dauernde Umleitung nicht nur unübersichtlich ist, sondern oft auch zu Konflikten führen kann. Ein gutes Beispiel hierfür ist der Rechenbefehl »EVAL«, bei dem eine Menge an Umleitungen nötig sein können, wenn mit Umgebungsvariablen gerechnet wird. Dazu dann noch eine Umleitung unerwünschter Meldungen ins "NIL-wana", und das Chaos ist perfekt. Ein unbedarfter Blick auf die beiden Listings läßt dies wohl erahnen. Die Variablen müssen also mit Vorsicht genossen werden, Fehler dürften sich vor allem am Anfang wohl nicht vermeiden lassen. Speziell zum Umgang mit »EVAL« hier noch einige Tips.

In der Verbindung aus »EVAL« und den "Umgebungsvariablen" eröffnen sich neue Möglichkeiten in der Programmierung von Scripts. Doch zuvor noch einige Worte zu »EVAL« selbst. Eine Übersicht der Operationen wurde bereits in der letzten Folge geliefert, die generellen Funktionen dürften klar sein. Als Ersatz für einen Taschenrechner ist »EVAL« wohl kaum geeignet, seine Stärke kommt vor allem in der Programmierung von Schleifen innerhalb von Scripts zum Tragen – und scheitern dort meist an dem Wirrwarr mit den Umgebungsvariablen.

Die Handhabung ist eigentlich recht praktikabel, allerdings nur, wenn die Parameter direkt übergeben werden. Späte-

stens, wenn der erste Operand aus einer Variablen stammt, geht das Chaos jedoch los. Richten Sie mit »SETENV« mal eine Variable »Var« mit dem Wert »1« an, und versuchen Sie, per »EVAL 1« zu addieren und das Ergebnis in einer anderen Variable abzulegen. Geht es nicht? Kann es auch nicht, denn bei einer Umleitung wird »EVAL« pingelig. Zum einen müssen nun der zweite Operand mit dem Schlüsselwort »VALUE2=« und die Operation mit »OP=« gekennzeichnet werden. Weiterhin wird noch ein »?“ benötigt, um die Sache ins Rollen zu bringen. Dies führt jedoch auch zur Ausgabe einer Kurzanleitung, die unterdrückt werden sollte. Experimentieren wir nun mit allen möglichen Positionen der einzelnen Parameter, so kommen wir auf genau eine Reihenfolge, in der wirklich alles klappt: Als erstes die Umleitung des ersten Operanden, dann die Umleitung des Hilfstestes, als nächstes der mit »TO« gekennzeichnete Name der Zielvariablen, danach Wert 2 und die Operation und als krönender Abschluß noch das Fragezeichen. Schauen wir uns das Monstrum nochmals in der Praxis an:

```
SETENV Var 1
EVAL <ENV:Var >NIL: TO ENV:
Result VALUE2=1 OP=?
GETENV Result
```

Ein ziemlich harter Brocken, oder? Wenigstens ist die Auswertung des Resultats mit »IF« etwas einfacher:

```
IF $Result EQ 2 VAL
ECHO "Richtig..."
ENDIF
```

Allerdings darf man bei »IF« die Option »VAL« nicht vergessen, wenn Zahlen verglichen werden, denn sonst würde der Vergleich als ASCII-Zeichen erfolgen, was nicht immer gute Ergebnisse liefert. Zu berücksichtigen ist auch, daß »EVAL« nicht nach der Formel »x:=x+1« rechnen kann. Bei Zählschleifen muß also erst in eine temporäre Variable berechnet und diese per »COPY« wieder auf die Zählvariable kopiert werden. Mathematisch also »y:=x+1«; »x:=y«, unter DOS also

```
EVAL <ENV:Count ... TO ENV:
Temp ... ?
COPY ENV:Temp ENV:Count
```

Schauen Sie sich hierzu nochmals die Beispiele gut an, und

merken Sie sich die verwendeten Strukturen gut – es gibt kaum eine Alternative. So ist es auch nicht verwunderlich, daß die die Schleifenstruktur in beiden Beispielen identisch ist. Listing 1 macht nicht viel mehr, als einen einfachen Text zehnmal auf den Bildschirm zu schreiben. In Listing 2 darf sich »EVAL« voll austoben und eine kleine Wertetabelle zaubern, bestehend aus Dezimal-, Hex- und Oktalzahlen und den entsprechenden ASCII-Zeichen, die alle aus der Zählvariable berechnet werden. Wie man sieht, können dem Lformat-String auch andere Zeichen beigefügt werden, hier ist es »:«, um die Ausgabe schön zu strukturieren. Die einzelnen Werte werden in separaten Variablen erfaßt, mit »JOIN« zu einer neuen Variablen zusammengefaßt und abschließend mit »GETENV« ausgegeben. (»JOIN« ist übrigens ein bislang vernachlässigter Befehl, mit dem bis zu 15 Dateien zu einer zusammengefaßt werden können.)

Nach soviel Arbeit mit dem CLI wird Ihnen sicherlich eines aufgefallen sein: Die Arbeit ist nicht unbedingt die bequemste. Dies merkt man vor allem dann, wenn man sich bei der Eingabe eines Befehls vertippt hat. Solange die Eingabe noch nicht bestätigt wurde, kann man sich mit [Backspace] wenigstens noch bis zum Fehler

zurückhangeln und dort mit der Korrektur beginnen, nach dem Drücken von [Enter] muß jedoch ganz von vorn mit dem Tippen begonnen werden.

Mit dem Shell zu neuen Ufern

Ärgerlich, aber bislang unvermeidlich. Ab Version 1.3 gibt es jedoch eine neue Version des Befehlsinterpreters, die wesentlich mehr Komfort bietet. Der Luxus nennt sich "Shell" (eigentlich englisch für Muschel, im EDV-Jargon aber auch Benutzeroberfläche) und ist ein vollwertiger Ersatz für den CLI. Auf den ersten Blick hat sich nach dem Aufruf des Shell wenig getan, die Überraschung kommt jedoch nach der Eingabe einiger Zeichen und Zeilen und anschließendem Spielen mit den Cursorasten: Die Eingabezeile kann nicht nur beliebig bearbeitet werden, es können sogar alte Eingaben wieder hervorgeholt werden. Eine Übersicht über diese komfortable Hilfe für Viel- und Falschtipper zeigt die Tabelle. Das ist jedoch noch nicht alles, was das Shell zu bieten hat...

Der Prompt eines Shell kann ebenso wie der des CLI frei gestaltet werden. Zusätzlich kann auch das aktuelle Verzeichnis angezeigt werden, was durch den Platzhalter »%S« geschieht. Ein neuer, nur im Shell

verfügbarer Befehl ist »ALIAS«. Mit ihm können beliebige Befehle umbenannt und automatisch mit Optionen versehen werden. Die generelle Syntax ist

ALIAS Alias-Name Befehl

Nehmen wir für die Beispiele an, daß Sie in Zukunft den Dir-Befehl mit »D« abkürzen wollen. Der einfachste Weg hierfür ist

ALIAS D DIR

Nach der Eingabe von »D« wird nun automatisch »DIR« ausgeführt, übergebene Parameter werden einfach hinten angehängt, so daß aus »D ALL« letztendlich »DIR ALL« wird. Solche festen Zusätze können auch bei der Definition verankert werden, nach

ALIAS D DIR ALL

wird bei Aufruf von »D« ein »DIR ALL« ausgeführt. Parameter können jedoch auch zwischen die Bestandteile der Definition eingefügt werden, hierzu dient der Platzhalter »[]«, an dessen Stelle beim Aufruf übergebene Parameter eingefügt werden. Wollen wir also optional noch ein Laufwerk angeben, hilft

ALIAS D DIR [] ALL

Leider können die Parameter nur auf eine der beiden Weisen plazierte werden, ein »D DFO: >PRT:« würde nicht zur erwünschten Druckerausgabe führen, da der Befehl als »DIR DFO: >PRT: ALL« interpretiert würde (und die Umleitung sowieso an der falschen Stelle stünde). Aber da kann man sich notfalls auch ein neues Alias definieren, beispielsweise

ALIAS DP DIR >PRT: [] ALL

Auf diese Weise kann man dem Betriebssystem ein völlig neues Aussehen geben, häufig verwendete Befehle abkürzen oder gar ein "deutsches" DOS zusammenbasteln.

Auf die Dauer ist es sicher recht aufwendig, bei jedem Aufruf des Shell erst einmal für das richtige "Betriebsklima" zu sorgen. Daher ist die Möglichkeit gegeben, feste Aktionen bei jedem Öffnen eines Shell-Fensters automatisch zu tätigen. Verantwortlich hierfür ist die Datei »Shell-Startup« im Verzeichnis »S:«. Alle in ihr enthaltenen Befehle werden beim Aufruf der Shell von alleine ausgeführt. Hier kann man also zum einen den Prompt de-

Editorbefehle von Shell und »NEWCON:«

Diese Tabelle zeigt alle Steuersequenzen, die im Shell oder bei der Verwendung des Gerätes »NEWCON:« möglich sind.

Bearbeitung der Eingabezeile

[Cursor-li],[Cursor-re]	: Bewegung des Cursors
[SHIFT+Cursor-links],[CTRL+A]	: Cursor an Zeilenanfang
[SHIFT+Cursor-rechts],[CTRL+Z]	: Cursor an Zeilenende
[BACKSPACE],[CTRL+H]	: Löscht Zeichen links vom Cursor
[DEL]	: Löscht Zeichen rechts vom Cursor
[CTRL+K]	: Löscht Zeile ab Cursor bis Zeilenende
[CTRL+U]	: Löscht Zeile ab Cursor bis Zeilenanfang

Bewegung im Eingabepuffer

[Cursor-auf],[Cursor-ab]	: Blättern im Puffer
[SHIFT+Cursor-ab],[CTRL+B]	: Sprung ans Pufferende
[SHIFT+Cursor-auf],[CTRL+R]	: Puffer nach Übereinstimmung mit den bisher eingegebenen Zeichen durchsuchen

Tabelle: Editierfunktionen in CLI und Shell


```
Prompt "%N.%S> "
alias xcopy copy [] clone
alias endshell endcli
alias pro execute s:spat protect []
alias sdate execute s:spat setdate []
alias ren execute s:spat rename []
alias clear echo ""E[0;0H"E[J"
alias reverse echo ""E[0;0H"E[41;30m"E[J"
alias normal echo ""E[0;0H"E[40;31m"E[J"
alias exit endcli
alias del delete
alias md mkdir
```

Listing: Ein Beispiel für die Gestaltung der Datei »Shell-Startup«

finieren und Alias-Zuweisungen vornehmen, zum anderen kann man aber auch beliebige andere Befehle ausführen lassen. Folgendes Listing zeigt ein Beispiel für Shell-Startup, welches ein gewisses MS-DOS-Feeling verwirklicht. Für das CLI ist eine ähnliche Starthilfe in Form der Datei »S:CLI-Startup«. Allerdings können dort nicht alle Optionen des Shell ausgenutzt werden.

Starthilfe für Shell und CLI

Verfügbar ist der ganze Komfort des Shell aber nur, wenn die Datei »L:Shell-Seg« resident gemacht und das Gerät »NEWCON:« mit »MOUNT« angemeldet wurde. Dies wird im Normalfall bereits in der Startup-Sequence erledigt, so daß bei der Verwendung der Original-Sequence keine Probleme zu erwarten sind. Das Icon zum Aufruf der Shell aus der Workbench erscheint im Hauptfenster der Boot-Diskette, das alte CLI ist noch über das Systemverzeichnis zugänglich, ein Anlaß für dessen Verwendung besteht allerdings nicht mehr. In der Startup-Sequence kann auch in der bereits bekannten Weise festgelegt werden, ob mit Shell oder Workbench gearbeitet werden soll. Der Befehl hierzu lautet dann jedoch nicht »NEWCLI«, sondern »NEWShell«. Die optionalen Parameter entsprechen denen von »NEWCLI«, allerdings sollte als Eingabegerät »NEWCON:« gewählt werden, um die editierbare Eingabezeile zu erhalten. (Dies geht auch bei »NEWCLI«, allerdings fehlen dann dort Features wie »ALIAS« oder der erweiterte Prompt.) Mit

```
NEWSHELL NEWCON:0/0/640/ 256/
"Willkommen im Shell"
```

würde also ein Shell mit maximaler Fenstergröße eröffnet. Das Schließen eines Shell-Fensters erfolgt mit »ENDSHELL«, es kann jedoch auch »ENDCLI« verwendet werden. Einen echten Befehl »ENDSHELL« gibt es nämlich gar nicht, wenn Sie einen genaueren Blick in »Shell-Startup« riskieren: Dort wird nämlich ganz einfach eine ALIAS-Definition vorgenommen. Daher können Sie den Befehl quasi selbst bestimmen, zum Beispiel ein def-

tiges »AUS« oder »SCHLUSS«, MS-DOS-Kennern sei »ALIAS EXIT ENDCLI« empfohlen...

Die Vorteile der neuen Version von AmigaDOS liegen also auf der Hand. Für alle, die jedoch vor den Kosten des Umstiegs zurückschrecken, gibt es auch günstige Alternativen in der Public Domain. Den Komfort von editierbaren Kommandozeilen oder Alias-Befehlen bieten eine Vielzahl von Shells, die auch unter 1.2 laufen, als Beispiel sei hier »xy-SHELL« genannt, welches auf Disk »yx« der FISH-Serie erschienen ist. Wesentlich interessanter ist jedoch das "Amiga Replacement Project" (ARP), welches unter anderem auf der KICK PD Disk 220 zu finden ist – es bietet einen kompletten Ersatz für die Kommandos der offiziellen Version von Commodore.

Seit 1987 versuchen die Programmierer des ARP, die "Schlampereien" von Commodore auszubügeln und AmigaDOS zu mehr Leistung und Komfort zu verhelfen. Mit gutem Erfolg, wie die Version 1.3

von ARP zeigt, die für alle DOS-Befehle optimierten Ersatz liefert. Schwerpunkte der Optimierung sind vor allem Geschwindigkeit und Größe der Programme. Die alternativen Befehle sind im Schnitt nur halb so groß wie die offiziellen Varianten. Der Platzbedarf wird zwar durch eine zusätzlich notwendige Library wieder ausgeglichen, in der Praxis werden jedoch wesentlich kürzere Zeiten in den Diskettenzugriffen erreicht, was nicht zu verachten ist. Dafür stellt die ARP-Library jedoch auch völlig neue Funktionen zur Verfügung, die insbesondere für Programmierer interessant sind und inzwischen auch von vielen Programmen, vor allem in der Public Domain, benutzt werden – die ARP-Library ist fast schon inoffizieller Bestandteil von AmigaDOS. ARP 1.3 ist jedoch nicht nur voll kompatibel zu AmigaDOS 1.3 (inklusive Shell, »RESIDENT« usw.), sondern bietet auch zusätzliche Erweiterungen, teilweise in Form neuer Befehle, teilweise aber auch in erweiterten Optionen wie etwa der Verwendung MS-DOS-kompatibler Wildcards (*). In Preis und Leistung ist das ARP in jedem Fall eine empfehlenswerte Alternative zum offiziellen DOS, sowohl zum Aufstieg von Version 1.2 als auch als Ersatz für 1.3.

User's Little Helper

Auch für die alltäglichen Arbeiten unter DOS gibt es eine Vielzahl von Hilfen. Insbesondere das Kopieren von Dateien und die Hege und Pflege der Verzeichnisbäume artet mit den normalen DOS-Befehlen in stressige Tipparbeit aus – besonders auf großen Festplatten. Aber auch hier bietet der Software-Markt Lebenshilfen, die man sogar wieder mit der Maus bedienen kann. Das Spektrum ist hier sehr groß, sowohl kommerzielle als auch Programme aus der Public Domain werben um die Gunst des Anwenders.

Bereits das Belegen der Funktionstasten mit häufig benötigten Kommandos kann erhebliche Erleichterung bedeuten. Stellvertretend hierfür sei der Gold Commander genannt. Viel Komfort beim Kopieren

Die Aufgabe

Als Hausaufgaben seien vorerst wieder die üblichen Experimente mit den neuen Features sowie deren Einsatz im eigenen System genannt. Konkret sollten Sie als Übung auch versuchen, das erste Beispiel soweit umzubauen, daß sowohl Anzahl der Durchläufe als auch der auszugebende Text als Parameter übergeben werden kann und daß bei ihrem Fehlen Vorgaben verwendet werden, beispielsweise zehnmal "Ich bin ein dummer Computer" oder ähnliche Weisheiten, die optional auch hörbar gemacht werden könnten. Nützen Sie eventuell auch »RESIDENT«, da bei 100 Durchläufen doch einiges an Diskettenarbeit nötig ist.

Können Sie sich auch daran wagen, eine komfortable Oberfläche für eine beliebige Programmiersprache oder Textverarbeitung zu schreiben, mit der beispielsweise Datendateien angezeigt, Backups erstellt, Arbeitsdateien ausgewählt oder Optionen gesetzt werden können; das alles in Form eines menügesteuerten Scripts und unter Ausnutzung von Steuerzeichen und anderen "Luxus-Befehlen". Dabei werden Sie vielleicht noch auf ein kleines Problem stoßen: Interaktives Setzen von Umgebungsvariablen ist mit »SETENV« leider nicht möglich. Nachdem wir jedoch schon genauer Bescheid wissen über die Struktur solcher Variablen, dürfte uns die Formulierung eines Auswegs nicht schwerfallen. Denken Sie mal an das Gerät "*" und eine Sequenz wie

```
ECHO "Dateiname: "
COPY * ENV:Dateiname
GETENV Dateiname
```

Ahnen Sie, wohin der Hase läuft? Dann versuchen Sie doch mal, ob Sie ein kleines Script namens »ASKENV« zustande bekommen, welches mit

```
EXECUTE ASKENV Variable
```

interaktives Arbeiten mit dem Environment erlaubt – natürlich mit Überwachung der Parameter und einer kleinen Anleitung auf dem Schirm. Viel Vergnügen...

und der Dateipflege liefern Programme wie die kommerziellen Programme »Disk Master« und »CLI-Mate« oder »A-Tree« aus der Public Domain (TBag 32). Mit ihnen können Dateien bequem kopiert oder Verzeichnisse verwaltet werden. Darüber hinaus bieten sie oft auch noch zusätzliche Funktionen wie das Editieren von Texten, die Anzeige von IFF-Bildern oder die Einbindung anderer Hilfsprogramme. Wenn sie auch noch den Start von Programmen erlauben, hat man schon wieder einen kompletten Ersatz für die Workbench. Das Angebot für

diese Programme ist so groß, daß wohl jeder die für seinen Geschmack passenden Hilfen findet.

Wenn wir schon beim Thema Ersatz sind: Auch zur normalen Workbench gibt es Alternativen. Da wäre zum einen die »Jazz Bench« (KICK PD Disk 220), die durch echtes Multitasking schnelleres Arbeiten erlaubt, zum anderen aber auch der »Browser« (FISH-Disk 180), der einen effektiven Kompromiß zwischen Workbench und CLI darstellt. Er überwindet das Problem, daß unter der Workbench nicht alle Dateien sichtbar sind dadurch,

daß er im Textmodus arbeitet und in den Fenstern wirklich alle Dateien anzeigt und dadurch zugänglich macht. Wie bei der Workbench können mit der Maus Dateien kopiert werden, für Handarbeiten steht jedoch immer noch eine Kommandozeile zur Verfügung. Darüber hinaus können noch eigene Menüleisten definiert werden, so daß der effektiven Arbeit kaum noch etwas im Wege steht. Der Browser ist also eine durchaus akzeptable Alternative zur Workbench.

Das Problem, das sich aus der Vielfalt an Arbeitshilfen ergibt, ist, alle Möglichkeiten effektiv

zu organisieren. Am besten ist es natürlich, wenn man eine Festplatte besitzt, auf der man das breite Angebot unterbringen kann. Die Auswahl, ob heute mit Shell, Workbench oder Browser gearbeitet werden soll, kann in der »Startup-Sequence« erfolgen, dort können auch andere Hilfsprogramme geladen werden. Für den reinen Diskettenbetrieb muß wohl für jede Oberfläche eine eigene Startdiskette erstellt werden. Das nötige Know-how haben wir ja im Laufe dieses Kurses erlangt...

(mm)

```

Listings
1: ; BEISPIEL1.S - Schleifenprogrammierung
2: ; mit Umgebungsvariablen und EVAL
3:
4: ECHO "**ec"
5: SETENV Count 0
6: LAB Loop
7:   EVAL <ENV:Count >NIL: TO ENV:Temp VALUE2=1 OP=+ ?
8:   COPY ENV:Temp ENV:Count
9:   ECHO "Ich bin ein Amiga..."
10:  IF NOT $Count EQ 10 VAL
11:    SKIP Loop BACK
12:  ENDIF
13: ECHO "Das war alles..."
Listing: CLI-Kurs

```

```

1: ; BEISPIEL2.S - EVAL in der Praxis
2:
3: ECHO "**ecWertetabelle*n-----*n"
4: SETENV Count 31
5: LAB Loop
6:   EVAL <ENV:Count >NIL: TO ENV:Temp VALUE2=1 OP=+ ?
7:   COPY ENV:Temp ENV:Count
8:   EVAL <ENV:Count >NIL: TO ENV:Ascii LFORMAT "%C" ?
9:   EVAL <ENV:Count >NIL: TO ENV:Dez LFORMAT "%i3 : " ?
10:  EVAL <ENV:Count >NIL: TO ENV:Hex LFORMAT "%x2 : " ?
11:  EVAL <ENV:Count >NIL: TO ENV:Oct LFORMAT "%o3 : " ?
12:  JOIN ENV:Dez ENV:Hex ENV:Oct ENV:Ascii TO ENV:Output
13:  GETENV Output
14:  IF NOT $Count EQ 127 VAL
15:    SKIP Loop BACK
16:  ENDIF
17: ECHO "**nEnde der Wertetabelle..."
Listing: CLI-Kurs

```

FIBU deluxe + OASE # 101

Suchen Sie schon lange eine leistungsstarke Finanzbuchhaltung? Jetzt gibt es die neueste Version der bereits über tausendfach bewährten "FIBU deluxe". Nicht nur der sagenhafte Preis von nur DM 49,- für das Komplettpaket (incl. 40-seitigem Handbuch) setzt neue Maßstäbe, auch die Leistungsstärke:

- FIBU deluxe + verwaltet bis zu 2000 Konten aller Art
- FIBU deluxe + ist mandantenfähig (mit universellen Steuersätzen oder Währungen)
- FIBU deluxe + gibt Ihnen AfA-Vorschläge für jedes Ihrer Anlagekonten
- FIBU deluxe + erstellt umfangreiche Auswertungslisten und druckt diese komfortabel aus
- FIBU deluxe + bietet Ihnen wahlweise einfache Maus- oder Tastaturbedienung
- FIBU deluxe + erstellt komplette Bilanzen, Journale, Kassenberichte oder GuV-Rechnungen
- FIBU deluxe + beinhaltet einen TEXTEDITOR und eine FAKTURA mit direkter Verbuchung

nur sagenhafte **DM 49,-**

EINKOMMENSTEUER '90

Endlich gibt es das neue Steuerprogramm 1990 mit sämtlichen neuen Steuerrichtlinien der Steuerreform. Dieses Programm erstellt Ihre komplette Lohn- und Einkommensteuererklärung 1990. Natürlich auch mit der aktuellen Lohnsteuertabelle für 1991. Es werden 99% aller Normal- und Sonderfälle abgedeckt. Mit jährlichem Updateservice!

nur vorbildliche **DM 59,-**

OASE # 109

AMIGA
Köln 90

Wir stellen aus
09.-11. November 1990
Halle 12, Stand 216

WOLF

software & design Tel.: 02541/2874
Deipe Stegge 187 FAX: 02541/71172
4420 Coesfeld Händleranfragen willkommen!

Verandkosten: Vorkasse (V-Scheck) DM 3,- (Ausland DM 6,-) / Nachnahme DM 7,- (Ausland DM 15,-)

SOFTWARE - WERBUNG - DESIGN



OASE SOFTWARE

Die Problematik ist unübersehbar. Das Programm "Clock" benutzt ein eigenes Fenster, das meistens im Weg ist und von größeren Windows überdeckt wird, falls man ein Hintergrund-Gadget anklickt.

Die Lösung des Problems heißt »CPointer«, abgeleitet von "Clock-Pointer" (frei übersetzt: Uhrzeiger). Wie schon am Namen zu erkennen ist, verändert das Programm den Mauszeiger so, daß in diesem die aktuelle Uhrzeit angezeigt wird. Wenn man »CPointer« startet, dauert es rund fünf Sekunden, bis der neue Mauszeiger zu sehen ist. Diese Verzögerung wurde eingebaut, damit der Anwender ein anderes Fenster als das des CLI aktivieren kann, um dort den Clock-Pointer zu positionieren.

Stefan Heller

Der Mauszeiger und die Zeit

Der Amiga zeigt jedem, was die Stunde geschlagen hat. Er besitzt dafür ein Programm und benutzt sogar ein eigenes Window – wie umständlich!

Beendet wird das Programm durch (längeres) Drücken der Funktionstaste F10.

Die eigentliche Routine »prtime«, die das Programm steuert, wurde so eingebaut, daß sie als CLI-Befehl wirkt. Es

stellt jedoch kein Problem dar, die Routine »prtime« in anderen Programmen zu verwenden.

Möchte man sie benutzen, muß »prtime« dann regelmäßig (zirka alle 1 bis 30 Sekunden)

aufgerufen werden, um die Uhrzeit zu "refreshen". Die Breite des Sprites ist auf 16 Pixel begrenzt. Daher wurde die Darstellung "HH:MM" für die Uhrzeit gewählt. HH stellt die Stunden von 0 bis 12 und MM die Minuten von 0 bis 59 dar.

Das Programm »CPointer« wurde mit dem DevPac-Assembler V2.0 geschrieben. Der Aufbau und die Benutzung der Systemfunktionen sind dem gut dokumentierten Listing zu entnehmen und dürften dem Anwender so keine Probleme bereiten.

Sicherlich ist dieses Programm ein gutes Beispiel dafür, wie man auf eine recht "einfache" Art zwei Fliegen mit einer Klappe schlagen kann. Vielleicht fallen Ihnen ähnlich gute Beispiele ein.

(vb)

Listings

```

1: ;*****
2: ;* CPointer
3: ;* CPointer veraendert den Mauszeiger und zeigt dort *
4: ;* die aktuelle Uhrzeit an
5: ;* (c) 1990 Stefan Heller & AMIGA DOS
6: ;* Sprache: DevPac Assembler V.2.0
7: ;*****
8:
9: ;*****
10: ;** die folgenden Angaben den eigenen Gegebenheiten **
11: ;** anpassen **
12:
13: include sys:include/exec/exec_lib.i
14: include sys:include/intuition/intuition_lib.i
15: include sys:include/libraries/dos_lib.i
16:
17: ;*****
18:
19: ib_ActiveWindow equ 52 ;Konstanten
20: REFRESH_TIME equ 50
21:
22: ;*****
23:
24: start lea iname,a1 ;Intuition-Library oeffnen
25: clr.l d0
26: CALLExec OpenLibrary
27: move.l d0,_IntuitionBase
28: beq ende1
29: lea dname,a1 ;DOS-Library oeffnen
30: clr.l d0
31: CALLExec OpenLibrary
32: move.l d0,_DOSBase
33: beq ende2
34: moveq #(dataend-data),d0
35: ;*****
36: moveq #2,d1 ;Speicher fuer Spritedaten
37: ;im Chipram reservieren
38: CALLExec AllocMem
39: move.l d0,adr ;Startadresse des Blocks retten
40: beq ende3
41: move.l adr,a1 ;Spritedaten in den
42: lea data,a0 ;reservierten Block kopieren
43: moveq #(dataend-data-1),d0
44:
45: .loop move.b (a0)+,(a1)+
46: dbf d0,.loop
47: bsr prtime ;und Uhrzeit hineinschreiben
48: move.l #250,d1 ;5 Sekunden warten, damit User
49: CALDOS Delay ;sein Fenster aktivieren kann
50: movea.l _IntuitionBase,a6
51: ;Zeiger des aktiven Windows
52: move.l ib_ActiveWindow(a6),win
53: ;holen und speichern
54: movea.l win,a0 ;"ClockPointer" aktivieren
55: movea.l adr,a1
56: moveq #16,d0 ;Groesse des Sprites y=16
57: moveq #16,d1 ;und x=16 (max. Groesse)
58: moveq #-8,d2 ;rel. Koordinaten des Hot
59: moveq #-6,d3 ;Spots (-8/-6)
60: CALLINT SetPointer
61:

```

Listing: CPointer

```

62:
63: wait moveq #REFRESH_TIME,d1 ;Hauptschleife
64: CALDOS Delay ;eine Sekunde warten
65: bsr prtime ;Uhrzeit refreshen
66: cmp.b #$4d,$bfec01 ;F10 gedrueckt?
67: bne wait ;wenn nein, weiter warten
68: movea.l win,a0 ;alten Mauszeiger aktivieren
69: CALLINT ClearPointer
70: moveq #(dataend-data),d0 ;Speicher freigeben
71: movea.l adr,a1
72: CALLExec FreeMem
73:
74:
75: ende3 movea.l _DOSBase,a1 ;DOS-Library schliessen
76: CALLExec CloseLibrary
77:
78: ende2 movea.l _IntuitionBase,a1 ;Intuition-Library schliessen
79: CALLExec CloseLibrary
80:
81: ende1 rts ;zurueck zum CLI
82:
83: ;*****
84: ;***** Unterprogramm: prtime *****
85: ;***** akt. Zeit in Pfeil schreiben *****
86:
87: prtime move.l #ds_Days,d1 ;akt. Zeit holen
88: CALDOS DateStamp
89: moveq #4,d0 ;Hintergrund (Doppelpunkt)
90: lea prtback,a0 ;in Arbeitsblock kopieren
91: lea prtwork,a1
92:
93: .loop move (a0)+,(a1)+
94: dbf d0,.loop
95: move.l ds_Minute,d4
96:
97: divu #60,d4 ;Minuten (seit 0:00) durch 60
98: ;teilen -> Stunden
99: d5 ;unteres Wort von d4 nach d5
100: move d4,d5 ;kopieren -> Stunden
101: clr d4 ;unteres Wort loeschen und gegen
102: swap d4 ;oberes vertauschen -> Rest=Minuten
103: divu #10,d4
104: ;Minuten in Einer- und Zehnerstellen
105: ;umrechnen
106: move d4,d0
107: moveq #1,d1
108: bsr print ;Zehnerstellen drucken
109: d4
110: move d4,d0
111: clr.l d1
112: bsr print ;Einerstellen drucken
113: #12,d5 ;Stunden in Zahl zw. 0 und 12
114: bsr hour ;umrechnen und drucken
115: sub.l #12,d5
116:
117:
118: hour cmpi.b #10,d5
119: blt hour10
120: sub.l #10,d5
121: moveq #1,d0 ;Zehnerstelle (=1)
122: moveq #3,d1
123: bsr print ;drucken
124:
125:

```

Listing: CPointer




```

126:
127:      hourlo    move d5,d0
128:      moveq    #2,d1
129:      bsr      print          ;Einerstellen drucken
130:
131:      copy      moveq #4,d0
132:
133: ;** Arbeitsblock (berechnete Bitmap) in Speicher **
134: ;** des Sprites kopieren **
135:
136:      lea      prtwork,a0
137:      movea.l  adr,a1
138:      adda.l   #56,a1
139:
140:      .loop    move (a0)+(a1)+
141:
142: ;** Bitmap nach Plane 1 von Sprite Bits in Plane 2 **
143: ;** loeschen **
144:
145:      clr      (a1)+
146:      dbf      d0,.loop
147:      rts
148:
149: ;***** Unterprogramm: print *****
150: ;***** best. Stelle in Arbeitsblock drucken *****
151:
152:      print    mulu #5,d0          ;Adresse der Zifferndaten
153:      lea      numtab,a0          ;berechnen
154:      adda.l   d0,a0
155:      lea      prtwork,a1
156:      moveq    #4,d2
157:
158:
159:      .loop    clr.l d0          ;ein Byte der Zifferndaten
160:      move.b   (a0)+,d0          ;holen und Bits verschieben
161:      tst.b    d1                ;Minuteneinerstelle (d1=0) ?
162:      beq      .trans            ;wenn ja, in Arbeitsblock kopieren
163:      lsl      #4,d0             ;wenn nein, 4 Bits nach links
164:      cmpi.b   #1,d1             ;verschieben
165:      beq      .trans            ;Minutenzehnerstelle (d1=1)
166:      lsl      #6,d0
167:      cmpi.b   #2,d1             ;Stundeneinerstelle (d1=2)
168:      beq      .trans
169:      lsl      #4,d0             ;Stundenzehnerstelle (d1>2)
170:

```

Listing: CPointer

```

171:
172:      .trans   or d0,(a1)+
173:      dbf      d2,.loop
174:      rts
175:
176: iname dc.b   "intuition.library",0      ;Librarynamen
177: even
178:
179: dname dc.b   "dos.library",0
180: even
181:
182: _IntuitionBase dc.l 0          ;Basiszeiger der Libraries
183: _DOSBase       dc.l 0
184: adr            dc.l 0
185:                ;Zeiger auf Speicherblock im Chipram
186: win            dc.l 0          ;Zeiger auf aktives Fenster
187: ds_Days        dc.l 0          ;DateStamp-Struktur:
188: ds_Minute      dc.l 0          ;Minuten seit 0:00
189: ds_Tick        dc.l 0
190:                ;nur ds_Minute wird von benutzt
191:
192: data dc.w     0,0              ;Spritdaten (Kreuz)
193:          dc.w  $80,$80,$80,$80,$80,$80,$80,$80
194:          dc.w  0,0,$1e3c,$1e3c,0,0
195:          dc.w  $80,$80,$80,$80,$80,$80,$80,$80
196:          dc.w  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
197:          dataend                ;Leerzeilen fuer Zeit
198:
199:
200: prtback dc.w   0,$100,0,$100,0      ;Doppelpunkt
201: prtwork ds.w   5                    ;Arbeitsblock
202: numtab  dc.b   7,5,5,5,7            ;Daten der Ziffern von 0
203:          dc.b   1,1,1,1,1
204:          dc.b   7,1,7,4,7
205:          dc.b   7,1,3,1,7
206:          dc.b   5,5,7,1,1
207:          dc.b   7,4,7,1,7
208:          dc.b   7,4,7,5,7
209:          dc.b   7,1,1,1,1
210:          dc.b   7,5,7,5,7
211:          dc.b   7,5,7,1,7
212: even
213: end
214:
215: ;***** Ende von CPointer *****
216:

```

Listing: CPointer



;bis 9

Achtung, Super-Service!

Als besonderen Service der AMIGA DOS gibt es ja die bekannten privaten Kleinanzeigen, in denen Soft- und Hardware angeboten werden kann oder die einfach der Kontaktaufnahme mit anderen Amiga-Interessierten dient. Für die Ausgaben 11 und 12 des Jahrgangs 1990 können wir Ihnen nun eine besondere Attraktion bieten:

Die kostenlose private Kleinanzeige!

Und so können Sie dieses tolle Angebot wahrnehmen:

Benutzen Sie einfach die im Heft befindliche Servicekarte und notieren Sie Ihre Kleinanzeige. Schicken Sie diese Postkarte dann an den Verlag, und Ihre Kleinanzeige erscheint in der nächsterreichbaren Ausgabe von AMIGA DOS!

Kleinanzeigen

Biete Software

STOP: ARISC HAT SOFTWARE FÜR DEINEN AMIGA SOWIE BESCHREIBUNG PD-SERIEN. INFO BEI ARISC: POSTFACH 102, 1165 WIEN

Public Domain/Shareware für AMIGA ATARI IBM/PC
Katalog 1.00 DM in Briefmarken
Jede Disketten 2.00 DM inkl.
MWSt. Bei: Brunosoft/Schreiber
1000 Berlin 51, Sommerstr. 37 **G**

Harddisk-Menü für alle Amigas,
Programme per Mausclick starten
49,- DM, Demo 10,- DM. IHS Bartsch,
Lindenstr. 21, 8900 Augsburg 1 **G**

PD-SOFT SCHWEIZ! Privatverkauf
oder Tausch. 2 Katalogdisks bei
Hugo Kneubühler, Hübeliweg 284.
CH-3078 Richigen,
Tel: 0 31 83 68 51

Computer Hard- und Software
z.B. Page Setter DTP Prog. 49,- DM
Liste anfordern - HSL Udo Lenzen
Klinkhammer 4, 4060 Viersen 11 **G**

VM's Public-Domain-Kiste
Amiga Public Domain, jede Disk.
2.00. Katalogdisk. 2,00 in Briefm.
V. Maciejek, Hebbelstr. 6,
4130 Moers 1 **G**

Biete Hardware

Verkaufe DTP-Paket AMIGA 2500
mit 5 MB RAM AOC-Monitor, Flick-
kerfixer, Postscript-Laserprinter
und div. Software!!!
NP 23'000 Fr VB 16'000 Fr.
René Bäder, Fliederstrasse 3,
6010 Kriens, Schweiz
Tel. 0 41/41 54 01 (24h)

512 KB-Erweiterung mit Uhr f.
A500 (Original A501) DM 100,-
VB, PRYBYLSKI, Engerstr. 31,
4900 Herford, Tel.: 052 21-45 41

MultiCom Soft- & Hardwareversand
Der Versand mit den vernünftigen
Preisen! Gratisliste bei:
MultiCom, Obere Mühle 1
586 Iserlohn (Tel. 0 23 71/1 37 34) **G**

Verschiedenes

Neueröffnung! Besuchen Sie uns!
Tgl. ab 16 h, GAME-PLAY, Öster-
wieher Str.70, 4837 Verl 1,
T: 0 52 46/15 72, MEGA Drive,
Engine, GAMEBOY, SNK, Spiele
ab 19,- DM, T.: 0 52 46/15 72 **G**

COMMODORE REPARATUREN
FRANKFURT/M.
TEL.: 0 69/45 27 22 **G**

Edgar Meyzies

Assembler ist keine Hexerei

Nachdem wir uns im letzten Heft mit der Erstellung der Programme beschäftigt haben, kommen wir jetzt zu den Programmbeispielen und weiteren Erklärungen.

Falls es für Sie verwirrend sein sollte, hier ein kurzer Überblick über die Kursteile:

Dies gehört immer noch zu **Teil 1**. Wir erfahren heute etwas über die Codierung und bekommen die Programme, an denen wir unsere Kenntnisse erforschen wollen, auf "dem silbernen Tablett serviert". Mit **Teil 2** geht es dann im nächsten Heft richtig los.

Es darf codiert werden

Sicherlich werden Sie der Anforderung, beide Programm-entwürfe nun in Assembler zu realisieren, sehr viel williger folgen als letztes bei der Anforderung zum Entwurf. Unsere beiden Lösungsversuche sind als Listing 2 und 3 abgedruckt. Die benötigten Vereinbarungen für den Zugriff auf Systemroutinen und anderes mehr finden Sie in Listing 1. Sie können auch die Include-Dateien des Aufsteigerkurses verwenden, nachdem Sie die Korrekturen vorgenommen haben, wie in Listing 2 vermerkt. Die aktuelle Fassung der Include-Dateien stellen wir Ihnen in der Databox zu dieser Ausgabe zur Verfügung.

Tools im Einsatz

Zur Übersetzung der Programme rufen Sie den Assembler wie folgt auf, vorausgesetzt Sie haben ein Projektverzeichnis (mit den Unterverzeichnissen »txt«, »obj«, »incl« und »bin«) angelegt und als aktuell bestimmt:

```
A68K txt/TestLauf1.asm obj
/TestLauf1.o -d
```

Die Option »-d« veranlaßt den Assembler, einen Hunk mit den verwendeten Symbolen (Namen) anzulegen. Übersetzen Sie das Programm auch ohne die Option, und vergleichen Sie die unterschiedlichen Längen. Ein Symbol-Hunk frißt Speicherkapazität, sowohl auf der Diskette als auch im Rechner.

Um das Programm zu binden, setzen Sie den Linker wie folgt ein:

```
BLINK obj/TestLauf1.o TO bin/
TestLauf1
```

Falls Sie schon jetzt den Debugger kennenlernen wollen, so starten Sie ihn bitte mit dem Aufruf:

```
Debug
```

laden das Modul mit:

```
l bin/TestLauf1
```

und führen es bis zu einem Label aus, zum Beispiel mit:

```
g TextAusgeben
```

Nach dem Laden des Moduls zeigt der Debugger an, daß drei Hunks entsprechend unserer Sections (»CODE«, »DATA«, »BSS«) geladen wurden. Den Adressen ist zu entnehmen, daß der BSS-Hunk sich im Chip-Memory befindet. Damit ist auch klar, wie der Assembler und der Linker auf Wünsche nach Aufteilung eines Programms im Speicher reagieren.

Zum Abschluß des Testlaufs regen wir an, mit dem Debugger zu spielen, um das Werkzeug richtig kennenzulernen. Zu beiden Ablaufplänen ist noch anzumerken, daß sie optimiert werden können, sowohl in der Länge als auch in der Ausführungsgeschwindigkeit. Reizt es Sie nicht, Dampf zu machen? Wir würden uns freuen, wenn Sie unsere Lösungen

Kursfahrplan
Teil 1: Einführung und Grundlagen
Teil 2: Benutzeroberfläche entwickeln
Teil 3: Steuerung realisieren
Teil 4: Dynamische Datenstrukturen
Teil 5: Requester mit Funktionen
Teil 6: Programmschnittstelle schaffen

verbessern. Versäumen Sie es auch nicht, Ihren, aber auch unseren Testideen nachzugehen. Es lohnt sich! Bei Fragen zur Codierung kann Ihnen ein Buch zur Programmierung des 68000 sicherlich wertvolle Hilfe leisten, zum Beispiel [10] und [11].

Ab auf die Workbench

Die bisher erstellten Programme laufen nur vom CLI aus. Wir wollen uns nun die Möglichkeit erschließen, Programme auch von der Workbench aus zu starten. Dazu steigen wir wieder in eine Analysephase ein und fordern uns selbst heraus, einen Startup-Code zu schreiben, der für unsere Programme Ablaufbedingungen herstellt, die den Start von der Workbench und vom CLI aus zulassen.

Das Entwicklungsziel kann wie folgt formuliert werden:

Der Startup-Code

- soll reentrant sein,
- ohne Anpassungen nutzbar sein,
- auf der Ebene einer Objekt-codedatei den eigentlichen Programmen zur Verfügung stehen,
- mit den Anwendungen nicht durch einen gemeinsamen Satz an globalen Objekten verzahnt sein,
- letztlich mindestens acht Argumente auswerten,
- die gewachsenen Konventionen für die Schnittstelle zu den Anwendungen einhalten,
- bei Start von der Workbench kein Fenster anlegen und
- ausbaubar sein.

Zu den Entwicklungszielen sind einige Erläuterungen angebracht: "nutzbar ohne An-

passung" bedeutet, daß der Objektcode so universell zu halten ist, daß er mit beliebigen Anwendungsprogrammen gebunden werden kann. Die Forderung bedeutet allerdings auch, daß der Startup-Code keine Fenster öffnet und auch nicht auf Argumente reagiert, die in Icons eingetragen sind. In diesem Zusammenhang ist auch die Forderung nach "Ausbaubarkeit" zu betrachten. Der Startup-Code muß strukturiert sein, um problemlos ergänzt zu werden. Der verwendete Assembler läßt bedingte Übersetzungen zu und bietet somit wesentliche Hilfen, um Ergänzungen leicht und gezielt einzubeziehen.

Einfache Schnittstelle

Der Startup-Code würde an Universalität verlieren, wenn seine völlige Integration in die Anwendungen gefordert wäre. Die Schnittstelle zwischen Anwendung und Startup-Code würde an Übersichtlichkeit einbüßen. Der Preis dafür ist eine größere Länge des resultierenden Programms. Einfachheit kostet halt ihren Preis.

Bei dem beabsichtigten Vorgehen gibt es nur drei Berührungspunkte mit der Anwendung:

- Aufruf aus dem Startup-Code mit dem symbolischen Namen »__main«, den die Anwendung mit »XDEF« exportiert.
- Übergabe der Parameter an die Anwendung (nach Auswertung) über den Stack:
 - »argc« = Anzahl der Argumente (bei CLI-Start) beziehungsweise »null« (bei Start von der Workbench)
 - »argv« = Adresse des Feldes, das die Adressen der Ar-

gumente enthält (CLI-Start) beziehungsweise Adresse der Startup-Message (WB-Start)

- Übernahme eines Fehlercodes von der Anwendung im Register »d0«.

Damit sind zugleich die wesentlichen Konventionen für die Programmierung von Startup-Codes beschrieben (siehe dazu ausführlich zum Beispiel [12]).

Startup-Code entwerfen

Im nächsten Schritt ist der Programmablaufplan an der Reihe. Da es sich hier um Systemprogrammierung handelt, wären wir nicht enttäuscht, wenn Sie sofort unseren Lösungsansatz aus Heft 09/90, Abbildung 5 studierten, ohne sich selbst der Mühe zu unterziehen, einen Plan zu erstellen. Dazu nun einige Kommentare:

Einem gestarteten Programm bietet die CLI-Umgebung Argumente (Adresse der eingegebenen Zeichenkette in »a0« und ihre Länge in »d0«) an. Selbst bei Start von der Workbench sind die Informationen über die Argumente zunächst zu sichern, da die Register a0 und d0 die »Arbeitspferde« der Systemroutinen sind.

Der Prozeßstruktur ist zu entnehmen, ob ein Programm von der Workbench aus gestartet wurde (siehe [12]). Als sicheres Erkennungsmerkmal gilt der vorhandene beziehungsweise fehlende Eintrag der Adresse einer »CommandLineInterface-Struktur«. Beide Strukturen finden Sie in Listing 4; sie sind in [12] und in [13] gut beschrieben. Die Prozeßstruktur können wir erst analysieren, wenn der Startup-Code ihre Adresse ermittelt hat.

In der Abbildung 5 aus Heft 09/90 werden Sie keine Verwendung der DOS-Library entdecken. Dennoch wird ihre Öffnung jetzt schon vorgesehen, um die Struktur des Startup-Codes insgesamt zu errichten. Es würde naheliegen, die Adresse der DOS-Library an die Anwendung weiterzugeben, wir verzichten jedoch aus obigen Gründen darauf. Bei einem maßgeschneiderten Startup-Code wäre anders vorzugehen.

Je nach Umgebung, von der aus unser Programm gestartet wurde, sind entweder die CLI-Argumente (CLI-Start) oder die Workbench-Argumente (WB-Start) auszuwerten. In der heu-

tigen Ausgabe gehen wir darauf noch nicht tiefer ein.

Bei CLI-Start ist zum Beispiel die übernommene Zeichenkette (Argumente retten) auszuwerten. Der Ballast ist zu entfernen, die Argumente sind zu zählen und ihre Adressen sind in einem Array festzuhalten. Das Array ist dynamisch anzulegen, damit der Code reentrant bleibt.

Bei WB-Start wartet ein Programm (Konvention) auf den Empfang einer Startup-Message von der Workbench. Erst dann beginnt es loszulegen. Die erhaltene Message bietet den Zugriff auf die zu übernehmenden Argumente (Icons). Bei WB-Start ist die Startup-Message an die Workbench zurückzusenden (siehe »Reply Msg«). Damit das Workbench-Programm nicht sofort damit beginnt, den Arbeitsspeicher freizugeben, den unser Programm belegt, schützen wir uns mit einem »Forbid()«. Schließlich wollen wir ja noch den Code zu Ende abarbeiten.

Der Startup-Code dient nur dazu, für unser eigentliches Programm die benötigten Ablaufbedingungen zu schaffen und diese hinterher wieder rückgängig zu machen. Mit »jsr _main« wird das Programm aufgerufen.

Startup codieren

Mit diesen Informationen versehen könnten Sie sich nun ans Werk machen, um Ihren Startup-Code zu erstellen. Probieren Sie es ruhig. Wenn es nicht weiter geht, dann »spitzeln« Sie halt mal in Listing 5.

Test und Integration

Der Startup-Code läßt sich zunächst ohne eine Anwendung testen. Dazu ist es lediglich erforderlich, den Import von »_main« zu sperren (Stern vor »XREF«). Auch der Debugger kann Ihnen gute Dienste leisten, wenn Sie den Aufstart vom CLI austesten wollen. Den Workbench-Mode unterstützt er leider nicht. (Hier könnte das Tool »ShowRegs« einspringen, das wir im Aufsteigerkurs entwickelt haben.) Ohne Aufruf einer Anwendung muß »Startup« sauber terminieren. Es darf kein Arbeitsspeicher belegt bleiben (Test mit »C:avail«). Inspizieren Sie insbesondere die Belegung des Stack, um den Gurus nicht ausgesetzt zu sein. Sollte der

Debugger mal nicht mehr wollen, dann muß es nicht unbedingt an Ihrem Programm liegen. Gelegentlich mag er einfach nicht mehr, ohne daß die Ursachen erkennbar sind.

Wenn der erste Testschritt erfolgreich verlaufen ist, dann kann die Integration in eine Anwendung erfolgen, die dadurch die Aufwertung zu einem Workbench-fähigen Programm erfährt. Dazu importieren Sie bitte in »Startup« den Namen »_main« und übersetzen das Modul erneut. Als Anwendung könnten Sie zum Beispiel »Startup1« wählen und »_main«, wie in Listing 3 bereits vorgesehen, exportieren (Stern löschen vor »XDEF«). Selbstverständlich ist auch dieses Modul erneut zu übersetzen. Schließlich sind beide Module zu einem Programm zu binden:

```
BLINK obj/Startup.o obj/TestL
auf1.o TO bin/TestStartup
```

Wir schreiben die Anweisungen für das Binden stets in eine Batch-Datei und führen sie mit »C:Execute...« aus, um uns die Tipperei zu ersparen.

Was bringt die nächste Ausgabe?

Im nächsten Heft soll der Startup-Code um die Auswertung der Argumente vervollständigt werden. Als angehenden Champion sollten Sie sich schon der Aufgabe stellen, herauszubekommen, wie dabei vorzugehen ist. Analysieren Sie dazu anhand des Be-

nutzerhandbuchs die Möglichkeiten der Eingabe von Argumenten auf der CLI-Kommandozeile. Es sollte nicht allzu schwer sein, die zu eliminierenden Zeichen herauszubekommen und die »reinen« Argumente in den Puffer »su_argvBuf« zu übertragen sowie zu zählen. In diesen Puffer gehört auch der Name des Programms (Listing 4, »CLI_CommandName«). Um sich selbst an die Auswertung der Workbench-Argumente zu machen, könnten Sie zum Beispiel [14] zu Rate ziehen. Auch dieses Thema gehen wir nächstes Mal an.

Schließlich werden wir uns dem File-Requester, unserem Hauptthema, zuwenden. Sie könnten in der Zwischenzeit in die Analyse- oder gar in die Entwurfsphase eintreten. Wir beabsichtigen zusammen mit Ihnen einen Requester zu realisieren, der von allen Programmen genutzt werden kann, die über eine entsprechende Schnittstelle verfügen. Welche Anforderungen würden Sie an den Requester stellen? Welche Entwicklungsziele ergeben sich für Sie?

Sie hätten besonders viel von diesem Kurs, wenn Sie möglichst eigene Vorstellungen entwickeln, diese für sich selbst in Entwicklungsziele umsetzen und neben unseren Lösungsvorschlägen einen eigenen Weg gehen. Wir freuen uns über Konkurrenz und auf Ihre Kritik!

(jb)

Literatur und Software

- [1] AmigaDOS 1/90 bis 5/90, sämtliche Ausgaben noch lieferbar
- [2] Kick PD 266 (Vers 2.61) oder Fred Fish 168 (Vers 2.42)
- [3] AmigaDOS 1/90
- [4] Kick PD 266 oder Fred Fish 110
- [5] Das Amiga Guru-Buch, 1989, S. 146 ff.
- [6] Fred Fish 251
- [7] Software Engineering kurz und bündig, 1986
- [8] AmigaDOS 1/90, S. 108 ff.
- [9] Das Amiga Guru-Buch, 1989, S. 466 ff.
- [10] AMIGA, Programmieren in Maschinensprache, 1989
- [11] M68000-Familie, Teil 1, 1985
- [12] Das Amiga Guru-Buch, 1989, S. 379 ff.
- [13] Amiga Intern Band 2, 1989
- [14] AMIGA ROM KERNEL, Libraries & Devices, 1990


```

1:
2: * aus Types.i
3:
4: sizeByte = 1      ; Laengen von Datentypen der CPU
5: sizeWord = 2
6: sizeLong = 4
7:
8: struktur macro
9:   struktZeig set 0
10:  endm
11:
12: byte macro
13:   \1 = struktZeig
14:   struktZeig set struktZeig+sizeByte
15:  endm
16:
17: word macro
18:   \1 = struktZeig
19:   struktZeig set struktZeig+sizeWord
20:  endm
21:
22: long macro
23:   \1 = struktZeig
24:   struktZeig set struktZeig+sizeLong
25:  endm
26:
27: aptr macro
28:   long \1
29:  endm
30:
31: bptr macro ; BPOINTER sind auch Langworte
32:   long \1
33:  endm
34:
35: strukt macro
36:   \1 = struktZeig
37:   struktZeig set struktZeig+\2
38:  endm
39:
40: struktLaenge macro
41:   \1 = struktZeig
42:  endm
43:
44: * aus Dos.i
45:
46: Close = -036 ; (file)(d1)
47: Open = -030 ; (name,accessmode)(d1,d2)
48: Read = -042 ; (file,buffer,length)(d1,d2,d3)
49: Write = -048 ; (file,buffer,length)(d1,d2,d3)
50:
51: modeNewFile = 1006
52:
53: CallDos macro ; vereinfacht Aufruf
54:   move.l a6,-(sp)
55:   move.l dosBase(globals),a6
56:   jsr \1(a6)
57:   move.l (sp)+,a6
58:  endm
59:
60:
61: * aus Exec.i
62:
63: AllocMem = -198 ; (byteSize,requirements) (d0,d1)
64: CloseLibrary = -414 ; (library) (a1)
65: FreeMem = -210 ; (memoryBlock,byteSize) (a1,d0)
66: OpenLibrary = -552 ; (libName, version) (a1,d0)
67:
68: Mem_Clear = $10000
69:
70: execBase = 4
71:
72: CallExec macro ; vereinfacht Aufruf
73:   move.l a6,-(sp) ; aus Exec
74:   move.l execBase,a6
75:   jsr \1(a6) ; Exec-Routine aufrufen
76:   move.l (sp)+,a6
77:  endm
78:
79:
80: Listing 1: Include-Datei zu Listing 2 und 3 als
81: Ersatz fuer die Includes des Aufsteiger-
82: kurses.

```

```

1: * Testlauf0
2:
3: include incl/Mini.i ; Listing 1
4:
5:
6: * Falls Includes aus Aufsteigerkurs vorhanden :
7:
8: * include incl/Types.i ; Makro bptr aus Listing 1
9:   ; hinzufuegen
10: * include incl/Exec.i ; Makro CallExec aendern wie
11:   ; in Listing 1
12: * include incl/Dos.i ; CallDos aendern gem. Listing1
13:
14:
15: * Globals
16:
17: struktur ; siehe Types.i
18:   aptr dosBase
19:   bptr winHdl

```

Listing: Assembler-Kurs-Programme

```

20:   long errorCode
21:   struktLaenge globals_size
22:
23: * Registergebrauch
24:
25: globals equ a5
26:
27:
28: CODE TL0,FAST
29:
30: Start
31:
32: GlobalsAnlegen
33:   moveq #globals_size,d0
34:   move.l #Mem_Clear,d1
35:   CallExec AllocMem
36:   movea.l d0,globals
37:   bne.s OpenDos
38:   moveq #21,d0
39:   bra Ende1
40:
41: OpenDos
42:   lea.l DosName,a1
43:   moveq #33,d0
44:   CallExec OpenLibrary
45:   move.l d0,dosBase(globals)
46:   bne.s OpenWindow
47:   move.l #22,errorCode(globals)
48:   bra Ende2
49:
50: OpenWindow
51:   move.l #WinDef,d1
52:   move.l #modeNewFile,d2
53:   CallDos Open
54:   move.l d0,winHdl(globals)
55:   bne.s TextAusgeben
56:   move.l #23,errorCode(globals)
57:   bra.s Ende3
58:
59: TextAusgeben
60:   move.l winHdl(globals),d1
61:   move.l #TestText,d2
62:   move.l #Dummy,d3 ; fuer Laengenberechnung
63:   sub.l d2,d3 ; Laenge in d3
64:   CallDos Write
65:
66: ZeichenLesen
67:   move.l winHdl(globals),d1
68:   lea.l TextPuffer,a0
69:   move.l a0,d2
70:   moveq #1,d3
71:   CallDos Read
72:
73: WindowSchliessen
74:   move.l winHdl(globals),d1
75:   CallDos Close
76:
77: Ende3
78:   movea.l dosBase(globals),a1
79:   CallExec CloseLibrary
80:
81: Ende2
82:   move.l errorCode(globals),d6
83:
84: GlobalsAufloesen
85:   moveq #globals_size,d0
86:   movea.l globals,a1
87:   CallExec FreeMem
88:   move.l d6,d0 ; errorCode
89:
90: Ende1
91:   rts
92:
93: DosName dc.b 'dos.library',0
94:   even
95:
96: DATA Windowdaten,FAST
97:
98: WinDef dc.b 'NEWCON:10/10/500/100/ Testlauf ',0
99:   even
100: TestText dc.b 'Bitte RETURN druecken',0,10
101: Dummy
102:
103: BSS Puffer,CHIP
104:
105: TextPuffer dc.b 8
106:
107: end
108:
109:
110: Listing 2: Das ausfuehrbare Programm zum Ablaufplan
111:   gemass Bild 3 (mit bewusstem Verzicht auf
112:   Strukturierung umfasst 280 Bytes.

```

```

1: * Testlauf1
2:
3: include incl/Mini.i ; Listing 1
4:
5: * Falls Includes aus Aufsteigerkurs vorhanden :
6: * siehe Listing 2
7:
8:
9: * Konstanten

```

Listing: Assembler-Kurs-Programme



+++ VESALIA TOP ANGEBOTE +++

WINNER II – die neue A2000 SCSI-Filecard

AutoBoot bereits unter Kickstart 1.2 direkt von FastFile

Durch SCSI-II 16-bit-Technik sind Datenübertragungen über 1 MB/Sekunde möglich. Abschaltbar mit durchgeführtem Bus.

Einführungspreise

32 MB Seagate ST 138-N/28 mS. Filecard	1065,-
48 MB Seagate ST 177-N/28 mS. Filecard	1198,-
40 MB Quantum 40S/19 mS. Filecard	1338,-
80 MB Quantum 80S/19 mS. Filecard	1838,-
105 MB Quantum 105S/19 mS. Filecard	2098,-
120 MB Quantum 120S/15 mS. Filecard	2298,-
170 MB Quantum 170S/15 mS. Filecard	2768,-
210 MB Quantum 210S/15 mS. Filecard	2998,-

Winner-Autoboot-Filecard

31 MB Filecard	898,-
47 MB Filecard	1098,-
66 MB Filecard	1298,-

Autoboot-Harddisk

31 MB A500/1000	1098,-
47 MB A 500/1000	1298,-
63 MB A 500/1000	1498,-

Filecard und Harddisk bereits formatiert und mit WB 1.3 installiert. Harddisk-Gehäuse für Amiga 500/1000 als Monitoruntersatz.

512 KB-Ramkarte

für A 500 129,-
abschaltbar, mit Uhr/Akku, Megabittechnik

512 KB-Ramkarte

für Amiga 500 99,-
abschaltbar, Megabittechnik, ohne Uhr.

2 MB-Ramkarte für A 500

498,-
Erweiterung auf 2,3 MB für Amiga 500. Abschaltbar, mit Uhr, alles gesockelt.

8 MB-Ramkarte für A2000

512 KB bestückt	425,-
1,0 MB bestückt	475,-
2,0 MB bestückt	595,-
4,0 MB bestückt	998,-
2 MB-Box für A500/A1000	668,-
4 MB-Box für A500/A1000	1098,-

3.5"-Winner-Drive

169,-
extern mit durchgeführtem Bus und abschaltbar, amigafarbenes Metallgehäuse

3.5" intern für Amiga 500

165,-
komplett mit original Auswurfaste

3.5" intern für Amiga 2000

145,-
Mit Zubehör und Einbauanleitung.

3.5" intern für Amiga 2000

219,-
Als drittes 3,5" Laufwerk, komplett mit Interface und 5,25" Einbaurahmen.

5.25" Laufwerk für alle Amiga

Mit Chinon FZ 506 oder FR 506 199,-
Mit Teak FD 55 GFR 229,-
extern mit durchgeführtem Bus, 40/80 Track schaltbar und abschaltbar.

5.25" intern für Amiga 2000

269,-
Als DF2-Laufwerk mit Interface und Bootselector DF0/DF2. 40/80 Track schaltbar.

Elektr. Bootselector

DF0-DF3 48,-
Es kann von allen Laufwerken gebootet werden, das interne LW ist abschaltbar. Kein Löten erforderlich.

DF2-Interface für Amiga 2000

89,-
Mit Bootselector DF0-DF2, für ein drittes Laufwerk im Amiga 2000 intern.

Winner-Midi A500/2000

89,-

Sounddigitizer

A 500/2000 89,-
Mit Software

ROM-ROM-Umschaltplatine

komplett mit ROM 1.2 99,-
komplett mit ROM 1.3 99,-

Amiga 2000 C 1998,-

mit zwei 3.5" Laufwerken

Citizen Swift 24 998,-

mit Colour Kit

NEC Pinwriter P60 1890,-

NEC Pinwriter P70 2390,-

Colour Kit P60 / P70 198,-

Digi-Splitt Junior 428,-

Der Testsieger. Vollautomatischer RGB-Splitter, jetzt mit S-VHS

Pal-Genlock V 1.3 548,-

Überblenden von einer Bildquelle zur anderen.

Y-C Genlock 1120,-

Baugleich Hama. RGB-Bandbreite 10 MHz. S-VHS Anschluß

OMTI-Adapter f. A 2000 65,-

Filecardblech für 3.5" HD 15,-

OMTI 5528 189,- Kabelsatz 10,-

AutoBoot-Platine 125,-

für Amiga 2000 mit Software und Anleitung

Autoboot-Set f. A 2000 359,-

OMTI 5528, Adapter, Kabelsatz, Autobootmodul und Software

Autoboot-Set f. A 500 359,-

OMTI 5528, Hostadapter mit Bus, Kabelsatz und Software

Amiga 500/1000

Autobootmodul 159,-

zur Nachrüstung von Harddisk

Festplattengehäuse

A 500 68,-

Schaltnetzteil 109,-

A2090A Turbo-Chip-Satz 149,-

doppelte Geschwindigkeit

A2090 alt TurboAutobootkarte 159,-

zusätzlich unter Kick. 1.2

Fast Lightning 29,-

Kopierprogramm für 2-4 Laufwerke

TurboCopy V 2.0 19,-

White Lightning 19,-

Kopierprogramm für 1 Laufwerk

BootBlockGenerator 19,-

Zum Erstellen eines eigenen Vorspanns in den Bootblock, auch mit Sound

De Luxe View 4.1 378,-

Digi View Gold 378,-

Quarterback 4.0 neu 125,-

Festplatten-Backup

3.5" No Name 2DD 10 Stück 11,-

3.5" No Name 2DD 100 Stück 99,-

Halle 10 Stand 310

Auch wir sind auf der

AMIGA
Köln 90

vom 8.-11. November

Vesalia Computer

Industriestraße 25 · 4236 Hamminkeln · Tel. 02852/1068 · Fax 02852/1802 · Mo. – Fr. 9–17 Uhr


```

10:
11: AnyVers = 33 ; Library-Version 1.2 +
12:
13: NoGlobals = 21 ; Fehlermeldungen
14: NoDosLib = 22
15: NoWindow = 23
16:
17: * Globals ; zusammengefasst in einer Struktur, um Code
18: ; reentrant zu halten
19: struktur ; siehe Types.i
20: aptr dosBase
21: bptr winHdl
22: long errorCode
23: struktLaenge globals_size
24:
25: * Registergebrauch
26:
27: globals equ a5
28:
29: * XDEF _main Export fuer spaeteren Test des StartUp-
Code
30:
31: CODE TLO,FAST ; Section
32:
33: _main
34: bra Start ; zum Hauptteil
35:
36:
37: Initialisieren ; Ergebnis:
38: *----- ; Fehlercode in d0
39: GlobalsAnlegen
40: move.l #globals_size,d0 ; Laenge Globals-Struk-
tur
41: move.l #Mem Clear,d1 ; mit null init.
42: CallExec AllocMem
43: move.l d0,globals ; a5 darauf richten
44: bne.s OpenDos ; da Globals angelegt
45: move.l #NoGlobals,d0 ; Fehlercode setzen
46: bra.s Init1 ; Routine verlassen
47:
48: OpenDos
49: lea.l DosName,a1
50: moveq #AnyVers,d0 ; 1.2 +
51: CallExec OpenLibrary
52: move.l d0,dosBase(globals) ; ablegen
53: bne.s OpenWindow ; da Window angelegt
54: moveq #NoDosLib,d0 ; Fehlercode setzen
55: bra.s Init2 ; da Fehler aufgetret.
56:
57: OpenWindow
58: move.l #WinDef,d1
59: move.l #modeNewFile,d2
60: CallDos Open
61: move.l d0,winHdl(globals) ; merken
62: beq.s Init3 ; da Fehler eingetreten.
63:
64: clr.l d0 ; kein Fehler
65: bra.s Init1 ; Routine verlassen
66:
67: Init3
68: move.l #NoWindow,d0 ; Fehlercode setzen
69: Init2
70: move.l d0,errorCode(globals) ; Fehlercode ablegen
71: Init1
72: rts
73:
74:
75: Abschliessen ; Ergebnis:
76: *----- ; d0 Fehlercode
77: WindowSchliessen
78: move.l winHdl(globals),d1
79: beq.s CloseLib ; kein Window da
80: CallDos Close
81:
82: CloseLib
83: tst.l dosBase(globals) ; DosLib open ?
84: beq.s GlobalsKopieren ; nein
85: move.l dosBase(globals),a1
86: CallExec CloseLibrary
87:
88: GlobalsKopieren
89: move.l errorCode(globals),d6 ; temporaer
90:
91: GlobalsAufloesen
92: move.l #globals_size,d0
93: move.l #globals,a1
94: CallExec FreeMem
95:
96: move.l d6,d0 ; Fehlercode
97: Abschl1
98: rts
99:
100: KernAusfuehren
101:
102: *-----
103: TextAusgeben
104: move.l winHdl(globals),d1
105: move.l #TestText,d2
106: move.l #Dummy,d3 ; um Textlaenge zu
107: sub.l d2,d3 ; ermitteln
108: CallDos Write
109:
110: ZeichenLesen
111: move.l winHdl(globals),d1
112: lea.l TextPuffer,a0
113: move.l a0,d2

```

Listing: Assembler-Kurs-Programme

```

114: moveq.l #1,d3
115: CallDos Read ; auf Eingabe warten
116:
117: rts
118:
119:
120: Start ; Hauptteil
121: *-----
122: bsr Initialisieren
123: tst.l d0 ; Fehler aufgetreten ?
124: bne.s St1 ; leider ja
125:
126: bsr KernAusfuehren ; der wesentliche Part
127: bra.s St2
128:
129: St1
130: cmp.w #NoGlobals,d0 ; etwa keine Globals ?
131: beq.s St3 ; Schade, Nix geht !
132:
133: St2
134: bsr Abschliessen
135:
136: St3
137: rts
138:
139: DosName dc.b 'dos.library',0
140: even
141:
142: DATA Windowdaten,FAST
143:
144: WinDef dc.b 'NEWCON:10/10/500/100/ Testlauf ',0
145: even
146: TestText dc.b 'Bitte RETURN druecken',0,10
147: Dummy
148: even
149:
150: BSS Puffer,CHIP
151:
152: TextPuffer dc.b 8 ; eigentlich nur ein Byte benoetigt
153:
154: end
155:
156:
157: Listing 3: Das ausfuehrbare Programm zum Ablaufplan
158: gemaess Bild 4 ist 308 Bytes lang. Die Zu-
159: nahme im Vergleich zu Listing 2 resultiert
160: aus den Bemuehungen um Strukturierung. Der
161: Aufwand scheint gerechtfertigt.

```

```

1:
2: * zu/aus Exec.i
3:
4: FindTask = -294 ; (name) (a1) ++
5: Forbid = -132 ; () ++
6: GetMsg = -372 ; (port) (a0) ++
7: Permit = -138 ; () ++
8: ReplyMsg = -378 ; (message) (a1) ++
9: WaitPort = -384 ; (port) (a0) ++
10:
11:
12: * Process zu Dos.i ++
13:
14: struktur
15: strukt pr Task,92
16: strukt pr MsgPort,34
17: word pr Pad
18: bptr pr SegList
19: long pr StackSize
20: aptr pr GlobVec
21: long pr TaskNum
22: bptr pr StackBase
23: long pr Result2
24: bptr pr CurrentDir
25: bptr pr CIS
26: bptr pr COS
27: aptr pr ConsoleTask
28: aptr pr FileSystemTask
29: bptr pr CLI
30: aptr pr ReturnAddr
31: aptr pr PktWait
32: aptr pr WindowPtr
33: struktLaenge pr_size
34:
35: * CommandLineInterface zu Dos.i ++
36:
37: struktur
38: long cli Result2
39: bptr cli SetName
40: bptr cli CommandDir
41: long cli ReturnCode
42: bptr cli CommandName
43: long cli FailLevel
44: bptr cli Prompt
45: bptr cli StandardInput
46: bptr cli CurrentInput
47: bptr cli CommandFile
48: long cli Interactive
49: long cli Background
50: bptr cli CurrentOutput

```



Listing: Assembler-Kurs-Programme


```

51: long cli_DefaultStack ; in Langworten
52: bptr cli_StandardOutput
53: bptr cli_Module
54: struktLaenge cli_size
55:
56:
57: Listing 4: Ergaenzungen zu Mini.i und zu den
58: Include-Dateien des Aufsteiger-
59: kurses.

1: * StartUp
2:
3: include incl/Mini.i ; Listing 1
4: include incl/Mini1.i ; Listing 4
5:
6: * Konstanten
7:
8: AnyVers = 33 ; beliebige Version ab 1.2
9: AnzArgs = 8 ; auf Commandline des CLI
10:
11: * Registergebrauch
12:
13: cmdLen equir d2
14: cmdBuf equir a2
15: ourProc equir a5
16: dosBase equir a4
17: suArgs equir a3
18: wBMsgPtr equir d6
19: retCode equir d7
20:
21:
22: * SUArgs nur bei CLI-Start benutzt
23: struktur
24: strukt su_argvAry,AnzArgs*4 ; ADR der Argumente
25: strukt su_argvBuf,256 ; die Argumente selbst
26: struktLaenge su_size
27:
28: * -----
29:
30: XREF _main ; Import aus einem anderen Modul
31:
32: * -----
33:
34: * Code beginnt hier
35:
36: ArgumenteRetten ; von Dos/WB uebergeben
37:
38: move.l d0,cmdLen
39: move.l a0,cmdBuf
40:
41: bsr ProcessErmitteln ; Ergebnis ourProc
42:
43: OpenDos
44:
45: lea.l DosName,a1
46: moveq #AnyVers,d0
47: CallExec OpenLibrary
48: movea.l d0,dosBase
49: bne.s CLIOderWB
50: moveq #20,retCode
51: bra RetCodeUebergeben
52:
53: CLIOderWB
54:
55: move.l pr_CLI(ourProc),d0
56: bne.s FromCLI
57: bsr WBStart
58: bra.s ArgumenteUebergeben
59:
60: FromCLI
61: bsr CLISTart
62: move.l suArgs,d1 ; SUArgs angelegt ?
63: bne.s ArgumenteUebergeben ; ja
64: moveq #20,retCode ; Speichermangel
65: bra CloseDos
66:
67: ArgumenteUebergeben
68: ; CLI WB
69: move.l dosBase,-(sp) ; suArgs 0
70: move.l suArgs,-(sp) ; 0 WBMSPtr
71: move.l a1,-(sp) ; 0 WBMSPtr
72: move.l a0,-(sp) ; argv WBMSPtr
73: move.l d0,-(sp) ; argc 0
74:
75: AnwendungAufrufen
76:
77: jsr main ; dem eigentlichen Programm
78: move.l d0,retCode ; von Anwendung uebernehmen
79:
80: StartUpBeenden
81:
82: jsr ProcessErmitteln ; ourProc wieder aktuell
83: move.l pr_ReturnAddr(ourProc),a0
84:
85: * STACK-Aufbau : -----
86: * | ??? |
87: * pr_ReturnAddr --> -----> vor Start
88: * | Ruecksprung-Adr. | unseres
89: * - 4 -----> Prozesses

```

Listing: Assembler-Kurs-Programme

```

90: * | dosBase |
91: * - 8 ----->
92: * | suArgs |
93: * - 12 ----->
94: * | wBMsgPtr |
95: * - 16 ----->
96:
97: suba.w #16,a0 ; Ruecksprungadr,
98: move.l a0,sp ; nun auf argv/WBMsgPtr
99: move.l (sp)+,wBMsgPtr
100: move.l (sp)+,suArgs
101: move.l (sp)+,a1 ; auf Ruecksprung-Adr.
102:
103: CloseDos
104:
105: CallExec CloseLibrary
106:
107: move.l wBMsgPtr,d0 ; WBStartet ?
108: beq.s SUArgsAufgeben ; nein, CLI
109: jsr ReplyWBMSPtr ; ja
110: bra.s RetCodeUebergeben
111:
112: SUArgsAufgeben
113:
114: move.l #su_size,d0
115: move.l suArgs,a1
116: CallExec FreeMem
117:
118: RetCodeUebergeben
119: move.l retCode,d0
120:
121: rts
122:
123: WBStart ; Output : d0 0
124: *----- a0 WBMSPtr
125: ; a1 WBMSPtr
126: ; a3 0
127:
128: lea.l pr_MsgPort(ourProc),a0
129: CallExec WaitPort
130: lea.l pr_MsgPort(ourProc),a0
131: CallExec GetMsg
132: move.l d0,wBMsgPtr
133:
134: ; hier WB-Argumente auswerten
135:
136: movea.l wBMsgPtr,a0
137: movea.l wBMsgPtr,a1
138: clr.l d0 ; immer
139: movea.l d0,a3 ; immer
140: rts
141:
142:
143: ReplyWBMSPtr
144: *-----
145:
146: CallExec Forbid
147: move.l wBMsgPtr,a1
148: CallExec ReplyMsg
149: rts
150:
151:
152: CLISTart ; Input : d2 cmdLen
153: *----- a2 cmdBuf
154: ; Output : d0 argc
155: ; a0 argv
156: ; a1 0
157: ; a3 suArgs
158:
159: SUArgsAnlegen
160:
161: move.l #su_size,d0
162: move.l #Mem_Clear,d1
163: CallExec AllocMem
164: movea.l d0,suArgs
165: beq CLI1
166:
167: ; hier CLI-Argumente auswerten
168:
169: clr.l d0 ; vorlaeufig, bis Argumente
170: suba.l a0,a0 ; ausgewertet werden
171: suba.l a1,a1 ; immer
172:
173: CLI1
174: rts
175:
176: ProcessErmitteln ; Output : ourProc
177: *-----
178: suba.l a1,a1 ; a1 = 0
179: CallExec FindTask
180: move.l d0,ourProc
181: rts
182:
183:
184: DosName dc.b 'dos.library',0
185:
186: end
187:
188:
189: Listing 5 : StartUp-Code (reentrant), der noch
190: nicht uebergebene Argumente/Icons
191: ausgewertet. Das eigentliche Programm
192: wird ueber "jsr _main" aufgerufen.

```

Listing: Assembler-Kurs-Programme



Dieses Programm hat folgende Aufgaben:

1. geschlossene Lautsprechergehäuse zweiter Ordnung auf sehr einfache Weise zu berechnen,
2. dem Anwender zu helfen, für ihn einen Bauvorschlag berechnen zu lassen und diesen mit dem Programm »Graphic-Dump« von der Workbench inklusive Stückliste für die Bestellung der Bretter ausdrucken zu lassen,
3. beliebig viele Daten von Lautsprecherchassis zu speichern, um bei Bedarf zu einem späteren Zeitpunkt berechnen zu lassen.

Nach dem Starten des Programms befinden Sie sich im Hauptmenü. Von hier aus können Sie zwischen »Volumenberechnung«, »Materialberechnung« und »Speicher« wählen.

a) »Volumenberechnung«: Als erstes müssen Sie das Äquivalenzvolumen (»Vas«) in Liter eingeben. Unter Äquivalenzvolumen versteht man das Boxinnenvolumen, das auf die Membran des Treibers die gleichen Rückstellkräfte wirken läßt wie die mechanische Einspannung des Chassis. Nach Eingabe dieses Wertes und Drücken von [Return] fragt der Computer Sie nach dem Ge-

Matthias Knopp

Wie groß wird die Box?

Schließt man einen Verstärker an die Audio-Ausgänge des Amiga an, hat man fast Super-Stereosound. Fast – es fehlen nämlich noch Boxen.

samtgütefaktor des Chassis (»Qts«); er errechnet sich aus der mechanischen Güte (»Qms«) und der elektrischen Güte (»Qes«). Die beiden Werte »Vas« und »Qts« sind in jedem guten Lautsprecherkatalog dem entsprechenden Chassis beigelegt. Nach Drücken von [Return] zeigt der Computer Ihnen das optimale Gehäuse-Nettovolumen auf dem Monitor an. Des weiteren errechnet er die optimalen Gehäuseinnenmaße in Millimeter im Verhältnis 0,6:1:1,6. Diese Dimensionierung der Verhältniszahlen sind so gewählt, daß keine Wand die dreifache Größe einer anderen hat. Nach Drücken von [Return] gelangen Sie wieder zum Hauptmenü zurück.

b) »Materialberechnung«: Als erstes fragt der Computer, ob Sie die zuletzt berechneten Werte oder neue Werte zur Materialberechnung verwenden wollen. Bei Punkt 1 setzt er die zuletzt berechneten Werte automatisch ein. Danach müssen Sie die gewünschte Materialstärke in Millimetern eingeben. Nach Drücken von [Return] zeigt Ihnen der Computer in einer kleinen Grafik, wie Sie die errechneten Gehäuseteile zusammensetzen. Darunter sehen Sie, in welcher Größe Sie die einzelnen Bretter kaufen müssen, damit sie so, wie in der Grafik gezeigt, zusammenpassen. Am besten eignen sich MDF- oder Spanplatten mit ei-

ner Stärke von 19 oder 22 mm. Bei Punkt 2 können Sie die gewünschten Innenmaße der Box selbst eingeben. Danach geht alles wie oben genannt weiter. Bei Punkt 3 kehren Sie zum Hauptmenü zurück.

c) »Speicher«: Bei Aufruf dieses Programnteils öffnen Sie eine Direktzugriffsdatei. Dann fragt Sie der Computer, ob Sie aus dieser Datei lesen oder in diese Datei schreiben wollen. Zum Weiterlesen müssen Sie das Seitensymbol anklicken! Um aus der Datei zu lesen, muß natürlich erst einmal etwas hineingeschrieben werden. Also gehen Sie auf »Schreiben«, geben Satznummer, an welcher Stelle es gespeichert werden soll, ein. Danach geben Sie den Hersteller, die Typenbezeichnung und danach die Werte »Vas« und »Qts« des Chassis ein. Dann gehen Sie zurück zum Speichermenü und wählen »Lesen«. Geben Sie die Satznummer an, die Sie lesen wollen, und schon erscheinen die an dieser Stelle gespeicherten Werte. Diese geben Sie dann im Menü »Volumenberechnung« ein, und schon haben Sie das optimale Gehäusevolumen des gespeicherten Chassis.

(jb)

```

Listings
*****
** Programm: Lautsprecher-Boxen-Berechnung **
** Autor : Matthias Knopp **
** Sprachen: AmigaBASIC **
*****

1: Vorbereitung:
2: PALETTE 0,0,0,0
3: PALETTE 1,1,5,0
4: PALETTE 2,1,0,0
5: PALETTE 3,1,1,0
6: MENU ON
7: ON ERROR GOTO Fehler
8:
9: Hauswahl:
10: CLS
11: COLOR 2,3
12: LOCATE 1,2:PRINT "Hauptauswahl "
13: COLOR 0,2
14: LOCATE 4,2:PRINT "> 1 <"
15: LOCATE 6,2:PRINT "> 2 <"
16: LOCATE 8,2:PRINT "> 3 <"
17: LOCATE 10,2:PRINT "> 4 >"
18: COLOR 2,0
19: LOCATE 4,10:PRINT "Volumenberechnung"
20: LOCATE 6,10:PRINT "Materialberechnung"
21: LOCATE 8,10:PRINT "Speicher"
22: LOCATE 10,10:PRINT "Ende"
23: COLOR 3,0
24: LOCATE 13,2:INPUT "Ihre Wahl: ";Wahl$
25: IF Wahl$<"1" OR Wahl$>"4" THEN BEEP:GOTO Hauswahl
26: IF Wahl$="1" THEN GOTO Volumen
27: IF Wahl$="2" THEN GOTO Material
28: IF Wahl$="3" THEN GOTO Speicher
29: IF Wahl$="4" THEN GOTO Ende
30:
31: Volumen:
32: CLS
33: COLOR 2,3:LOCATE 1,2:PRINT "Volumenberechnung "
34: COLOR 3,0
35: LOCATE 3,10:PRINT "Äquivalenzvolumen (Vas) in liter:"
36: LOCATE 3,50:INPUT Vas
37: LOCATE 5,10:PRINT "Gütefaktor (Qts):"
38: LOCATE 5,50:INPUT Qts
39: CLS
Listing: Boxenberechnung.BAS

```



```

40: COLOR 2,3:LOCATE 1,2:PRINT "Volumenberechnung "
41: COLOR 3,0
42: V=Vas/1000
43: Qtc=.707
44: Vbox=V/(((Qtc/Qts)^2)-1)
45: L=Vbox*1000:X=L:Q=X^(1/3)
46: B=Q*10:H=1.6*Q*10:T=.6*Q*10
47: Br=B*10:Ho=H*10:Ti=T*10
48: LOCATE 4,15:PRINT "Benötigtes Nettovolumen"
49: LOCATE 7,15:PRINT "Liter"
50: LOCATE 12,15:PRINT "Optimale Gehäusemasse (innen)"
51: LOCATE 15,15:PRINT "Breite: ";Br;" mm"
52: LOCATE 17,15:PRINT "Höhe : ";Ho;" mm"
53: LOCATE 19,15:PRINT "Tiefe : ";Ti;" mm"
54: LOCATE 22,15:PRINT "Bitte beachten Sie die Abmessungen des Chassis!"
55: LOCATE 22,64:INPUT "RETURN";x
56: CLS:GOTO Hauswahl
57:
58: Material:
59: CLS
60: COLOR 2,3:LOCATE 1,2:PRINT "Materialberechnung"
61: COLOR 3,0
62: LOCATE 3,3:PRINT "> 1 < berechnete Werte"
63: LOCATE 5,3:PRINT "> 2 < neue Werte"
64: LOCATE 7,3:PRINT "> 3 < zurueck"
65: LOCATE 10,3:INPUT "Ihre Wahl: ";Wert$
66: IF Wert$<"1" OR Wahl$>"3" THEN GOTO Material
67: IF Wert$="1" THEN GOTO Almat
68: IF Wert$="2" THEN GOTO Neumat
69: IF Wert$="3" THEN GOTO Hauswahl
70:
71: Almat:
72: CLS
73: LOCATE 5,25:PRINT "Materialstaerke in mm: "
74: LOCATE 5,50:INPUT M
75: CLS
76: PALETTE 0,1,1,1
77: PALETTE 1,0,0,0
78: PALETTE 2,0,0,0
79: PALETTE 3,0,0,0
80: LINE(198,28)-(302,102),B
81: LINE(198,28)-(240,10)
82: LINE(302,28)-(344,10)
83: LINE(302,102)-(344,84)
84: LINE(240,10)-(344,10)
85: LINE(344,10)-(344,84)
86: LINE(208,26)-(242,12)
Listing: Boxenberechnung.BAS

```

Fortsetzung auf Seite 71

PD-WORKSHOP

Im Trend

Die Programmplut in der PD nimmt kein Ende. Fast täglich erreichen mich Anrufe der verschiedensten PD-Händler, daß in Kürze mit einer neuen Diskettensammlung zu rechnen ist. Kurzum, Programmvielfalt ist für die PD-Freunde angesagt.

Dazu passend finden Sie an dieser Stelle ab dieser Ausgabe unseren PD-WORKSHOP zum Heraustrennen und Sammeln. Wir präsentieren Ihnen Top-Programme in ausführlicher Form. Damit schlagen wir zwei Fliegen mit einer Klappe. Zum einen erhalten Sie, liebe Leser, eine Referenz zum Sammeln, zum anderen nehmen wir den "schwarzen Schafen" unter den PD-Vertreibern den Wind aus den Segeln, die deutsche Anleitungen zu den verschiedensten PD-Programmen für nicht weniger als 10,- DM vertreiben.

Zum Sammeln des Workshops stehen Ihnen unsere bewährten Sammelordner zur Verfügung.

Falls Sie weitere Anregungen oder Verbesserungsvorschläge haben, können Sie sie gerne an mich richten. Viel Spaß bei der Lektüre wünscht Ihnen

Vera Brinkmann

Vera Brinkmann
Redakteurin AMIGA DOS



Bezugsquellen

Die hier vorgestellten Programme wurden uns freundlicherweise von den Firmen

- A.P.S. Electronic
Sonnenborstel 31
3071 Steimbke
Tel.: 05026/1700

und

- Herrmanns & Kommelter
Vom-Bruck-Platz 45
4150 Krefeld 1
Tel.: 02151/399833

zur Verfügung gestellt. Diese beiden Firmen haben die entsprechenden Disketten vorrätig. Weitere Vertreiber können Sie der Anbieterliste in der PD-Werkzeugkiste entnehmen.

Inhalt

SID-Workshop

Rund ums Programm	S. 64
Nachrichtenzeile	S. 64
Device-Schalter	S. 64
Screens und Farbe	S. 65
Anzeigemodi	S. 65
Menüs	S. 65
File-Menü	S. 66
Flag-Menü	S. 66
Kommandoschalter	S. 66
Schalter über Schalter	S. 67
Fast geschafft	S. 67
Konfigurations-File	S. 68
Selector-Workshop	S. 70

Themen

Der PD-WORKSHOP informiert Sie diesmal ausführlich zu den Programmen

»SID« – ein Utility der Spitzenklasse von Timm Martin. Hangeln Sie sich mit dieser Benutzeroberfläche durch die verschwiegsten Verzeichnisse Ihres Amiga. Lernen Sie dieses Programm von A bis Z kennen.

»Selector« – ein nützliches Hilfsmittel von Nico Francois. Keine Chance für den umständlichen Programmstart.

Literatur

Hinweise zu PD-Programmen finden Sie noch in folgenden Büchern

- | | |
|---|---|
| <p>[1] Ram S., Hartwig J.: Das große Amiga Public Domain Buch; technicSupport 1988, ISBN 3-926847-01-8.</p> <p>[2] Leithaus R., Hertwig J.: Das zweite Amiga Public Domain Buch; technicSupport 1988, ISBN 3-926847-05-0.</p> <p>[3] Leithaus R., Hertwig J.: Das dritte Amiga Public Domain Buch; technicSupport 1989, ISBN 3-926847-06-9.</p> | <p>[4] Leithaus R., Hertwig J.: Das goldene Amiga Public Domain Buch; technicSupport 1989, ISBN 3-926847-08-5.</p> <p>[5] Laub J., Wenzl J.: Amiga Public Domain Dokumentation; Markt&Technik 1989, ISBN 3-89090-242-1.</p> <p>[6] Röhrich P., Sanio G., Torns-dorf M.: Die besten Public Domain & Shareware Programme; Data Becker 1990, ISBN 3-89011-368-0.</p> |
|---|---|

Mit diesem Utility kann man alle Arbeiten, die man sonst mit dem CLI erledigt, bequem mit der Maus steuern (ähnlich wie bei Diskmaster und CLI-Mate). »SID« bietet zusätzlich die Möglichkeit, die Konfiguration nach den Wünschen des Anwenders zu gestalten.

»SID« wird seit Oktober 1988 von Timm Martin entwickelt und ständig verbessert. Das Programm ist Shareware, und der Autor bittet um Einsendung von 25\$. Durch die Einsendung der Share-Gebühr (inklusive Name, Adresse und Versionsnummer) werden Sie registrierter Benutzer. Die Gegenleistung: Ihnen werden die jeweils neueste Version von »SID«, alle Unterstützungsprogramme (wie der geplante interaktive Konfigurationseditor), die neueste Version der Anleitung mit Updates und eine Auswahl von PD-Programmen von ihm und seinen Freunden zugesandt. Sie werden über neu geplante Funktionen und Änderungen frühzeitig informiert. Die Upgrades erhalten Sie wesentlich früher (zirka einen Monat) als Sie diese in irgendeiner PD-Serie finden.

Rund ums Programm

»SID« wird entweder über das Icon der Workbench oder über den CLI aufgerufen. Findet der Aufruf über den CLI statt, können Sie folgendermaßen Parameter übergeben:

SID [Dir1] [Dir2]

»Dir1« und »Dir2« werden dann in die Directory-Liste eingeladen. Die Konfigurationsdatei (Name: »SID.config«) sollte im aktuellen oder im Verzeichnis »s:« stehen. Diese Datei wird nach dem Start geladen und ausgeführt. Damit ist es möglich, fast alle Einstellungen von »SID« zu ändern und Ihren eigenen Wünschen anzupassen.

Nach dem Start öffnet sich ein Fenster auf der Workbench. Hier können Sie alle Funktionen aufrufen. Die System-Gadgets des Fensters reagieren wie gewohnt. Das Größen-Gadget ist nicht benutzbar, dazu jedoch später mehr. Die Farben entsprechen den Standardeinstellungen der Workbench:

Robert Münch

SID-Workshop

In der Public Domain hat sich im Bereich der Utilities ein Programm hervor getan: »SID«.

Farbe 0 = Blau
Farbe 1 = Weiß
Farbe 2 = Schwarz
Farbe 3 = Orange

Ferner sehen Sie zwei große Bereiche im Fenster, hier werden die Directories und File-Namen angezeigt. Die Art der Anzeige der Daten kann im Config-File geändert werden. Das dafür zuständige Flag heißt »SeparateDirs«. Die beiden kleinen Fenster über der Directory-Liste enthalten den Pfad zum momentan angezeigten Directory. Die aktiven Funktionen erscheinen in Orange. Die ausgewählten Kommandos beziehen sich immer auf die aktuelle Liste. Klicken Sie nun einmal auf »DF0:« unter der linken Directory-Liste. Damit wird das Directory der Diskette in »DF0:« gelesen und angezeigt. Um zwischen der aktiven und der inaktiven Liste zu tauschen, gibt es verschiedene Möglichkeiten:

Einfach mit der linken oder rechten Maustaste in die inaktive Liste klicken. Sie können aber auch in die in der Mitte befindlichen Schieberegler oder in die Pfeile klicken. Möchten Sie einen Eintrag auswählen, klicken Sie einfach

mit der linken Maustaste auf den Eintrag. Dieser wird dann invers dargestellt; um ihn zu deselektieren, einfach noch einmal anklicken. Bei mehreren Einträgen empfiehlt sich, die linke Maustaste festzuhalten und einfach hoch und runter zu fahren. Alle dabei überfahrenen Einträge werden nun selektiert oder deselektiert (je nachdem, wie ihr Zustand vorher war). Der übliche »Parent-knopf« (um in das übergeordnete Verzeichnis zu gelangen) wurde bei »SID« in der rechten beziehungsweise linken Seite der Directory-Liste versteckt. Um nun in der linken Directory-Liste in das übergeordnete Verzeichnis zu gelangen, klicken Sie an den linken Rahmen der Liste, und in der rechten Liste genau umgekehrt.

Der Pfad kann natürlich auch per Mausklick und »zu Fuß« eingegeben werden. Dazu klicken Sie in den Pfadnamen über der Directory-Liste und tippen den Pfad ein (mit [Return] abschließen). Rechts und links neben den Kommando-Gadgets unterhalb der Directory-Liste verstecken sich zwei Schalter, mit denen verlorengegangene Einträge wieder zurückgeholt werden können

(Undo-Funktion). Diese werden genauso gehandhabt wie der Parent-Schalter. Eine weitere Möglichkeit der Pfadeingabe besteht über Tastenkombinationen. Mit [Cursor-links] kann in der linken Pfadangabe und mit [Cursor-rechts] in der rechten ein Eintrag gemacht werden. Mit [Esc] verläßt man dieses Menü. [Return] ohne Pfadangabe bewirkt das Einlesen des aktuellen Verzeichnisses.

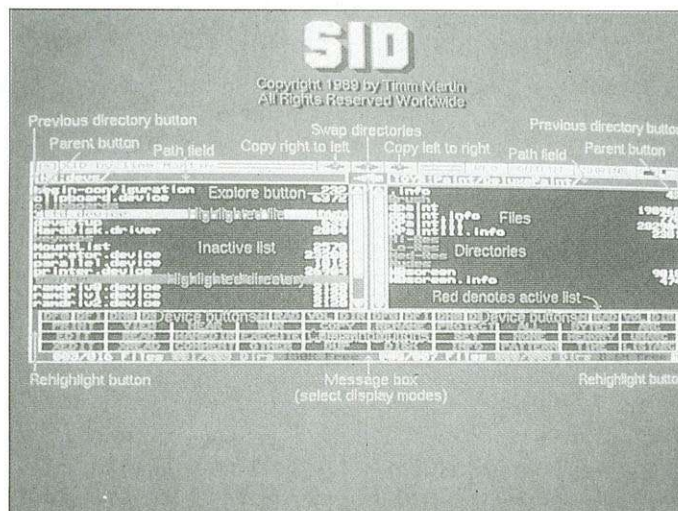
Nachrichtenzeile

Die Zahlenkolonnen in der Nachrichtenzeile beziehen sich alle auf die aktuelle Directory-Liste.

Ein Wechsel in ein Unterverzeichnis kann per Doppelklick auf dem entsprechenden Verzeichniseintrag durchgeführt werden. Um ein Überschreiben der Verzeichnisse beim Wechsel zu vermeiden, stellt »SID« einen sogenannten Suchschalter zur Verfügung (der Pfeil in der Mitte der beiden Pfadeingabefelder). Per Mausklick wird der Durchsuchmodus aktiviert. Ein weiterer Klick deaktiviert diesen Modus. Mit den Pfeilen über dem Suchschalter kann man die Inhalte der Liste vertauschen beziehungsweise kopieren. Der Doppelpfeil vertauscht die beiden Listen, alle angewählten Files und Verzeichnisse bleiben erhalten. Die einzelnen Pfeile zeigen immer in die Datenflußrichtung, das heißt, der Pfeil nach links kopiert die rechte Verzeichnisliste in die linke und umgekehrt.

Device-Schalter

Unter jeder Verzeichnisliste befinden sich zehn Device-Schalter. Acht davon sind auch als solche benannt (zum Beispiel: »DF0:«, »RAM:« und so weiter) und zeigen den Inhalt des entsprechenden Device an. Der Dir- und der Vol-Schalter haben spezielle Funktionen. Der Vol-Schalter zeigt Ihnen eine Liste aller gemounteten Geräte. Das sind nicht nur physikalische Geräte (Diskettenlaufwerke und Festplatten), sondern auch logische (mit dem Kommando »Assign« erstellte) Geräte. Mit dem Dir-Schalter wird das in dem Pfadeingabefeld angezeigte Ver-



Informationen zum Programm

zeichnis noch einmal gelesen. Diese Funktion ist notwendig, da »SID« das Verzeichnis nach einer Aktion nicht wieder neu einlädt. Der Stop-Requester beendet die gerade laufende Aktion und fährt im Programm fort.

Screens und Farbe

Wem die Window-Größe von »SID« nicht zusagt, der sollte im Menü »Environment« unter dem Punkt »Window« nachsehen. Hier können Sie die Größe variieren. Sie wählen zwischen drei Standardwerten. Mit »Specify Size« erstellen Sie eigene Definitionen. In dem erscheinenden Requester ist die Zeilenanzahl veränderbar. Sie sollten dabei aber Ihr Chip-RAM nicht vergessen, denn zusätzliche Screens verbrauchen viel Speicher. Der Schalter »Shrink« in der rechten oberen Ecke macht aus »SID« ein Workbench-Fenster mit der Höhe der Titelzeile. Durch Anklicken des Fensters »Expand« versetzen Sie die Anzeige in den Urzustand. Bei aktivem Fenster genügt auch ein Druck auf die Taste [E].

Mit der Option »Screen« im Menü »Environment« kann man auch eigene Fenster erstellen. Hier stehen Ihnen drei Punkte zur Verfügung:

- »Workbench« ist voreingestellt und benötigt am wenigsten Speicher.
 - »Custom« erzeugt einen eigenen hochauflösenden Screen (640 x 256 Pixel). Dieses Fenster läßt sich nicht bewegen, Sie müssen es mit der Ziehleiste sowie mit den Depths-Schaltern bewegen.
 - »Interlace« baut einen Screen mit einer Auflösung von 640 x 512 Pixel auf.
- Im selbsterstellten Screen können Sie die Farben ändern. Im Menü »Environment« ist der Menüpunkt »Colors« anzuwählen. Dieser Requester wird wie der Color-Requester des Programms »Preferences« von der Workbench bedient.

Anzeigemodi

Hinter den Einträgen in der Verzeichnisliste können verschiedene Daten erscheinen. »SID« kennt sechs verschiedene Anzeigemodi.

Bit Funktion Auswirkung

H	Hidden	File wird nicht angezeigt
S	Script	Batchdatei kann unter Kickstart1.3 wie ein Programm aufgerufen werden (ohne EXECUTE).
P	Pure	Programm kann resident gemacht werden
A	Archived	File wurde nicht geaendert seit dem letzten Backup
R	Read	File kann gelesen werden
W	Write	File kann beschrieben werden
E	Execute	File ist ausfuehrbar
D	Delete	File kann geloescht werden

Abb. 1. Unterschiedliche Protectionbits haben unterschiedliche Auswirkungen bei gesetztem Bit

- Größe des Files in Byte »Size«
- Datum »Date«
- Zeit »Time«
- Bit-Konstellation »Bits«
- Kommentar »Note«
- Beschreibung »Desc«

Um einen dieser Modi auszuwählen, klicken Sie mit der linken Maustaste in die Nachrichtenzeile. Daraufhin erscheinen die oben erwähnten Modi und können per Mausklick (links) angewählt werden. Normalerweise wird das Directory beim Moduswechsel nicht noch einmal neu eingelesen, die Daten werden sofort angezeigt. Eine Ausnahme ist der Modus »Desc«. Hier wird das Directory nochmals eingelesen, da angezeigt wird, um welches File es sich handelt (ASCII, Binary und so weiter). Der Doppelpfeil am Ende einiger Einträge zeigt Ihnen an, daß er länger ist und überdeckt wird.

Folgende File-Arten erkennt das Programm:

- Amiga-Basic-Programm
- ARC-File
- Logisches Device (Assignment)
- Directory
- Ausführbares Programm (Executable Object)
- Font-Header
- IFF-Grafik
- IFF-Sound
- IFF-Musik
- IFF-Text
- .info Icon
- Lattice Objekt
- LHArc-File
- Manx Objekt
- Multiplan Spreadsheet
- PageStream Dokument
- PAK-File
- ProPage Dokument
- Superbase Datenbankdefinition (Superbase File Def)
- Superbase Abfrage (Superbase Query)
- Volume (Diskettenname von

- DF0: usw.)
- WARP-File
- WordPerfect Text
- ZOO-File

Files, die sich nicht zuordnen lassen, bekommen entweder eine Bezeichnung »ASCII file« oder »Binary file«. »SID« bietet eine Funktion, die per Doppelklick auf einen Eintrag ausgelöst wird und je nach File-Art eine andere Auswirkung hat. Bei einem ASCII-File wird beim Doppelklick mittels der Funktion »Read« der Inhalt des Files angezeigt. Da »Pak« und »Warp« keine List-Funktion besitzen, wird bei einem solchen File die Funktion »Info« ausgeführt.

Manchmal werden Sie mit dem Input-Requester zu einer Eingabe aufgefordert. Dieser Requester hat entweder zwei oder vier Schalter. »Accept« erscheint ganz links im Requester und hat die Funktion, Ihr Kommando zu akzeptieren und auszuführen. Der Schalter »Stop!« erklärt sich von selbst. Aktivierte Kommandos können mit »Previous« auf ein zuvor ausgewähltes File angewendet werden. »Next« schaltet zum nächsten File weiter.

Die entsprechenden Tastatur-Äquivalente:

- »Accept« = [Return]
- »Stop« = [Esc]
- »Previous« = [Ctrl] Cursor Links
- »Next« = [Ctrl] Cursor Rechts

Menüs

Beginnen wir beim Programm-Menü mit vier verschiedenen Menüpunkten. Da Sie die Möglichkeit haben, in »SID« fast alle Funktionen und Voreinstellungen zu verändern, lädt der Menüpunkt »Configuration« Ihr Config-File (»Load«); zusätzlich können Sie es edi-

tieren (»Edit«) und damit Ihren Wünschen angleichen. Falls Sie »Edit« angeklickt haben, wird der von Ihnen bevorzugte oder der voreingestellte Text-editor gewählt, um das Config-File zu editieren. »Information« liefert Ihnen die Version des Programms sowie die Adresse des Programmierers. »Last Error« zeigt den zuletzt aufgetretenen Fehler in der Nachrichtenzeile auf. Als Tastatur-Äquivalente können Sie »?« oder »/« benutzen. Mit »Last Message« wird die letzte Meldung des Programms noch einmal angezeigt. In der jetzigen Version funktioniert dies nur bei der Funktion »Byte«.

Im Systemmenü können Sie drei Programme angeben, die per Menüauswahl ausgeführt werden. Nach Beendigung befinden Sie sich wieder im »SID«. Die Programme sind im Config-File einzutragen.

Über »Calculator« wird der von Ihnen eingestellte Taschenrechner nachgeladen. Mit »Command« ist es möglich, Amiga-DOS-Kommandos auszuführen. Nach Anwahl dieses Punktes erscheint ein Requester, in den Sie Ihre Eingaben machen können. Bei mehreren Kommandos sind diese durch [Ctrl-N] zu trennen, dies ist die Sequenz für eine neue Zeile (Newline) im CLI. »SID« öffnet bei der Ausführung des/der Kommandos ein entsprechendes Ausgabefenster. Dieses arbeitet genauso wie der CLI und muß auch mit »Endcli« verlassen werden. Der Menüpunkt »Newcli« macht genau das gleiche wie der äquivalente CLI-Befehl. »Preferences« ruft das gleichnamige Programm von der Workbench auf.

Das Diskmenü verfügt nur über zwei Optionen. »Fit« gibt eine Meldung aus, ob die momentan angewählten Files der aktiven Liste in dem Directory der inaktiven Liste genug Platz haben, um dorthin kopiert zu werden. Dabei wird sogar die Diskettenblockgröße berücksichtigt. Damit ist gemeint, daß eine Diskette 488 Byte Daten pro Block speichern kann, bei einer Festplatte sind es 512 Byte Daten pro Block. Mit »Relabel« können Sie der Diskette einen neuen Namen geben.

Die Funktion »Copy As« fragt nach dem Namen, unter dem das ausgewählte File gespeichert werden soll. Ein bereits

Tastatur	Kommando
[Ctrl-A]	ALL
[Ctrl-O]	COMMENT
[Ctrl-D]	DISK
[Ctrl-U]	DUP
[Ctrl-E]	EDIT
E (nur wenn SID verkleinert wurde)	EXPAND
[Ctrl-I]	INFO
[Ctrl-M]	MAKEDIR
[Ctrl-N]	NONE
[Ctrl-P]	PATTERN
[Ctrl-R]	RENAME
[Ctrl-S]	SHRINK
Tastatur	Menue
[Ctrl-!] oder [Ctrl-1] [Ctrl-Paragraph] oder [Ctrl-2]	COMMAND (System) COPY AS (File)
[Ctrl-C]	CREATE (File)
[Ctrl-L]	RELABEL (Disk)
[Ctrl-Y]	SELECT BY DATE (File)
Tastatur	Erlaeuterung
[Cursor links] (Cl)	aktiviert linkes Pfadeingabefeld
[Cursor rechts] (Cr)	aktiviert rechtes Pfadeingabefeld
[Shift-Cl]	aktiviert linke Verzeichnisliste
[Shift-Cr]	aktiviert rechte Verzeichnisliste
[Ctrl-Cursor hoch]	laedt vorheriges Verzeichnis der aktiven Liste
[Ctrl-Cl]	waehlt selektierte Files der linken Liste wieder an
[Ctrl-Cr]	waehlt selektierte Files der rechten Liste wieder an

Abb. 2. Übersicht über Tastenbelegung

bestehender Name wird ohne Nachfrage überschrieben. Dieser Mißstand kann mit dem Setzen des Flags »Only New« im gleichnamigen Menü behoben werden. »Create« erstellt ein neues File zum Editieren.

Das File-Menü

Falls Sie einmal alle Programme markieren möchten, die zwischen zwei bestimmten Daten (keine Computerdaten) liegen, so benutzen Sie am besten die Funktion »Select By Date«. Im Requester werden zwei Daten durch Leerzeichen getrennt eingegeben (Format: Tag:Monat:Jahr; dies gilt nur, wenn das Flag »DateMDY« im Config-File gleich NO ist!), und dann markiert »SID« alle Files, die in diesem Zeitraum liegen. Wollen Sie alle Programme bis zu einem Datum markiert wissen, stellen Sie dem Datum einfach ein Minuszeichen voran. Sollen alle Programme nach einem bestimmten Da-

tum markiert werden, stellen Sie das Minuszeichen hinter das Datum. In diesem Menü kann die Konfiguration des Programms geändert werden. Da »SID« die Möglichkeit hat, Files und ganze Disketten zu archivieren, stehen mit »Archiv Method« fünf Archivierungsmethoden zur Verfügung: »Arc«, »Lharc«, »Pak«, »Warp« und »Zoo«.

Flag-Menü

Das Flag »Byte Count« gibt an, welche Bytes bei der gleichnamigen Funktion angezeigt werden. Mit »Actual« wird die aktuelle Anzahl von Bytes der angewählten Files und Verzeichnisse angezeigt. »Occupied« liefert die Anzahl der wirklich belegten Bytes auf der Diskette. Bei dieser Anzeigemethode werden zusätzlich noch die Header- und Directory-Bytes angezeigt. Das Flag »Copy« kann entweder auf »Overwrite« oder auf »Only New« gesetzt

werden. Beim ersten werden vorhandene Files im Zielverzeichnis einfach überschrieben. Wählen Sie jedoch »Only New« werden nur die Files kopiert, die nicht im Zielverzeichnis zu finden sind. Mit »Delete« wird ein Sicherheitsabfrage-Requester eingeschaltet, der immer beim Löschen von Files erscheint. Entsprechend gilt das Flag »Directory Delete« für Verzeichnisse. Ist das Flag »Error Stop« gesetzt, erscheint bei einem Fehler ein Error-Requester und macht Sie auf den Fehler aufmerksam.

Die Option »Diskette« ermöglicht das sofortige Laden eines Directorys nach Einlegen der Diskette. Mit dem Flag »Autoload« wird das Directory der Diskette immer in die aktuelle Liste geladen. »Load/Switch« bewirkt das Einladen in die aktive Liste und den Wechsel der inaktiven zur aktiven Liste. Damit werden die Verzeichnisse abwechselnd, einmal links, einmal rechts, geladen. Zu gu-

ter Letzt haben wir noch »Ignore«; es wird kein Directory geladen.

Die Funktion »Hidden Files« nutzt das von AmigaDOS nicht unterstützte Hidden-Flag aus. Wenn Sie also dieses Flag bei einem File setzen, wird es von »SID« nicht mehr angezeigt. Nützlich ist diese Funktion, um Dateien, die man nicht in der Verzeichnisliste sehen möchte, auszusortieren.

Mit »Path« wird der absolute Pfad des Directorys angezeigt. Bei »Run« können Sie entscheiden, ob ein mit »Run« gestartetes Programm im Vordergrund oder im Hintergrund laufen soll. Der Vordergrundaufruf ist bei den Programmen nützlich, die eine Kommandozeileingabe benötigen.

Kommando-schalter

»SID« bietet Ihnen 30 verschiedene Kommandoschalter. Die meisten beziehen sich auf die ausgewählten Verzeichnisse und Files. Diese werden mit den dafür vorgesehenen Funktionen ausgeführt. Auftretende Fehler werden in der Nachrichtenzeile ausgegeben.

■ Systemkommandos

Diese Schalter wirken weder auf ein Verzeichnis noch auf ein File. Mit »Memory« wird der verfügbare Speicherplatz, unterteilt in Fast- und Chip-RAM, angezeigt. »Time«, soweit vorhanden, sorgt dafür, daß die aktuelle Zeit mit Datum angezeigt wird.

■ List-Kommandos

Diese Schalter wirken auf angewählte Verzeichnisse und Files. »All« wählt alle Verzeichnisse und Files an. Mit »None« werden alle angewählten Verzeichnisse und Files demarkiert. »Pattern« erlaubt die Selektion der Verzeichnisse und Files, die dem Auswahlmuster entsprechen. Es werden die Amiga-DOS-Pattern »#« und »?« unterstützt.

»()« überprüft, ob die Zeichenkette oder Teile der Zeichenkette, die zwischen den Klammern stehen, in einem Eintrag vorkommen. Hierzu ein Beispiel: Die Angabe »ROB# (BERT)« findet ROB, ROBERT und ROBERTBERT.

»!« arbeitet wie die logische Funktion OR, die aus den mei-

sten Programmiersprachen bekannt ist. Somit findet »#?« (doc|asm)« alle Namen, die auf ».doc« oder ».asm« enden.

»%« – Diese Funktion findet jeden Buchstaben oder keinen. Ein Beispiel:

»AmigaD(SI%)« findet AmigaDOS oder AmigaDOSMagazin und so weiter. Hier können Sie sehen, daß verschiedene Separatoren auch kombinierbar sind.

»'« – Der Apostroph dient der Suche nach einem Platzhalterzeichen. Auch hierzu ein Beispiel:

»Bild('?)« findet die Namen Bild(1), Bild(2), Bild(a)...

»SID« stellt für Leute aus der MS-DOS-Welt auch deren heißgeliebtes Sternchen (*) zur Verfügung. Dieses Sternchen steht für jeden Buchstaben sowie jede Buchstabenkette. Einträge, die dem Suchmuster nicht entsprechen und schon angewählt waren, werden nicht demarkiert.

■ Directory-Kommandos

»Disk« gibt Informationen über die Diskette oder das Device der aktuellen Liste aus. Wenn Sie »Disk« mit der rechten Maustaste anklicken, werden die Informationen für die inaktive Liste ausgegeben. Mit dem Drücken der linken Maustaste oder irgendeiner Taste auf der Tastatur schließen Sie das Fenster.

»Makedir« erstellt ein neues Directory in der aktiven Liste. Sie müssen den Namen des neuen Verzeichnisses eingeben, dabei darf dieser nicht bereits in diesem Verzeichnis vorhanden sein. Dieses neues Verzeichnis kann sofort eingelesen werden. Die entsprechende Funktion ist nur über Tastatur verfügbar. [Ctrl-<] oder [Ctrl-] liest das soeben erstellte Verzeichnis in die linke Liste, [Ctrl->] oder [Ctrl-] in die rechte Liste ein. Wer zusätzlich ein Icon zu diesem Directory haben möchte, muß nur im Konfigurations-File das Flag »MakeDirIcon« setzen. Bei diesem Flag muß der Name mit Pfad einer Icon-Datei angegeben werden. Am besten kopieren Sie das File »Empty.info« von der Workbench in das Verzeichnis »s:« und nennen es »SID.info«. So haben Sie alle Daten beieinander. Der Name Ihres neuen Directorys darf 25 Zeichen nicht überschreiten,

da die Endung ".info" noch angehängt werden muß.

Die folgenden Schalter agieren immer nur mit einem angewählten Eintrag. Dies bedeutet jedoch nicht, daß Sie nicht mehrere Einträge gleichzeitig anwählen können.

Schalter über Schalter

Mit »Comment« können Sie einen File-Kommentar anlegen oder einen bereits existierenden Kommentar editieren. »SID« kopiert übriges File-Kommentare mit.

Der Schalter »Delete« arbeitet wie der gleichnamige CLI-Befehl. Ein Requester erscheint und fragt nach, ob Sie sich auch wirklich sicher sind. Wie weiter oben beschrieben, können Sie das Erscheinen dieses Requesters ein- und ausschalten (im Flag-Menü).

Mit »Dup« werden alle angewählten Einträge im gleichen Verzeichnis dupliziert. Im Requester müssen Sie den neuen Namen des duplizierten Files eingeben, denn zwei Files dürfen nicht den gleichen Namen in einem Directory haben. Directories können in der momentanen Version (V1.06) noch nicht dupliziert werden. Ausgewählte Files werden mit »Edit« mit Ihrem Texteditor, den Sie im Konfigurations-File festgelegt haben, eingelesen und von dort editiert. Die Größe des Files wird übrigens sofort auf den neuesten Stand gebracht, um Änderungen zu verdeutlichen.

»Execute« startet die gewählten Files mit dem Kommando »Execute« des CLI. Sie müssen nur sicherstellen, daß die Files, die ausgeführt werden sollen, auch Batch- oder Scriptfiles sind.

Digitalisierter Sound kann mittels »Hear« mit dem im Konfig-File eingetragenen Soundplayer abgespielt werden.

»Info« gleicht fast dem Schalter »Disk«, er gibt jedoch Informationen über die angewählten Files und Verzeichnisse aus. Mit »Prev« erhalten Sie die Daten des vorherigen Files und mit »Next« die des folgenden. Mit »Stop!« kommen Sie aus dem Requester heraus.

Der Inhalt eines archivierten Files wird mit »Listarc« angezeigt. »SID« erkennt dabei eigenständig, welche Archivie-

rungsmethode benutzt wurde. Die Angaben, wo und wie die Archivierungsprogramme stehen beziehungsweise den Inhalt ausgeben, werden im Konfigurations-File angegeben. Die Archivierungsmethoden von »Pak« und »Warp« lassen ein Anzeigen des Inhalts im Moment noch nicht zu. Kann »SID« die Methode nicht feststellen, erscheint ein Requester, in dem Sie per Hand das richtige Programm angeben können.

»Other« ruft ein Programm Ihrer Wahl auf (Dies wird auch im Konfig-File angegeben).

Die Funktion »Protect« benötigen Sie, falls Sie die sogenannten Protectionbits (das sind solche, die in der Liste als »--RWED« auftauchen) ändern möchten. Sie können gleich mehrere Files auswählen und dann in gewohnter Weise mit einem Requester die Bits für jedes File einzeln ändern. AmigaDOS kennt acht sogenannte Protectionbits, die in Tabelle 1 näher erläutert werden. Der entsprechende Requester besitzt acht Schalter, wobei jeder für ein Bit steht. Wollen Sie eines dieser Bits ändern, so klicken Sie auf den entsprechenden Schalter. Im unteren Teil des Requesters werden die alte sowie die neue Protection-Maske angezeigt (»Old: --RWED« »NEW: S--RWED«). Möchten Sie Ihre alte Maske wieder haben, brauchen Sie nur auf das Wort »Old« zu klicken; so müssen Sie nicht jedes Bit wieder einzeln zurücksetzen. In diesem Requester gibt es noch vier weitere Schalter. Die drei »Previous«, »Next« und »Stop!« müßten Ihnen bekannt sein. Der Schalter »Protect« ändert die Bitanordnung für das entsprechende File.

Mit »Read« können Sie sich Text-Files anzeigen lassen. »SID« lädt die Files in den eigenen Textanzeiger ein. Binäre Zeichen werden als Leerzeichen ausgegeben. Den Text

können Sie mit Ziehbalken und Pfeilen an der Seite weiter durchscrollen. Daneben können Sie auch Maus oder Tastatur benutzen. »Previous«, »Next« und »Stop!« funktionieren wie gewohnt. Die entsprechenden Tastaturäquivalente:

[Cursor hoch] – vorherige Zeile
[Cursor runter] – nächste Zeile
[Shift Cursor hoch] – vorherige Seite
[Shift Cursor runter] – nächste Seite
[Alt Cursor hoch] – Textanfäng
[Alt Cursor runter] – Textende

Fast geschafft

Der Schalter »Rename« benennt angewählte Files um, der Name darf nicht bereits im gleichen Verzeichnis vorhanden sein.

»Run« startet ein File. Wenn Sie Parameter für das Programm eingeben müssen, klicken Sie auf »Req« rechts oben im Fenster. Benötigt das angewählte Programm während des Ablaufs Eingaben, stellen Sie bitte sicher, daß Sie »Foreground« im Flag-Menü unter »Run« angewählt haben.

»Unarc« versetzt archivierte Files in ihren Urzustand. Dazu klicken Sie das archivierte File an, dann »Unarc«, und »SID« stellt fest, welche Archivierungsmethode benutzt wurde. Falls die Methode nicht erkannt wird, kann das entsprechende Programm im Requester eingegeben werden. Die neuen Files können gleich weiter bearbeitet werden. Möchten Sie Ihre Files in einem anderen Verzeichnis oder Device ablegen, so wählen Sie als inaktive Liste das gewünschte Ziel und klicken danach auf den Suchschalter.

Um IFF-Grafiken anzeigen zu lassen, wählen Sie »View« an. Mit »Xedit« laden Sie die entsprechenden Dateien in Ihren angegebenen Hex-Editor. Files

Art Information

SIZE	Groesse der einzelnen Files
DATE	Datum der Erstellung des Files
TIME	Zeit der Erstellung des Files
BITS	Protectionbits der Files
NOTE	Filekommentar
DESC	Art des Files (ASCII usw.)

Abb. 3. Was sagen die Infos zu den File-Namen noch aus

werden mit »Xread« in Hex-dump ausgegeben. »SID« benutzt das gleiche Fenster wie für das Kommando »Read«. 16 Bytes pro Zeile werden dargestellt, und die korrespondierenden ASCII-Zeichen werden rechts daneben angezeigt. Die sechsstellige Hex-Zahl ganz links zeigt die Adresse des ersten Zeichens in dieser Zeile an.

■ Befehle für Files

»Arc« archiviert mit der angewählten Methode Files. Im Eingabe-Requester wird nach dem Namen des gepackten Files gefragt. »Bytes« untersucht alle angewählten Files und Verzeichnisse nach ihrer Größe und addiert diese zu einer Summe auf. Bei Verzeichnissen werden alle Unterverzeichnisse mit berücksichtigt. Welche Daten »Bytes« ausgegeben soll, bestimmen Sie im Flag-Menü unter dem Punkt »Bytes«. Mit »Print« wird das im Konfig-File angegebene Druckprogramm gestartet und die entsprechenden Files ausgedruckt. Dieses Programm muß in der Lage sein, mehrere Files hintereinander auszudrucken, da sonst Probleme auftreten können. Stellt man im Konfigurations-File dem Namen des Programms ein »Run« voran, ist »SID« sofort wieder einsatzbereit, und der Druckvorgang läuft im Hintergrund ab.

Die Funktion »Set« ist für das Setzen der Protectionbits für mehrere Programme gedacht. Der Requester entspricht dem von »Protect«. Die Bits besitzen lediglich drei Zustände. In der Maske wird der Anfangsbuchstabe eines Bits gesetzt (S,A,D usw.). Bei Minus ist dieses Bit nicht gesetzt. Bei einem Sternchen wird das entsprechende Bit ignoriert. Ein Beispiel für das Archivierungs-Flag:

A.Bit löschen : (-)
A.Bit setzen : (A)
A.Bit ignorieren : (*)

■ Befehle für Listen

Die selektierten Files werden mit »Copy« von der aktiven in die inaktive Liste kopiert. Dazu muß jedoch kein Verzeichnis in die inaktive Liste geladen sein. »SID« kopiert übrigens alle File-Kommentare sowie die Protectionbits mit. Mit »Move« werden die angewählten Files von einem Verzeichnis in ein ande-

res verschoben. In der hier vorgestellten Version werden Verzeichnisse nur innerhalb desselben Devices bewegt. Existierende Files werden nicht berücksichtigt.

Wenn der Schalter »Req« angeschaltet ist, fragt »SID« bei jedem Befehl nach dem Kommando und den Argumenten, die Sie ausführen wollen. Der Text wird für den nächsten Aufruf zwischengespeichert, damit können Sie temporäre Änderungen an dem Programm vornehmen. »Output« weist das Programm im Zustand »on« an, für jedes zu startende Programm ein CLI-Fenster zu öffnen. Dort können Sie dann sämtliche Ergebnisse der gestarteten Programme beobachten. Die beiden Schalter befinden sich in der rechten oberen Bildschirmcke. Der Status beim Programmstart kann im Konfig-File geändert werden. Tabelle 2 zeigt die Tastaturbelegung.

Konfigurations-File

Das Konfigurations-File muß den Namen »SID.config« haben und im Directory »s« stehen. Es kann mit jedem ASCII-Editor geändert und editiert werden. Die Angaben, die Sie in diesem File machen, haben alle das Format »Schlüsselwort=Wert«. Sie müssen unbedingt darauf achten, daß Sie die Schreibweise der Schlüsselwörter exakt genauso schreiben wie angegeben (Groß- und Kleinschreibung wird unterschieden). Wir unterscheiden vier Kategorien: Flags (hier können Sie nur ja (YES) oder nein (NO) eingeben), numerische Einträge, Texteingaben und Einträge wie beispielsweise Pfadangaben.

■ Flag-Einträge

»AbsolutePath« (YES voreingestellt) gibt an, ob der Pfad absolut ist (das heißt beispielsweise der Name der Diskette wird mit angezeigt) oder ob nur das Device (DF0: und so weiter) angezeigt wird.

»BeginShrunk« (NO voreingestellt) zeigt, ob »SID« nach dem Start als kleines Window oder mit voller Größe erscheinen soll.

»BytesOccupied« (YES voreingestellt): Dieses Flag bezieht sich auf das Kommando »Bytes«. Bei YES wird die An-

zahl der belegten Bytes auf dem Device angezeigt. Ist die Einstellung NO, so werden die aktuellen Bytes der Files angezeigt.

»CopyOverwrite« (YES voreingestellt): Bei YES werden beim Kopieren die im Ziel-Directory schon existierenden überschrieben. Bei NO werden diese Files nicht überschrieben.

»DateDMY« (YES voreingestellt): Bei YES wird das Datum im Format »Monat-Tag-Jahr« angezeigt, bei NO hingegen im Format »Tag-Monat-Jahr«.

»DeleteCheck« (YES voreingestellt): Bei YES erscheint vor der Ausführung des Kommandos »Delete« eine Sicherheitsabfrage. »DirDelete« (YES voreingestellt): Synonym zu »DeleteCheck« jedoch für Directories.

»EnglishDates« (YES voreingestellt): In den Verzeichnislisten werden die Daten im englischen Format ausgegeben (TODAY, YESTERDAY und so weiter). Bei NO werden alle Daten in dem Format »MM/TT/JJ« ausgegeben.

»ErrorStop« (YES voreingestellt): Bei einem Fehler erscheint ein Requester zum Abbruch des laufenden Kommandos.

»InsertMode« (YES voreingestellt) wechselt zwischen den Modi »Einfügen« und »Überschreiben«.

»LoadDiskette« (NO voreingestellt) gibt an, ob das Verzeichnis einer neu eingelegten Diskette in die aktive Liste geladen werden soll.

»LoadSwitch« (YES voreingestellt) arbeitet mit »LoadDiskette« zusammen. Er gibt an, ob die eingelesenen Verzeichnisse immer in die aktive Liste oder in die inaktive Liste geladen werden und diese dann zur aktiven Liste ernannt wird.

»QuitCheck« (NO voreingestellt) fragt bei YES nach, ob Sie das Programm wirklich verlassen wollen.

»Requester« (NO voreingestellt): Bei YES erscheint für jedes Kommando ein Requester zum Verändern der Befehle oder Argumente.

»ScreenBehind« (NO voreingestellt): Wenn Sie »SID« auf einem eigenen Screen betreiben, gibt dieses Flag an, ob der Screen hinter oder vor dem

Workbenchscreen erscheinen soll (NO = davor; YES = dahinter).

»ShowHidden« (YES voreingestellt): Haben Sie bei einigen Files das H-Protectionbit gesetzt, so können Sie das Programm anweisen, diese Files anzuzeigen (YES) oder nicht anzuzeigen (NO).

»ShowOutput« (NO voreingestellt) gibt an, ob ein CLI-Fenster für jedes von Ihnen gestartete Programm geöffnet werden soll (YES) oder nicht (NO).

»Wait« (NO voreingestellt): Bei NO werden gestartete Programme im Hintergrund abgearbeitet, bei YES wird das Programm in einen Wartezustand gebracht, und Sie können mit dem gestarteten Programm kommunizieren.

»WindowStay« (YES voreingestellt): Wenn Sie die Fenstergröße von »SID« verändern, erinnert sich das Programm an die letzte Position des Windows (YES) und bringt das Window bei Vergrößerung genau wieder an dieser Stelle auf den Bildschirm.

■ Numerische Einträge

Bei diesen Einträgen sind nur Zahlen erlaubt.

»Color=0..3« (-1 voreingestellt): Falls Sie »SID« auf einem eigenen Screen betreiben, haben Sie hier die Möglichkeit, die Farben einzustellen. Der Eintragungswert kann zwischen 0 und 4095 liegen (für jede Farbe eine Zahl). Setzen Sie hingegen -1 ein, so werden die Farben der Workbench benutzt.

»FullLeft« (0 voreingestellt): Dies ist der Wert für die linke Kante des größten Fensters.

»FullTop« (0 voreingestellt): der Wert für die obere Kante des größten »SID«-Fensters.

»HalfLeft« (0 voreingestellt): dies ist der Wert für die linke Kante des halben Fensters. Der Eintrag ist für spätere Applikationen gedacht, da »SID« immer die Breite des Workbench-screens annimmt.

»HalfTop« (0 voreingestellt): der Wert für die obere Kante des halben Fensters.

»LaceLeft« (0 voreingestellt): der Wert für die linke Kante des Interlace-Fensters.

»LaceTop« (0 voreingestellt): der Wert für die obere Kante des Interlace-Fensters.

»OutputHeight=« (120 voreingestellt) gibt die Höhe des CLI-Fensters an, das »SID« öffnet.

»OutputLeft=« (0 voreingestellt): die linke Kante des CLI-Fensters.

»OutputTop=« (80 voreingestellt): die obere Kante des CLI-Fensters.

»OutputWidth=« (640 voreingestellt): Breite des CLI-Fensters

»ShrinkLeft=« (344 voreingestellt): die linke Kante des verkleinerten Fensters auf dem Workbenchscreen.

»ShrinkTop=« (0 voreingestellt): die obere Kante des verkleinerten Fensters.

»SpecifyLeft=« (0 ist voreingestellt): die linke Kante des User-Fensters (dieses Fenster können Sie selbst einstellen, siehe »Specify« weiter oben).

»SpecifyRows=« (10 voreingestellt): die Anzahl der Zeilen, die das User-Fenster haben soll (kommt nur zum Einsatz, wenn Sie bei »WindowHeight=Specify« eintragen).

»SpecifyTop=« (0 voreingestellt) die obere Kante des User-Fensters.

■ Zeicheneingaben

Bei diesen Eingaben wird alles, was nach dem Gleichheitszeichen folgt, berücksichtigt (auch Leerzeichen)

»Arc=« (»c:arc a« voreingestellt): Hier geben Sie den Pfad zu dem Arc-Programm an, das Sie benutzen wollen. Da »SID« fünf verschiedene Archivierungsmethoden kennt, gibt es auch fünf verschiedene Einträge.

»AutoDir1=« »AutoDir2=« (nichts ist voreingestellt): Falls das Programm beim Start sofort selbstständig ein oder zwei Directories einlesen soll, so kann dies hier angegeben werden.

»Button1=« bis »Button8=« Das Programm besitzt acht verschiedene Device-Schalter. Wenn Sie keine Festplatte besitzen und die Schalter 3 bis 5 umbenennen wollen, so können Sie hier einen Pfad für ein Directory angeben. Diese Eingaben veranlassen »SID«, bei Aufruf des korrespondierenden Schalters den angegebenen String auszuführen.

»Calculator=« (»c:calculator« voreingestellt): Hier geben Sie das gewünschte Rechnerprogramm an.

»Command=« (keine Voreinstellung): Hier geben Sie den CLI-Befehl ein, der durch die Funktion »Command« aus dem Systemmenü ausgeführt wird. Sie müssen hier nichts eingeben, dies ist nur dazu gedacht, falls Sie immer nur einen bestimmten CLI-Befehl ausführen wollen.

»DiskEdit=« (»c:DiskX« voreingestellt): Der Name des Disketteneditors, den Sie verwenden möchten, wird hier eingetragen.

»Execute=« (keine Voreinstellung): Argumentzeile für Batch-Dateien. Dieser Eintrag wird meist leer sein, da Sie voraussichtlich verschiedene Batch-Dateien starten werden.

»FileEdit=« (»c:e« voreingestellt): Hier geben Sie Pfad und Name des von Ihnen verwendeten ASCII-Editors ein.

»FileXEdit=« (»c:newzap« voreingestellt) der von Ihnen gewählte Hex-Editor, um binäre Files zu editieren.

»Hear=« (»c:hear« voreingestellt): Ein Sound-Abspielprogramm für IFF- und RAW-Daten.

»LHArc=« (»c:LhArc a« voreingestellt): LHArc-Programm, um Files zu archivieren.

»ListArc=« (»c:arc l« voreingestellt): »ListArc-Programm«, um den Inhalt von archivierten Files anzuzeigen.

»ListLHArc=« (»c:LHArc l« voreingestellt): ListLHArc-Programm, um den Inhalt der archivierten Files anzuzeigen.

»ListPAK=« (keine Voreinstellung): Dieser Eintrag ist für spätere Versionen von »PAK« gedacht, da das Programm im Moment das Listen des Inhalts von archivierten Pak-Files noch nicht zuläßt.

»ListWarp=« (keine Voreinstellung): Hier gilt das gleiche wie bei »ListPAK«.

»ListZoo=« (»c:zoo l« voreingestellt): Zoo-Programm, um den Inhalt der archivierten Files zu zeigen.

»MakeDirIcon=« (keine Voreinstellung): Die Pfadangabe zu dem File, das Sie als Icon für den Befehl »MakeDir« benutzen wollen.

»NewCLI=« (»c:Newshell «NEWCON:0/0/640/96/Type ENDCLI to Quit« voreingestellt): Der Aufruf, um ein neues CLI-Fenster aufzurufen.

»Other=« (»c:cc« voreingestellt): Hier können Sie den

Pfad zu einem Programm Ihrer Wahl angeben (zum Beispiel Kopierprogramm oder Textverarbeitung und so weiter).

»Pak=« (»c:pak« voreingestellt): Der Pfad und Name zu dem PAK-Archivierungsprogramm, das Sie benutzen wollen.

»Pattern=« (»# ?c« voreingestellt): Diese Eingabe erscheint in dem Requester, wenn Sie den Schalter »Pattern« anwählen.

»Preferences=« (»sys:Prefs/Preferences« voreingestellt): Pfad zum Programm »Preferences«.

»Print=« (»run c:Print« voreingestellt): Name und Pfad Ihres gewählten Druckprogramms.

»Run=« (keine Voreinstellung): Argumentzeile für Programme, die von »SID« mit dem Schalter »Run« gestartet werden.

»SeparateDirs=« (NO voreingestellt): Bei NO werden Verzeichnisse und Files gemischt angezeigt. Setzen Sie hingegen YES ein, so werden zuerst immer die Directories und dann die Files angezeigt.

»Set=« (»*****«) voreingestellt): Diese Maske erscheint bei dem Aufruf des Kommandos »Set« im entsprechenden Requester. Die Bits sind wie folgt angeordnet: »HSPARWED«.

»UnArc=« (»c:arc x« voreingestellt): Programm, um archivierte Files wieder zu entpacken.

»UnLHArc=« (»c:lharc x« voreingestellt): Wie »UnArc« jedoch für das Programm »LHArc«.

»UnPak=« (keine Voreinstellung): Da Pal-Files sich selbst bei ihrem Aufruf entpacken, bleibt dieser Eintrag vorerst leer.

»UnWarp=« (»c:warp write« voreingestellt): Warp-Programm, das Sie zum Entpacken archivierter Files benutzen wollen.

»UnZoo=« (»c:zoo x/l« voreingestellt): Programm mit dem Sie »gezooete« Files wieder entpacken wollen.

»View=« (»c:superview -c« voreingestellt): Name und Pfad des von Ihnen benutzten IFF-Grafikanzeigeprogramms (-c verhindert das Colorcycling).

»Warp=« (»c:warp read 0 79« voreingestellt): Warp-Programm, mit dem Sie eine Diskette packen wollen.

»Zoo=« (»c:zooa« voreingestellt): Zoo-Programm, um Files zu archivieren.

■ Spezielle Eintragungen

»ArchiveMethod=« (»Arc« voreingestellt): Hier können Sie die Archivierungsmethode, die »SID« nach dem Programmstart verwenden soll, einstellen. Sie können folgende fünf Arten einstellen: »Arc«, »Lharc«, »Pak«, »Warp« und »Zoo«. Diese Einstellung hat nichts mit dem Entpacken oder Listen des Inhalts von archivierten Files zu tun, da das Programm die Methode selbst bestimmt.

»ArrowDirection=« (»Flow« voreingestellt): Dieses Flag gibt die Richtung des Pfeils in der Mitte des Bildschirms an. Geben Sie »Flow« an, so zeigt der Pfeil in Datenflußrichtung, bei »Active« hingegen fließen die Daten in Richtung der aktiven Seite.

»DeviceButtons=« (»df0df1Ram Raddh0dh1dh2dh3« voreingestellt): die Namen der acht Device-Schalter von »SID«. Diese Namen müssen ohne Leerzeichen direkt hintereinander angegeben werden. Die Eingabe muß übrigens immer genau drei Zeichen lang sein, und Sie müssen alle acht Device-Schalter angeben.

»EntryType=« (»Size« voreingestellt): Diese Voreinstellung gibt an, welche Informationen zu den File-Namen noch mit ausgegeben werden. Die Angaben können Sie Tabelle 3 entnehmen.

»ScreenType=« (»Workbench« voreingestellt): Dieser Eintrag gibt an, auf welchem Screen das SID-Fenster erscheinen soll.

Dieser Workshop war fürs erste zwar ziemlich lang, aber wir hoffen, Ihnen mit diesem Überblick bei der Arbeit mit »SID« weitergeholfen zu haben. Die Funktionsvielfalt dieses Programms sollte Sie nicht abschrecken, die Qualitäten noch näher kennenzulernen. Denn Probieren geht über Studieren. (vb)

PD-Info

Programm: SID

Merkmal: Shareware

Der Autor bittet um eine Gebühr von 25\$

Autor: Timm Martin

P.O. Box 10084

Largo, FL 34643

Quelle: Fish 338

Das Programm »Selector« gibt Ihnen diese Möglichkeit. Um in den Genuß von Selector zu kommen, fügen Sie den Aufruf des Programms in Ihre Startup-Sequence ein. Danach haben Sie die Möglichkeit, alle Programme der Bootdiskette oder der Festplatte (wenn diese autobootend ist) über das Auswahlmenü zu starten. Sie müssen also nicht mehr in den CLI wechseln oder unendlich viele Unterverzeichnisse öffnen, um an das Icon des entsprechenden Programms zu gelangen. »Selector« läuft erst richtig zu guter Form auf, wenn Sie eine Festplatte besitzen. Eine Begrenzung der Einträge besteht nicht, da Sie mit einer Form von »Overlays« arbeiten können.

»Selector« wird wie folgt aus dem CLI aufgerufen:

```
Selector [-h][<Scriptfile-name>]
```

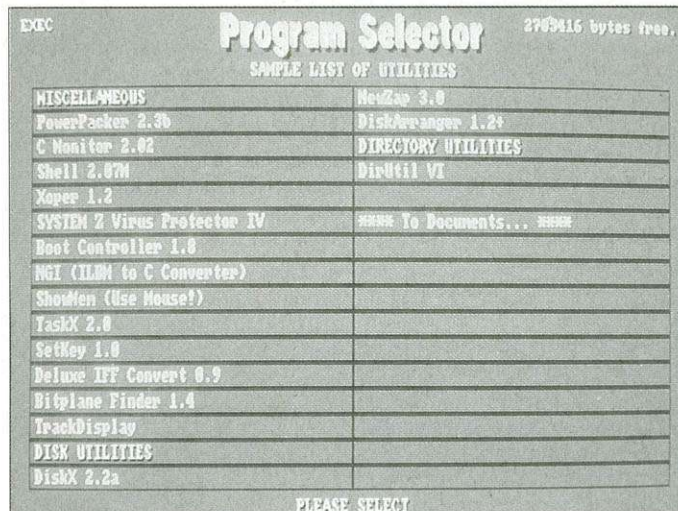
Zur kurzen Erläuterung: Setzen Sie das Flag »-h«, so werden die beiden Spalten des Auswahlfensters mit gleich vielen Einträgen gefüllt. Bei Scriptfilenamen müssen Sie den Pfad- und File-Namen des Scriptfiles angeben. Erfolgt keine Angabe, so wird als Filename »S:Selector.programs« benutzt.

Sie können in der Aufruf-Sequenz auch mehrere Anweisungen hintereinander angeben. Diese müssen dann durch ein Semikolon getrennt werden. Einige Zeichen haben eine besondere Bedeutung, wenn Sie diese als erstes Zeichen eines Kommandos benutzen. Hierzu einige Beispiele:

»*« Sie können dieses Zeichen benutzen, wenn Sie das Programm im Multitasking betreiben wollen. Ist Multitasking unerwünscht, so lassen Sie das Sternchen einfach weg.

»#« Befindet sich dieses Zeichen an erster Stelle, so wird »Selector«, nach der Aufruf-Sequenz beendet (immer!). Diese Funktion ist nützlich, falls Sie den durch das angewählte Programm belegten Speicher benötigen.

»<« Bei diesem Zeichen wird der Text, der direkt dahinter folgt, in einem String-Gadget ausgegeben. Damit haben Sie hier die Möglichkeit, Argumente an das nachfolgende Kommando (in der Aufruf-Sequenz)



Mit »Selector« lassen sich eine Vielzahl Programme per Menü starten

Robert Münch

Selector-Workshop

Programme komfortabel per Menü zu starten, ist der Wunsch vieler User.

zu übergeben. Ein Beispiel: Der Aufruf »!Welches File editieren ? ;c:Microemacs« hat zur Folge, daß die Eingabe, die Sie bei »Welches File editieren?« gemacht haben, an den Aufruf von Microemacs weitergegeben wird. Sie haben somit den Aufruf des Editors mit angehängtem Filenamen als Parameter angeführt.

»<« Hiermit kann ein zweites Scriptfile geladen werden. Das Programm versucht das File, das direkt hinter diesem Zeichen angegeben ist, zu laden und in gewohnter Weise darzustellen. Diese Funktion ermöglicht Ihnen das Erstellen von Untermenüs, in denen Programme gestartet werden können, die Sie nicht so oft brauchen.

»@« Schreiben Sie den »Klammeraffen« unter eine Textzeile, so erscheint diese in Orange und kann daher als eine Überschrift für eine Gruppe von Programmen benutzt werden. Auch hierzu ein kurzes Beispiel:

UTILITIES

@
Assembler

*dh0:assembler/devpac/gemnam2

...
TEXTVERARBEITUNG
@

Dieses Beispiel zeigt, daß nun alle Einträge, die unter Utilities stehen, zu dieser Gruppe gehören. Mit dem Wort Textverarbeitung, wird die neue Überschrift festgelegt.

»Selector« bietet Ihnen 32 Gadgets (in der PAL-Version), abzüglich der Anzahl der Überschriften, in denen Sie einen Programmaufruf angeben können. Das Fenster des Programms beinhaltet auch die Tiefen-Gadgets in der rechten oberen Ecke, diese sind jedoch unsichtbar. Links oben in der Ecke ist ebenfalls ein Gadget eingebaut. Klicken Sie dieses Gadget an, so wechselt es der Reihe nach von »Run«, »Exec« nach »Launch«. Die Bedeutungen im einzelnen:

»Run« startet das Programm im Multitask, daß heißt, »Selector« kann nach dem Aufruf sofort weiter benutzt werden.

»Execute« startet ein Programm gemäß der Aufruf-Sequenz. »Selector« erscheint je-

doch erst bei Beendigung des gestarteten Programms wieder. Mit »Launch« wird nach dem Start eines Programms Selector beendet. Wenn dieses Programm beendet ist, wird mit dem nächsten Befehl in der Startup-Sequence weitergemacht. Die Funktion »Launch« können Sie auch erzwingen, egal welcher Modus gerade eingeschaltet ist, indem Sie dem Aufruf des Programms ein »#« voranstellen.

Reichlich Info

In der rechten oberen Ecke finden Sie die momentan freie Anzahl an Bytes, diese Anzeige wird ständig auf den aktuellen Stand gebracht. Drücken Sie nun einmal die rechte Maustaste innerhalb des Selector-Fensters. Sie sehen, daß sich das Fenster auf die Größe einer Titelzeile reduziert. Um das Selector-Fenster wieder zurückzuholen, drücken Sie einfach noch einmal die rechte Maustaste innerhalb des kleinen Windows. Durch Anklicken des Schließsymbols, verlassen Sie »Selector«, und der Amiga fährt mit der Abarbeitung der Startup-Sequence fort. Die Installation des Programms erweist sich als sehr einfach. Folgende Files sollten Sie in die angegebenen Verzeichnisse kopieren: Die Files »Selector«, »More« und »Run« auf Diskette/Festplatte in das Verzeichnis »c«. Das File »Selector.doc« in das Hauptverzeichnis Ihrer Diskette/Festplatte.

Wenn Sie das gemacht haben, müssen Sie nur noch das Scriptfile »Selector.programs« erstellen und dieses in das Directory »s« kopieren. Der Aufruf in der Startup-Sequence erfolgt durch den Eintrag »Selector«.

Damit soll's für dieses Mal genügen. Ich hoffe, wir haben Ihnen ein gutes und nützliches PD-Programm vorgestellt, das Ihre tägliche Arbeit am Amiga etwas komfortabler gestaltet.

(vb)

PD-Info

Programm: Selector

Merkmal: Freeware

Autor: Nico Francois

Quelle: Fish 302


```

87: LINE(299,26)-(333,12)
88: LINE(237,12)-(338,12)
89: LINE(305,26)-(204,26)
90: LINE(306,26)-(306,101)
91: LINE(340,11)-(340,86)
92: LOCATE 8,30:PRINT "a,b"
93: LOCATE 7,40:PRINT "c,d"
94: LOCATE 3,33:PRINT "e,f"
95: U=Br+M+M:V=Ho+M+M
96: W=Ti:X=Ho+M+M
97: Y=Br:Z=Ti
98: LOCATE 15,15:PRINT "teile a,b = ";U;"mmn * ";V;"mmn
  * ";M;"mm"
99: LOCATE 17,15:PRINT "teile c,d = ";W;"mmn * ";X;"mmn
  * ";M;"mm"
100: LOCATE 19,15:PRINT "teile e,f = ";Y;"mmn * ";Z;"mmn
  * ";M;"mm"
101: LOCATE 22,86:INPUT F
102: GOTO Vorbereitung
103:
104: Neumat:
105: CLS
106: CLEAR
107: COLOR 2,3:LOCATE 1,2:PRINT "Masseingabe fuer Mater
ialberechnung"
108: COLOR 3,0
109: LOCATE 3,3:PRINT "Breite in mm   :"
110: LOCATE 5,3:PRINT "Hoehe in mm   :"
111: LOCATE 7,3:PRINT "Tiefe in mm   :"
112: LOCATE 3,25:INPUT Br
113: LOCATE 5,25:INPUT Ho
114: LOCATE 7,25:INPUT Ti
115: GOTO Altmat
116:
117: Speicher:
118: OPEN "R",#1,"Speicher",56
119: FIELD #1,20 AS Her$,20 AS Typ$,8 AS Aqu$,8 AS Gute$
120: Anf:
121: CLS
122: COLOR 2,3:LOCATE 1,2:PRINT "Speicher"
123: COLOR 3,0
124: LOCATE 3,3:PRINT "> 1 <  Daten lesen"
125: LOCATE 5,3:PRINT "> 2 <  Daten schreiben"
126: LOCATE 7,3:PRINT "> 3 <  Zurueck"
127: LOCATE 10,3:INPUT "Ihre Wahl: ";Wahl$
128: IF Wahl$<"1" OR Wahl$>"3" THEN GOTO Speicher
129: IF Wahl$="1" THEN GOTO Lesen
130: IF Wahl$="2" THEN GOTO Schreiben
131: IF Wahl$="3" THEN CLOSE #1:GOTO Hauswahl

```

Listing: Boxenberechnung.BAS



```

133: Lesen:
134: CLS
135: COLOR 2,3:PRINT"Lesen":COLOR 3,0:PRINT
136: INPUT"Adresse Nr. (-1 fuer Ende):";code%
137: IF code%=-1 THEN Anf
138: GET #1,code%
139: IF EOF(1) THEN BEEP:GOTO Lesen
140: PRINT "Hersteller: ",Her$
141: PRINT "Typ" : ",Typ$"
142: PRINT "Vas" : ",Aqua$"
143: PRINT "Qts" : ",Gute$"
144: PRINT
145: INPUT "Weitere Daten lesen (j/n)";Antw$
146: IF UCASE$(Antw$)="J" THEN GOTO Lesen
147: GOTO Anf
148:
149: Schreiben:
150: CLS
151: COLOR 2,3:PRINT"Schreiben":COLOR 3,0:PRINT
152: INPUT"Adresse Nr. (-1 fuer Ende):";code%
153: IF code%=-1 THEN Anf
154: INPUT "Hersteller: ";EingHer$
155: INPUT "Typ" : ";EingTyp$"
156: INPUT "Vas" : ";EingAqua$"
157: INPUT "Qts" : ";EingGute$"
158: PRINT
159: LSET Her$=EingHer$
160: LSET Typ$=EingTyp$
161: LSET Aqua$=EingAqua$
162: LSET Gute$=EingGute$
163: PUT #1,code%
164: INPUT "Weitere Daten schreiben (j/n)";Antw$
165: IF UCASE$(Antw$)="J" THEN GOTO Schreiben
166: GOTO Anf
167:
168: Fehler:
169: CLS
170: PALETTE 1,1,0,0
171: PALETTE 2,1,0,0
172: PALETTE 3,1,0,0
173: WINDOW 2,"ERROR",(0,0)-(315,70),18,-1
174: PRINT "Fehler Nr. :";ERR
175: PRINT "Chassis nicht verwendbar"
176: PRINT "RETURN startet Programm"
177: INPUT K:WINDOW CLOSE 2
178: RESUME Vorbereitung
179:
180: Ende:
181: SYSTEM

```

Listing: Boxenberechnung.BAS

Warum nicht gleich bei uns?

TOP - Aktualität bei der **PD!**

Alle gängigen Serien und das Neuste von ACS über Fish bis Taifun.
Weit über 7000 Disketten, mehr als 200 Reihen! Gute unbekannte
Serien wie Best of Public Dom.(- 61), Docs by Dr. Knox, Saar (- 79!)
TuCs Digis. Vollgepackte deutsche Katalogdisk incl. Versand: **DM 5.-**

Alle Programme zum Buch "Public Domain Schatztruhe" von "technicSupport" vollständig auf **10 Disks** nur **DM 25,- !!**

Die 10 besten PD-Disks eines Monats
aus allen Serien
von uns zusammengestellt
zum Abo-Preis von nur **DM 25,- !!**

TOP - Qualität bei der **Hardware!**

Golden Image Supermaus, Schnupperpreis	DM 79,-
Umschaltplatine Kickstart 2.0/1.2/1.3	auf Anfr.
512 KB RAM für A500, Uhr + Abschalter	DM 125,-
3.5 LW extern mit Bus	DM 179,-
5.25 LW extern (TEAC), 40/80 Tr., Bus	DM 225,-
2 MB für A500, voll autokonfigurierend	DM 549,-
Supra 2 MB/8MB für A2000, voll autokonfig.	DM 749,-
Grafikkarte Highgraph V, 50 Hz Vollbild	DM 898,-
Live! 2000 Echtzeit-Digitizer, Spitzentest	DM 949,-
Quantum HD für A2000, 42 MB, SCSI, 19 ms	DM 1199,-
Quantum HD für A500, 42 MB, SCSI, 19 ms	DM 1349,-
Modems von US Robotics, Supra, Best u.a.	auf Anfr.

Versandkosten bei PD/kl. Hardware DM 6,- (Vork.)/ DM 10,- (Nachnahme), sonst nach Gewicht. Fordern Sie unsere aktuellen Preislisten an!

Gabriele von Thienen

Handeln mit neuen Medien

Postfach 100 648, 1000 Berlin 10

Tel: 030 - 322 63 68 bis 22.00 Uhr Fax: 030 - 321 31 99

Modula-2

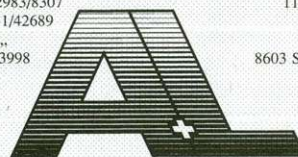
M2Amiga ist das auf dem Commodore Amiga am weitesten verbreitete Modula-2 System mit den meisten Werkzeugen und Bibliotheken und einer Riesenauswahl von PD-Disketten.

Compiler	SFr.	DM	Treasures-Libs	SFr.	DM
M2Amiga	270.00	342.00	AmigaTreasures	158.00	201.78
Demodiskette			FileTreasures	158.00	201.78
(Fish-Disk 113)	10.00	10.00	ModulaTreasures	78.00	102.60
			MathTreasures	78.00	102.60
			Treasures-Demodisk	10.00	10.00
Werkzeuge	SFr.	DM			
Source-Level-					
Debugger	180.00	228.00	Report-Libs	SFr.	DM
Automatisches Make	80.00	108.30	IntuitionReport	80.00	108.30
M2APSE	80.00	108.30	DeviceReport	80.00	108.30
M2APSE Demodisk	10.00	10.00	GraphicReport	80.00	108.30
M2Decoder	80.00	108.30	Report-Demodisk	10.00	10.00
Objektconverter	80.00	108.30			
Sourcecode + RTS	80.00	108.30	PD-Disketten	SFr.	DM
IFF-Bibliothek	80.00	108.30	AMOK PD-Disk, je	10.00	10.00
Speed-Editor	80.00	108.30	Treasures-PD, je	10.00	10.00

Die genannten Preise sind unverbindlich
M2Amiga Produkte sind auch im guten Fach- und Versandhandel erhältlich.

Die Modula-2 Leute:

Bundesrepublik Deutschland:
Beech Tree Systems GmbH
5788 Winterberg, Tel. 02983/8307
3300 Braunschweig, 0531/42689
SW-Datentechnik GmbH.,
2085 Quickborn, 04106/3998
Advanced Applications,
GmbH, 7500 Karlsruhe,
Tel. 0721/700912
Interplan-Muhlert
8000 München GmbH,
Tel. 089/1234066



Österreich:
ICA Elektronische Geräte Ges.mbH,
1160 Wien, Tel. 0222/4545010

Schweiz:
Frei-Elektronik,
8603 Schwerzenbach. 01/945 54 32

Generalvertrieb:
A+L AG
Dädleriz 61

CH-2540 Grenchen
Tel. (0041/0)65/52 03 11
Fax (0041/0)65/52 03 79

Andreas Polk

Freie Fahrt für Modula2

In diesem Kursteil wollen wir uns den Prozeduren und Funktionsprozeduren widmen. Sie erfahren, was Prozeduren sind, wofür sie gebraucht werden und wie man Variablen an sie übergibt.

Prozeduren sind einzelne Programmteile, die eine bestimmte Aufgabe übernehmen. Eine Prozedur kann als alleinstehendes Programm angesehen werden, das innerhalb eines übergeordneten Programms eine bestimmte Aufgabe übernimmt. Somit umfaßt eine Prozedur also eine Ansammlung von Befehlen. Diese Befehle werden unter einem bestimmten Namen zusammengefaßt, dem Prozedurnamen. Sollen dann später in einem Programm die Befehle ausgeführt werden, so muß nach einmaliger Definition der Prozedur nur noch der Prozedurname aufgerufen werden.

Das hört sich alles etwas kompliziert an. Betrachten wir einfach einmal das Listing 1 als Beispiel. In der ersten Zeile wird wie immer der Modulname angegeben. Dann folgen die schon bekannten Importanweisungen. In diesem Falle werden nur die beiden Befehle »WriteString« und »WriteLn« aus dem Modul »InOut« importiert. Nun kommt die Definition der Prozedur. Sie wird eingeleitet durch das Schlüsselwort »PROCEDURE«.

Hinter diesem Schlüsselwort wird der Name der Prozedur angegeben. Der Name lautet in diesem Falle »InfoText«. Abgeschlossen wird die Zeile wieder durch ein Semikolon. Der Rest ist ein ganz normales Programm. Zuerst finden wir also die Importanweisungen und die Variablendeklarationen. Diese sind optional und können weggelassen werden, falls sie nicht benötigt werden. Da in unserem Beispiel alle Befehle bereits im Hauptpro-

gramm importiert wurden und keine Variablen gebraucht werden, können diese beiden Teile hier weggelassen werden. Bei Bedarf könnte noch die Typen- und Konstantendeklaration erfolgen, aber auch die wird in dieser Prozedur nicht gebraucht. Nun schließt sich der Hauptteil der Prozedur, der Programmcode, an. Wie in einem Hauptprogramm auch, wird er durch das Befehlswort »BEGIN« eingeleitet.

Dann folgen die Befehle, in unserem Fall die Befehle »WriteString« und »WriteLn«, die einen Infotext ausgeben. Am Ende der Prozedur finden Sie wie immer ein »END«, mit der Angabe des Prozedurnamens, also hier »InfoText«. Beachten Sie, daß nach dem Namen der Prozedur kein Punkt, sondern lediglich ein Semikolon folgt. Der Punkt kennzeichnet immer das Ende eines Programms, das Semikolon nur das Ende einer Prozedur, einer Schleife oder eines Befehls.

Nachdem die Funktion definiert wurde, beginnt das Hauptprogramm. Es wird wie gewohnt durch »BEGIN« eingeleitet. Danach finden wir den Aufruf der eben definierten Prozedur. Der Name wurde ja bereits in der Definition nach dem Schlüsselwort »PROCEDURE« angegeben. Er lautet in diesem Falle »InfoText«. Die Prozedur wird also durch »InfoText« aufgerufen.

Beim Programmablauf werden nun an dieser Stelle alle Befehle ausgeführt, die in der Prozedur »InfoText« angegeben wurden, in diesem Falle wird also der Infotext ausgegeben. Die Angabe des Prozedurnamens wird wie gewohnt durch ein Semikolon beendet.

Kursfahrplan	
Teil 1: Der Compiler M2Amiga	
Teil 2: Die ersten Befehle	
Teil 3: Schleifen und Verschachteln	
Teil 4: Prozeduren und Variablen	
Teil 5: Weitere Datentypen	
Teil 6: Zeiger und Datenstrukturen	
Teil 7: Modulkonzept	
Teil 8: Systemprogrammierung	

Prozeduren sind also im Endeffekt nichts anderes als vom Benutzer geschriebene Befehle; sie werden genauso gehandhabt wie bereits vorhandene. Unser Programm wird durch den Befehl »END« und der Angabe des Programmnamens beendet. Sie wissen nun, wie eine Prozedur aufgebaut ist und wie man sie aufruft. Welchen Sinn hat eine Prozedur denn nun?

Der erste Vorteil besteht darin, daß ein Programm so abstrakter und besser lesbar wird.

So weiß der Anwender sofort, wenn er das Programm durchschaut, daß der Befehl »InfoText« einen Text ausgibt, der dem Anwender Informationen über das Programm gibt. Würde er mehrere Befehle sehen, müßte er diese erst alle durchschauen, bis er deren Sinn erkennt. Zugegeben, in diesem Beispiel hätte er das recht schnell raus, doch stellen Sie sich einmal eine Prozedur vor, die die Fakultät einer Zahl berechnet. Der Aufruf einer Prozedur »Fakultaet« ist besser lesbar als die Programmzeilen, die die Fakultät einer Zahl berechnen. Außerdem kann so eine Prozedur von verschiedenen Stellen aus aufgerufen werden. Unter diesem Gesichtspunkt fällt auch noch die vereinfachte Programmentwicklung. Ein Programm, welches ein Problem lösen soll, wird erst einmal in grobe Teile unterteilt, die den Rahmen des Programms festlegen. Diese einzelnen Teile werden wieder in kleinere Teile unterteilt, bis man dann hinterher auf Befehlsebene ist. Nehmen wir als Beispiel ein Programm, welches ein Fenster öffnet, einen Text ausgeben soll und dann nach Druck einer bestimmten Taste das Fenster wieder schließt.

Ein sinnvolles Schema wäre in diesem Falle folgendes:

```
MODULE FensterDemo;
Importliste;
Deklarationen;
BEGIN
  OeffneFenster;
  SchreibeText;
  Wiederhole
    WarteAufTaste;
  BisTastegleichVergleichsta-
    steist;
  SchliesseFenster;
END FensterDemo.
```

Jemand, der dieses Ablaufschema in die Hand bekommt, würde sofort erkennen, was das Programm macht. Er erfährt, daß zuerst ein Fenster geöffnet wird. Dann erfährt er, daß ein Text ausgegeben wird. Wollen wir es nun realisieren, so müßten wir dieses Schema so lange ergänzen, bis wir ein ablauffähiges Programm erhalten. Als erstes fällt hier natürlich die Konstruktion

```
Wiederhole WarteAufTaste Bi-
sTastegleichVergleichsta-
steist;
```

auf. Wir haben im letzten Kursteil eine Konstruktion kennengelernt, die genau dieses Schema besitzt. Erinnern Sie sich? Es war die Repeat-Until-Schleife.

Somit können wir diesen Programmteil schon durch folgenden Teil etwas detaillierter beschreiben:

```
REPEAT WarteAufTaste; UNTIL
Taste=Vergleichstaste;
```

Wir entwickeln nun für den Programmteil »OeffneFenster« eine Prozedur, die ein Fenster öffnet. Für »SchreibeText« schreibt man nun eine Prozedur, die einen Text in diesem Fenster ausgibt. Genauso verfährt man für die anderen Befehle. Somit bleibt das Hauptprogramm gut lesbar. In Listing 2 sehen Sie dieses Programm so, wie es mit Prozeduren realisiert würde. Da man zum Öff-

nen eines Fensters unter Intuition programmieren müßte, was wir bis jetzt noch nicht gemacht haben, habe ich nur die Prozedurköpfe angegeben, nicht aber die Befehle, die die einzelnen Aufgaben übernehmen. Trotzdem können Sie an diesem Beispiel erkennen, daß das Programm durch die Verwendung von Prozeduren wesentlich übersichtlicher und besser lesbar geworden ist.

Kommen wir nun zu dem zweiten Vorteil, den uns die Verwendung von Prozeduren bietet. Eine Befehlsfolge muß nur einmal definiert werden, sie kann dann von jeder Stelle des Hauptprogramms aufgerufen werden. Dieser Vorteil ist gravierender, als er sich anhört. Nehmen wir einmal an, Sie schreiben ein Programm, in dem an vielen verschiedenen Stellen die Fakultät einer Zahl berechnet wird. Man könnte nun überall dort die Zeilen einfügen, die diese Aufgabe übernehmen. Einfacher ist es jedoch, einmal eine Prozedur zu definieren, die dieses übernimmt. Im Programm muß an den entsprechenden Stellen nur noch die Prozedur »Fakultaet« aufgerufen werden. Hier noch einige Punkte, die Sie beachten sollten:

- Für Prozedurnamen gelten genau die gleichen Regeln wie für die Variablennamen.
- Eine Prozedur wird immer erst ausgeführt, wenn sie im Programm aufgerufen wird. Bisher wurde eine Prozedur immer vom Hauptprogramm aufgerufen, wobei der Prozedur jedoch keine Werte übermittelt wurden. Normalerweise wird eine Prozedur aber Werte aus dem aufrufenden Programm brauchen. Aus diesem Grunde wurde die Übergabe von Werten an eine Prozedur implementiert.

Die Übergabe von Werten an Prozeduren

Dies kann auf zwei Arten geschehen. Zum einen können die übergebenen Werte von der Prozedur verändert werden, zum anderen können die Werte nur als Werteparameter übergeben werden. Das heißt, daß die Werte der Prozedur bekanntgegeben werden, diese aber von der Prozedur nicht verändert werden können. Wir

wollen eine Prozedur schreiben, die einen Text in einer bestimmten Anzahl ausgibt. Zugegeben, dieses Programm ist nicht sehr sinnvoll, zeigt aber, wie man einer Prozedur einen Wert übergibt. Der Prozedur soll in unserem Beispiel der Wert übergeben werden, der definiert, wie oft der Text ausgegeben werden soll.

Die Realisierung dieses Programms sehen Sie in Listing 3. Eingeleitet wird die Definition der Prozedur wieder durch das Befehlswort »PROCEDURE«. Dann folgt der Name der Prozedur. Nun kommt nicht, wie bisher, das abschließende Semikolon, sondern eine Klammer. Innerhalb dieser Klammer wird eine Variablendeklaration vorgenommen. In diesem Falle wird eine Variable namens »n« definiert, die vom Typ »CARDINAL« ist. Beim Aufruf der Prozedur muß vom Hauptprogramm nun immer eine Variable des Typs »CARDINAL« übergeben werden. Dies kann durch Angabe des Wertes direkt geschehen oder über die Angabe einer anderen Variablen. Schauen wir uns einmal den Aufruf dieser Prozedur vom Hauptprogramm aus an. Im ersten Beispiel wird in der Zeile

```
TextAusgabe(5);
```

die Prozedur »TextAusgabe« mit dem Parameter »5« aufgerufen. Dieser Wert wird der Prozedur in Klammern übergeben. Genau an dieser Stelle wird nun der Variablen »n« dieser Wert übergeben, das heißt, »n« bekommt in dem Augenblick des Prozeduraufrufes den gleichen Wert zugeteilt, den der Werteparameter hat. Bei dem ersten Aufruf erhält »n« also den Wert »5«. Die Variable »n« kann nun innerhalb der Prozedur beliebig verändert werden, was in unserer Prozedur aber nicht geschieht. Beim zweiten Aufruf wird der Werteparameter nicht direkt übergeben, sondern in Form einer Variablen. Es geschieht genau das gleiche. Der Variablen »n« wird der Wert zugewiesen, den »Anzahl« hat. In diesem Fall ist das die Zahl Zehn.

Beachten Sie, daß, wenn nun der Wert der Variablen »n« geändert würde, dies nicht den Wert der Variablen »Anzahl« beeinflussen würde. »Anzahl« behält immer seinen Wert bei,

AMIGA-DOS-REFERENZ

Die vorgefertigten Prozeduren und Funktionsprozeduren

Im Modulall-Standard sind einige Prozeduren und Funktionsprozeduren enthalten. Einige haben Sie schon in einem der vorherigen Kursteile kennengelernt. An dieser Stelle soll nun eine vollständige Auflistung der zur Verfügung stehenden Prozeduren folgen, so daß Sie hier immer nachschlagen können, falls Sie eine bestimmte Prozedur suchen.

Variable1:=ABS(Variable2);

»Variable1« und »Variable2« müssen vom gleichen Typ und entweder »INTEGER« oder »REAL« sein. Die Funktion »ABS« liefert den Betrag von »Variable2«.

Variable1:=CAP(Variable2);

»Variable1« und »Variable2« müssen beide vom Typ »CHAR« sein. »CAP« liefert den Großbuchstaben von »Variable2«.

Variable1:=CHR(Variable2);

»Variable1« ist vom Typ »CHAR«, »Variable2« vom Typ »CARDINAL«. »CHR« liefert das Zeichen, das den ASCII-Wert »Variable2« besitzt.

DEC(Variable, Wert);

Variable ist vom Typ »INTEGER, CARDINAL, CHAR« oder ein Aufzählungstyp. »DEC« vermindert Variable um den angegebenen Wert. Wert kann auch weggelassen werden. Somit wird Variable um eins verringert.

EXCL(Menge, Element);

Menge ist eine Menge, Element ist ein Element dieser Menge. »EXCL« entfernt das angegebene Element aus der angegebenen Menge.

Variable1:=FLOAT(Variable2);

»Variable1« ist vom Typ »REAL«, »Variable2« vom Typ »CARDINAL« oder »INTEGER«. »FLOAT« liefert die Realzahl von »Variable2«.

Halt;

Beendet das Programm. Sie sollten lieber die Prozeduren des Moduls Arts verwenden, da diese flexibler und leistungsfähiger sind.

Variable:=HIGH(Feld);

Variable ist vom Typ »CARDINAL«, »Feld« ist ein Feld. »HIGH« liefert den maximalen Indexwert von »Feld«.

INC(Variable, Wert);

Variable ist vom Typ »INTEGER, CARDINAL, CHAR« oder

ein Aufzählungstyp. »INC« erhöht Variable um den angegebenen Wert. Wert kann auch weggelassen werden. Somit wird Variable um eins erhöht.

INCL(Menge, Element);

Menge ist eine Menge, Element ist ein Element dieser Menge. »EXCL« fügt das angegebene Element in die angegebene Menge ein.

Variable:=MAX(Typ);

»Typ« ist ein Typ, »Variable« eine Variable dieses Typs. »MAX« liefert die obere Grenze, die eine Variable dieses Typs annehmen kann.

Variable:=MIN(Typ);

»Typ« ist ein Typ, »Variable« eine Variable dieses Typs. »MIN« liefert die untere Grenze, die eine Variable dieses Typs annehmen kann.

Variable1:=ODD(Variable2);

»Variable1« ist vom Typ »BOOLEAN«, Variable vom Typ »INTEGER« oder »CARDINAL«. »ODD« liefert »TRUE«, wenn »Variable2« durch zwei teilbar ist. Sonst liefert »ODD« »FALSE«.

Variable:=ORD(Element);

Variable ist vom Typ »CARDINAL«, Element vom Typ »INTEGER«, »CARDINAL, CHAR« oder ein Aufzählungstyp. »ORD« liefert die Ordnungszahl von Element.

Variable1:=SIZE(Variable2);

»Variable1« ist vom Typ »CARDINAL«, »Variable2« ist ein beliebiger Typ. »SIZE« liefert die Anzahl an Bytes, die »Variable2« im Speicher belegt.

Variable1:=TRUNC(Variable2);

»Variable1« ist vom Typ »INTEGER« oder »CARDINAL«, »Variable2« vom Typ »REAL«. »TRUNC« schneidet die Nachkommastellen von »Variable2« ab und liefert in »Variable1« das ganzzahlige Ergebnis.

Element:=VAL(Typ, Variable1);

»Element« ist ein Element einer Menge, »Typ« ist vom Typ »INTEGER, CARDINAL, CHAR« oder ein Aufzählungstyp. »Variable1« ist die Ordnungszahl eines Elementes des angegebenen Typs. »VAL« liefert das zu »Variable2« angegebene Element des angegebenen Typs.

die Prozedur kann also nicht durch Ändern der Variablen »n« den Wert der Variablen »Anzahl« beeinflussen. Die Variable innerhalb der Prozedur darf sogar den gleichen Namen haben wie die Variable, die übergeben wird. So darf beispielsweise der Prozedur »TextAusgabe« die Variable »n« übergeben werden (diese muß vorher natürlich deklariert und initialisiert sein). Innerhalb der Prozedur darf »n« beliebig verändert werden, ohne daß die Variable »n« des aufrufenden Programms verändert wird. Die Variable »n« der Prozedur ist also eine vollkommen andere, als die des aufrufenden Programms. Für jede Variable wird ein Speicherplatz reserviert. Wird nun einer Variablen einer Prozedur ein Wert zugewiesen, so wird ein neuer Speicherbereich für den Wert dieser Variablen reserviert. Wichtig ist nur, daß Sie wissen, daß durch Ändern eines Wertes einer Prozedur nicht der entsprechende Werteparameter des Hauptprogramms beeinflusst wird. Dieser behält immer bis zum Ende des Prozedurablaufes seinen Wert bei. Manchmal ist es aber gerade erwünscht, daß eine Prozedur auch eine Variable des Hauptprogramms verändern kann. Dies muß dann aber explizit angegeben werden, und zwar dadurch, daß vor den übergebenen Parametern ein »VAR« steht. Schauen wir uns zuerst einmal ein Beispiel in Listing 4 an. Die Prozedur »Swap«, die hier interessant ist, vertauscht den Inhalt zweier Integer-Variablen. Schauen wir uns diese Prozedur etwas genauer an, denn hier finden wir gleich zwei Neuigkeiten. Zuerst folgt wieder das Befehlswort »PROCEDURE« gefolgt von dem Namen der Prozedur, der hier »Swap« (engl.: tausche) heißt. Dann folgt die Klammer mit den zu übergebenden Parametern. Hier fällt zuerst auf, daß vor der Angabe der Variablen das Befehlswort »VAR« gestellt ist. Dies bedeutet nun, daß ein Ändern der Variablen in der Prozedur nun auch die Variablen im Hauptprogramm beeinflusst. Genauer genommen heißt das, daß über den angegebenen Variablennamen die übergebene Variable verändert wird. Somit steht der hier angegebene Variablenname für

die übergebene Variable im Hauptprogramm. Das wird gleich noch etwas deutlicher, wenn wir das Hauptprogramm zusammen durchgehen. Eine weitere Neuigkeit ist die Tatsache, daß dieser Prozedur nicht nur eine Variable, sondern gleich zwei übergeben werden. Die Angabe der Variablen geschieht dabei wie bei der Variablendeklaration. Sie werden durch ein Komma voneinander getrennt. Der Prozedur »Swap« werden also zwei Integer-Werte übergeben. Zuerst wird eine Variable namens »dummy« definiert. Sie ist auch vom Typ »INTEGER«. Im Programmcode wird dieser Variablen zuerst der Wert der Variablen »i« zugewiesen. Anschließend erhält »i« den Wert von »j«. Damit »j« nun noch den Wert von »i« erhält, wird »j« der Wert von »dummy« zugewiesen, der ja bekanntlich »i« ist. Somit sind die beiden Variablen vertauscht worden, die Prozedur ist beendet. Schauen wir uns nun einmal an, wie die Prozedur vom Hauptprogramm aus aufgerufen wird. Zuerst wird nach der Eingabe zweier Variablen gefragt, die »Zahl1« und »Zahl2« heißen. Nach der Ausgabe des Hinweises, daß die Variablen nun vertauscht werden, wird die Prozedur »Swap« aufgerufen. Als Parameter werden die beiden Variablen »Zahl1« und »Zahl2« angegeben. Sie werden durch ein Komma getrennt. Beide müssen vom Typ »INTEGER« sein, da dies in der Prozedurdefinition so angegeben ist. Wäre dies nicht der Fall, würde der Compiler einen Fehler ausgeben. Beim Aufruf der Prozedur »Swap« wird der Variablen »i« der Wert von »Zahl1«, und der Variablen »j« der Wert von »Zahl2« zugewiesen. Da es sich hier um Variablenparameter handelt, was durch das »VAR« gekennzeichnet wird, werden die Variablen »Zahl1« und »Zahl2« immer abgeändert, wenn auch »i« oder »j« abgeändert wird. Wird beispielsweise »i« um eins erhöht, erhöht sich auch der Wert von »Zahl1« um eins, wird »j« um zwei verringert, verringert sich auch »Zahl2« um zwei. Im ersten Befehl der Prozedur »Swap« wird der Variablen »dummy« der Wert von »Zahl1« zugewiesen. Dies beeinflusst

weder »Zahl1« noch »Zahl2«. Dann wird im zweiten Befehl der Variablen »j« der Wert von »i« zugewiesen. Somit erhält »Zahl1« nun den Wert von »Zahl2«, da »j« ja den Wert von »Zahl2« besitzt und beim Abändern von »i« auch »Zahl1« abgeändert wird. Anschließend erhält »j« den Wert von »dummy«, also »Zahl2« den Wert, den »Zahl1« beim Aufruf der Prozedur hatte. Wir mußten diesen Wert in der Variablen »dummy« zwischenspeichern, da »i« (und somit auch »Zahl1«) ja in der Prozedur verändert werden. Dann ist die Prozedur beendet, und die Werte der beiden Variablen »Zahl1« und »Zahl2« wurden ausgetauscht. Anschließend werden im Hauptprogramm die Werte der beiden Variablen ausgegeben. Dann wird das Programm beendet. Halten wir erst einmal fest, was Sie bisher gelernt haben:

- Werden einer Prozedur Variablenparameter übergeben, was durch ein vorangestelltes »VAR« gekennzeichnet wird, so bedeutet ein Verändern der Variablen auch gleichzeitig ein Verändern der übergebenen Variablen. Die Variablenparameter sind also quasi nur Platzhalter für die übergebenen Variablen.
- Einer Prozedur werden mehrere Parameter übergeben, indem man sie durch Kommata trennt. Bei der Übergabe von Werteparametern werden für die Variablen neue Speicherplätze reserviert, in denen dann die Werte der Variablen gespeichert werden. Somit können durch deren Änderung nicht die Werte der übergebenen Variablen verändert werden, da diese einen anderen Speicherbereich besitzen. Dies ist bei Variablenparametern nicht mehr der Fall. Hier wird der Variablen der Prozedur der Speicherplatz zugewiesen, den die übergebene Variable besitzt. Somit bedeutet ein Abändern dieses Wertes in der Prozedur auch ein Ändern der übergebenen Variablen, da deren Wert in derselben Speicherstelle abgespeichert wird. Eben habe ich behauptet, daß es möglich ist, mehrere Parameter zu übergeben. Man kann dieses Spiel fast beliebig weiter treiben. So müssen die übergebenen Parameter nicht immer desselben Typs sein,

sondern können durchaus unterschiedliche Typen besitzen. Einer Prozedur können sowohl Werte- als auch Variablenparameter übergeben werden. Eine solche Prozedurdeklaration sähe dann beispielsweise so aus:

```
PROCEDURE
VieleParameter(VAR i, j : INTEGER; VAR k : LONGINT; l, m : CARDINAL);
```

Der Prozedur »VieleParameter« müssen in diesem Falle fünf Parameter übergeben werden. Die ersten beiden müssen vom Typ »INTEGER« sein. Hierbei handelt es sich um Variablenparameter, was durch das vorangestellte »VAR« deutlich wird. Beim dritten Parameter handelt es sich ebenfalls um einen Variablenparameter. Diesmal muß jedoch eine Variable des Typs »LONGINT« übergeben werden. Der vierte und fünfte Parameter muß vom Typ »CARDINAL« sein. Hierbei handelt es sich allerdings nur um Werteparameter, da hier das Befehlswort »VAR« fehlt. Beachten Sie, daß zwischen den einzelnen Parameterangaben Semikolons stehen. Sind die Variablen allerdings gleichen Typs und gleicher Parameterart, so werden sie nur durch Kommata getrennt. Die Deklaration geschieht also im Grunde genauso wie beim Deklarieren von Variablen im Hauptprogramm. Ein Aufruf der Prozedur sähe dann beispielsweise so aus:

```
VieleParameter(Int1, Int2, LongInt1, Card1, Card2);
```

wobei »Int1« und »Int2« vom Typ »INTEGER«, »LongInt1« vom Typ »LONGINT« und »Card1« vom Typ »CARDINAL« sein müßten. Das Abändern der Variablen »i«, »j«, »k« in der Prozedur beeinflusst somit »Int1«, »Int2« und »LongInt«, »Card1« und »Card2« werden nicht durch Ändern von »l« beziehungsweise »m« beeinflusst. Beachten Sie noch, daß beim Aufruf der Prozedur kein Semikolon die einzelnen Parameter trennt, sondern ein Komma. Hier noch einmal die Unterschiede zwischen den beiden Parameterarten:

- Werteparameter nennt man Parameter, die nicht innerhalb einer Prozedur durch Abändern der entsprechenden Variablen verändert werden können.

- Variablenparameter können innerhalb einer Prozedur durch Abändern der entsprechenden Variablen verändert werden. Variablenparameter werden durch ein vorangestelltes »VAR« gekennzeichnet.
- Mehrere Variablen desselben Typs werden in der Prozedurdefinition durch Kommata getrennt angegeben.
- Variablen unterschiedlichen Typs oder bei unterschiedlicher Parameterart werden durch Semikolon getrennt.
- Alle Parameter werden beim Aufruf einer Prozedur durch Kommata getrennt angegeben, unabhängig von deren Typ und Parameterart.

Funktionsprozeduren

Schauen wir uns Listing 5 einmal an. Der Prozedur »Fak« müssen zwei Parameter übergeben werden. Die erste Zahl stellt die Zahl dar, deren Fakultät berechnet werden soll. Hierbei handelt es sich um ei-

nen Werteparameter, da der Wert der Zahl nicht geändert werden soll.

Der zweite Parameter ist ein Variablenparameter. In ihr wird das Ergebnis der Berechnung gespeichert. Beide Parameter sind vom Typ »LONGCARD«. Die Prozedur berechnet nun die Fakultät von Zahl und speichert das Ergebnis in »Erg« ab. Im Hauptprogramm wird die Prozedur mit zwei Variablen aufgerufen. Wurde die Prozedur abgearbeitet, so befindet sich in Ergebnis das Resultat der Rechnung. Dies wird dann ausgegeben. Sie sehen, eigentlich spricht nichts gegen die Realisierung solch einer Funktion. Sie ist sauber programmiert und arbeitet einwandfrei. Trotzdem sollte man für die Realisierung dieses Problems lieber eine Funktionsprozedur verwenden. Eine Funktionsprozedur ist eine Prozedur, welche einen Wert zurückgibt. Dies ist ähnlich wie in der Mathematik, wo eine Funktion einem x-Wert den zugehörigen y-Wert zuordnet.

Genauso funktioniert eine Funktionsprozedur.

Sie ordnet einem oder mehreren Parametern ein Ergebnis zu. Somit eignet sich dieser Prozedurtyp hervorragend zur Realisierung der Prozedur »Fakultaet«. Wie sieht eine Funktionsprozedur aus? Sie wird genauso definiert wie jede andere Prozedur, lediglich eine Erweiterung wird vorgenommen. Hinter der Klammer im Prozedurkopf folgt nun eine Anweisung, die angibt, von welchem Typ das Ergebnis ist. Der Prozedurkopf hat nun folgendes Aussehen:

```
PROCEDURE Name(Parameter :
Typ) : Ergebnistyp;
```

Sie sehen, hinter dem Doppelpunkt wird also angegeben, welchen Typ das Ergebnis hat. Hier muß kein Variablenname angegeben werden. Natürlich muß die Prozedur auch wissen, welchen Wert sie als Ergebnis liefern soll. Dies geschieht durch eine neue Anweisung, die immer am Ende einer Funktionsprozedur ste-

hen muß. Dies ist die Anweisung »RETURN«. Ihr folgt in Klammern die Angabe einer Variablen, deren Wert als Ergebnis dem Hauptprogramm zurückgegeben wird. Der Typ der Variablen muß natürlich dem Typ entsprechen, der im Prozedurkopf hinter den Parametern angegeben wird. Der Aufruf einer Funktionsprozedur sieht, verglichen mit dem Aufruf einer Prozedur, etwas anders aus. Eine Funktionsprozedur wird wie ein Operator gehandhabt. Der Aufruf einer Funktionsprozedur hat also folgendes Aussehen:

```
Ergebnis:=FunktionsProzedur(Parameter);
```

Es findet also eine Zuweisung an eine Variable statt. Diese Variable muß natürlich dem Rückgabotyp der Funktionsprozedur entsprechen, ansonsten bricht der Compiler mit einer Fehlermeldung ab. Beachten Sie, daß der Parameter nicht denselben Typ haben muß wie das Ergebnis. So ist es möglich, die Fakultät eines

WISSEN IST MACHT

AMIGA Köln 90

Auf geht's zu den Masterclasses der

Hier eine Übersicht über das laufende Programm

ANIMATION

- Erfried Prenissl, H. Czerny
- Von der Idee zum fertigen Film
 - Planung, Konzeption und Durchführung komplexer Animationen
 - Unterschiede, Möglichkeiten und Einsatzgebiete von D'Paint III Anim., Sculpt Animate 4d, Turbo Silver, Tips und Tricks

Uhrzeit
Fr. 14:00-17:00
Sa. 10:00-13:00
So. 14:00-17:00

GRAFIK UND ANIMATION

- Gabriele Lechner, Walter Friedhuber
- Produktion eines Werbespots mit Deluxe Paint III
 - Konstruktion von 3D-Objekten in Turbo Silver
 - 3D-Techniken mit "Imagine"
 - Tips und Tricks in Kameraführung, Trickfilm-inszenierung, u. a.

Uhrzeit
Fr. 10:00-13:00
Sa. 14:00-17:00
So. 10:00-13:00

Preis pro Masterclass: DM 70, – im Vorverkauf, DM 80, – an der Kasse

Köln Messe, 8.-11. November 1990

MUSIK/MIDI

- Chris Hülsbeck, Peter Thierolf
- Die Komposition von Musik für professionelle Spiele
 - Umgang mit einem Editor für Musikdaten und interne Amiga-Sounds
 - Einbinden des Amiga in den MIDI-Datenverbund
 - Kompositionstechniken mit einem MIDI-Sequencer
 - Praktische Beispiele

Uhrzeit
Fr. 14:00-17:00
Sa. 10:00-13:00
So. 14:00-17:00

VIDEO

- Jürgen Schneider
- Amiga-Konfiguration im Videobereich (Videoequipment, Soft- und Hardware)
 - Elemente der Videoproduktion
 - Beispiele von Studioproduktionen
 - Anwendungen für Fortgeschrittene

Uhrzeit
Fr. 10:00-13:00
Sa. 14:00-17:00
So. 10:00-13:00

Coupon bis spätestens 10. Oktober einsenden an: AMI Shows Europe GmbH, Zugspitzstr. 2a, 8011 Vaterstetten. Ich bestelle im voraus und spare damit DM 10, – pro Masterclass

Name _____ Anschrift _____ Telefon _____

Den gewünschten Kurs und die Anzahl der Karten markiere ich im dafür vorgesehenen Kästchen mit der entsprechenden Ziffer. Da ich im voraus bezahlen muß, lege ich einen V-Scheck über DM _____ bei.

Uhrzeit	Freitag, 9.11.90		Samstag, 10.11.90		Sonntag 11.11.90	
10:00-13:00	☐ Grafik	☐ Video	☐ Grafik/Animation	☐ Musik/MIDI	☐ Grafik	☐ Video
14:00-17:00	☐ Grafik/Animation	☐ Musik/MIDI	☐ Grafik	☐ Video	☐ Grafik/Animation	☐ Musik/MIDI

Integer-Wertes zu berechnen und das Ergebnis als Cardinal-Wert zurückzugeben. Schauen wir uns anhand von Listing 6 einmal an, wie man die Berechnung der Fakultät einer Zahl mit Hilfe einer Funktionsprozedur realisiert. Die Prozedur »Fak« erhält nur noch einen Parameter, und zwar die Zahl, deren Fakultät berechnet werden muß. Dies muß ein Longcard-Wert sein. Als Ergebnis soll ebenfalls ein Longcard-Wert zurückgegeben werden. Dazu dient die Erweiterung nach der Parameterliste. Dann werden die Variablen »i« und »Erg« deklariert. In »Erg« wird das Ergebnis gespeichert. Der Programmteil der Prozedur funktioniert von der Berechnung her wie bisher. Neu ist der letzte Befehl am Ende der Prozedur. Wie bereits gesagt, muß eine Prozedur ja mit dem Befehl »RETURN« beendet werden. Die Variable »Erg« enthält das Ergebnis der Berechnung. Also muß deren Wert zurückgegeben werden. Dies geschieht mit dem letzten Befehl »RETURN(Erg)«. »Erg« wird als Ergebnis dem aufrufenden Programm übergeben. Schauen wir uns nun noch den Aufruf der Funktionsprozedur an. Er findet in der Zeile

```
WriteCard(Fak(Zahl),1);
```

statt. »Zahl« wird »Fak« als einziger Parameter übergeben. Daraufhin wird das Ergebnis berechnet und dem Befehl »WriteCard« übergeben, der diesen Wert direkt ausgibt. Wir hätten das gleiche auch in zwei Befehlen erreichen können. Nehmen wir an, eine Variable namens »Ergebnis« vom Typ »LONGINT« wäre definiert, so würden die beiden Befehle

```
Ergebnis:=Fak(Zahl); WriteCard(Ergebnis,1);
```

das gleiche bewirken. Der Variablen »Ergebnis« würde das Ergebnis der Funktionsprozedur übergeben und anschließend würde dieser Wert ausgegeben. Die Realisierung dieses Problems mit einer Funktionsprozedur ist wesentlich eleganter und eindeutiger. So weiß der Benutzer sofort, daß eine Funktionsprozedur einen Wert zurückgibt. Außerdem kann man Programmcode sparen, da eine Funktionsprozedur als Operator eingesetzt werden kann. Allerdings lassen sich Funktions-

prozeduren nur verwenden, wenn **ein** Wert zurückgegeben wird. Soll eine Prozedur mehr als einen Wert zurückgeben, so müssen Sie eine Prozedur mit mehreren Variablenparametern verwenden.

Lokal oder global?

Zuerst möchte ich den Begriff 'sichtbar' erläutern. Ist eine Variable sichtbar, so bedeutet dies, daß sie dem Programm bekannt ist. Man kann also mit einer sichtbaren Variablen Berechnungen ausführen oder deren Wert ausgeben. Dies ist nicht immer selbstverständlich.

Entsprechend versteht man unter dem Sichtbarkeitsbereich den Bereich, in dem eine Variable sichtbar ist. Dieser Bereich umfaßt in der Regel eine Ansammlung von Befehlen, beispielsweise eine Prozedur oder das Hauptprogramm. Soviel zu diesen beiden Begriffen. Kommen wir nun zu den Begriffen lokal und global. Die Attribute lokal beziehungsweise global sagen aus, ob eine Variable nur in der aktuellen Prozedur sichtbar ist oder auch in der aufrufenden.

Schauen wir uns dazu am besten einmal ein Beispiel (Listing 7) an. Im Hauptprogramm werden zwei Variablen deklariert. Dies sind die Variablen »i« und »j«. Ihnen wird im Hauptprogramm jeweils der Wert »eins« zugeordnet. Diese Werte werden dann im Hauptprogramm erst einmal ausgegeben. Dann wird die Prozedur »Proz1« aufgerufen. Ihr wird ein Parameter übergeben. Dieser Parameter heißt hier »a«. »a« wird der Wert der Variablen »i« zugeordnet. Weiterhin werden die Variablen »i« und »b« deklariert. Ihnen wird im Anweisungsteil dieser Prozedur jeweils der Wert »2« zugewiesen. Dann wird eine weitere Prozedur namens »Proz2« aufgerufen. Ihr wird ebenfalls ein Parameter übergeben, der innerhalb der Prozedur ebenfalls »a« heißt. »a« bekommt den Wert von »i« zugewiesen. In »Proz2« werden weiterhin die Variablen »i« und »c« deklariert. Sie erhalten den Wert »3«. Diese Werte werden dann ausgegeben.

Anschließend wird wieder zur übergeordneten Prozedur (»Proz1«) zurückgekehrt, welche wiederum direkt zum

Hauptprogramm zurückkehrt. Wie bereits gesagt, erhalten die Variablen »i« und »j« im Hauptprogramm den Wert »1«. Dann wird die Prozedur »Proz1« aufgerufen. Da das Hauptprogramm die Prozedur aufruft, sind die Variablen »i« und »j« sichtbar, sie sind also relativ zur Prozedur »Proz1« global. Doch halt, die Variable »i« wird in »Proz1« ja noch einmal deklariert. Dies ist durchaus möglich.

Der Compiler merkt sich nun den Wert, den die globale Variable »i« hatte und erzeugt eine neue Variable »i«, die aber nur in dieser Prozedur sichtbar ist. Wird nun »i« abgeändert, so beeinflußt das nur die lokale Variable »i«, nicht aber die globale. Wird dagegen »j« abgeändert, so bezieht sich dieses Änderung auf die globale Variable »j«. Im Hauptprogramm hätte nun »j« einen neuen Wert erhalten. Des weiteren wird »Proz1« noch die Variable »a« übergeben und die Variable »b« deklariert. Diese beiden Variablen sind im Hauptprogramm nicht sichtbar, aber in der aktuellen Prozedur.

Sie sind somit lokal. Dies erklärt nun auch die Ausgaben der Prozedur »Proz1«. »i« wurde der Wert »2« zugeordnet (dies ist jedoch nur lokal, die globale Variable »i« des Hauptprogramms besitzt immer noch den Wert »1«), »j« besitzt noch denselben Wert wie im Hauptprogramm. »a« und »b« wurden ebenfalls der Wert »2« zugewiesen, einmal in der Prozedur und einmal durch Parameterübergabe. Die Prozedur »Proz1« ruft nun »Proz2« auf.

Sie erhält ebenfalls einen Parameter, der auch hier wieder »a« heißt. Somit wird eine neue Variable »a« deklariert, die nur in dieser Prozedur sichtbar ist. Kehrt das Programm wieder zu der übergeordneten Prozedur zurück, so hat »a« der »Proz1« wieder seinen alten Wert. Weiterhin werden wieder »i« und »c« deklariert. »c« ist ganz neu, diese Variable ist nur in dieser Prozedur sichtbar. Würde man versuchen, in »Proz1« die Variable »c« auszugeben, so würde der Compiler melden, daß »c« nicht sichtbar ist. Das liegt daran, daß »c« erst in der Prozedur »Proz2« deklariert wird und auch nur dort sichtbar bleibt. Die Variable »i« wird erneut deklariert, und ihr wird der Wert »3« zugewiesen. Sie

ist lokal, also nur in »Proz2« sichtbar. In »Proz1« ist »i« zwar auch sichtbar, jedoch handelt es sich hier um eine ganz andere Variable »i«. Sie hat den Wert »2«. Ändert man »i« in »Proz2« ab, so wird »i« in »Proz1« nicht beeinflußt (Wert: »2«), genau wie »i« aus dem Hauptprogramm (Wert: »1«). Nun werden in dieser Prozedur die Werte »i,j« und »a,b« und »c« ausgegeben. Doch halt, »j« und »b« wurden doch gar nicht neu deklariert. Das ist zwar richtig, doch handelt es sich hierbei um globale Variablen. »j« stammt aus dem Hauptprogramm und hat immer den Wert »1«. Würde man diese Variable in »Proz2« verändern, so würde sich das auch auf das Hauptprogramm auswirken. »b« wurde in »Proz1« deklariert. Diese Prozedur hat »Proz2« erst aufgerufen. Somit ist »b« relativ zu »Proz2« global. Eben habe ich gesagt, daß »b« aber in »Proz1« lokal ist. Auch dies ist richtig.

Sie ist immer relativ. So ist »b« in »Proz1« relativ zum Hauptprogramm lokal, aber relativ zu »Proz2« global, da »Proz2« eine Prozedur von »Proz1« ist. Somit ergeben sich die Werte, die dann in »Proz2« ausgegeben werden. Anschließend kehrt die Prozedur »Proz2« wieder zur Prozedur »Proz1« zurück.

Überlegen Sie nun einmal, welche Werte »i,j,a,b« und »c« haben. »c« ist nicht mehr sichtbar, da »c« nur in »Proz2« sichtbar ist. »a« hat den Wert »2«, da »a« in »Proz2« ebenfalls lokal ist. Somit ist »a« aus »Proz2« nicht mehr sichtbar, dafür aber »a« aus »Proz1«, und das hat den Wert »2«. »b« hat ebenfalls den Wert »2«, da »Proz2« »b« nicht verändert hat. Wäre dies der Fall, so hätte »b« auch hier den neuen Wert angenommen. »j« hat wieder den Wert »1«. Hier gilt das gleiche wie für »b«. »i« hat den Wert »2«. Hier gilt das gleiche wie für »a«. Die Prozedur »Proz1« kehrt letztendlich auch wieder zum Hauptprogramm zurück. Nun sind die Variablen »b« und »a« ebenfalls nicht mehr sichtbar, da sie nur in Prozedur »Proz1« deklariert und somit sichtbar waren. Nur noch die Variablen »i« und »j« sind sichtbar. »i« hat wieder den Wert »1«, da »i« in »Proz1« relativ zum Hauptprogramm lokal war und diese Prozedur nun beendet ist. So-

mit ist nicht mehr »i« aus der »Proz1« sichtbar, sondern »i« aus dem Hauptprogramm, welches den Wert »1« besitzt. Haben Sie alles verstanden? Hier noch einmal einige Regeln:

– Die Begriffe 'lokal' und 'global' sind relativ. So kann eine Variable durchaus lokal und global zugleich sein.

– Eine Variable ist immer in einer untergeordneten Prozedur sichtbar, also zu ihr global. Eine untergeordnete Prozedur ist eine Prozedur, die innerhalb einer anderen Prozedur (die übergeordnete) definiert wird. In Listing 7 ist »Proz2« »Proz1« untergeordnet.

– Wird in einer untergeordneten Prozedur eine Variable deklariert, die bereits in der übergeordneten sichtbar ist, so wird die Variable der übergeordneten Prozedur so lange nicht mehr sichtbar, bis die untergeordnete Prozedur beendet wurde.

Die Variable der übergeordneten Prozedur behält dabei ihren Wert bei, unabhängig von dem Wert der untergeordneten Prozedur. Wenn Sie diese drei Regeln immer im Kopf haben, dürfte es Ihnen nicht allzu schwer fallen, zu beurteilen, wann eine Variable welchen Wert hat. Vielleicht werden Sie sich fragen, warum es so wichtig ist, ob eine Variable nun lokal oder global ist. Dies ist deshalb sehr wichtig, da dadurch bestimmt wird, wann der Wert einer Variablen den Wert einer übergeordneten Variablen mit demselben Namen beeinflusst. Dies ist immer dann möglich, wenn die Variable relativ zu

der übergeordneten Prozedur global ist. Eine lokale Variable beeinflusst eine übergeordnete Variable nie! Eine Variable kann nur global zu ihrer übergeordneten Prozedur sein. Eine Prozedur ist dann übergeordnet, wenn in ihr eine andere Prozedur definiert wird. Wird in dieser Prozedur wieder eine Prozedur definiert, so ist sie zu dieser neu definierten übergeordnet, zu der, in der sie aber definiert wurde, untergeordnet. Schauen wir uns dazu einmal Listing 8 an. Es sollte eigentlich genau das gleiche verrichten wie Listing 7.

Der Compiler wird das Programm nicht definieren. Er wird abbrechen mit der Meldung, daß die Variable »b« in der Prozedur »Proz2« nicht sichtbar ist. »Proz1« ruft zwar »Proz2« auf, und in diesem Falle wäre »b« dann auch sichtbar, aber leider ist die Prozedur »Proz1« der Prozedur »Proz2« nicht übergeordnet. Somit ist »b« in »Proz2« nicht sichtbar, und der Compiler gibt einen Fehler aus. Woher soll er auch wissen, daß »Proz2« von »Proz1« aufgerufen wird, und »b« dann sichtbar wäre? Beachten Sie, daß die Prozedur »Proz2« vor der Prozedur »Proz1« definiert werden muß, da »Proz1« diese aufruft. Wäre dies nicht der Fall, so würde der Compiler mit der Fehlermeldung abbrechen, daß die Prozedur »Proz2« nicht sichtbar ist, da sie erst später definiert wird. Sie sehen, auch Prozeduren, ja sogar Module können sichtbar und nicht sichtbar sein!

(tb)

Listings

```
1: MODULE ProzedurDemo;
2:
3: FROM InOut IMPORT WriteString, WriteLn;
4:
5: PROCEDURE InfoText;
6: BEGIN
7:   WriteString("Dieses Programm wurde von A. Polk fr
8:               AmigaDOS");
9:   WriteLn;
10:  WriteString("geschrieben. Es demonstriert den
11:              Gebrauch einer Prozedur.");
12:  WriteLn;
13: END InfoText;
14:
15: BEGIN
16:   InfoText;
17: END ProzedurDemo.
```

```
1: MODULE TextFenster;
2:
3: (* Importliste *)
4:
5: VAR Taste : CHAR;
6:
```

Listing: Modula2-Kurs



**AMIGA -- BTX
MULTITERM
!!! PRO !!!**

Der absolute BTX Software-Decoder.

**Kopierschutz passe', jetzt kommt:
Project D
Disketten Utility der Sonderklasse**

Das Original Kopierprogramm direkt aus USA.
Mit Update-Service.

Preise:

Multiterm an A-Koppler / Modem	125,-DM	incl. deutscher Anleitung	80,-DM
Multiterm Pro	149,-DM		
Multiterm an Post-Modem DBT03	195,-DM	DiGi-View 4.0	298,-DM
Multiterm Pro	229,-DM	Pagestream 1.8	339,-DM
Sep. Interface Amiga-DBT-03 nur	75,-DM	Pagesetter II	189,-DM
		weitere Software zu Superpreisen !!!!	
		Kickstart ROM 1.3	59,-DM

Das richtige Modem !!!!

Best 2400 Plus (Btx-fähig)*	429,-DM	Neuheiten !!!
Discovery 2400C*	359,-DM	Interne PC-Karte f. Amiga 500
U.S. Robotics HST 14.400*	1560,-DM	Interne 20 MB Festplatte
U.S. Robotics Dual Standard*	2390,-DM	

Weitere Modelle Robotics/Lightspeed etc. auf Anfrage
*Betrieb am Postnetz der BRD und West-Berlin bei Strafe verboten.
512K Speicher mit Uhr f. A500 189,-DM
Info über MS-DOS Rechner und weitere Neuheiten anfordern.

Für XT / AT-Karten

Flugsimulator 4.0
Die neueste Version von Microsoft.
Direkt aus USA. 129,-DM

M. Kirschbaum Medienberatung

Schubertstr. 3, 4320 Hattingen

Tel. + Btx: 02324/82249 --- Fax: 02324/83722

MVC

Musik Video Computer

**Public-Domain-
Dschungel?
Nicht bei uns!!!**

Fordern Sie unsere 4 deutschen Katalogdisketten an (DM 6,- in Briefmarken).

Alle gängigen Serien auf Lager! Natürlich topaktuell!

Und jetzt der Hammer: Jede 3,5"-Disk.

nur DM 1,90

Jede 5 1/4"-Disk.

nur DM 1,10

Sonderserien plus DM 0,60

Wir liefern Ihren Auftrag spätestens 1 Tag nach Eingang aus!!!

Daß wir nur mit Verify kopieren, ist für uns selbstverständlich!

Testen Sie uns doch einfach!!!

MVC

Musik Video Computer – Alles für und mit AMIGA
Hammer Str. 103 · 4730 Ahlen · Telefon/BTX 023 82/25 03
Telefax 023 82/25 04

Ordnung und Übersicht schaffen die beliebten DMV Sammelmappen



Bitte Bestellkarte benutzen

DMV Verlag · Postfach 250 · 3440 Eschwege


```

7: (* Variablendeklaration, Typen- und Konstantenfest-
   legung etc *)
8:
9: PROCEDURE OeffneFenster;
10: BEGIN
11: END OeffneFenster;
12:
13: PROCEDURE SchreibeText;
14: BEGIN
15: END SchreibeText;
16:
17: PROCEDURE WarteAufTaste;
18: BEGIN
19: END WarteAufTaste;
20:
21: PROCEDURE SchliesseFenster;
22: BEGIN
23: END SchliesseFenster;
24:
25: BEGIN
26:   OeffneFenster;
27:   SchreibeText;
28:   REPEAT WarteAufTaste
29:   UNTIL ORD(Taste)=0CH;
30:   SchliesseFenster;
31: END TextFenster.

```

```

1: MODULE Ausgabe;
2:
3: FROM InOut IMPORT WriteString, WriteLn;
4:
5: VAR Anzahl : CARDINAL;
6:
7: PROCEDURE TextAusgabe(n : CARDINAL);
8:
9: VAR i : CARDINAL;
10:
11: BEGIN
12:   FOR i:=1 TO n DO
13:     WriteString("Hallo");
14:     WriteLn;
15:   END;
16: END TextAusgabe;
17:
18: BEGIN
19:   Anzahl:=10;
20:   TextAusgabe(5);
21:   WriteLn;
22:   TextAusgabe(Anzahl);
23: END Ausgabe.

```



```

1: MODULE Swap;
2:
3: FROM InOut IMPORT WriteString, WriteInt, ReadInt,
   WriteLn;
4:
5: VAR Zahl1, Zahl2 : INTEGER;
6:
7: PROCEDURE Swap(VAR i, j : INTEGER);
8:
9: VAR dummy : INTEGER;
10:
11: BEGIN
12:   dummy:=i;
13:   i:=j;
14:   j:=dummy;
15: END Swap;
16:
17: BEGIN
18:   WriteString("Geben Sie die erste Zahl ein: ");
19:   ReadInt(Zahl1);
20:   WriteString("Geben Sie die zweite Zahl ein: ");
21:   ReadInt(Zahl2);
22:   WriteString("Tausche Zahlen");
23:   WriteLn;
24:   Swap(Zahl1, Zahl2);
25:   WriteString("Zahl1 lautet nun: ");
26:   WriteInt(Zahl1, 6);
27:   WriteLn;
28:   WriteString("Zahl2 lautet nun: ");
29:   WriteInt(Zahl2, 6);
30:   WriteLn;
31: END Swap.

```

```

1: MODULE Fakultaet1;
2:
3: FROM InOut IMPORT WriteString, WriteLn, ReadLongCard,
   WriteCard;
4:
5: VAR Zahl, Ergebnis : LONGCARD;
6:
7: PROCEDURE Fak(Zahl : LONGCARD; VAR Erg : LONGCARD);
8:
9: VAR i : LONGCARD;
10:
11: BEGIN
12:   Erg:=1;

```

```

13:   FOR i:=1 TO Zahl DO
14:     Erg:=Erg*i;
15:   END;
16: END Fak;
17:
18: BEGIN
19:   WriteString("Geben Sie eine Zahl ein: ");
20:   ReadLongCard(Zahl);
21:   Fak(Zahl, Ergebnis);
22:   WriteCard(Zahl, 1);
23:   WriteString("! ist ");
24:   WriteCard(Ergebnis, 1);
25:   WriteLn;
26: END Fakultaet1.
27:

```

```

1: MODULE Fakultaet2;
2:
3: FROM InOut IMPORT WriteString, WriteLn, ReadLongCard,
   WriteCard;
4:
5: VAR Zahl : LONGCARD;
6:
7: PROCEDURE Fak(Zahl : LONGCARD) : LONGCARD;
8:
9: VAR i, Erg : LONGCARD;
10:
11: BEGIN
12:   Erg:=1;
13:   FOR i:=1 TO Zahl DO
14:     Erg:=Erg*i;
15:   END;
16:   RETURN(Erg);
17: END Fak;
18:
19: BEGIN
20:   WriteString("Geben Sie eine Zahl ein: ");
21:   ReadLongCard(Zahl);
22:   WriteCard(Zahl, 1);
23:   WriteString("! ist ");
24:   WriteCard(Fak(Zahl), 1);
25:   WriteLn;
26: END Fakultaet2.
27:

```

```

1: MODULE LokGlob1;
2:
3: FROM InOut IMPORT WriteString, WriteInt, WriteLn;
4:
5: VAR i, j : INTEGER;
6:
7: PROCEDURE Proz1(a : INTEGER);
8:
9: VAR b, i : INTEGER;
10:
11:   PROCEDURE Proz2(a : INTEGER);
12:
13:   VAR c, i : INTEGER;
14:
15:   BEGIN
16:     c:=3;
17:     i:=3;
18:     WriteString("i=");
19:     WriteInt(i, 1);
20:     WriteLn;
21:     WriteString("j=");
22:     WriteInt(j, 1);
23:     WriteLn;
24:     WriteString("a=");
25:     WriteInt(a, 1);
26:     WriteLn;
27:     WriteString("b=");
28:     WriteInt(b, 1);
29:     WriteLn;
30:     WriteString("c=");
31:     WriteInt(c, 1);
32:     WriteLn;
33:   END Proz2;
34:
35: BEGIN
36:   i:=2;
37:   b:=2;
38:   WriteString("i=");
39:   WriteInt(i, 1);
40:   WriteLn;
41:   WriteString("j=");
42:   WriteInt(j, 1);
43:   WriteLn;
44:   WriteString("a=");
45:   WriteInt(a, 1);
46:   WriteLn;
47:   WriteString("b=");
48:   WriteInt(b, 1);
49:   WriteLn;
50:   WriteString("Rufe Prozedur 2 auf");
51:   WriteLn;
52:   Proz2(3);
53: END Proz1;
54:

```

Listing: Modula2-Kurs

Listing: Modula2-Kurs


```

55: BEGIN
56:   WriteString("Hauptprogramm");
57:   WriteLn;
58:   i:=1;
59:   j:=1;
60:   WriteString("i=");
61:   WriteInt(i, 1);
62:   WriteLn;
63:   WriteString("j=");
64:   WriteInt(j, 1);
65:   WriteLn;
66:   WriteString("Rufe Prozedur 1 auf");
67:   WriteLn;
68:   Proz1(2);
69:   WriteString("Wieder Hauptprogramm");
70:   WriteLn;
71:   WriteString("i=");
72:   WriteInt(i, 1);
73:   WriteLn;
74:   WriteString("j=");
75:   WriteInt(j, 1);
76:   WriteLn;
77: END LokGlob1.

```



```

1: MODULE LokGlob1;
2:
3: FROM InOut IMPORT WriteString, WriteInt, WriteLn;
4:
5: VAR i, j : INTEGER;
6:
7: PROCEDURE Proz2(a : INTEGER);
8:
9: VAR c, i : INTEGER;
10:
11: BEGIN
12:   c:=3;
13:   i:=3;
14:   WriteString("i=");
15:   WriteInt(i, 1);
16:   WriteLn;
17:   WriteString("j=");
18:   WriteInt(j, 1);
19:   WriteLn;
20:   WriteString("a=");
21:   WriteInt(a, 1);
22:   WriteLn;
23:   WriteString("b=");
24:   WriteInt(b, 1);
25:   WriteLn;
26:   WriteString("c=");
27:   WriteInt(c, 1);
28:   WriteLn;
29: END Proz2;
30:
31: PROCEDURE Proz1(a : INTEGER);
32:
33: VAR b, i : INTEGER;
34:
35: BEGIN
36:   i:=2;
37:   b:=2;
38:   WriteString("i=");
39:   WriteInt(i, 1);
40:   WriteLn;
41:   WriteString("j=");
42:   WriteInt(j, 1);
43:   WriteLn;
44:   WriteString("a=");
45:   WriteInt(a, 1);
46:   WriteLn;
47:   WriteString("b=");
48:   WriteInt(b, 1);
49:   WriteLn;
50:   WriteString("Rufe Prozedur 2 auf");
51:   WriteLn;
52:   Proz2(3);
53: END Proz1;
54:
55: BEGIN
56:   WriteString("Hauptprogramm");
57:   WriteLn;
58:   i:=1;
59:   j:=1;
60:   WriteString("i=");
61:   WriteInt(i, 1);
62:   WriteLn;
63:   WriteString("j=");
64:   WriteInt(j, 1);
65:   WriteLn;
66:   WriteString("Rufe Prozedur 1 auf");
67:   WriteLn;
68:   Proz1(2);
69:   WriteString("Wieder Hauptprogramm");
70:   WriteLn;
71:   WriteString("i=");
72:   WriteInt(i, 1);
73:   WriteLn;
74:   WriteString("j=");
75:   WriteInt(j, 1);
76:   WriteLn;
77: END LokGlob1.

```

Listing: Modula2-Kurs

Haben Sie Fragen zum Thema AMIGA und AMIGA DOS?

Dann rufen Sie doch einfach mal an!

Das AMIGA-DOS-Team freut sich darauf, Ihnen bei Fragen zum AMIGA und zu diesem Magazin Hilfestellung zu leisten.

In diesem Sinne steht Ihnen das AMIGA-DOS-Team an jedem **Dienstag**

in der Zeit von **17.00 bis 20.00 Uhr** zur Verfügung, um Ihre Fragen zu beantworten.

Wählen Sie einfach eine der nachfolgenden Nummern, um direkt den betreffenden Redakteur zu erreichen.



Markus Matejka: 05651/809-741

Jürgen Borngießer: 05651/809-742

Vera Brinkmann: 05651/809-743

Claus Daschner: 05651/809-744

Thomas Baum: 05651/809-740

Wir freuen uns auf Ihren Anruf!

Ihr AMIGA-DOS-TEAM

Andreas Vorgeitz

Push it !!!

Wenn Ihre "grauen Zellen" trotz der sommerlichen Temperaturen noch nicht glühen, unser AmigaBASIC-Listing schafft es bestimmt.

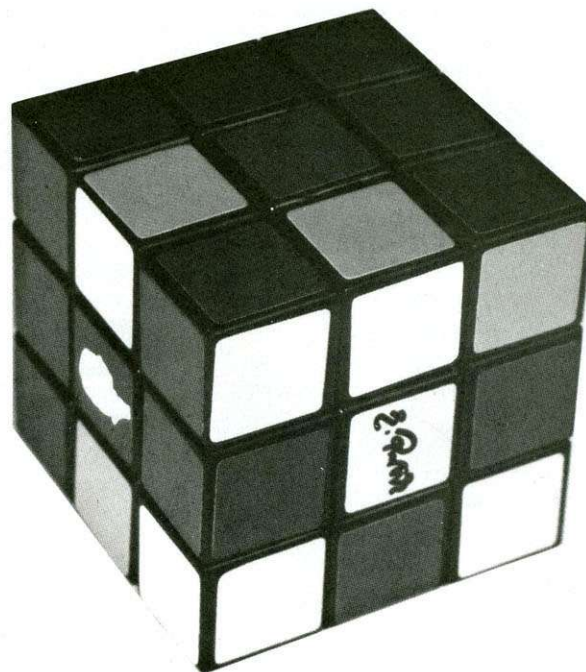
Das Programm »Push it !!!« ist eine grafisch reizvoll gelöste Variante des rubikschen Würfels. Hier gilt es, Farbflächen durch horizontales und vertikales Verschieben so zu plazieren, daß ein vorgegebenes Muster erreicht wird.

Die Vorgabe, die mit jeder Spielstufe schwieriger zu bewältigen ist, wird im rechten, oberen Bildschirmteil angezeigt. Um die Farbflächen zu verschieben, muß zuerst die Zeile beziehungsweise Spalte angewählt werden. Dies geschieht mit einem Pfeilsymbol, das mit Hilfe des Joysticks (Port 2) bewegt werden kann. Eine seitliche Bewegung des Joysticks läßt den Pfeil im Uhrzeigersinn (rechts) oder gegen den Uhrzeigersinn (links) um das Spielfeld rotieren. Drückt man den Joystick nach Anwählen der Zeile oder Spalte nach oben, bewegen sich die Farb-

felder vom Pfeil weg, umgekehrt zum Pfeil hin. Über das Spielfeld hinausgeschobene Flächen werden auf der gegenüberliegenden Seite in das Spielfeld wieder hineingeschoben.

Nerven sparen durch Zwischenspeichern

Hat man nun alle Farbflächen der Vorlage entsprechend angeordnet, drückt man den Feuerknopf. Dadurch gelangt man in den nächsten Level. Wenn noch nicht alle Flächen korrekt geschoben wurden, kann man mit einem Druck auf den Feuerknopf die aktuelle Spielfeldposition in einem Puffer ablegen. Hat man sich gänzlich "verschoben", kann der im Puffer befindliche Spielstand mit [Return] jederzeit zurückgeholt werden.



Das Spiel »Push it !!!« hat 11 verschiedene Spielstufen (Levels). Die Stufen 1 (easy), 4 (medium) und 7 (hard) können zu Spielbeginn durch Drücken der entsprechenden Buchstabetaste ausgewählt werden.

»Push it !!!« ist auf allen Amigas mit mindestens 512 kByte Speicher lauffähig. Zur Eingabe des Listings starten Sie bitte Ihren AmigaBASIC-Interpreter. Dieser befindet sich auf der Extras-Diskette. Geben Sie

das Listing ohne Zeilennummern ein, und speichern Sie das Programm vor dem ersten Starten unbedingt ab. Zum Abspeichern drücken Sie die rechte Maustaste und wählen im »Project«-Menü den Punkt »Save« an. Es öffnet sich ein Fenster, in dem Sie den Namen des Programms eingeben können. Anschließend können Sie über den Menüpunkt »Start« mit der Programmausführung beginnen.

(tb)

Listings

```

1: REM *****
2: REM *          PUSH IT !!!          *
3: REM *          by Andreas Vorgeitz   *
4: REM *          (c) 1990 DMV-Verlag   *
5: REM *          Sprache: AMIGABASIC  *
6: REM *****
7:
8: graphicsetup:
9: SCREEN 1,320,256,4,1
10: WINDOW 2,,,0,1
11: WINDOW OUTPUT 2
12: DIM c(3)
13: DIM bp%(400)
14: DIM sp%(200)
15: DIM pfeil%(1000)
16: DIM P(7,7),pp(7,7),bufp(7,7)
17: DIM z(7,8),s(7,8)
18: FOR d=0 TO 8
19: READ r,g,b:PALETTE d,r/15,g/15,b/15
20: NEXT
21:
22: DATA 5,3,1, 15,13,11, 0,0,0, 15,0,15, 0,15,0
23: DATA 15,15,0, 0,0,15, 15,0,0, 15,7,0
24:
25: c(1)=0:c(2)=15:c(3)=15
26: GOSUB nextpal
27:
28: Bigpiece:
29: LINE (0,0)-(15,15),2,bf
30: LINE (2,2)-(13,13),4,bf
31: LINE (0,15)-(15,20),2,bf
32:
33: x=1:y=1:dir=1:GOSUB colline
34: dir=4:GOSUB colline
35: x=14:y=14:dir=2:GOSUB colline
36: dir=3:GOSUB colline
37:

```

Listing: Push it !!!

38: x=1:dir=1

39: FOR y=17 TO 19

40: GOSUB colline

41: NEXT

42:

43: GET (0,0)-(15,20),bp%

44: CLS

45:

46: smallpiece:

47: LINE (2,2)-(10,10),2,b

48: LINE (3,3)-(8,8),13,bf

49: LINE (0,0)-(8,8),2,b

50: LINE (1,1)-(7,7),6,bf

51:

52: GET (0,0)-(10,10),sp%

53: CLS

54:

55: Pfeile:

56: FOR d=0 TO 3:FOR t=0 TO 2

57: READ x,y

58: AREA (x,y)

59: NEXT

60: AREAfill

61: GET (0,0)-(12,12),pfeil%(d*200)

62: CLS

63: NEXT

64:

65: DATA 0,0,12,0,6,6, 12,0,12,12,6,6

66: DATA 12,12,0,12,6,6, 0,12,0,0,6,6

67:

68: GOSUB ShowTitle

69: ON dif GOTO level1,level4,level7

70:

71: level1:

72: level=1:RESTORE dlevel1

73: c(1)=15:c(2)=4:c(3)=15:steps=6

74: GOSUB makearray

75:

76: level2:

77: c(1)=0:c(2)=15:c(3)=0:steps=25

Listing: Push it !!!




```

78: GOSUB makearray
79:
80: level3:
81: c(1)=15:c(2)=15:c(3)=0
82: GOSUB makearray
83:
84: level4:
85: level=4:RESTORE dlevel4
86: c(1)=15:c(2)=0:c(3)=15:steps=15
87: GOSUB makearray
88:
89: level5:
90: c(1)=0:c(2)=5:c(3)=15
91: GOSUB makearray
92:
93: level6:
94: c(1)=15:c(2)=0:c(3)=5:steps=30
95: GOSUB makearray
96:
97: level7:
98: level=7:RESTORE dlevel7
99: c(1)=5:c(2)=15:c(3)=15:steps=30
100: GOSUB makearray
101:
102: level8:
103: c(1)=15:c(2)=10:c(3)=15:steps=25
104: GOSUB makearray
105:
106: level9:
107: c(1)=0:c(2)=15:c(3)=15
108: GOSUB makearray
109:
110: level10:
111: c(1)=15:c(2)=15:c(3)=15
112: GOSUB makearray
113:
114: level11:
115: c(3)=3
116: GOSUB makearray
117:
118: GOTO ende
119:
120: nextpal:
121: FOR d=0 TO 7:PALETTE d+9,c(1)/15,c(2)/15,c(3)/15
122: FOR t=1 TO 3
123: IF c(t)>3 THEN c(t)=c(t)-2
124: NEXT
125: NEXT
126: RETURN
127:
128: colline:
129: x1=x:y1=y
130: FOR d=0 TO 6
131: FOR t=0 TO 1
132: PSET (x1,y1),d+9
133: IF dir=1 THEN x1=x1+1
134: IF dir=2 THEN y1=y1-1
135: IF dir=3 THEN x1=x1-1
136: IF dir=4 THEN y1=y1+1
137: NEXT
138: NEXT
139: RETURN
140:
141: makearray:
142: buffer = 0:zug = 0
143: GOSUB nextpal
144: FOR pk=0 TO 7:FOR pl=0 TO 7
145: P(pk,pl)=0:pp(pk,pl)=0:bufp(pk,pl)=0
146: NEXT:NEXT
147:
148: READ z
149: IF z=0 THEN END
150:
151: FOR d=1 TO z
152: READ o,oz
153: ON o GOSUB box, linie, punkt
154: NEXT
155: GOTO Start
156:
157: box:
158: FOR t = 1 TO oz
159: READ x1,y1,x2,y2,c:dx=SGN(x2-x1):dy=SGN(y2-y1)
160: FOR x=x1 TO x2 STEP dx
161: FOR y=y1 TO y2 STEP dy
162: IF x=<7 AND x>=0 AND y=<7 AND y>=0 THEN P(x,y)=c
163: NEXT
164: NEXT
165: NEXT
166: RETURN
167:
168: linie:
169:
170: FOR t=1 TO oz
171: READ x1,y1,l,dir,c
172: IF dir=1 THEN dy=-1:dx=0
173: IF dir=2 THEN dy=0:dx=1
174: IF dir=3 THEN dy=1:dx=0
175: IF dir=4 THEN dy=0:dx=-1
176: FOR k=0 TO l-1
177: x=x1+k*dx:y=y1+k*dy
178: IF x=<7 AND x>=0 AND y=<7 AND y>=0 THEN P(x,y)=c
179: NEXT
180: NEXT
181: RETURN
182:
183: punkt:
184: FOR t=1 TO oz
185: READ x1,y1,c
186: IF x1=<7 AND x1>=0 AND y1=<7 AND y1>=0 THEN P(x1,y1)=c
187: NEXT

```

Listing: Push it !!!

```

188: RETURN
189:
190: Start:
191: CLS
192: LOCATE 1,1:PRINT "Level:";level
193: LOCATE 1,21:PRINT "Znge: 0"
194: LOCATE 17,27:PRINT "Puffer:"
195: FOR x=0 TO 7:FOR y=0 TO 8
196: z(x,y)=0:S(x,y)=0
197: NEXT:NEXT
198: FOR x=0 TO 7:FOR y=0 TO 7
199: IF P(x,y)<>0 THEN
200: S(x,S(x,0)+1)=y:S(x,0)=S(x,0)+1
201: z(y,z(y,0)+1)=x:z(y,0)=z(y,0)+1
202: PUT (48+16*x,48+16*y),bp%,PSET
203: LINE (50+16*x,50+16*y)-(61+16*x,61+16*y),P(x,y)+2,bf
204: PUT (200+8*x,16+8*y),sp%,PSET
205: LINE (201+8*x,17+8*y)-(207+8*x,23+8*y),P(x,y)+2,bf
206: pp(x,y)=P(x,y)
207: px=x:py=y
208: END IF
209: NEXT
210: NEXT
211: pz=3
212: GOSUB move
213: GOSUB drawpfeil
214:
215: joywait:
216: FOR zeit=0 TO 75:NEXT
217: IF STICK(3)<>0 THEN joy=STICK(3):GOTO moveplates
218: IF STICK(2)<>0 THEN joy=STICK(2):GOTO movecursor
219: IF STRIG(3)=-1 THEN GOTO testme
220: IF INKEY$=CHR$(13) AND buffer=1 THEN BEEP:GOTO putbuf
221: GOTO joywait
222:
223: move:
224: FOR d=0 TO steps
225: FOR t=0 TO 100:NEXT
226: dir=INT(RND(8)*2)
227: IF dir=0 THEN GOSUB movez:GOTO movenext
228: GOSUB rnds
229: dir=INT(RND(8)*2)
230: IF dir=0 THEN GOSUB sup ELSE GOSUB sdown
231:
232: movenext:
233: NEXT:RETURN
234:
235: movez:
236: GOSUB rndz
237: dir =INT(RND(8)*2)
238: IF dir=0 THEN GOSUB zleft ELSE GOSUB zright
239: RETURN
240:
241:
242: rndz:
243: z=INT(RND(5)*8):IF z(z,0)=0 THEN GOTO rndz
244: RETURN
245: rnds:
246: S=INT(RND(5)*8):IF S(S,0)=0 THEN GOTO rnds
247: RETURN
248:
249: zleft:
250: merker =P(z(z,1),z)
251: FOR k= 1 TO z(z,0)-1
252: P(z(z,k),z)=P(z(z,k+1),z)
253: NEXT
254: P(z(z,z(z,0)),z)=merker
255: GOSUB zeileneu
256: RETURN
257: zright:
258: merker=P(z(z,z(z,0)),z)
259: FOR k=z(z,0) TO 2 STEP-1
260: P(z(z,k),z)=P(z(z,k-1),z)
261: NEXT
262: P(z(z,1),z)=merker
263: GOSUB zeileneu
264: RETURN
265: sup:
266: merker=P(S,S(S,1))
267: FOR k=1 TO S(S,0)-1
268: P(S,S(S,k))=P(S,S(S,k+1))
269: NEXT
270: P(S,S(S,S(S,0)))=merker
271: GOSUB spalteneu
272: RETURN
273: sdown:
274: merker=P(S,S(S,S(S,0)))
275: FOR k=S(S,0) TO 2 STEP-1
276: P(S,S(S,k))=P(S,S(S,k-1))
277: NEXT
278: P(S,S(S,1))=merker
279: GOSUB spalteneu
280: RETURN
281:
282: zeileneu:
283: FOR k=0 TO 7
284: IF P(k,z)<>0 THEN
285: LINE (50+16*k,50+16*z)-(61+16*k,61+16*z),P(k,z)+2,bf
286: NEXT
287: spalteneu:
288: FOR k=0 TO 7
289: IF P(S,k)<>0 THEN
290: LINE (50+16*S,50+16*k)-(61+16*S,61+16*k),P(S,k)+2,bf
291: NEXT
292:
293: movecursor:
294: GOSUB drawpfeil
295: IF joy=-1 THEN GOTO clockwise

```

Listing: Push it !!!

LISTING

```

296: IF joy=1 THEN GOTO aclockwise
297: GOTO joywait
298:
299: clockwise:
300: ON pz GOSUB clock1,clock2,clock3,clock4
301: GOSUB drawpfeil
302: GOTO joywait
303:
304: clock1:
305: IF py=0 OR px=7 THEN GOTO clock1a
306: IF P(px+1,py-1)<>0 THEN px=px+1:py=py-1:pz=4:RETURN
307: clock1a:
308: IF px=7 THEN GOTO clock1b
309: IF P(px+1,py)<>0 THEN px=px+1:RETURN
310: clock1b:
311: pz=2:RETURN
312: clock2:
313: IF px=7 OR py=7 THEN GOTO clock2a
314: IF P(px+1,py+1)<>0 THEN px=px+1:py=py+1:pz=1:RETURN
315: clock2a:
316: IF py=7 THEN GOTO clock2b:
317: IF P(px,py+1)<>0 THEN py=py+1:RETURN
318: clock2b:
319: pz=3:RETURN
320: clock3:
321: IF px=0 OR py=7 THEN GOTO clock3a
322: IF P(px-1,py+1)<>0 THEN px=px-1:py=py+1:pz=2:RETURN
323: clock3a:
324: IF px=0 THEN GOTO clock3b
325: IF P(px-1,py)<>0 THEN px=px-1:RETURN
326: clock3b:
327: pz=4:RETURN
328: RETURN
329: clock4:
330: IF px=0 OR py=0 THEN GOTO clock4a
331: IF P(px-1,py-1)<>0 THEN px=px-1:py=py-1:pz=3:RETURN
332: clock4a:
333: IF py=0 THEN GOTO clock4b
334: IF P(px,py-1)<>0 THEN py=py-1:RETURN
335: clock4b:
336: pz=1:RETURN
337:
338: aclockwise:
339: ON pz GOSUB aclock1,aclock2,aclock3,aclock4
340: GOSUB drawpfeil
341: GOTO joywait
342:
343: aclock1:
344: IF py=0 OR px=0 THEN GOTO aclock1a
345: IF P(px-1,py-1)<>0 THEN px=px-1:py=py-1:pz=2:RETURN
346: aclock1a:
347: IF px=0 THEN GOTO aclock1b
348: IF P(px-1,py)<>0 THEN px=px-1:RETURN
349: aclock1b:
350: pz=4:RETURN
351: aclock2:
352: IF px=7 OR py=0 THEN GOTO aclock2a
353: IF P(px+1,py-1)<>0 THEN px=px+1:py=py-1:pz=3:RETURN
354: aclock2a:
355: IF py=0 THEN GOTO aclock2b:
356: IF P(px,py-1)<>0 THEN py=py-1:RETURN
357: aclock2b:
358: pz=1:RETURN
359: aclock3:
360: IF px=7 OR py=7 THEN GOTO aclock3a
361: IF P(px+1,py+1)<>0 THEN px=px+1:py=py+1:pz=4:RETURN
362: aclock3a:
363: IF px=7 THEN GOTO aclock3b
364: IF P(px+1,py)<>0 THEN px=px+1:RETURN
365: aclock3b:
366: pz=2:RETURN
367: RETURN
368: aclock4:
369: IF px=0 OR py=7 THEN GOTO aclock4a
370: IF P(px-1,py+1)<>0 THEN px=px-1:py=py+1:pz=1:RETURN
371: aclock4a:
372: IF py=7 THEN GOTO aclock4b
373: IF P(px,py+1)<>0 THEN py=py+1:RETURN
374: aclock4b:
375: pz=3:RETURN
376:
377:
378: moveplates:
379: zug = zug + 1
380: LOCATE 1,26:PRINT zug
381: z=py:s=px
382: IF joy=-1 THEN GOTO moveup:
383: IF joy=1 THEN GOTO movedown:
384: GOTO joywait
385: moveup:
386: ON pz GOSUB sdown,zleft,sup,zright
387: GOTO joywait
388: movedown:
389: ON pz GOSUB sup,zright,sdown,zleft
390: GOTO joywait:
391:
392: drawpfeil:
393: ON pz GOTO drawpfeil1,drawpfeil2,drawpfeil3,drawpfeil4
394: drawpfeil1:
395: PUT (49+16*px,38+16*py),pfeil%(0),XOR
396: RETURN
397: drawpfeil2:
398: PUT (62+16*px,49+16*py),pfeil%(200),XOR
399: RETURN
400: drawpfeil3:
401: PUT (49+16*px,64+16*py),pfeil%(400),XOR
402: RETURN
403: drawpfeil4:
404: PUT (36+16*px,49+16*py),pfeil%(600),XOR
405: RETURN

```

```

406:
407: testme:
408: testflag=0
409: FOR dx=0 TO 7:FOR dy=0 TO 7
410: IF P(dx,dy)<>pp(dx,dy) THEN testflag=1
411: NEXT:NEXT
412: IF testflag=0 THEN level=level+1:RETURN
413:
414: getbuf:
415: FOR bx=0 TO 7:FOR by=0 TO 7
416: bufp(bx,by)=P(bx,by)
417: IF P(bx,by)<>0 THEN
418: PUT (200+8*bx,149+8*by),sp%,PSET
419: LINE (201+8*bx,150+8*by)
- (207+8*bx,156+8*by),P(bx,by)+2,bf
420: END IF
421:
422: NEXT:NEXT
423: buffer = 1
424: GOTO joywait
425:
426: putbuf:
427: bufs=S
428: FOR bx=0 TO 7:FOR by=0 TO 7
429: P(bx,by)=bufp(bx,by)
430: NEXT
431: S=bx
432: GOSUB spaltenue
433: NEXT
434: S=bufs
435: GOTO joywait
436:
437: ende:
438: CLS
439: PRINT " THE
440: PRINT "END !"
441: GOSUB Bigtext
442: END
443:
444: ShowTitle:
445:
446: CLS
447: PRINT "Push"
448: PRINT "It !!"
449:
450: GOSUB Bigtext
451: LOCATE 25,1:PRINT "(E)asy,(M)edium,(H)ard"
452: getlevel:
453: le$=INKEY$
454: IF le$="h" THEN dif=3
455: IF le$="m" THEN dif=2
456: IF le$="e" THEN dif=1
457: IF dif=0 THEN GOTO getlevel
458: CLS:RETURN
459:
460: Bigtext:
461: FOR x=0 TO 56
462: FOR y=0 TO 16
463: col=POINT(x,y)
464: IF col>0 THEN PUT (8*x,24+8*y),sp%,PSET
465: NEXT:NEXT
466: RETURN
467:
468: dlevel1:
469: DATA 1, 2,3, 0,0,3,2,1, 0,1,3,2,2, 0,2,3,2,3
470:
471: DATA 3, 1,3, 1,0,2,1,-1, 5,0,6,1,-1, 3,3,4,4,5
472: DATA 2,3, 0,4,2,3,6, 2,7,4,2,6, 7,4,2,3,6, 3,6
473: DATA 2,1,4, 5,1,4, 1,6,6, 6,6,6, 5,6,6, 2,6,6
474:
475: DATA 2, 2,4, 0,1,5,2,5, 3,0,5,3,6, 4,3,5,4,5
476: DATA 1,4,5,1,6, 3,2, 1,1,5, 2,2,1
477:
478: dlevel4:
479: DATA 3, 1,1, 0,0,2,2,1, 2,4, 0,4,5,2,4, 4,0,5,3,4
480: DATA 0,1,3,2,5, 1,0,3,3,5, 3,1, 1,1,3
481:
482: DATA 3, 3,8, 2,0,2, 0,2,2, 5,0,2, 0,5,2, 2,7,2, 7,2,2
483: DATA 5,7,2, 7,5,2, 1,4, 1,1,2,2,5,
5,5,6,6,5, 1,5,2,6,5
484: DATA 5,1,6,2,5, 2,4, 3,1,2,2,1, 1,3,2,3,1, 3,6,2,2,1
485: DATA 6,3,2,3,1
486:
487: DATA 1, 2,4, 0,0,4,2,1, 0,1,4,2,2, 0,2,4,2,3
488: DATA 0,3,4,2,4
489:
490: dlevel7:
491: DATA 3, 1,2, 0,2,3,5,-1, 4,1,7,5,12, 2,6
492: DATA 2, 0,4,2,-1, 1,1,3,2,-1, 4,1,3,2,12, 1,6,3,2,-1
493: DATA 4,6,3,2,12, 2,7,4,2,12, 3,6, 5,2,5, 2,5,5
494: DATA 3,2,12, 4,5,-1, 0,6,0, 7,1,0
495:
496: DATA 2, 2,10, 1,0,3,3,1, 5,0,3,3,2, 0,1,3,2,1
497: DATA 4,1,3,2,2, 1,4,3,3,3, 0,5,3,2,3, 5,4,3,3,4
498: DATA 4,5,3,2,4, 3,0,7,3,12, 0,3,7,2,12, 3,1, 3,3,-1
499:
500: DATA 1, 2,8, 0,0,4,2,13, 1,1,4,2,11, 2,2,4,2,9
501: DATA 3,3,4,2,7, 3,4,4,2,7, 2,5,4,2,9, 1,6,4,2,11
502: DATA 0,7,4,2,13
503:
504: DATA 1, 1,14, 0,0,1,1,7, 2,0,3,1,9, 4,0,5,1,11
505: DATA 6,0,7,1,13, 6,2,7,3,11, 6,4,7,5,9, 6,6,7,7,7
506: DATA 4,6,5,7,9, 2,6,3,7,11, 0,6,1,7,13, 0,4,1,5,11
507: DATA 0,2,1,3,9, 2,2,5,5,4, 3,3,4,4,1
508:
509: DATA 3, 1,1, 2,3,3,4,5, 2,11, 0,5,3,3,4, 0,4,2,2,3
510: DATA 1,5,2,2,3, 1,6,3,2,2, 3,5,3,2,-1, 4,1,3,3,2
511: DATA 4,6,2,2,3, 3,7,2,2,3, 6,5,3,3,4, 6,2,3,3,10
512: DATA 7,4,4,3,6, 3,6, 1,7,4, 5,7,4, 7,2,10, 5,2,2
513: DATA 2,7,2, 4,4,-1

```



OMEGA Datentechnik Tel.: 0441 / 71109

Junkerstr. 2 2900 Oldenburg, 24 Std. Service
Ihr autorisierter Commodore Partner in Oldenburg

AMIGA 3000 mit 16MHz, 25MHz und 30MHz Preis auf Anfrage
AMIGA 3000 Speichererweiterung jedes weitere 1 MB 149,- DM
Wir führen die Spezial RAMs für AMIGA 3000 und A2620 / 30

AMIGA 2000 C 1775,- DM
AMIGA 500 798,- DM
AMIGA 500 C mit 1 MB ChipRAM und Uhr 955,- DM
A 2088 PC-XT Karte mit Laufwerk 575,- DM
A 2286 PC-AT Karte mit Laufwerk 1598,- DM
A 2630 Prozessorkarte mit 30 MHz 3698,- DM
A 2091 mit 40 MB Quantum 1498,- DM
A 1950 Multisync Monitor 1098,- DM
A 590 20 MB Festplatte für AMIGA 500 855,- DM
AMIGA 2000 8 MB Karte MicroBotics "8Up" / 2 MB 649,- DM
AMIGA 500 512 kB Erweiterung, abschaltbar 95,- DM
AMIGA 500 512 kB Erweiterung, mit Uhr, abschaltbar 109,- DM
128 kB RAM Erweiterung auf 640 kB für A 2088 129,- DM
ALF2 Filecard 60 MB, 25ms, RLL 1249,- DM
ALF2 Filecard 84 MB, 25ms, SCSI 1598,- DM
ALF2 Filecard 135 MB, 20ms, SCSI 2498,- DM
ALF2 Filecard 180 MB, 20ms, SCSI 2698,- DM
Systech Wechselplatte mit 44MB Medium 1498,- DM
Profisampler 56 kHz V2 125,- DM
Professional 68030 Board, 68881 20 MHz, 1 MB RAM 2198,- DM
PageStream 1.8 DTP Software 298,- DM
NEU O.M.A. Assembler Version 1.8 149,- DM
Test AMIGA Magazin 8/90 gut Seite 122/123 DEMO 6,- DM
A.D.D.A. V 16 Soundsampler in 16 Bit Technik für Amiga 2000 und AMIGA 3000, bitte fordern Sie Infos an.

Original Commodore Ersatzteile

A 500	DM
Netzteil	135,-
IC 8372	185,-
IC Denise	38,-
IC Paula	55,-
Orig. Floppy intern	228,-

Versandpauschale: 7,50 DM

Außerdem bietet unser Haus einen autorisierten 48-Stunden-Reparaturservice für Commodore, Atari, Panasonic, Toshiba und andere Home- und Personalcomputer. Händler fordern bitte unsere Händlerpreisliste an.

Audio Video Service Lukowiak GmbH & Co.
Löhner Str. 157
4971 Hüllhorst-Tengern
Tel.: (05744) 1092/1093
Fax.: (05744) 2890

YC-FARBSPLITTER YC-GENLOCK



Zwei Geräte für den Profi.
Garantiert beste Qualität und Verarbeitung. Bei OPTIVISION inklusive Service und Beratung. Fordern Sie ausführliche Unterlagen an: OPTIVISION. Der AMIGA & VIDEO Spezialist.

OPTIVISION
Heckenerstr. 16 D-5469 Windhagen
☎ 02645/4424 Fax 02645/3395

6000 PD

AMIGA PD incl. 3,5" MF2DD
schon ab 1,60 DM je Diskette

Profitieren auch Sie von unserer Erfahrung
Wir versenden noch am Tag des Bestelleingangs.
Wir haben 5 Jahre AMIGA Erfahrung,
Ultraschnell - Preiswert und stets Top Aktuell.
ABO Möglichkeit, Serienz. B. Fish, Kickstart, ANTARES usw.

AMIGA PD incl. 3,5" MF2DD
1 - 79 Je PD Disk 1,90 DM
80 - Je PD Disk 1,80 DM
AMIGA PD Disk incl. 5,25" Disk
10 - 99 Je PD Disk 1,20 DM
100 - Je PD Disk 1,00 DM

2 Katalogdisketten 5,- DM, bei Vorkasse (in Briefm.)
oder fordern Sie unser Info-Material an 1,- DM
Bestellannahme Mo.-Fr. 10 - 20 Uhr, Sa. 11 - 14 Uhr
Vorkasse + 4,- DM, Nachnahme + 8,- DM,
Ausland + 16,- DM bei Vorkasse.

Versand-Service
Bis 13 Uhr bestellt
bis 16 Uhr versandt



R. Dombrowski
Postfach 710462
2000 Hamburg 71
Tel: 040/6428225
FAX: 040/6426913

MacSoft — AMIGA SHOP
Hardware — Software — Schulung — PD

Public Domain

DISKETTE **4,- DM**
AUF 2 DD NUR
Über 4500 PD-Disk! Immer aktuell!
24-Std.-Versand-Service
Katalog-Disketten anfordern, 7,- DM
Selber abholen, NN gespart!
Hardware-Zusammenstellung auf Wunsch.
Lassen Sie sich Ihren
persönlichen Amiga anfertigen.
Fragen Sie nach unseren
Amiga-Einsteiger-Kursen.

Telefon 02 31 / 51 60 10
Mo.-Fr. 10-13, 15-20 Uhr, Sa. 10-16 Uhr
Hannövrer Str. 82, 4600 Dortmund 1
BTX * mac soft amiga #
Besuchen Sie unseren Messestand auf
der ORGATEC in Köln — Stand 505

Amiga Software in Köln

Public Domain PD:

Einzeldiskette 3,50 DM, 10 - 20 Disketten 3,20 DM, ab 20
Disketten 2,90 DM. Wir führen jeweils 5 Spiele oder
Anwenderpakete zu je 29,- DM.
Wir führen alle Serien wie z. B. Cactus, ACS, RPD, FISH,
Kickstart, Taifun, Killroy (ab 18 Jahren), Ruhrsoft u.v.a.
Ebenfalls veröffentlichen wir monatlich 10 Disketten auf
unserer eigenen Serie COLONIA PD, die bis jetzt 46
Disketten umfaßt.
Versand per Nachnahme mit UPS. Versandkosten trägt
der Besteller.

ACHTUNG PROGRAMMIERER:

Wir kaufen Ihre Programme, Spiele oder Anwender zu
vernünftigen Honoraren für die kommerzielle Vermark-
tung oder unsere Serie COLONIA PD. Rufen Sie uns an!

"DON MANOLO" - eine Pacman-Variante
mit insgesamt 63 Level • Preis: **DM 19,90**

Öffnungszeiten: Mo. - Fr. 10.00 Uhr bis 18.30 Uhr
Samstag 10.00 Uhr bis 14.00 Uhr

Pielago Software

Olpener Straße 438 • 5000 Köln 91
Telefon 02 21 - 89031 62

JEDES TEIL WOANDERS EINKAUFEN ?- BEI SCHAEFER BRAUCHST DU NUR EINMAL LAUFEN !

AMIGA-FLACHBETT-SCANNER DM 948,00
Kann als BILDERFASUNGSGERÄT/Kopierer und Thermodrucker eingesetzt werden. Scannbreite
200 Punkte/Zoll, Scanzeit 10 Sekunden. Abgabe des ganzen Bildes im iFF. Auflösungen 320x200,
640x512 werden unterstützt. Ausschnittvergrößerungen möglich. Binär u. 16 Grau-Darstellungen.

VIDEOTEXT-DECODER DM 298,00
Man kann das VIDEOSIGNAL eines Recorders, ScartTV oder Tuners dazu verwenden, den freien Service
des Teletextes im iFF- oder ASCII-Format abzugeben. Super-Grafik-Darstellung.

MINI MAX 500 DM 258,00
Erweiterung bis 2 MB Preise für Aufrüstsätze siehe Gigatron-Insertat

AMIGA-ACTION-REPLAY DM 198,00
(Eurosyste) Erweiterungsmodul für den AMIGA 500
- Programmierer, Spieltrainer, Zeilupe, Virus Detector, Statusanzeige und vieles mehr.
ZWEITLAUFWERK "Senator 3,5" (mit Trackanzeige DM 259,00) DM 239,00
ZWEITLAUFWERK 5,25" DM 258,00
MIDI-MASTER / Midi-Interface (Amiga-Typ angeben) DM 99,00
PRO-SAMPLER und DATEL-JAMMER DM 169,00
GENIUS oder GOLDEN-IMAGE-MAUS voll Amiga-kompatibel DM 69,50
SYNCRX EXPRESS incl. neueste Hardware DM 149,00
- Weiterentwicklung des Burst-Nibbler bei
Benutzung externer Laufwerke -
DEVPACK-ASSEMBLER (M+T-Verlag) DM 149,00
AMIGA-DEUTSCH - Deutsche DOS-Befehle einlesen DM 29,95
SPRACHENKIT - Vokabel-Trainer in Deutsch, Latein,
Engl., Französisch, Span., Italien., Japanisch!
AMIGA-Ersatzteilepreise wegen starker Schwankungen
derzeit nur auf Anfrage.

Alle Bücher von Markt und Technik - umfangreiche Software, Hardware und Ersatzteile
auf Anfrage - Versand nur gegen Vorkasse • DM 5,00 oder Nachnahme • DM 8,00.

(CLS) - COMPUTERLADEN SCHAEFER
Klingelhof 111, 5600 Wuppertal-2, Tel.: 0202/508121
Geschäftszeiten: Mo.+Di., Do.+Fr. 14-18.30 Uhr, Samstags 10-13 Uhr

DELTA-SOFT

6000 PD-DISKETTEN !!!
WIR FRIEREN DIE PREISE EIN !!!
JEDE DISK NUR DM

1,70

4 Katalogdisketten gegen 4,- DM Rückporto !

HOTLINE TEL.: (02 28) 66 54 43
F. Krüger - Postfach: 450 023 - 5305 Alfter 1
DELTA-SOFT - IM TANNENBUSCH 38-5300 BONN 1

Btx- Btx- Btx- Btx-

IBM-kompatible PCs XT/AT,
C 64, AMIGA und Atari ST

Der große Versender Deutschlands, aus BTX

Programme ab: **10,- DM**

Ihr Ansprechpartner für: * Spiele *

* Stengel

Wir kaufen und verkaufen:
Gebraucht-Spiele (nur Originale)

Info unter Tel.: 0521/763918
ab 17 Uhr anfordern
BTX: 0521/763918 - 0001/2

Computershop H.-J. Stengel
Brachhofstraße 21B, 4800 Bielefeld 16

Speichererweiterungen

ANRUFEN !!!! LOHNT SICH !!!!
TAGESPREISE !!!!!

Amiga 500	- 512 KB intern, abschaltbar	ab DM 159,-
Amiga 500	- 1 MB intern, absch., Uhr	ab DM 398,-
Amiga 500	- 2 MB intern, absch., Uhr	ab DM 598,-
Amiga 1000	- 2 MB extern, absch.,	ab DM 798,-
Amiga 2000	- 2 MB intern, absch.,	ab DM 898,-

Alle Speichererweiterungen sind autokonfigurierend, abschaltbar und mit sehr schnellen RAM's (100ns und schneller) ausgerüstet!
Durch Megabit-Technologie: minimaler Strombedarf.
****12 Monate Garantie****

Laufwerk 3.5" intern f. Amiga 2000	DM 169,-
Laufwerk 3.5" extern, durchgeschl. Bus, abschaltb	DM 198,-

B & S Computer-Vertriebs GmbH
Beethovenstr. 33 : 4172 Straelen 1
Tel: 02834/1249 : Fax: 02834/6979

SUPERPACK 50

50 PD-Programme der Extraklasse!
Return to Earth, Kampf um Eriador, Risk, Broker, Paranoid, Lucky Loser, Faktura, MS-Text, Videodatei, Plattenliste, Superliga, Haushaltsbuch, MCAD, Wizard of Sound, CLI-Pack, Virus Stop, Werner-Spiel, Latein, R.O.M., StarTrek, Alf, Core Wars, Label, Amiga-Paint, Giroman, Blizzard, VirusControl, Tetric, Moria, Battleforce, Peters Quest, Super-Bilder, Billard, Einkommensteuer, DSort III, Fix-Disk, Universaldatei, Quickmenü, Diskey, Mandelbrot, Silver Bilder, Astronomie, Super-Print, Calc, Atlantis, Schach, Labelpaint! Fast alle mit dt. Anleitungen!
Alle zusammen nur DM 79,-
GRATIS-KATALOGDISK

NEU SUPERPACK II DM 49,-
Banner, Power Packer, Xytronic, ASDG-RAM, Sonix-Player u. -Musik, Fußballmanager, Steinschlag, MR-Backup, Bibel-Quiz, Bootmaster, Boulder 1.3, Roll On, Pacey, Pente, Tumbler Street, Labelprint, Thundercopy, Trucking, SchreibM, Hyperadress, Ultra-Paint, Quizmaster, Pinnball, Database-Wizard, Jazzbench!
PATRICK PAWLOWSKI, Software-Service
Ellerbruch 19, 2177 Wingst, 04778/7294
Versandkosten: Vorkasse DM 3,50/NN DM 6,00

PUBLIC DOMAIN CENTER
Postfach 3142
5840 Schwerte

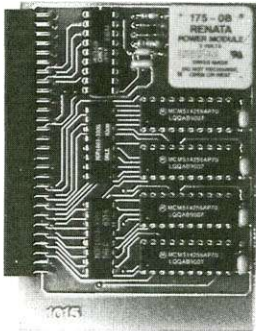
Volltreffer ...

Jede nur

1,90 DM

Info anfordern!

AMIGA 500 512 k Speichererweiterung



- Megabit Technologie
- Abschaltbar inkl. Schalter
- Uhr und Batterie

DM 119,-

ohne Uhr	DM 98,-	Kickstart Umschaltplatine	DM 29,-
2 MB A 500	DM 459,-	3,5"-Laufwerk extern	DM 178,-
2 MB A 2000	DM 598,-		

Hardware Design Udo Neuroth
Postfach 525, 4250 Bottrop
Tel. 020 41 / 204 24 ab 16 Uhr

GNE

- 5.25" TEAC Profilaufwerk extern 269,-**
1,67 MB unformatiert, 880 KB im AMIGAFORMAT, 100% kompatibel zu internem 3,5" Diskchange, abschaltbar, 40/80 Track unschaltbar original standardmäßiger durchgeschliffener Bus, Amiga-fertiges extrem robustes Stahlblechgehäuse, direkt MS-DOS + PCXT/AT Karten kompatibel, mindestens 80 cm langes zugestütztes Rundkabel, Stromversorgung über Amiga, passender BOOTSELEKTOR im Preis von 279,- enthalten, inklusive WRITE PROTECT Schalter, qualitätsgeprüft, ausführliche deutsche Bedienungsanleitung
- 5.25" TEAC intern A2000 229,-**
Technische Daten wie 5.25" extern, wahlweise DF1 oder DF2, kostet 249,-; wegen zusätzlicher Modifikationsplatine! unkomplizierter Einbau innerhalb 2 Minuten OHNE LÖTEN!!!
- 3.5" TEAC Profilaufwerk extern 229,-**
Technische Daten wie 5.25" Laufwerk extern, Write Protect Schalter und Bootselektor im Preis enthalten!
- 3.5" TEAC intern A2000 169,-**
Technische Daten kompatibel zu 3,5" Laufwerk extern! für A500 als DFO, für A2000 als DFO-DF1, für A1000 als DFO! unkomplizierter Einbau innerhalb 2 Minuten OHNE LÖTEN!!!
- BOOTSELEKTOR DF1/DF2/DF3 15,-**
SOUNDVERTEILER 15,-
A500 512KB + Uhr/abschaltbar 149,-
STAR LC 24-10 Farbband STAR LC 24-10: 9,- DM 698,-
STAR XB 24-10 1398,-
NEC P2 PLUS Farbband NEC P2200,-: 8,50 DM 849,-
NEC P60/70 auf Anfrage
Umschaltplatine mit Kick 1.3 98,-
Wir liefern nur deutsche Geräte mit Seriennummer und Super-Service!
1 Jahr Garantie auf alle Produkte dieser Anzeige!

GNE - GREBE NEUMANN ELEKTRONIK
Am Stein 10, 5419 Raubach, 02684-5566/5572
Telex: 869987 Fax: 02684/5448
HÄNDLERANFRAGEN ERWÜNSCHT!!!

02162/12073 • HAMO
Fachversand für Hard und Soft

Testen Sie unsere Leistungsfähigkeit rund um den **AMIGA**.
Wir führen nur Originalsoftware und geprüfte Markenhardware, zu **HAMO**-Preisen.

512 KB Erweiterung, Megabit, Akku-Uhr, abschaltbar	159,-
3,5" Diskettenlaufwerk, slimline abschaltbar, Bus	189,-
5,25" Diskettenlaufwerk, 40/80 Tracks, Bus, abschaltbar	249,-
HD A 590 - 20 MB Festplatte für AMIGA 500	828,-
Speichererweiterung 2 MB für AMIGA 2000 intern	689,-
Ersatzmaus für alle AMIGA-Modelle (REISWARE)	77,-

688 ATTACK SUBMARINE	dt 59,90	FIRE AND BRIMSTONE	dt 64,50
5th GEAR	dt 14,50	FOOTBALL MANAGER WORLD CUP ED 48,50	
ANARCHIE	dt 49,50	GREAT COURTS	dt 59,50
ASTRO MARINE CORPS (A.M.C.)	dt 49,50	HAMMERFIST	dt 64,50
ATOMIX	dt 51,50	IMPERIUM	dt 64,50
B.A.T.	dt 69,50	INTERNATIONAL 3D TENNIS	dt 64,50
BATTLECHESS - chinese -	dt 64,50	ITALIAN 1990 - WINNERS EDIT.	dt 54,50
BACK TO THE FUTURE 2	dt 64,50	KHALAAN	dt 64,50
BERLIN 1948 EAST vs WEST	dt 64,50	KICK OFF 2 WORLD CUP EDITION	dt 57,50
CHAMPIONS OF KRYNN Am 1 MB	dt 64,50	KLAX	dt 49,50
CHRONOQUEST 2	dt 69,50	LEISURE SUIT LARRY 2	dt 84,50
CASTLE MASTER	dt 56,50	LEISURE SUIT LARRY 3	dt 84,50
CODENAME ICEMAN	dt 94,50	LEGEND OF FEARHAIL	dt 69,50
CONQUEST OF CAMELOT	dt 94,50	MIDWINTER	dt 64,50
EMLYN HUGHES INT. SOCCER	dt 62,50	MIGHT AND MAGIC 2	dt 73,50
EUROPE SPACE SIMUL. (E.S.S.)	dt 69,50	RAILROAD TYCOON	dt a. Anfr.
E-MOTION	dt 58,50	THEIR FINEST HOUR	dt 69,50
F16 COMBAT	dt 59,50	ULTIMA 5	dt 69,50
F16 FALCON	dt 74,50	X-COPY mit HARDWARE	dt 58,50
F19 STEALTH FIGHTER	dt 69,50	ZOMBIE	dt 53,50

VERSAND: VORKASSE: +5,-, NN: +7,-, AUSLAND: AUFANFRAGE
BESTELLUNGENTELEFONISCH ODER SCHRIFTLICH AN:
HAMO - K. Rösge, Rahserstr. 235
4060 Viersen 1, Telefon: 02162/12073, Fax: 12074
DRUCKFEHLER UND PREISÄNDERUNGEN VORBEHALTEN.

ALSDORFER PUBLIC DOMAIN CENTER

PD Serien	Public Domain Preise		
ACS	Stück	3,5"	5,25"
Amok	1 - 50	2,50 DM	1,50 DM
AmSel	51 - 100	2,40 DM	1,40 DM
Auge	ab 101	2,30 DM	1,30 DM
Bavarian			
Cactus			
DDD			
FAUG	Tu.C.	Stück	4,00 DM
Fish	Ollis Games	Stück	4,00 DM
Franz	Germany	Stück	4,50 DM
Kickstart	Zeeka!	Stück	5,00 DM
Pfalz			
Pudomix			
RFD			
Saar			
Taifun			
TBAG			
u. v. a.			

Public Domain Sonderserien

FRUITMANIA, der einarmige Bandit in einer Superausführung, incl. Versand nur 34,95 DM
Professional Music Artist erstellen Sie eigene Musikstücke wie ein Profi 34,95 DM
Professional Print 2.0, das Druckerutilite mit umfangreichen Möglichkeiten 79,00 DM
3 Katalogdisketten mit Beschreibung vieler Serien und Programme, incl. Versand 15,00 DM

Hardware und Prolissoftware auf Anfrage, 48 Std. Versand
Versandkosten 8,00 DM + NN, Ausland nur Vorkasse + 15,00

WALTER KAMINSKI
Geilenkirchener Straße 4, 5110 Alsdorf
Tel. 02404/22963 Fax 02404/24285

Endlich!!
gibt es Shareware, wo die Autoren
an jeder Disk beteiligt sind. Von

SHARE

bekommt der Autor pro Disk, die
verkauft wird, 2,- DM
Unterstützen Sie unsere Idee.
2 Katalogdisketten komplett
deutsch + Antivirendisk +
1 Spiel erhalten Sie gegen 15,- DM
Vorkasse bei:

Herrmanns & Kommelter
Vom-Bruck Platz 45
Telefon 021 51 / 39 98 33
4150 Krefeld 1

AMIGA PD

Riesiges Angebot ausgesuchter PD Software

- Jedes Programm mit DEUTSCHER ANLEITUNG
- SUPER SICHER - Alles 100 % Virentfrei

zum Sonderpreis ab 1,-

Immer auf dem NEUESTEN STAND

- Zu jeder AMIGA-AUSGABE die getesteten PD PROGRAMME auf DREI 3,5" Disketten in einem Monatlichen ABO zusammengefaßt für:

DM 19,90 per Bankeinzug DM 22,90 per Nachnahme

Jetzt einmalig für Sie zum Schnäppchenpreis 9,90

- Fordern Sie unseren umfangreichen Katalog für nur DM 3,- an.

Hotline 0781 - 57734
von 18.00 - 20.00 Uhr

GERMAN-SOFT
PD - UTE - URLBAUER

Alt. C-84 / C-128 / AMIGA PD
Am Rothweg 9, D-7800 Ottenberg 18
Tel. 0781 / 57734



Spiele für Amiga

SPICHERERWEITERUNG AMIGA 500

auf 1 MB, inklusive Uhr, abschaltbar 178,00 DM
mit dem Spiel DUNGEON MASTER (dtisch.) 248,00 DM

Wir bekommen täglich Neuheiten für den Amiga! Fordern Sie doch einfach mal die neueste Preisliste an, Sie werden überrascht sein! Hier ein paar Beispiele:

NORDIC POWER	TITEL	PREIS
Der Freezer für den Amiga 500!	Aquonaut	74,80 DM
	Fighter Bomber	89,80 DM
	It came from the Desert	84,80 DM
	Kaiser	109,00 DM
Fordern Sie ausführliche Unterlagen an!	Little Computer People	19,80 DM
	Treasure Island Dizzy	19,80 DM
	Rockstar eats my Hamster	19,80 DM
	Space Harrier II	59,90 DM

X-Copy II + Cyclone + Hardware nur 69,00 DM

Versandkosten:

5,00 DM bei Warenwert unter 100,00 DM
Warenwert über 100,00 DM Versandkosten frei!

COMPY/SHOP

Gneisenastr. 29
4330 Mülheim/Ruhr
0208-497169 + 496178

AURIGA TECHNOLOGIE

Mainastr. 38
8000 München 60
Tel.: 089/8203651

Tel. Bestellannahme Mo. - Fr. 8.00 bis 18.00
unter Tel.: 0821/462511 in Augsburg

Speichererweiterung A2000:

Amiga 2000 interne Speicherkarte bis 8MB, aufrüstbar in 2MB Schritten mit 41000 oder 511000, autoconfigurierend, Testsoftware, Handbuch
2MB bestückt: DM 699.-- 4MB: DM 999.--

FileCards Amiga 2000:

ALF2 SCSI-Kontroller, autoboot ab Kickstart 1.3, hohe Datentransferrate, einfache Bedienbarkeit durch mitgelieferte ALF2-Software
mit Quantum P40S, 19/11ms, 42MB 1585.--
mit Seagate ST1096, 24ms, 83MB 1599.--

Atari-ST-Emulator MEDUSA DM 489.--
GFA-Basic3.5 Interpr.-Compiler DM 329.--
M2Amiga Modula-2 Compiler V3.3 DM 329.--

Drucker:

Citizen SWIFT24, 24 Nadeln DM 799.--
Farboption SWIFT24 DM 120.--
Einzelblatteinzug SWIFT24 DM 245.--
FontCard SWIFT24, je Stück DM 99.--

Commodore® Ersatzteil Service

✱ Wir liefern
für **Händler** und Privat-
anwender preiswert und prompt

✱ Rufen Sie uns an: (02331-43001)
oder schreiben Sie uns:

CIK-Computertechnik • Ingo Klepsch
Berliner Straße 49b • D-5800 Hagen 7

TELEFAX: 02331-42499

DATATRANS

Endlich die Daten Übertragungsmöglichkeit:

Vom Commodore C64 / C 128 zum AMIGA 500/1000/2000

Das Kit besteht aus folgenden Komponenten:

- 1.) Software Diskette für C 64 / C 128
- 2.) Übertragungskabel Länge ca. 2 Meter
- 3.) Anpassungsprogramm für die Empfangsseite
- 4.) Ausführliche Bedienungsanleitung
- 5.) Tel. Hotline für spezielle Fragen



Unser Übertragungs-Kit ist auch für PC's +
ATARI-ST's bestimmt!

Preis 60,- DM incl. Versandkosten.
Händler Anfragen erwünscht!

HVK Kunze Hollandstr. 119 4600 Dortmund 12
Tel. 0231/259090
(c) by Witte Soft FAX 0231/201565

COMPUTER SHOP

Wie?... Wir verschenken keine Amigas!
Aber wir haben immer gute Angebote!
Von A 500 bis A 2500, ob Filecard ...
oder PC-AT-Karten oder Zubehör

Was?... Alles für den Amiga bis Professional!
Anrufen lohnt sich!!!

Wo?... PV. Computer-Shop
Bahnhofstr. 278, 8623 Wetzikon (ZH)
Tel. 01/9 30 79 54
Schweiz

HARDWARE • SOFTWARE • BERATUNG

AMIGA - PD CENTER

8500 DISKETTEN

ANGEBOT DES MONATS

KATALOG-SET

5 voll gepackte Disketten incl. Aquarium!
Hochaktuell und komplett in deutsch!!!
Integrierter Update-Service!

ANTI - VIRUS

komplett neu überarbeitete AVD (Anti-Viren-
Disk mit diversen Updates (VirusX), vielen
neuen Progr. und deutschen Anleitungen!

TIME - SPEZIAL

Sonderausgabe der TIME-Serie -2 Disketten-
mit diversen Updates, neuen Programmen
und natürlich mit neuer ST-Musik!

alles komplett für nur DM 30,- (Vorkassell)

A.P.S. -electronic-
Sonnenborstel 31
D-3071 Steimbke
Tel.: 05026/1700
FAX 05026/1615



Info-Disk mit akt. Angeboten, News,
Preisauschreiben, Soundtracks,
Serien-Infos, u.s.w. für DM 2,- (VK)

Ihre Hotline für AMIGA-DOS-TIP

Tel.: 089/4 39 10 87 bis -89

Fax: 089/4 39 10 80

DONAU-SOFT

24 h-Schnellversand

Neutrale Disketten

3,5" 2DD (100 % errorfree)

	von Sentinel	von SONY/Collossus
bis 99 Stück	1,50 DM	1,85 DM
ab 100 Stück	1,30 DM	1,70 DM
ab 500 Stück	1,15 DM	1,50 DM

Laufwerke mit allen Extras

3,5" intern.....	155,- DM
3,5" extern, abschaltbar, Busdurchführung.....	209,- DM
5,25" extern, wie 3,5" + 40/80-Trackumschaltung.....	269,- DM

Sim City (dt.).....	79,- DM	GFA-Basic 3.5.....	208,- DM
B.A.D.....	77,- DM	GFA-Compiler 3.5.....	129,- DM
DPaint III dt.....	240,- DM	512 KB-Erw. (A500).....	179,- DM
Zoetrope 1.1.....	189,- DM	2 MB-Erw. (A500).....	548,- DM

Vorkasse: + 5,- DM, Nachnahme: + 8,- DM, Ausland: + 10,- DM

MAIK HAUER

Postfach 14 01, 8858 Neuburg Fax: 0 84 31/4 98 00
Tel.: 0 84 31/4 97 98 (bis 22 Uhr) BTX: *Donau-Soft #

Jingle Machine Teil 2

Die »Jingle Machine« von Oliver Heggelbacher war Programm des Monats in Heft 9/90. Geschrieben wurde das Assemblerfile mit dem Assembler »Devpac 2«. Weil das Listing zu lang war, mußten wir uns dazu entschließen, es in zwei Hälften zu teilen und in den Ausgaben 9/90 und 10/90 abzdrukken; Teil 2 bekommen Sie also jetzt hier präsentiert. Wer Teil 1 noch nicht hat, kann diesen entweder dem Heft 9/90 entnehmen oder direkt bei der Redaktion (Redaktion AMIGA DOS, DMV-Verlag, Postfach 250, D-3440 Eschwege, Kennwort »Jingle Machine«) anfordern; bitte den an sich selbst adressierten, ausreichend frankierten DIN-A4-Umschlag nicht vergessen.

Noch ein Hinweis für diejenigen, die sich sofort an die Arbeit machen wollen:

Der Quelltext kann mit einem beliebigen Editor eingetippt werden, er sollte danach an den ersten Teil angehängt werden. Das ganze Programm muß dann mit dem »Devpac 2« assembliert werden.

»Jingle Machine« befindet sich fertig assembliert, also als lauffähiges Programm, auf der Databox zu Heft 9/90.

Für diejenigen, die Teil 1 noch nicht kennen, hier ein kurzer Überblick: Die Jingle Machine erlaubt das Einlesen von Samples, die zur Untermalung von Video-Filmen oder selbst erstellten Musik-Sequenzen dienen können. Gruppen von Samples können als sogenannte »Sets« abgespeichert und wieder geladen werden. So, und nun an die Arbeit mit Teil 2.

(jb)

Listings

```

1179      move.b   EchoVL2,d1
1180      sub.b    EchoVol,d1
1181      bcc.s     .echvol
1182      clr.b     d1
1183 .echvol  move.b   d1,EchoVL2
1184      PUSHM    d0/a0-a2
1185      bsr       InstallTone
1186      POPM     d0/a0-a2
1187 .noecho  rts
1188
1189 DoLes    moveq    #0,d0
1190      move.b    Lesley,d0
1191      cmp.b     #128,d0
1192      blo.s     .lelin
1193      subi.w    #128,d0
1194      mulu      d0,d0
1195      lsr.w     #2,d0
1196      addi.w    #128,d0
1197 .lelin    add.w     lestate,d0
1198      move.w    d0,lestate
1199      rts
1200
1201 DoPhase  cmp.b     #254,chused(a1)
1202      bne.s     .nophase
1203      moveq     #0,d1
1204      move.b     Phase,d1
1205      cmp.b     #128+64,d1
1206      bhi.s     .aphase
1207      cmp.b     #128-64,d1
1208      bhs.s     .sphase
1209      subi.w    #128-64,d1
1210      bra.s     .phcont
1211 .aphase  subi.w    #128+64,d1
1212      phcont    asr.w     #3,d1
1213      add.w     chper(a1),d1
1214      move.w    d1,ac_per(a2)
1215 .nophase  rts
1216
1217
1218 .sphase  moveq     #1,d2
1219      subi.w    #128,d1
1220      tst.w     d1
1221      bpl.s     .nd2s
1222      neg.w     d1
1223      neg.w     d2
1224 .nd2s    addq.b    #1,d1
1225      lsr.b     #3,d1
1226      mulu      d1,d1
1227      subq.b    #1,d1
1228      eor.b     #63,d1
1229      move.b     chph(a1),d3
1230      addq.b    #1,d3
1231      cmp.b     d3,d1
1232      bhi.s     .norset
1233      clr.b     d3
1234 .norset  move.b    d3,chph(a1)

```

Listing: JingleMachine



```

1235      tst.b     d3
1236      beq.s     .dsph
1237      cmp.b     #1,d3
1238      beq.s     .nper
1239      bra.s     .nophase
1240 .dsph     add.w     chper(a1),d2
1241      move.w    d2,ac_per(a2)
1242      rts
1243 .nper     move.w    chper(a1),ac_per(a2)
1244      rts
1245
1246 ;----- Audio-Interrupt -----
1247 SIVec    equ $70
1248
1249 InitsInt  move.l    SIVec,OldsInt
1250      lea       DMABase,a3
1251      move.w    #1111000000,intena(a3)
1252      move.l    #AudInt,SIVec
1253      rts
1254
1255 ResetsInt  clr.b     Hold
1256      bsr       KeyUp
1257      lea       DMABase,a3
1258      move.w    #1111000000,intena(a3)
1259      move.l    OldsInt,SIVec
1260      rts
1261
1262
1263 DELCH     MACRO
1264      clr.b     chused(a0)
1265      clr.w     chvol(a0)
1266      ENDM
1267
1268 AudInt     PUSHM     a0/d0-d1
1269      lea       DMABase,a0
1270      move.w    intreqr(a0),d0
1271      and.w     #1111000000,d0
1272      move.w    d0,d1
1273      lsr.w     #7,d1
1274      move.w    d1,dmacon(a0)
1275      move.w    d0,intena(a0)
1276      move.w    d0,intreq(a0)
1277
1278      lea       Channel,a0
1279      btst      #0,d1
1280      beq.s     .ch2
1281      DELCH
1282      .ch2     lea       chlen(a0),a0
1283      btst      #1,d1
1284      beq.s     .ch3
1285      DELCH
1286      .ch3     lea       chlen(a0),a0
1287      btst      #2,d1
1288      beq.s     .ch4
1289      DELCH
1290      .ch4     lea       chlen(a0),a0
1291      btst      #3,d1
1292      beq.s     .chfin
1293

```

Listing: JingleMachine


```

1294      DELCH
1295 .chfin POPM      a0/d0-d1
1296      rte
1297
1298 ***** Bildschirm Aufbau *****
1299 ; Jingletablelle
1300 BuildJD
1301      bsr      SetNMode
1302      moveq    #2,d7
1303      moveq    #1,d0
1304      moveq    #11,d2
1305      moveq    #1,d1
1306      moveq    #16,d3
1307      moveq    #5,d4
1308 BuildJ2 PUSHM      d0-d7
1309      bsr      MakeFr
1310      POPM      d0-d7
1311      addi.w    #13,d0
1312      addi.w    #13,d2
1313      dbra     d7,BuildJ2
1314      moveq    #0,d0
1315 BuildJ3 bsr      PrSample
1316      addq.b    #1,d0
1317      cmp.b     #48,d0
1318      blo.s     BuildJ3
1319      rts
1320
1321 BuildEdit
1322      bsr      SetNMode
1323      MAKEFRA 0,20,39,27,2
1324      MAKEFRA 16,22,21,23,10
1325      MAKEFRA 23,22,29,26,10
1326      MAKEFRA 31,22,38,26,13
1327      MAKEFRA 9,22,12,22,5
1328      lea      EditText,a0
1329      bsr      Print
1330      bsr      PrMem
1331      rts
1332
1333 BuildPlay
1334      bsr      SetNMode
1335      MAKEFRA 0,20,39,27,2
1336      MAKEFRA 1,21,4,26,13
1337      MAKEFRA 18,21,21,21,13
1338      MAKEFRA 18,23,21,23,13
1339      MAKEFRA 18,25,21,25,13
1340      MAKEFRA 35,22,38,22,6
1341      MAKEFRA 35,26,38,26,6
1342      MAKEFRA 29,22,29,26,10
1343      MAKEFRA 12,26,13,26,13
1344      lea      PlayText,a0
1345      bsr      Print
1346      rts
1347
1348 PrSample
1349      PUSH      d0
1350      bsr      GetPSPos
1351      moveq    #2,d1
1352      moveq    #11,d4
1353      moveq    #0,d5
1354      addq.b    #1,d0
1355      bsr      PrNum
1356      POP       d0
1357 PrSample2
1358      bsr      GetPSPos
1359      PUSH      d0
1360      lea      Sample,a1
1361      mulu     #slen,d0
1362      lea      fname(a1,d0),a1
1363      lea      PrStr,a0
1364      addq.b    #3,d2
1365      move.b    #1,(a0)+
1366      move.b    d2,(a0)+
1367      move.b    d3,(a0)+
1368      move.b    #2,(a0)+
1369      move.b    #1,(a0)+
1370      move.b    #0,(a0)+
1371      tst.b     used-fname(a1)
1372      bne.s     sused
1373      move.l    #"not ",(a0)
1374      move.l    #"used",4(a0)
1375      move.w    #$00ff,8(a0)
1376      bra.s     prncf
1377 sused      move.l    #$20202020,(a0)
1378      move.l    #$20202020,4(a0)
1379      move.w    #$00ff,8(a0)
1380      moveq    #0,d0
1381 prncop     move.b    (a1)+,(a0)+
1382      tst.b     (a1)
1383      beq.s     prncf
1384      addq.b    #1,d0
1385      cmp.b     #8,d0
1386      blo.s     prncop
1387 prncf      lea      PrStr,a0
1388      bsr      Print
1389      POP       d0
1390      rts
1391
1392 GetPSPos
1393      moveq    #0,d2
1394      move.b    d0,d2
1395      lsr.b     #4,d2
1396      mulu     #13,d2

```



Listing: JingleMachine

```

1397      addq.b    #1,d2
1398      move.b    d0,d3
1399      and.b     #15,d3
1400      addq.b    #1,d3
1401      rts
1402
1403 ; Speicher
1404 PrMem      moveq    #2,d1
1405      CALLEXEC AvailMem
1406      moveq    #7,d1
1407      moveq    #23,d2
1408      moveq    #24,d3
1409      moveq    #1,d4
1410      moveq    #11,d5
1411      bsr      PrNum
1412      moveq    #4,d1
1413      CALLEXEC AvailMem
1414      moveq    #7,d1
1415      moveq    #23,d2
1416      moveq    #26,d3
1417      moveq    #1,d4
1418      moveq    #11,d5
1419      bsr      PrNum
1420      rts
1421
1422 ; Fehlermeldungen, a0=Text
1423 PrError    lea      PETxBuf,a1
1424      COPYSTR  a0,a1
1425      move.b    #255,(a1)
1426      lea      PETxt,a0
1427      bsr      Print
1428      move.b    #-1,peflag
1429      rts
1430 DelError
1431      tst.b     peflag
1432      bne.s     .cont
1433      rts
1434 .cont      clr.b     peflag
1435      lea      PEDelTx,a0
1436      bra      Print
1437
1438 SECTION JMData,DATA
1439 PETxt      dc.b     1,1,18,2,9,11
1440      dcb.b     38,32
1441      dcb.b     0
1442      dcb.b     1,1,18
1443 PETxBuf    dcb.b     40,0
1444 PEDelTx    dc.b     1,1,18,2,8,11
1445      dc.b     "      - Jingle Machine v1.2 - "
1446      dc.b     0,255
1447 peflag     dc.b     0
1448      EVEN
1449 SECTION JMCode,CODE
1450
1451 ; Rahmen zeichnen: d0,d1-d2,d3 (in Zeichen), d4=1. Farbe
1452 MakeFr     lsl.w     #3,d0
1453      lsl.w     #3,d1
1454      lsl.w     #3,d2
1455      lsl.w     #3,d3
1456      subq.w    #1,d0
1457      subq.w    #1,d1
1458      addq.w    #8,d2
1459      addq.w    #8,d3
1460      PUSHM     d0-d4
1461      move.l    RastPort,a1
1462      move.w    d4,d0
1463      addq.w    #1,d0
1464      CALLGRAF SetAPen
1465      POPM      d0-d4
1466      PUSHM     d0-d4
1467      CALLGRAF RectFill
1468      POPM      d0-d4
1469      subq.w    #1,d0
1470      subq.w    #1,d1
1471      addq.w    #1,d2
1472      addq.w    #1,d3
1473      bsr.s     Border
1474      subq.w    #1,d0
1475      subq.w    #1,d1
1476      addq.w    #1,d2
1477      addq.w    #1,d3
1478      bsr.s     Border
1479      rts
1480
1481 Border     PUSHM     d0-d4
1482      move.l    RastPort,a1
1483      move.w    d0,bx1
1484      move.w    d1,by1
1485      move.w    d2,bx2
1486      move.w    d3,by2
1487      move.w    d4,bcol
1488      move.w    d4,d0
1489      CALLGRAF SetAPen
1490      move.w    bx1,d0
1491      move.w    by2,d1
1492      CALLGRAF Move
1493      move.w    bx1,d0
1494      move.w    by1,d1
1495      CALLGRAF Draw
1496      move.w    bx2,d0
1497      move.w    by1,d1
1498      CALLGRAF Draw
1499      move.w    bcol,d0

```

Listing: JingleMachine


```

1500      addq.w #2,d0
1501      CALLGRAF SetAPen
1502      move.w bx2,d0
1503      move.w by2,d1
1504      CALLGRAF Draw
1505      move.w bx1,d0
1506      addq.w #1,d0
1507      move.w by2,d1
1508      CALLGRAF Draw
1509      moveq #1,d0
1510      CALLGRAF SetAPen
1511      move.w bx1,d0
1512      move.w by1,d1
1513      CALLGRAF WritePixel
1514 bend  POPM    d0-d4
1515      rts
1516
1517 bx1    dc.w 0
1518 by1    dc.w 0
1519 bx2    dc.w 0
1520 by2    dc.w 0
1521 bcol    dc.w 0
1522
1523 ***** hauseigene Gadgetverwaltung!!! *****
1524
1525 JinGadgets
1526      moveq #0,d0
1527      bsr    ClearGad
1528      moveq #1,d0
1529      moveq #1,d1
1530      moveq #2,d3
1531      moveq #1,d4
1532 .loop1  moveq #1,d2
1533 .loop2  bsr    InitGad
1534      addq.b #1,d2
1535      cmp.b #17,d2
1536      blo.s .loop2
1537      add.b #13,d1
1538      cmp.b #40,d1
1539      blo.s .loop1
1540      moveq #0,d0
1541      move.b d0,jnum
1542      bsr    GetGadPt
1543      bsr    InvSW2
1544      rts
1545
1546 SetGadgets
1547      moveq #48,d0
1548      bsr    ClearGad
1549      moveq #1,d1
1550      moveq #22,d2
1551      bsr    InitReg
1552      moveq #5,d1
1553      moveq #22,d2
1554      bsr    InitReg
1555      MAKEGAD 2,9,22,4,1
1556      MAKEGAD 1,7,20,32,1
1557      MAKEGAD 3,31,22,8,1
1558      MAKEGAD 3,31,23,8,1
1559      MAKEGAD 3,31,24,8,1
1560      MAKEGAD 3,31,26,8,1
1561      move.b jnum,d0
1562      bsr    JSelect
1563      rts
1564
1565 PlayGadgets
1566      moveq #48,d0
1567      bsr    ClearGad
1568      MAKEGAD 2,1,22,1,1
1569      MAKEGAD 2,2,22,1,1
1570      MAKEGAD 2,3,22,1,1
1571      MAKEGAD 2,4,22,1,1
1572      moveq #6,d1
1573      moveq #21,d2
1574      bsr    InitReg
1575      moveq #9,d1
1576      moveq #21,d2
1577      bsr    InitReg
1578      MAKEGAD 2,18,21,4,1
1579      MAKEGAD 2,18,23,4,1
1580      MAKEGAD 3,35,22,4,1
1581 ; Echoteil
1582      moveq #23,d1
1583      moveq #22,d2
1584      bsr    InitReg
1585      moveq #26,d1
1586      moveq #22,d2
1587      bsr    InitReg
1588      MAKEGAD 5,29,26,1,1
1589      MAKEGAD 5,29,25,1,1
1590      MAKEGAD 5,29,24,1,1
1591      MAKEGAD 5,29,23,1,1
1592      MAKEGAD 5,29,22,1,1
1593
1594      MAKEGAD 2,18,25,4,1 ; 64=Mono
1595      moveq #12,d1
1596      moveq #21,d2
1597      bsr    InitReg ; Phase
1598      moveq #15,d1
1599      moveq #21,d2
1600      bsr    InitReg ; Lesley
1601      MAKEGAD 2,12,26,2,1 ; PhAct
1602      MAKEGAD 3,35,26,4,1 ; 68=exit

```

Listing: JingleMachine

```

1603      rts
1604
1605 ; Ja/Nein - Sicherheitsabfrage
1606 Requester
1607      MAKEFRA 12,6,26,11,13
1608      MAKEFRA 16,10,18,10,10
1609      MAKEFRA 20,10,22,10,10
1610      lea    ReqTxt,a0
1611      bsr    Print
1612      move.b gused,d0
1613      move.b d0,regyes
1614      addq.b #1,d0
1615      move.b d0,regno
1616      MAKEGAD 3,16,10,3,1 ; yes
1617      MAKEGAD 3,20,10,3,1 ; no
1618 .reqloop
1619      bsr    MouseControl
1620      cmp.b regyes,d0
1621      beq.s .yes
1622      cmp.b regno,d0
1623      bne.s .reqloop
1624      moveq #-1,d0
1625      bra.s .no
1626 .yes  moveq #0,d0
1627 .no   PUSH    d0
1628      move.b regyes,d0
1629      bsr    ClearGad
1630      bsr    SetNMode
1631      move.l RastPort,a1
1632      moveq #0,d0
1633      CALLGRAF SetAPen
1634      moveq #0,d0
1635      moveq #0,d1
1636      move.w #319,d2
1637      move.w #139,d3
1638      CALLGRAF RectFill
1639      bsr    BuildJD
1640      move.b jnum,d0
1641      bsr    GetGadPt
1642      bsr    InvSW2
1643      POP     d0
1644      rts
1645
1646 SECTION JMData,DATA
1647 regyes dc.b 0
1648 regno  dc.b 0
1649 ReqTxt dc.b 1,13,7,2,9,14,"Are you sure?",0
1650      dc.b 1,16,10,2,6,11,"YES",0,1,20,10,"NO",0,255
1651      EVEN
1652 SECTION JMCode,CODE
1653
1654 ClearGad
1655      move.b d0,gused
1656      bsr    GetGadPt
1657 .loop1  clr.b gtyp(a0)
1658      lea    glen(a0),a0
1659      addq.b #1,d0
1660      cmp.b #ganz,d0
1661      blo.s .loop1
1662      rts
1663
1664 rho    equ 4 ; Hoehe der Schieberegler
1665 InitReg
1666      moveq #4,d0
1667      moveq #2,d3
1668      moveq #rho,d4
1669      bsr    InitGad
1670      PUSHM  d0-d4
1671      PUSH    a0
1672      bsr    SetNMode
1673      POP     a0
1674      move.w gx1(a0),d0
1675      move.w gy1(a0),d1
1676      move.w gx2(a0),d2
1677      move.w gy2(a0),d3
1678      subq.w #1,d0
1679      addq.w #1,d2
1680      subq.w #3,d1
1681      addq.w #3,d3
1682      PUSHM  a0/d0-d3
1683      subq.w #1,d0
1684      subq.w #1,d1
1685      addq.w #1,d2
1686      addq.w #1,d3
1687      moveq #6,d4
1688      bsr    Border
1689      moveq #7,d0
1690      move.l RastPort,a1
1691      CALLGRAF SetAPen
1692      POPM    a0/d0-d3
1693      PUSH    a0
1694      move.l RastPort,a1
1695      CALLGRAF RectFill
1696      POP     a0
1697      bsr    SkTeil
1698      moveq #0,d3
1699      bsr    DrawReg
1700      POPM    d0-d4
1701      rts
1702 InitGad lea    Gadgets,a0
1703      clr.w  d5
1704      move.b gused,d5
1705      mulu   #glen,d5

```

Listing: JingleMachine


```

1706      lea      (a0,d5),a0
1707      move.b   d0,gtyp(a0)
1708      clr.b    gstate(a0)
1709      move.b   d1,gx(a0)
1710      move.b   d2,gy(a0)
1711      move.b   d3,gbr(a0)
1712      move.b   d4,gho(a0)
1713      PUSHM    d0-d4
1714      lsl.w    #3,d1
1715      move.w    d1,gx1(a0)
1716      lsl.w    #3,d2
1717      move.w    d2,gy1(a0)
1718      lsl.w    #3,d3
1719      add.w     d1,d3
1720      subq.w    #1,d3
1721      move.w    d3,gx2(a0)
1722      lsl.w    #3,d4
1723      add.w     d2,d4
1724      subq.w    #1,d4
1725      move.w    d4,gy2(a0)
1726      POPM     d0-d4
1727      add.b     #1,gused
1728      rts
1729
1730 ;----- Mausabfrage, Gadget angeklickt? -----
1731 MouseControl
1732      bsr.s     Mouse2
1733      cmp.b     #1,d0
1734      beq.s     .nderr
1735      cmp.b     #2,d0
1736      beq.s     .nderr
1737      PUSHM     a0/d0-d1
1738      bsr      DelError
1739      POPM      a0/d0-d1
1740 .nderr
1741 Mouse2
1742      move.l    UserPort,a0
1743      CALLEXEC GetMsg
1744      tst.l     d0
1745      beq.s     Mouse2
1746      move.l    d0,a0
1747      move.w    32(a0),d0
1748      move.w    34(a0),d1
1749      move.w    d0,mousex
1750      move.w    d1,mousey
1751      lsr.w     #3,d0
1752      lsr.w     #3,d1
1753      move.b    d0,mousecx
1754      move.b    d1,mousecy
1755      move.w    24(a0),d0
1756      move.w    d0,mouseq
1757      move.l    20(a0),d1
1758      PUSH     d1
1759      move.l    a0,a1
1760      CALLEXEC ReplyMsg
1761      POP      d1
1762      btst     #4,d1
1763      bne      RegMove
1764 rmcont
1765      btst     #10,d1
1766      bne.s    KeyPressed
1767      btst     #3,d1
1768      beq.s    Mouse2
1769      move.w    mouseq,d0
1770      cmp.b     #104,d0
1771      beq      GadSearch
1772      cmp.b     #232,d0
1773      beq      RegRelease
1774      cmp.b     #105,d0
1775      beq      dosnd
1776      cmp.b     #233,d0
1777      beq      KeyRel
1778      moveq     #-1,d0
1779      rts
1780 KeyPressed
1781      move.w    mouseq,d0
1782      btst     #7,d0
1783      bne.s    KeyRel
1784 dosnd
1785      moveq     #-2,d0
1786      rts
1787 KeyRel
1788      moveq     #-3,d0
1789      rts
1790 RegMove
1791      moveq     #0,d0
1792      move.b    rnum,d0
1793      cmp.b     #-1,d0
1794      beq      rmcont
1795      bsr      GetGadPt
1796      move.w    mousey,d1
1797      sub       gy1(a0),d1
1798      bcc.s     .tlow
1799      clr.w     d1
1800      cmp.w     #rho*8,d1
1801      blo.s     .thigh
1802      move.w    #rho*8-1,d1
1803      eor.w     #rho*8-1,d1
1804      lsl.w     #8,d1
1805      divu      #rho*8,d1
1806      bra      Switch
1807 RegRelease
1808      move.b    rnum,d0
1809      moveq     #1,d1

```

Listing: JingleMachine

```

1809      bsr      GetGadPt
1810      move.b    #-1,rnum
1811      rts
1812
1813 GadSearch
1814      lea      Gadgets,a0
1815      moveq     #0,d0
1816      move.b    mousecx,d1
1817      move.b    mousecy,d2
1818 .loop1
1819      tst.b     gtyp(a0)
1820      beq.s     .nohit
1821      move.b    gx(a0),d3
1822      move.b    gy(a0),d4
1823      move.b    gbr(a0),d5
1824      move.b    gho(a0),d6
1825      add.b     d3,d5
1826      add.b     d4,d6
1827      cmp.b     d3,d1
1828      blo.s     .nohit
1829      cmp.b     d4,d2
1830      blo.s     .nohit
1831      cmp.b     d5,d1
1832      bhs.s     .nohit
1833      cmp.b     d6,d2
1834      bhs.s     .nohit
1835      bra.s     GadHit
1836 .nohit
1837      lea      glen(a0),a0
1838      addq.b    #1,d0
1839      cmp.b     #ganz,d0
1840      blo.s     .loop1
1841      moveq     #-1,d0
1842      rts
1843 GadHit
1844      move.b    gtyp(a0),d1
1845      cmp.b     #3,d1
1846      beq.s     Blink
1847      cmp.b     #2,d1
1848      beq.s     Klick
1849      cmp.b     #5,d1
1850      beq.s     SetOn
1851      cmp.b     #4,d1
1852      beq.s     RegHold
1853      rts
1854 SetOn
1855      moveq     #-1,d1
1856      bra.s     Switch
1857 Blink
1858      move.b    gstate(a0),d1
1859      eor.b     #255,d1
1860      bsr.s     Switch
1861      moveq     #0,d1
1862      addq.l    #1,d1
1863      cmp.l     #10000,d1
1864      bne.s     .loop
1865      move.b    gstate(a0),d1
1866      eor.b     #255,d1
1867      bra.s     Switch
1868 RegHold
1869      move.b    d0,rnum
1870      bsr      RegMove
1871      rts
1872 GetGadPt
1873      moveq     #0,d2
1874      move.b    d0,d2
1875      mulu      #glen,d2
1876      lea      Gadgets,a0
1877      lea      (a0,d2),a0
1878      rts
1879 Switch
1880      bsr.s     GetGadPt
1881      move.b    gstate(a0),d2
1882      move.b    d1,gstate(a0)
1883      cmp.b     #2,gtyp(a0)
1884      beq.s     InvertSW
1885      cmp.b     #3,gtyp(a0)
1886      beq.s     InvertSW
1887      cmp.b     #5,gtyp(a0)
1888      beq.s     InvertSW
1889      cmp.b     #4,gtyp(a0)
1890      beq.s     DispReg
1891      rts
1892 InvertSW
1893      cmp.b     d1,d2
1894      beq.s     isabort
1895      bsr      SetXMode
1896      PUSHM     a0/d0
1897      move.w    gx1(a0),d0
1898      move.w    gy1(a0),d1
1899      move.w    gx2(a0),d2
1900      move.w    gy2(a0),d3
1901      CALLGRAF RectFill
1902      bsr      SetNMode
1903      POPM      a0/d0
1904      isabort
1905      rts
1906 DispReg
1907      cmp.b     d1,d2
1908      beq.s     .abort
1909      moveq     #0,d3
1910      move.b    d2,d3
1911      bsr.s     DelReg
1912      moveq     #0,d3
1913      move.b    d1,d3

```

Listing: JingleMachine


```

1912      bsr.s    DrawReg
1913 .abort    rts
1914
1915 DelReg    PUSHM    a0/d0-d3
1916          bsr      SetNMode
1917          moveq     #7,d0
1918          move.l    RastPort,a1
1919          CALLGRAF  SetAPen
1920          POPM     a0/d0-d3
1921          PUSHM    a0/d0-d3
1922          move.w    d3,d1
1923          eor.b     #255,d1
1924          mulu     #rho*8,d1
1925          lsr.w     #8,d1
1926          add.w     gyl(a0),d1
1927          move.w    gx1(a0),d0
1928          move.w    gx2(a0),d2
1929          move.w    d1,d3
1930          subq.w    #3,d1
1931          addq.w    #3,d3
1932          move.l    RastPort,a1
1933          CALLGRAF  RectFill
1934          POPM     a0/d0-d3
1935          PUSHM    a0/d0-d3
1936          bsr      SkTeil
1937          POPM     a0/d0-d3
1938          rts
1939
1940 DrawReg    PUSHM    a0/d0-d3
1941          bsr      SetNMode
1942          POPM     a0/d0-d3
1943          PUSHM    a0/d0-d3
1944          move.w    d3,d1
1945          eor.b     #255,d1
1946          mulu     #rho*8,d1
1947          lsr.w     #8,d1
1948          add.w     gyl(a0),d1
1949          move.w    gx1(a0),d0
1950          move.w    gx2(a0),d2
1951          subq.w    #3,d1
1952          lea       RCTab,a0
1953          moveq     #6,d3
1954 .dreg2     move.b   (a0)+,d4
1955          PUSHM    a0/d0-d3
1956          moveq     #0,d0
1957          move.b     d4,d0
1958          move.l    RastPort,a1
1959          CALLGRAF  SetAPen
1960          POPM     a0/d0-d3
1961          PUSHM    a0/d0-d3
1962          move.l    RastPort,a1
1963          CALLGRAF  Move
1964          POPM     a0/d0-d3
1965          PUSHM    a0/d0-d3
1966          move.w    d2,d0
1967          move.l    RastPort,a1
1968          CALLGRAF  Draw
1969          POPM     a0/d0-d3
1970          addq.w    #1,d1
1971          dbra     d3, .dreg2
1972          POPM     a0/d0-d3
1973          rts
1974 RCTab      dc.b    1,2,2,4,2,2,3
1975          EVEN
1976
1977 SkTeil     PUSH     a0
1978          move.l    RastPort,a1
1979          moveq     #5,d0
1980          CALLGRAF  SetAPen
1981          POP      a0
1982          move.w    gyl(a0),d1
1983          move.w    gy2(a0),d2
1984          subq.w    #1,d1
1985          addq.w    #3,d2
1986 .loop1     PUSHM    a0/d1-d2
1987          move.w    gx1(a0),d0
1988          subq.w    #1,d0
1989          CALLGRAF  Move
1990          POPM     a0/d1-d2
1991          PUSHM    a0/d1-d2
1992          move.w    gx2(a0),d0
1993          addq.w    #1,d0
1994          CALLGRAF  Draw
1995          POPM     a0/d1-d2
1996          addq.w    #4,d1
1997          cmp.w     d2,d1
1998          bsr.s     .loop1
1999          rts
2000
2001 SetXMode   PUSHM    a0/d0-d3
2002          move.l    RastPort,a1
2003          moveq     #15,d0
2004          CALLGRAF  SetAPen
2005          moveq     #3,d0
2006          CALLGRAF  SetDrMd
2007          POPM     a0/d0-d3
2008          rts
2009
2010 SetNMode   move.l    RastPort,a1
2011          moveq     #0,d0
2012          CALLGRAF  SetDrMd
2013          rts
2014

```



Listing: JingleMachine

```

2015
2016 ***** Screens und Fenster *****
2017 OpenScreen
2018     lea       GfxName,a1
2019     clr.l     d0
2020     CALLExec  OpenLibrary
2021     move.l    d0, GfxBase
2022     lea       IntName,a1
2023     clr.l     d0
2024     CALLExec  OpenLibrary
2025     move.l    d0, IntuitionBase
2026     lea       JScreen,a0
2027     CALLINT   OpenScreen
2028     move.l    d0, ScreenHd
2029     beq       nosmem
2030     move.l    d0,a1
2031     addi.l    #52c,d0
2032     move.l    d0, ViewPort
2033     move.l    Sc0(a1),Plane0
2034     move.l    Sc4(a1),Plane1
2035     move.l    Sc8(a1),Plane2
2036     move.l    Sc(a1),Plane3
2037     move.l    ViewPort,a0
2038     lea       ColTab,a1
2039     moveq     #16,d0
2040     CALLGRAF  LoadRGB4
2041     lea       JinWind,a0
2042     CALLINT   OpenWindow
2043     move.l    d0, WindowHd
2044     beq.s     nowmem
2045     move.l    d0,a0
2046     move.l    50(a0),RastPort
2047     move.l    86(a0),UserPort
2048     moveq     #0,d0
2049     rts
2050
2051 CloseScreen
2052     move.l    WindowHd,a0
2053     CALLINT   CloseWindow
2054     nowmem    move.l    ScreenHd,a0
2055     CALLINT   CloseScreen
2056     nosmem    move.l    IntuitionBase,a1
2057     CALLExec  CloseLibrary
2058     move.l    GfxBase,a1
2059     CALLExec  CloseLibrary
2060     moveq     #1,d0
2061     rts
2062
2063 ***** Scrolling *****
2064 IFD Scrolling
2065
2066 InitScroll
2067     lea       GFTxt,a0
2068     lea       ScFont,a1
2069     moveq     #32,d0
2070 .instl      PUSHM    a0-a1/d0
2071     move.b     d0,6(a0)
2072     bsr        Print
2073     POPM       a0-a1/d0
2074     move.l     Plane0,a2
2075     moveq     #7,d1
2076 .copl       move.b   (a2),(a1)+
2077     lea        40(a2),a2
2078     dbra       d1, .copl
2079     addq.b     #1,d0
2080     bcc.s     .instl
2081     move.b     #32,6(a0)
2082     bsr        Print
2083     move.l     #ScrTxt,ScrPt
2084     move.l     #ScFont,LetPt
2085     clr.b     sccount
2086     clr.b     scspeed
2087
2088 ;----- UserCopList aufbauen
2089 CWAIT      MACRO
2090     move.l     UCLList,a1
2091     CALLGRAF  CWait
2092     move.l     UCLList,a1
2093     CALLGRAF  CBump
2094     ENDM
2095 CMOVE      MACRO
2096     move.l     UCLList,a1
2097     CALLGRAF  CMove
2098     move.l     UCLList,a1
2099     CALLGRAF  CBump
2100     ENDM
2101 copmem     equ    $10003
2102
2103     move.l     #ucl_SIZEOF,d0
2104     move.l     #copmem,d1
2105     CALLExec  AllocMem
2106     move.l     d0,UCLList
2107     move.l     d0,a0
2108     moveq     #127,d0
2109     CALLGRAF  UCopperListInit
2110     move.w     #scyp,d0
2111     moveq     #0,d1
2112     moveq     #0,d2
2113     moveq     #15,d3
2114 .icloop     PUSHM    d0-d3
2115     CWAIT
2116     POPM      d0-d3
2117     PUSHM     d0-d3

```

Listing: JingleMachine


```

2118      move.w    #$182,d0
2119      move.w    d2,d1
2120      andi.w    #$0ff,d1
2121      ori.w     #$f00,d1
2122      CMOVE
2123      POPM      d0-d3
2124      addq.w     #1,d0
2125      addi.w     #$111,d2
2126      dbra      d3,.icloop
2127      move.w     #$ffe,d0
2128      move.w     #$fff,d1
2129      CWAIT
2130      move.l     UCList,a1
2131      move.l     ViewPort,a0
2132      move.l     a1,UCopIns(a0)
2133      CALLINT    RethinkDisplay
2134
2135      ;----- InterruptServer
2136      pri       equ 100
2137      type      equ 3
2138      intnum     equ 5
2139      is_Data    equ 14
2140      is_Code    equ 18
2141      ln_Type    equ 8
2142      ln_Pri     equ 9
2143      ln_Name    equ 10
2144      DataLen    equ 300
2145
2146      move.l     #DataLen,d0
2147      move.l     #$10001,d1
2148      CALLEXEC   AllocMem
2149      move.l     d0,a2
2150      addi.l     #32,d0
2151      move.l     d0,is_Data(a2)
2152      move.b     #pri,ln_Pri(a2)
2153      move.b     #type,ln_Type(a2)
2154      move.l     #IntAnf,is_Code(a2)
2155      move.l     a2,IntStruc
2156      move.l     a2,a1
2157      move.l     #intnum,d0
2158      CALLEXEC   AddIntServer
2159      rts
2160
2161      ;----- Scroll-Int entfernen
2162      RemScroll
2163      move.l     IntStruc,a1
2164      move.l     #intnum,d0
2165      CALLEXEC   RemIntServer
2166      move.l     IntStruc,a1
2167      move.l     #DataLen,d0
2168      CALLEXEC   FreeMem
2169      rts
2170
2171      IntAnf     PUSHM    a0-a6/d0-d7
2172      bsr.s      DoScroll
2173      POPM      a0-a6/d0-d7
2174      clr.l      d0
2175      rts
2176
2177      ;----- Scrolling durchfuehren
2178      DoScroll
2179      tst.w      scdelay
2180      bne.s      .scpause
2181      moveq      #0,d7
2182      move.b     scspeed,d7
2183      .scr2      bsr.s      DoScroll2
2184      dbra      d7,.scr2
2185      rts
2186      .scpause
2187      subq.w     #1,scdelay
2188      rts
2189
2190      DoScroll2
2191      bsr.s      MoveLine
2192      bsr.s      PrLet
2193      move.b     sccount,d0
2194      addq.b     #1,d0
2195      and.b      #31,d0
2196      move.b     d0,sccount
2197      tst.b      d0
2198      beq        NewLetter
2199      rts
2200
2201      slofs      equ 160
2202      scyp       equ 232
2203
2204      MoveLine
2205      move.l     Plane0,a0
2206      lea        (scyp+1)*40(a0),a0
2207      lea        LinBuf,a1
2208      lea        ScpBuf,a3
2209      moveq      #15,d0
2210      .coplin    move.w     #slofs,d1
2211      add.b      d0,d1
2212      add.w      #slofs,(a1)
2213      cmp.w      (a1),d1
2214      bhi.s      .noshi
2215      sub.w      d1,(a1)
2216      move.l     a0,a2
2217      lsl        (a3)
2218      REPT 20
2219      roxl      -(a2)
2220      ENDR
2221      .noshi    lea        40(a0),a0

```



Software-Autoren für die Amiga-Computer gesucht

Haben Sie nicht auch schon einmal daran gedacht, ein gutes Programm, das Sie selbst geschrieben haben, zu veröffentlichen?

Warum sollten nicht auch andere Leser in den Genuß Ihrer Mini-Dateiverwaltung, Grafikerweiterung, Tips, Tricks, Tools, Utilities, Simulationen, Games usw. kommen?

Wirklich gute Software, die den Anforderungen unserer Leser genügt, wird von uns entsprechend honoriert.

Sie sollten jedoch bei der Einsendung Ihres Programms einige Punkte beachten.

Wenn Sie nachstehendes befolgen, wird Ihre Post zügig und ohne große Rückfragen und Verzögerungen bearbeitet.

Senden Sie uns Ihr Programm mit

- (a) allen benötigten Files auf der mit dem Programmnamen bezeichneten Diskette,
- (b) den kompletten Ausdrucken/Listings aller Files der Diskette,
- (c) einer Beschreibung Ihres Programms und
- (d) einer genauen Bedienungsanleitung.

Die Bedienungsanleitung und die Beschreibung sollten als Textdatei mit auf der Programm-Diskette enthalten sein. Wichtig für uns zu wissen wäre noch, mit welcher Konfiguration Sie arbeiten, welchen Drucker Sie benutzen, ob Sie ein zweites Laufwerk angeschlossen haben usw.

Wenn Sie der Meinung sind, ein solches Programm geschrieben zu haben, dann nichts wie einschicken an den

DMV-Verlag
Redaktion AMIGA DOS
Postfach 250
3440 Eschwege

LISTING

```

2222     tst.w    (a1)+
2223     tst.w    (a3)+
2224     dbra     d0,.coplin
2225     rts
2226
2227 PrLet     lea     ScpBuf,a0
2228     move.l    LetPt,a1
2229     move.b    sccount,d0
2230     lsr.b     #2,d0
2231     eor.b     #7,d0
2232     moveq     #0,d1
2233     bset      d0,d1
2234     moveq     #15,d3
2235 .plloop   move.b    (a1),d2
2236     btst      #0,d3
2237     bne.s     .pll2
2238     tst.b     (a1)+
2239 .pll2     and.b    d1,d2
2240     beq.s     .plcont
2241     bset      d3,(a0)
2242 .plcont   addq.l    #2,a0
2243     cmp.b     #8,d3
2244     bne.s     .plc2
2245     tst.b     (a0)+
2246 .plc2     dbra     d3,.plloop
2247     rts
2248
2249 NewLetter  move.l    ScrPt,a0
2250     moveq     #0,d0
2251 newlet2   moveq     (a0)+,d0
2252     cmp.b     #255,d0
2253     beq.s     .tobegin
2254     cmp.b     #1,d0
2255     beq.s     .newspeed
2256     cmp.b     #2,d0
2257     beq.s     .pause
2258     move.l    a0,ScrPt
2259     sub.b     #32,d0
2260     lsl.w     #3,d0
2261     lea       ScFont,a1
2262     lea       (a1,d0),a1
2263     move.l    a1,LetPt
2264     rts
2265
2266 .tobegin  move.l    #ScrTxt,ScrPt
2267     clr.b     scspeed
2268     bra.s     NewLetter
2269
2270 .newspeed move.b    (a0)+,scspeed
2271     bra.s     newlet2
2272
2273 .pause    move.b    (a0)+,d0
2274     lsl.w     #2,d0
2275     move.w    d0,scdelay
2276     bra.s     newlet2
2277
2278
2279 SECTION   JMData,DATA
2280 IntStruc  dc.l 0
2281 ScrPt     dc.l 0
2282 LetPt     dc.l 0
2283 UCList    dc.l 0
2284
2285 sccount   dc.b 0
2286 scspeed   dc.b 0
2287 scdelay   dc.w 0
2288 LinBuf     dcb.w 16,0
2289 ScpBuf     dcb.w 16,0
2290
2291 GFTxt     dc.b 1,0,0,2,1,0,32,0,255
2292
2293 ScrTxt     dc.b 1,4," JINGLE ",2,40," MACHINE ",2,60
2294     dc.b 1,1," - written in 1989/90 by Oliver Hegge
2295     lbacher"
2296     dcb.b 12,32
2297     dc.b 1,2,169,2,5,"1",2,5,"9",2,5,"9",2,5,"0",2,
2298     5,32
2299     dc.b 2,5,"D",2,5,"M",2,5,"V",2,5,32,2,50
2300     dc.b 1,2,"      special greetings to Sven-Silvius
2301     Schad..."
2302     dcb.b 12,32
2303     dc.b 255
2304     EVEN
2305 SECTION   JMCode,CODE
2306
2307     ENDC
2308 ;-----PRINT-Befehle-----
2309 ; Zahl ausdrucken: d0=Zahl d1=Ziffern
2310 ; d2,d3 = Locate , d4,d5 = Farbe
2311 PrNum      PUSHM    d0-d5
2312     lea       PrNBuf,a0
2313     move.b    #1,(a0)+
2314     move.b    d2,(a0)+
2315     move.b    d3,(a0)+
2316     move.b    #2,(a0)+
2317     move.b    d4,(a0)+
2318     move.b    d5,(a0)+
2319     cmp.b     #7,d1
2320     blo.s     prn6
2321     bsr.s     div7st
2322     cmp.b     #6,d1
2323     blo.s     prn5
2324     bsr.s     div6st
2325     cmp.b     #5,d1

```

Listing: JingleMachine

```

2323     blo.s     prn4
2324     divu      #10000,d0
2325     bsr.s     wrchar
2326 prn4      cmp.b    #4,d1
2327     blo.s     prn3
2328     divu      #1000,d0
2329     bsr.s     wrchar
2330 prn3      cmp.b    #3,d1
2331     blo.s     prn2
2332     divu      #100,d0
2333     bsr.s     wrchar
2334 prn2      cmp.b    #2,d1
2335     blo.s     prn1
2336     divu      #10,d0
2337     bsr.s     wrchar
2338 prn1      bsr.s     wrchar
2339     move.b    #0,(a0)+
2340     move.b    #255,(a0)
2341     lea       PrNBuf,a0
2342     bsr.s     Print
2343     POPM      d0-d5
2344     rts
2345
2346 wrchar     add.b    #S30,d0
2347     move.b    d0,(a0)+
2348     clr.w     d0
2349     swap      d0
2350     rts
2351
2352 div7st     moveq     #S30,d4
2353 .divst     sub.l     #1000000,d0
2354     bcs.s     .divwc
2355     addq.b     #1,d4
2356     bra.s     .divst
2357 .divwc     add.l     #1000000,d0
2358     move.b    d4,(a0)+
2359     rts
2360 div6st     moveq     #S30,d4
2361 .divst     sub.l     #100000,d0
2362     bcs.s     .divwc
2363     addq.b     #1,d4
2364     bra.s     .divst
2365 .divwc     add.l     #100000,d0
2366     move.b    d4,(a0)+
2367     rts
2368
2369 PrNBuf     dcb.b 16,0
2370
2371 ; Textstruktur fuer Print:
2372 ; 01,x,y - Locate (0-39,0-31)
2373 ; 02,fg,bg - Farbe
2374 ; 255 - Ende
2375 Print      move.b    (a0),d0
2376     cmp.b     #255,d0
2377     beq.s     prfin
2378     cmp.b     #1,d0
2379     beq.s     ploc
2380     cmp.b     #2,d0
2381     beq.s     pcol
2382     bra.s     pprint
2383 prfin      rts
2384 ploc       move.b    (a0)+,d0
2385     moveq     #0,d0
2386     move.b    (a0)+,d0
2387     lsl.w     #3,d0
2388     move.w    d0,prx
2389     moveq     #0,d0
2390     move.b    (a0)+,d0
2391     lsl.w     #3,d0
2392     move.w    d0,pry
2393     bra.s     Print
2394 pcol       move.b    (a0)+,d0
2395     move.b    (a0)+,txfg
2396     move.b    (a0)+,txbg
2397     bra.s     Print
2398
2399 pprint     move.l    a0,txpt
2400     PUSH      a0
2401     move.l    RastPort,a0
2402     lea       TxtStruc,a1
2403     moveq     #0,d0
2404     moveq     #0,d1
2405     CALLINT   PrintIText
2406     POP       a0
2407 fblank     tst.b     (a0)+
2408     bne.s     fblank
2409     bra.s     Print
2410
2411 ***** Programmvariablen/strukturen *****
2412 SECTION    JMData,DATA
2413
2414 _IntuitionBase dc.l 0
2415 _GfxBase       dc.l 0
2416 _ArpBase       dc.l 0
2417 _CiaaBase      dc.l 0
2418 _WindowHd     dc.l 0
2419 _RastPort      dc.l 0
2420 _ViewPort      dc.l 0
2421 _Plane0        dc.l 0
2422 _Plane1        dc.l 0
2423 _Plane2        dc.l 0
2424 _Plane3        dc.l 0
2425 _UserPort      dc.l 0
2426 _FileHd        dc.l 0

```

Listing: JingleMachine


```

2427 DCode          dc.l 0
2428 OldSInt        dc.l 0
2429 Timer          dc.w 0
2430 CINTDelay       dc.w 4096
2431
2432 SBufAdr  dc.l 0
2433 SBufLen  dc.l 0
2434 SBufOld  dc.l 0
2435 mousex   dc.w 0
2436 mousey   dc.w 0
2437 mousecx  dc.b 0
2438 mousecy  dc.b 0
2439 mouseq   dc.w 0
2440 SampPt   dc.l 0
2441 ASampPt  dc.l 0
2442 EchoCou  dc.w 0
2443 EchoC2   dc.w 0
2444 lestate  dc.w 0
2445
2446 gused     dc.b 0
2447 jnum      dc.b 0
2448 jbnun     dc.b 0
2449 abortfl   dc.b 0
2450 dvflag    dc.b 0
2451 rnum      dc.b 0
2452 Filter    dc.b 0
2453 Decay     dc.b 0
2454 Hold      dc.b 0
2455 Mono      dc.b 0
2456 Phase     dc.b 0
2457 PhAct     dc.b 0
2458 Lesley    dc.b 0
2459 MainVol   dc.b 0
2460 actchan   dc.b 0
2461 EchoDel   dc.b 0
2462 EchoVol   dc.b 0
2463 EchoRep   dc.b 0
2464 EchoSta   dc.b 0
2465 EchoVL2   dc.b 0
2466 prflag    dc.b 0
2467 EVEN
2468
2469 TxtStruc
2470 txfg      dc.b 1
2471 txbg      dc.b 0
2472 txmod     dc.b 1,0
2473 prx       dc.w 0
2474 pry       dc.w 0
2475 txfont    dc.l 0
2476 txpt      dc.l 0,0
2477
2478 JScreen   dc.w 0,0,320,256
2479           dc.w 4
2480           dc.b 1,0
2481           dc.w 2,15
2482           dc.l 0,JSCT,0,0
2483 JSCT      dc.b "Jingle Machine",0
2484           EVEN
2485 ColTab    dc.w $000,$fff,$ccc,$999,$666,$8cf,$58e,$24c
2486           dc.w $00b,$ff0,$fa0,$f60,$c00,$0f8,$0c3,$090
2487
2488 JinWind   dc.w 0,0,320,256
2489           dc.b 1,0
2490           dc.l $10000011000
2491           dc.l $10001101000000000
2492           dc.l 0,0,0
2493 ScreenHd
2494           dc.l 0,0
2495           dc.w 320,256,320,256,15
2496
2497 EditText
2498           dc.b 1,16,22,2,15,11,"LENGTH",0
2499           dc.b 1,23,22,"MEMORY",0
2500           dc.b 1,23,23,2,4,11,"CHIP:",0

```

Listing: JingleMachine



```

2501           dc.b 1,23,25,2,8,11,"FAST:",0
2502           dc.b 1,1,27,2,12,3,"VOL RATE",0
2503           dc.b 1,1,20,2,5,3,"NAME:",0
2504           dc.b 1,9,22,2,10,6,"LOOP",0
2505           dc.b 1,31,22,2,6,14,"LOAD SET",0
2506           dc.b 1,31,23,"SAVE SET",0
2507           dc.b 1,32,24,2,11,14,"CLEAR",0
2508           dc.b 1,31,26,2,12,14,"END EDIT",0
2509           dc.b 255
2510
2511 PlayText
2512           dc.b 1,1,21,2,2,4,"CHAN",0
2513           dc.b 1,1,22,2,10,4,"1234",0
2514           dc.b 1,6,26,2,12,3,"VL DC",0,1,15,26,"LE",0
2515           dc.b 1,18,21,2,4,14,"FILT",0
2516           dc.b 1,18,23,"HOLD",0
2517           dc.b 1,18,25,"MONO",0
2518           dc.b 1,12,26,"PH",0
2519           dc.b 1,35,22,2,12,7,"EDIT",0
2520           dc.b 1,35,26,2,12,7,"QUIT",0
2521           dc.b 1,23,20,2,10,12," ECHO ",0
2522           dc.b 1,23,27,2,10,3,"DL VL",0
2523           dc.b 1,29,22,2,5,11,"C",0,1,29,23,"3",0,1,29,24
           , "2", 0
2524           dc.b 1,29,25,"1",0,1,29,26,"0",0
2525           dc.b 255
2526
2527 IntName    INTNAME
2528 GfxName    GRAFNAME
2529           EVEN
2530
2531 ; Filerequester
2532 SamReq     dc.l SamRTex
2533           dc.l SamFile
2534           dc.l SamDir
2535           dc.l 0
2536           dc.b 64+32,0
2537           dc.l 0
2538           dc.l 0
2539 SamRTex    dc.b "Load Sample",0
2540 SamFile    dcb.b 34,0
2541 SamDir     dcb.b 34,0
2542           EVEN
2543
2544 SetReq     dc.l SetRTex
2545           dc.l SetFile
2546           dc.l SetDir
2547           dc.l 0
2548           dc.b 0,0
2549           dc.l 0
2550           dc.l 0
2551 SetRTex    dc.b "Save Settings",0
2552 SetFile    dcb.b 34,0
2553 SetDir     dcb.b 34,0
2554           EVEN
2555
2556 SetKenn    dc.l 0
2557 DatName    dcb.b 70,0
2558 SetFN      dcb.b 34,0
2559 SetDR      dcb.b 34,0
2560
2561 ; Datenspeicher reservieren
2562           SECTION JMBuff,BSS
2563 PrStr      ds.b 80
2564 Sample     ds.b slen*sanz
2565 Gadgets    ds.b glen*ganzz
2566 GadBuf     ds.b ganz
2567 ScFont     ds.b 224*8
2568           EVEN
2569 Channel    ds.b chlen*chanzz
2570
2571 ; that's all folks
2572 END

```

Listing: JingleMachine

SEHR GUT,*
jetzt noch besser!

Digi Tiger II
der ganz besondere
Videodigitizer

- Jetzt in der Version 2.0
 - 2 - 64 Farben-Modus
 - optimierter HAM-Modus
 - Farbbilder in 25 - 85 s, alle Auflösungen
 - SW-Bilder in 1 - 5 s
 - integrierter RGB-Splitter
 - Hotline und Update-Service
 - Infos gratis, Demodiskette für 10 DM Schein
- weiterhin nur 698,-**

* siehe Kickstart 5/90

**Funkbilder für IBM-kompatible PCs XT/AT, C64 und 128,
AMIGA und Atari ST
Fernschreiben, Morsen und
Radio-Kurzwellen-Nachrichten.**

Haben Sie schon einmal das Piepsen von Ihrem Radio auf Ihren Bildschirm sichtbar gemacht? Hat es Sie schon immer interessiert wie man Wetterkarten, Meteosat-Bilder, Wetter-Nachrichten, Presseagenturen, Botschaftsdienste usw. auf einem Computer sichtbar macht? Ja? -, dann lassen Sie sich eine Einsteiger-Information schicken oder bestellen Sie einfach gleich.

Steckfertige Karten mit eingebautem Filterkonverter. Alle gängigen Betriebsarten, selbsterkennende Auswertung und Abstimmung. Stufenlose Shiften und Baudrate. Sondereinheit für verschlüsselte Sendungen und Codeanalyse. Drucken, Speichern, automatische Aufzeichnung. Senden und Empfangen von Funkfernschreiben, Morsezeichen und Faksimile-Bilder. Aufzeichnen, Überarbeiten, Speichern und Drucken.

Unser Angebot - eine Komplett-Ausrüstung mit Anleitung für den Einsteiger für Funkfern-schreib-, Morse- und Bilder-Empfang. Modul einstecken, mit Lautsprecheranschluß verbinden, einschalten und los geht's.

Super-Sonder-Angebot

BONITO-Supercom ab 298,00 DM

Bitte Info Nr. 2 anfordern bei

Bonito, Peter Walter

Gerichtsweg 3, 3102 Hermannsburg, Telefon 050 52/60 52



Liebe Leser,
in dieser Rubrik finden Sie neben Informationen zu den in der AMIGA DOS vorgestellten Programmen und Produkten Rat und Hilfestellung zu Ihren kleineren und größeren Programmierproblemen. Wir sind natürlich jederzeit bemüht, die eingehenden Leserfragen zu beantworten. Doch haben Sie bitte Verständnis, daß wir nicht alle eingehenden Briefe persönlich beantworten können. Oft erreichen uns mehrere Briefe zum gleichen Thema, einer davon wird dann stellvertretend für die anderen in unserer Zeitschrift beantwortet.

Dringende Probleme, die das Heft betreffen, lassen sich möglicherweise besser telefonisch regeln. Rufen Sie dienstags unsere Hotline an. Von 17 bis 20 Uhr stehen wir Ihnen mit Rat und Tat zur Seite, wenn Sie eine der folgenden Nummern wählen:
0 56 51/8 09-7 40 (bis 744)

Ihre AMIGA-DOS-Redaktion

SIE FRAGEN, WIR ANTWORTEN

In unserer Leserbriefecke veröffentlichen wir Anfragen, die für die Leser der AMIGA DOS interessant sind. Falls Sie uns schreiben wollen, richten Sie Ihre Zuschrift an die folgende Adresse:

**DMV-Verlag
Redaktion AMIGA DOS
Leserservice
Postfach 250
3440 Eschwege**

Wenn Sie eine Antwort zur Problemecke haben, schicken Sie sie unter dem Kennwort "Problemecke" ein.

Eine Bitte haben wir noch: Bitte schreiben Sie Ihre vollständige Adresse nicht nur auf den Umschlag, sondern auch in den Brief. Legen Sie am besten einen an Sie selbst adressierten, ausreichend frankierten Rückumschlag bei. Nun aber zu den Briefen.

Copyright auf Musikstücke

Ich bin Musiker und habe meine erste CD herausgebracht. Seit längerem besitze ich einen Amiga 500, und ich habe mir überlegt, ihn zu musikalischen Projekten einzusetzen. Software-Angebote für Musik gibt es ja genug. Nun aber folgende Frage: Inwieweit darf ich die Programme denn nun anwenden?

Darf ich die Musik-Sounds aus Programmen wie zum Beispiel »Aegis Sonix« oder »Dynamic Drums« für eine zum Verkauf angebotene Platte verwenden? Ich denke mir, daß ich über MIDI- oder Amiga-Tastatur sowieso eigene Musikstücke erstelle. Bezieht sich die auf den Programmen angepriesene Urheberrechtsangabe nun auf das Programm oder kann man mir auch verbieten, die Klangfarben des Programms zu verwenden? Wenn ja, könnte ja keiner "den Hit" schreiben, es sei denn, er würde alle Programme selber schreiben.

Thomas Janak,
Offenbach/Main

Die rechtliche Seite sieht so aus, daß auch Musikstücke bei einem Musikprogramm dem Urheberrecht unterliegen und nicht ohne weiteres kommerziell genutzt werden dürfen. Selbst wenn Sie nur bestimmte Instrumente benutzen wollen, müssen Sie bei dem Hersteller nachfragen. Eine private Nutzung ist ohne Nachfrage möglich.

(Red.)

Star LC-10 und Amiga

In der Ausgabe 7/90 stand ein Aufruf für die Lösung eines Druckerproblems. Dort hatte Herr Andruschow Probleme

mit einem Star LC-10 Farbdrukker. Ich selbst besitze den gleichen Drucker, also könnte ich ihm Tips geben. Er könnte mal probieren, das Farbband zu erneuern. Mit einem neuen Farbband lassen sich etwa vier hochwertige Bilder erzeugen, danach ist es nur noch zu gebrauchen, um einen globalen Überblick zu erhalten. Es hat noch niemand gesagt, daß Farbbänder lange halten. Die Ursache dafür ist, daß die aufgetragenen Farben sich beim Drucken zurück auf das Farbband schmieren. Daß manche Gebiete heller erscheinen, hat auch mit der Abnutzung des Bandes zu tun.

Herr Andruschow sollte sich in der »Preferences« auf jeden Fall den Treiber vom »Epson JX-80« auswählen. Dieser steht nicht auf der Workbench 1.3, sondern nur auf der Extras-Diskette der Workbench 1.2. Am besten wäre es jedoch, wenn er einen Treiber für den Star LC-10 hätte.

Der Zufall wollte, daß ich auf einer Diskette bei dem englischen Blatt »Amiga Computing«, Monat August, drei fertige Workbench-Printertreiber für den Star LC-10 fand. Es ist mit diesen Druckertreibern möglich, in verschiedenen Modi zu drucken, wie Farbe, Graustufen oder Schwarzweiß. Eine andere Lösung könnte das Programm »Turbo Print II« oder »Turbo Print Professional«

bieten. Diese Programme unterstützen den Star LC-10 hundertprozentig.

C.Bulgarelli,
Maastricht (Niederlande)

Vielen Dank für die Hinweise. Interessierte Leser können sich im Buch- oder Zeitschriftenhandel die entsprechende Ausgabe besorgen.

(Red.)

Zum Leserbrief des Herrn K. Bühler

Liebe Redaktion, den Tips dieses Autors muß ich zum großen Teil widersprechen! Im einzelnen:

1. Man spart keine Zeit, wenn man den Monitor ausschaltet! Der Amiga hat dafür keinerlei Sensoren. Man spart dagegen Zeit, wenn man dem Prozessor weniger zu arbeiten gibt. Also nicht den Screen mit den meisten Bitplanes in den Vordergrund holen, alle Screens am besten mit der Maus ganz unter den unteren Rand ziehen.
2. Zur Abfrage der Mausposition bietet das AmigaBASIC die voll ausreichende Funktion »MOUSE (0/1/2)«. Es ist gegen alle Regeln sauberen Programmierens, direkt in irgendwelchen Systemstrukturen herumzupeeken und -poken, wenn man das gar nicht nötig hat.

Die Problemecke

Manche Probleme lassen sich selbst in der Redaktion nicht zufriedenstellend lösen. Zu diesem Zweck haben wir die Problemecke eingeführt, in der Leser spezielle Fragen an andere Leser richten können. Wer eine Antwort auf die Fragen hat, schickt diese bitte unter dem Stichwort "Problemecke" an die Redaktion (Anschrift siehe Einleitung).

Programme für Amateurfunker

Seit der Ausgabe 1/90 bin ich begeisterter Leser ihrer Zeitschrift und habe eine Anregung, die sicherlich auf großes Interesse stoßen wird. Ich suche für den Amiga 500 Programme, die dem Amateurfunker helfen. Da wären zum Beispiel: Fernschreiben senden und empfangen, Morsedekoder, Logbuchführung und so weiter.

Da es für den C-64 solche Programme in Hülle und Fülle gibt, müßte es so etwas

auch für den Amiga geben. Wer kann helfen?

Erich Schirle,
Aalen

Kein Kick mit Doppelklick

Seit einiger Zeit hängt sich mein Amiga 500 auf der Workbench bei jedem zweiten oder dritten Doppelklick auf. Das sieht dann so aus, daß nach dem Doppelklick auf ein Icon erst einmal gar nichts passiert. Wenn ich dann die rechte Maustaste drücke oder mit der linken ein anders Icon beziehungsweise Gadget oder sonstwas anklicke, friert der Mauszeiger und alle anderen Tasks fest. Wenn ich jetzt die Diskette aus dem Laufwerk nehme und wieder einlege, tut der Amiga in dem Moment, in dem die grüne LED aufleuchtet, so, als wäre nichts passiert, und lädt das angeklickte Programm. Wenn ich aber meinen Drucker (Präsident 6320) angeschlossen und auf »Online« habe, passiert mir das nie. Außerdem funktionieren einige Programme (zum Beispiel Textcraft) nur, wenn ich den Drucker auf »Online« habe, sonst hängt sich der

Computer wie in der Workbench auf. Am Drucker kann es nicht liegen, wenn ich den Druckerstecker abziehe, funktioniert es auch nicht. Die Ausdrücke von sämtlichen Programmen, die über das »Printer-Device« ausdrucken, kann man vergessen, da ungefähr die Hälfte fehlt. Den richtigen Druckertreiber habe ich natürlich eingestellt. Ein Virus kann es nicht sein, denn bei Freunden funktioniert es einwandfrei.

Wer kann mir einen Tip oder Hilfe geben?

Frank Barwich,
Bleckede

Vortex-Laufwerk am Amiga

Aus meiner »CPC-Zeit« besitze ich noch ein Vortex F1-X-Laufwerk. Natürlich würde ich gerne dieses Laufwerk als externes Zweitlaufwerk betreiben. Ich bin aber kein versierter Bastler. Eine Anfrage bei der Firma Vortex verlief negativ. Wissen Sie eine Firma oder eine Adresse, die den Umbau vornehmen kann?

Bodo Staron,
Garching

Diverses

Ich habe zwei Fragen, die den Amiga betreffen.

1. Wie kann ich die Laufwerks-LED an- und ausschalten (Laufwerk DF0: bis DF3:)?

2. Wie kann ich die externe 2-MByte-Speichererweiterung exklusive als RAM-Floppy verwenden?

Edwin Schneelee,
Frankenthal

Uhr verschwunden

Vor ein paar Tagen hatte ich auf meinem Amiga 2000 wieder einmal einen jener Katastrophenabstürze. Totaler Grafiksalat und der Guru wollte auch nicht mehr... Nun, seitdem ist die interne Uhr verschwunden. Bei jedem »SetClock« Load/Save bekomme ich ein "Battery Backed up Clock not found" zurück. Es geht zwar auch ohne Uhr, aber irgendwie vermisste ich sie doch. Also, wer kann mir sagen, wie ich sie wieder zum Leben erwecke?

Sönke Tesch,
Elmshorn

3. Dasselbe gilt für die Zentrierung von Text. Reicht einem eine zeichenweise Zentrierung, kommt man durch Anwendung von »Locate« und »Tab« schon hin. Und wenn man pixelweise positionieren will, muß man gefälligst den Cursor mit »Move« (Graphics.Library) setzen und nicht durch direktes Poken!

Petter Kittel,
Frankfurt

Raubkopie getestet?

Liebe Redaktion, es lief mir kalt den Rücken herunter, als ich in der AMIGA DOS 7/90, Seite 108, das Bildschirmfoto von »Yuppi« entdeckte. Dort stand zu lesen »Cracked by Helix«. Haben Sie etwa eine Raubkopie getestet?

Der Artikel soll ja angeblich PD-Programme testen. Es wäre nicht das erstemal, daß ein geklautes Programm ein-

fach zu PD erklärt wird, ohne den Autor zu fragen. (Mit Commodore Textcraft und gewissen Kickstart-Versionen ist das illegalerweise schon passiert!)

Peter Kittel,
Frankfurt

Auf den ersten Blick scheint das Foto von einer Raubkopie zu sein. Doch Untersuchungen zeigten, daß das Programm vom gleichen Autor geschrieben wurde, der den Satz »Cracked by Helix« in das Spielfeld geschrieben hat. Warum der Autor das getan hat, wird auch uns ein Rätsel bleiben. Vielleicht sollte es eine Art Aufwertung des Programms sein.

(Red.)

Was kostet ein Amiga?

Ich bin sehr an einen Amiga 500 interessiert. Jedoch muß ich gestehen, daß ich keine Ahnung von Computern habe.

Deshalb meine Frage: Was benötige ich als Einsteiger alles für den Amiga 500? Welche Software empfiehlt sich am besten; ich möchte gerne spielen und auch damit arbeiten (Textverarbeitung und so weiter). Wieviel wird das Ganze dann kosten, und wo kann ich den Amiga günstig beziehen? Lohnt es sich, einen Amiga 500 privat gebraucht zu kaufen, und was müßte ich dabei beachten?

Helmut Reitberger,
München

Als Einsteiger ist man oft überfordert, was man für einen Einstieg in die Computer-Szene benötigt. Hier erst einmal ein paar Hinweise für die Vorgehensweise.

Für den Anfang reicht das Grundgerät aus, das Sie für zirka 850 DM in Kaufhäusern bekommen. Dann empfiehlt sich ein zweites Diskettenlaufwerk für zirka 200 DM. Hinsichtlich weitergehender Fra-

gen oder irgendwelcher Umbauarbeiten am Computer ist man beim Fachhändler besser aufgehoben, da dieser oft kompetenter als die meisten Kaufhäuser ist.

Wer von Privat einen Computer kaufen will, der muß nur darauf achten, daß der Computer richtig läuft und daß alle benötigte Software für den Betrieb dabei ist.

Welche die beste Software ist, läßt sich nicht ohne weiteres sagen, da es Geschmackssache ist. Eine Übersicht der Textverarbeitungsprogramme finden Sie in der AMIGA DOS 7/90. Was die neuesten Spiele angeht, so finden Sie diese monatlich in der AMIGA DOS beschrieben.

(Red.)

Die Redaktion behält sich vor, Leserbriefe in gekürzter Form wiederzugeben.

Franz-Josef Reichert

Vielseitige Requester (Teil 3)

Heute erfahren Sie, wie die Intuition-Schnittstelle in die verschiedenen Sprachen eingebunden werden kann.

Halbzeit! In den letzten beiden Folgen unserer Artikelserie hatten wir Ihnen drei nützliche Listings vorgestellt, die Ihnen bei der Entwicklung eines komfortablen Intuition-Benutzer-Interfaces für Ihre eigenen Programme gute Dienste leisten konnten. Bisher kamen allerdings nur die C-Programmierer unter Ihnen in den Genuß dieses Komforts. Heute wollen wir unser zu Beginn gegebenes Versprechen wahr machen und alle Routinen zu einer echten »shared-library« zusammenfassen, wonach sie aus jeder Programmiersprache heraus angesprochen werden können.

Lediglich eine Voraussetzung muß dazu erfüllt werden: Interpreter beziehungsweise Compiler und Linker Ihres Entwicklungssystems müssen die Möglichkeit bieten, Aufrufe von Library-Routinen nach Amiga-Konvention einzubinden. Uns ist aber keine Programmierumgebung bekannt, in der dieses ausgeschlossen wäre. Selbst im von Commodore mitgelieferten AmigaBASiC-Interpreter wurde dieses Feature von der ersten Stunde an unterstützt.

Über Libraries und Devices...

Zunächst wollen wir einige grundsätzlichen Dinge zur Library-Programmierung klären. Als offenes System bietet der Amiga nicht nur auf Hardware-Ebene nahezu unbegrenzte Erweiterungsmöglichkeiten, wie beispielsweise durch das kaum noch überschaubare Angebot an Einsteckkarten für die unterschiedlichsten Anwendungen, sondern auch auf Software-Ebene. Seine Entwickler haben ihm hier nicht nur ein leistungsfähiges Multitasking-Betriebssystem mit auf

den Weg gegeben, sondern außerdem mit der consequenten Verwirklichung eines durchdachten und modularen Konzeptes der System-Software dafür Sorge getragen, daß zukünftige Erweiterungen und Verbesserungen ohne Probleme zu bewerkstelligen sind. Grundlegende Elemente der Amiga-Philosophie sind kleine, austauschbare, erweiterbare und wartungsfreundliche Software-Module, die »Libraries« und die »Devices«. Damit wurde das Betriebssystem nicht nur in überschaubare Funktionsbereiche gegliedert, sondern auch für die Zukunft beliebig ausbaufähig gemacht. Die Devices zur Kommunikation mit Hardware und Peripheriegeräten, die eigentlich nur den Spezialfall einer Library darstellen, sollen uns hier nicht weiter interessieren. Weitere Informationen zu diesem Themenkomplex finden sich zuhauf in [1], Kapitel 31-42 sowie [2], Teil B und [6].

Die Bibliotheken für die Funktionen

Wir wollen uns hier mit dem Grundtypus auseinandersetzen, der Library im Amiga. Was ist aber eigentlich eine »Function Library«, wie sie mit vollem Namen heißt? Ins Deutsche übertragen bedeutet dieser Begriff »Funktionen-Bibliothek«. Bei dem Wort Bibliothek denkt jeder sofort an eine Leihbibliothek. Genauso, wie man sich hier Bücher zum Lesen ausleihen kann, können sich Programmierer in einer Funktionen-Bibliothek Routinen zur Benutzung ausleihen, die sie ansonsten selber schreiben müßten. Warum aber für häufig auftretende Programmieraufgaben das Rad jedesmal neu erfinden? Wenn ein Pro-

grammierer in seinem Programm ein bestimmtes Ziel verfolgt, können ihm vorgefertigte Funktionen aus der Library die Arbeit ganz oder zumindest teilweise abnehmen, außerdem bietet dieses Konzept noch eine Vielzahl weiterer Vorteile: In einem Multitasking-System können Funktionsbibliotheken mehreren quasi gleichzeitig laufenden Programmen (Tasks) zur Verfügung stehen, sie brauchen aber nur einmal in den Arbeitsspeicher geladen zu werden. Für diese Eigenschaft wird auch oft der Namenszusatz »shared« verwendet, was übersetzt geteilt bedeutet. Auch beansprucht eine solche Library nur einmal Platz auf einem Massenspeichermedium, sei es im ROM, auf Festplatte oder Diskette. Sie braucht nur geladen zu werden, wenn ein Programm sie tatsächlich benötigt, die Library wird dann geöffnet.

Solange aber kein Programm Funktionen aus dieser Library

benötigt, kann sie bedenkenlos aus dem Arbeitsspeicher entfernt werden, man spricht vom Schließen der Library. Insgesamt ist das Library-Konzept ein gewaltiger Schritt nach vorne, wenn es um effizientes Haushalten mit System-Ressourcen geht.

Die Frage lautet »resident oder transient?«

Routinen für das Öffnen und Schließen von Libraries stellt uns die »exec.library«, bezeichnenderweise selber eine Library, mit ihren beiden Funktionen »OpenLibrary()« und »CloseLibrary()« zur Verfügung. Aber woher weiß »Exec«, welche Library geöffnet werden soll? Als echte Funktion erwartet »OpenLibrary()« natürlich Aufrufparameter und liefert ein Ergebnis zurück. Der erste Parameter ist sinnvollerweise der Name der zu öffnenden Library, der zweite eine Versionsnummer, dazu später noch mehr. »Exec« sucht also nach der Library und liefert bei Erfolg als Resultat die Basisadresse der Library zurück. Das ist ein Bezeichner, unter dem die Library angesprochen werden kann, er dient auch als Parameter für die Funktion »CloseLibrary()«, wenn die Arbeit mit der Library beendet ist. Aber wo befinden sich überhaupt die Libraries?

Hier müssen wieder zwei Fälle unterschieden werden: Zum einen enthält das Betriebssystem-ROM des Amigas bereits einige fest verdrahtete Libraries, die beim Systemstart eingebunden werden. Sie werden als residente (permanent gespeicherte) Libraries bezeichnet, da sie ständig zur Verfügung stehen und auch nicht mehr entfernt werden können. Eine andere Kategorie stellen die sogenannten transienten (vorübergehend gespeicherten) Libraries dar. Sie werden nur bei Bedarf von Diskette oder einem anderen Massenspeicher eingebunden und verfügbar gemacht. Das Betriebssystem sucht sie an einer ganz bestimmten Stelle, nämlich im Directory »libs« der Startdiskette. Dies ist das Tor für jede Library-Erweiterung. Alle dort befindlichen Programmmodule, die bestimmten Konventionen genügen, können als neue Li-

Eine kleine Übersicht:

Unser mehrteiliger Werkstatt-Artikel beschäftigt sich mit der Erstellung von eigenen Libraries. In den Beiträgen finden Sie folgende Themen:

- Überblick, Beschreibung und Listings des File-Requesters
- Beschreibung und Listings zu Auswahlbox und Colorpalette
- Wissenswertes zu Libraries, Listing des Library-Rumpfes, Zusammenbau aller Routinen als Amiga-Library
- Anwendungsbeispiele, Linker-Library, Aufruf aus allen Programmiersprachen

library geladen und benutzt werden (siehe [3], Seite XII ff.). Die Funktionen zum Öffnen und Schließen nehmen auch gleichzeitig die Aufgabe eines Schutzprotokolls wahr: Zugriffe auf Funktionen der Library dürfen von einem Programm aus erst nach erfolgreichem Öffnen stattfinden. Eine geöffnete, transiente Library darf auch nicht mehr entfernt werden. Hat jedoch das letzte Programm seine Arbeit mit der Library beendet, kann der beanspruchte Speicherraum im Bedarfsfalle wieder anderen Zwecken zugeführt werden, die Library wird entfernt.

In diesem Zusammenhang noch eine Warnung: Auch wenn es bei den residenten Libraries scheinbar keinen Sinn macht, sollten die genannten Aufrufkonventionen trotzdem unbedingt eingehalten werden und alle Anforderungen auf den erwarteten Rückgabewert überprüft werden! Es könnte ja durchaus sein, daß zukünftige Betriebssystemversionen eine bislang im ROM residente Library aus Platzmangel oder sonstigen Gründen auf Diskette auslagern. Wer findet dann noch die fälschlicherweise ganz sicher im ROM vermutete Library? Nur derjenige, der sie auch ordnungsgemäß öffnet! Das gleiche gilt auch für das Schließen: Eine nicht geschlossene, transiente Library hängt nutzlos herum, obwohl sie keiner mehr wirklich benötigt, und beansprucht unsinnigerweise wertvolle Ressourcen. Alle Eventualitäten sind dagegen nur durch die konsequente Einhaltung der Aufrufkonventionen abgedeckt, also ein wichtiger Schritt in Richtung zukünftssichere Programmierung (siehe auch [2], Teil A und [4], Kap. 2.1.2).

Als aufmerksame Leser haben Sie natürlich sofort einen Widerspruch im letzten Abschnitt entdeckt! Da »Exec«, wie ja bereits oben gesagt, selbst eine Library ist, wie können wir dann ihre Funktionen benutzen, ohne sie selbst vorher explizit geöffnet zu haben? Hinter dieser scheinbaren Problematik verbirgt sich aber nichts Geringeres als der Schlüssel zum gesamten Amiga-Betriebssystem: »Exec« als zentralste und wichtigste Library verfügt als einzige über einen absoluten Zeiger auf ihre Basisadresse, diese Adresse wird

beim Systemstart in der Speicherstelle vier abgelegt! Dies ist aber auch wirklich die einzige Ausnahme, welche die Benutzung einer Library unter Umgehung des ansonsten immer geforderten Schutzprotokolls erlaubt. Für alle anderen Systemadressen sollten niemals irgendwelche absoluten Annahmen gemacht werden!

Funktionseinsprünge relativ zur Basisadresse

Mit dem Öffnen der Library haben wir den ersten Schritt getan, um uns ihre Funktionen zugänglich zu machen. Wie werden nun die eigentlichen Funktionsaufrufe realisiert? Aus dem vorher Gesagten folgt, daß auch Library-Funktionseinsprünge keine festen Adressen zugewiesen werden können. Dies ist noch aus einem ganz anderen Grund unmöglich: Im Amiga wird der Speicherplatz dynamisch verwaltet, und kein Programm (und damit auch keine Library) kann vorhersehen, an welcher Adresse es letztlich im Arbeitsspeicher geladen wird. Hier

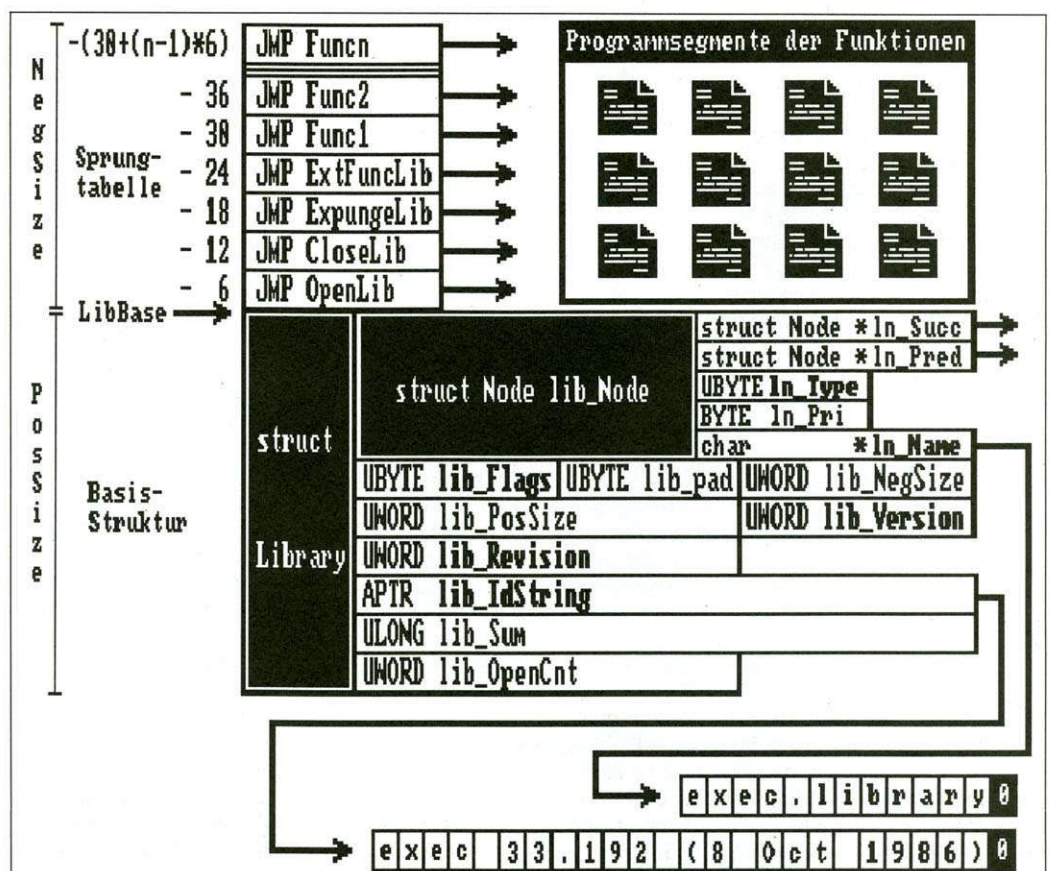
kommt uns die Library-Basisadresse zu Hilfe. Nach den einschlägigen Programmierrichtlinien erfolgen Funktionseinsprünge über die Sprungtabelle relativ zur Basisadresse der Library mit negativem Offset (siehe [1], Kap. 14, Seite 229 ff.). Was bedeutet dies konkret? Statt eine Funktion direkt anzuspringen, verzweigen wir zunächst an eine bestimmte Stelle im Speicherbereich vor der Basisadresse der Library. Hier treffen wir für jede Funktion auf eine unbedingte Sprunganweisung (»JMP <absexp>«), die in die gewünschte Routine verzweigt. Da eine Sprunganweisung mit 32-Bit-Adresse (»<absexp>«) genau sechs Bytes Speicherplatz beansprucht, unterscheiden sich die Offset-Nummern aufeinanderfolgender Funktionen jeweils um den Wert sechs. Es genügt also, den (negativen) Funktions-Offset und die Adresse der Basisstruktur zu kennen, um eine Library-Funktion aufzurufen. Die Basisadresse wird nach Konvention immer in das Prozessorregister A6 geladen. Der Aufruf der Exec-Funktion OpenLibrary() gestaltet sich

also in Maschinensprache wie folgt:

```
1 move.l $4,a6 ; Basisadresse laden
2 jsr _LVOOpenLibrary(a6) ; relative Verzweigung
```

Basisstruktur einer Library

Der Offset (»_LVOOpenLibrary«) ist extern definiert und beträgt \$fdd8 (entspricht -552), das Präfix _LVO steht für Library Vector Offset. Nach Durchlauf sorgt ein abschließendes »rts« in der Library-Funktion für die Rückkehr zum aufrufenden Programm. Wie sieht nun die eigentliche Basisstruktur einer Library aus, deren Adresse die Funktion »OpenLibrary()« zurücklieft? Im (negativen) Speicherbereich vor der Struktur befindet sich, wie bereits festgestellt, die Sprungtabelle auf die Library-Funktionen. Im positiven Bereich finden wir die »Struktur Library« selbst (Bild 1). Sie stellt einen Datenbereich für Verwaltungsaufgaben zur Verfügung, im folgenden wollen wir die Bedeutung der einzelnen Felder erläutern (siehe auch [1], Kap. 14, Seite 236; [3], Kap. 1, Seite 26 ff.):



Die Struktur-Library im Amiga

»lib_Node« ist eine Node-Struktur, die zur Verkettung in der Library-Liste von Exec dient. Über diese Liste werden Libraries gefunden, die bereits in den Speicher eingeladen sind. Die Node-Struktur enthält einen Zeiger »In_Name«, der auf den Namen der Library verweist.

»lib_Flags« enthält Kontrollbits, die Auskunft über den momentanen Zustand der Library geben. Die zur Zeit verwendeten Flags sind im Anschluß beschrieben.

»lib_Pad« dient nur zur Ausrichtung der Struktur auf Langwortgrenze.

»lib_NegSize« beschreibt die Größe der Sprungtabelle in Bytes.

»lib_PosSize« steht für die Größe der eigentlichen Library-Struktur. Diese muß nicht immer konstant sein, sie kann im Bedarfsfalle beliebig erweitert werden.

»lib_Version« ist die Versionsnummer der Library. Sie wird beim Öffnen der Library überprüft und kann als Parameter beim Aufruf von »OpenLibrary()« angegeben werden. Ist die gefundene Versionsnummer zu niedrig, schlägt der Aufruf fehl. Eine neue Versionsnummer wird immer dann fällig, wenn die Library derart überarbeitet wurde, daß sie nicht mehr abwärtskompatibel zu den Vorgängerversionen ist (aufwärtskompatibel muß dagegen jede Library sein, das heißt, einmal dokumentierte Features werden auch in den Nachfolgeversionen weiterhin unterstützt!). Durch Prüfen der Versionsnummer können Programme Inkompatibilitätsrisiken schon im Vorfeld erkennen und sicher vermeiden. Sinnvolle, wichtige Versionsnummern sind beispielsweise:

0 jede beliebige Version
32 Kickstart 1.1 PAL (nur noch historische Bedeutung)
33 Kickstart 1.2 (alte Amiga 500/1000/2000)
34 Kickstart 1.3 (entspricht 1.2 mit Autoboot)
36 Kickstart 2.0
(siehe auch [1], Kap. 14, S. 231 und [4], Kap. 4.6 S. 97 ff.)

Als Faustregel bei der Angabe von Versionsnummer gilt immer: So hoch wie nötig, so niedrig wie möglich! Gerade beim Sprung von V_33 nach V_34 ist die explizite Anforderung von V_34 nur dann wirklich notwendig, wenn die neu hin-

zugekommene Autoboot-Eigenschaft des Betriebssystems Voraussetzung für eine sichere Funktion des Programms ist. Dies dürfte allerdings nur für spezielle Gerätetreiber wesentliche Bedeutung haben. Vorschnell für alle Programme gefordert, hat die V_34 (und spätere) dagegen den Effekt, daß eben diese Programme nicht auf Amigas mit Kickstart 1.2 laufen. Besonders bei Verwendung des Symbols »LIBRARY_VERSION« aus dem Include-File »exec/types.h« sollte man sich darüber Gedanken machen.

»lib_Revision« ist die Revisionsnummer der Library. Bei jeder kleineren Überarbeitung durch den Systemprogrammierer wird diese erhöht. Dieser Eintrag hat nur statistische Bedeutung und ist für den Anwendungsprogrammierer ebenso wie das nächste Feld weitestgehend uninteressant.

»lib_IdString« zeigt auf einen Identifikations-String, der nähere Informationen zu Erstellungsdatum und Versionsnummer im Klartext enthält, zum Beispiel »intuition 33.702 (30 Sep 1986)«.

»lib_Sum« ist die Prüfsumme der Library zur Konsistenzkontrolle, siehe weiter unten ([3], Kap. 1, Seite 151 ff.).

»lib_OpenCnt« wird jedesmal um eins erhöht, wenn ein Task die Library öffnet. Analog erzeugt ein Schließen der Library die Subtraktion von eins.

Es folgt die Bedeutung der Flagbits für »lib_Flags«. Diese werden ausschließlich vom Betriebssystem behandelt (siehe [3], Kap. 1 Seite 107 ff.).

»LIBB_SUMMING (0)« Dieses Bit zeigt an, daß gerade eine neue Prüfsumme für die Library errechnet wird. Die Prüfsumme wird wortweise über die im RAM befindliche Sprungtabelle (von Library-lib_NegSize bis Library) gezogen, über sie läßt sich feststellen, ob alle Datenfeldinhalte noch gültig sind oder ob sie neu initialisiert werden müssen, weil sie etwa durch einen Systemabsturz in Mitleidenschaft gezogen wurden.

»LIBB_CHANGED (1)« zeigt an, daß die Prüfsumme aufgrund einer gewollten Änderung an der Library nicht mehr gültig ist. Anstelle eines »Inkonsistenz-Alerts« (Guru #81000003) bei Prüfung mit

negativem Ergebnis wird sie von »Exec« neu errechnet und in »lib_Sum« eingetragen.

»LIBB_SUMUSED (2)« Durch das gesetzte Flag wird bestimmt, daß die Prüfsumme überhaupt berücksichtigt wird. Bei gelöschtem Bit wird die Prüfsumme ignoriert und permanente Konsistenz der Library vorausgesetzt, ohne sie explizit zu überprüfen. »LIBB_DELEXP (3)« zeigt an, daß die Library bei nächster Gelegenheit entfernt werden soll (steht für delayed expunge, das heißt verzögerte Löschung). Dies kann natürlich nur dann stattfinden, wenn kein Task sie mehr geöffnet hält, »lib_OpenCnt« also Null ist. Dieses Flag wird gesetzt, wenn das Betriebssystem aufgrund von Speicherplatznöten eine »garbage collection« durchführt, wobei nicht mehr benötigte Strukturen, also etwa unbenutzte Libraries, Devices, Fonts und so weiter, sofort entfernt werden und der von ihnen beanspruchte Speicherplatz freigegeben wird (siehe [4], Kap. 2.8.1, Seite 56 ff.).

Damit haben wir die Verwaltungsstrukturen und -mechanismen der Libraries abgehandelt. Was uns für eine komplette Library noch fehlt, sind die eigentlichen Library-Funktionen, ohne die das vorher Beschriebene ja wenig Sinn macht. Welche Anforderungen werden an sie gestellt?

Wiedereintrittsfähige Funktionen

Zunächst müssen wir berücksichtigen, daß in einer »shared library« mehrere Tasks quasi gleichzeitig ein- und dieselbe Funktion abarbeiten können. Damit dies reibungslos funktioniert, muß ihr Programmsegment (Code- und Datenbereich) einer ganz speziellen Eigenschaft genügen, die im Fachjargon »wiedereintrittsfähig«, »reentrant« oder mit »pure code« bezeichnet wird. Folgende Vorgaben müssen dazu erfüllt werden:

– Auf globale Variablen darf niemals willkürlich schreibend zugegriffen werden, auch statische Variablen im Datensegment dürfen nur gelesen werden. Wenn unbedingt globale Kontrollstrukturen angelegt werden müssen, so muß für

den Zugriff auf diese ein Schutzprotokoll definiert werden, das gegenseitige Störungen der konkurrierenden Tasks ausschließt (siehe [1], Kap. 20, Seite 299 ff.).

– Lokale, modifizierbare Variablen müssen im Stack-Bereich (welcher ja zum Stack-Bereich des aufrufenden Tasks gehört) angesiedelt sein oder in einem dynamisch allokierten Datensegment. Letzteres vor allem bei größerem Platzbedarf oder speziellen Anforderungen an den Speichertyp (zum Beispiel »MEMF_CHIP« für Grafik- und Sounddaten oder Langwortausrichtung für DOS-Strukturen).

– Alle Systemressourcen (Strukturen, Speicher, Devices, andere Libraries und so weiter) müssen über die passenden Schutzprotokolle angefordert und nach Gebrauch sofort wieder freigegeben werden. Nur so lassen sich gegenseitige Störungen und »deadlocks« (Task A fordert Resource b und belegt bereits Resource a, auf die Task B zugreifen will und gleichzeitig Resource b blockiert -> System hängt) sicher vermeiden (siehe [4], Kap. 2.6, Seite 50 ff.).

– Selbstmodifizierender Code ist ohnehin verboten, wenn 680xx-Kompatibilität gewährleistet bleiben soll. Absolut tödlich wirkt er in einer Library-Routine!

– Außerdem natürlich keine Zeitschleifen, kein Zugriff auf undokumentierte Systemstrukturen, keine Direkteinsprünge ins ROM und die Nutzung konfigurationsspezifischer Eigenschaften (zum Beispiel Erweiterungsspeicher, PAL/NTSC, zweites Laufwerk und so weiter). Kurz: Bemühen Sie sich um saubere und multitasking-gerechte Programmierung! (siehe [4], Kap. 2, Seite 37 ff.) Nicht nur der eigene Funktionscode muß dieser Eigenschaft genügen, sondern natürlich auch etwa aufgerufene Funktionen aus System-Libraries (immer gewährleistet) oder aus den Standard-Libraries Ihres Entwicklungssystems (nicht immer gewährleistet, daher im Einzelfall zu prüfen). Unsere drei Funktionen »BuildFileRequester()«, »BuildColorPalette()« und »BuildBoolRequester()« aus den beiden letzten Folgen wurden unter Beachtung dieser Vorgaben entwickelt.

Zusätzlich zu den nach außen hin sichtbaren Funktionen müssen in jeder Library noch vier interne Routinen zur Verfügung stehen, die der Verwaltung dienen sowie Vor- und Nachbereitungsaufgaben wahrnehmen. Für diese sind die Einsprung-Offsets -6, -12, -18 und -24 definiert, womit die erste benutzerdefinierte Funktion also erst ab Offset -30 stehen kann. Den internen Routinen kommen folgende Aufgaben zu:

Die internen Verwaltungsroutinen

»OpenLib« (-6): wird im Verlauf von »OpenLibrary()« aufgerufen und inkrementiert »lib_OpenCnt«. Bei dieser Gelegenheit wird »LIBB_DELEXP« gelöscht, falls es überhaupt gesetzt war.
 »CloseLib« (-12): wird im Verlauf von »CloseLibrary()« aufgerufen und dekrementiert »lib_OpenCnt«. Sollte kein Benutzer-Task mehr vorhanden und ein »delayed-expun-

ge« (siehe oben) bereits angefordert sein, so wird gleich in die nächste Routine verzweigt.
 »ExpungeLib« (-18): Die Library wird explizit entfernt, aller mit ihr verbundener Speicher (Sprungtabelle, Basisstruktur) und angeforderte System-Ressourcen werden wieder freigegeben. Für nachfolgende Tasks muß sie bei Bedarf erneut geladen werden. Diese Funktion hat natürlich nur Sinn bei transienten Libraries; bei residenten muß sie zwar vorhanden sein, liefert aber immer null zurück.
 »ExtFuncLib« (-24): Ist für zukünftige Erweiterungen vorgesehen, sollte immer null zurückliefern.

Wir kommen nun zum Zusammenbau aller besprochenen Strukturen und Funktionen zu unserer fertigen Library. Theoretisch könnten wir alle benötigten Strukturen von Hand erstellen, in den Arbeitsspeicher laden und unsere Library in die Liste von »Exec« einbinden. Wesentlich einfacher und eleganter erreichen wir dies allerdings über das Betriebssystem.

Wir linken unsere Funktionen zusammen mit einer Resident-Struktur in ein Lademodul, das wir im Directory »libs« der Startdiskette ablegen.

Das Resident-Modul

Über die Resident-Struktur baut das Betriebssystem unsere Library automatisch auf, sobald wir sie über ein explizites »OpenLibrary()« ansprechen (siehe [1], Kap. 14, Seite 234 ff.; [4], Kap. 10.2.1, Seite 192 ff.). Folgende Einträge müssen dazu initialisiert werden:

»rt_MatchWord«: An dieser Kennung wird das Resident-Modul identifiziert. Sie entspricht dem Assembler-Befehl »ILLEGAL« und hat damit den Wert \$4AFC.

»rt_MatchTag« ist ein Zeiger auf die Resident-Struktur selbst. Zusammen mit dem ersten Eintrag folgt daraus eine eindeutige Identifizierung, die

eine Bearbeitung gemäß den Vorgaben in »rt_Flags« (siehe unten) nach sich zieht.

»rt_EndSkip« zeigt auf das Ende des geschützten Datenbereichs.

»rt_Flags«: Hier sind zwei Bits und deren mögliche Kombinationen definiert:

»RTB_AUTOINIT« (7): Ist dieses Bit gesetzt, so zeigt der Eintrag »rt_Init« (s.u.) auf eine Langworttabelle mit vier Einträgen, die zur Initialisierung der Library über die Exec-Funktion »MakeLibrary()« (siehe [3], Kap. 1, Seite 105 ff.) folgende Werte in dieser Reihenfolge enthält:

- die Größe der Basisstruktur,
- einen Zeiger auf die Funktionstabelle.

Die Funktionstabelle wird immer mit -1 abgeschlossen und besteht entweder aus absoluten 32-Bit-Referenzen zu den Library-Funktionen (in der geforderten Reihenfolge, einschließlich interner Funktionen) oder – platzsparender – aus 16-Bit-Word-Offsets der Funktionen, relativ zum Tabel-

DONAU-SOFT

24 h-Schnellversand

Ihr Amiga-PD-Partner

● ab 2,50 DM ●

Alle gängigen Serien sind lieferbar		3 ausführliche Katalogdisketten mit Kurzbeschreibung aller Programme gegen 10,— DM (V-Scheck/Briefmarken) anfordern! gratis zu unseren Katalogen: Viruskiller, CLJ-Wizard + Turbo Backup Das große Amiga-PD-Handbuch Band I-IV + alle 42 Disks + 3 Katalogdisketten (Einzelpreise erfragen) 299,-
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	
bei Serienabnahme: ab 2,50 DM		
Preise incl. 3,5" DD-Disks – Mit Qualitätsgarantie – Wir kopieren nur mit doppeltem Verify. Alle Disks sind: – 100 % Virus- und Error frei – etikettiert.		Pakete für Einsteiger und Anwender (jeweils 10 Disketten) Einsteiger 1,2; Spiele 1,2,3; Sound; Grafik; Modula II; Glanzlichter jedes Einzelpaket 35,— DM 3 Pakete nach Wahl nur 99,— DM
Leerdisketten 3,5" 2DD von Sentinel ab 1,15 DM Sony ab 1,50 DM		Floppy 3 1/2" int. 155,— DM Floppy 3 1/2" ext. } absaltbar 209,— DM Floppy 5 1/4" ext. } mit allen Extras 269,— DM
+ DM 5,— bei Vorkasse, + DM 8,— bei Nachnahme Ausland: + DM 10,— (nur Vorkasse)		
MAIK HAUER Postfach 1401, 8858 Neuburg Fax: 08431/49800 Tel.: 08431/49798 (bis 22 Uhr) BTX: *Donau-Soft #		

Public Domain Studio Nürnberg GmbH

Werderstr.4, 8500 Nürnberg 20

Tel.: 0911 / 45 77 54 Fax: 0911 / 446 72 79

Ca. 10000 Public Domain + Shareware Disketten für Amiga, Atari ST und PC.
 Gratis Katalog und Info für Ihren Computer anfordern !!!

DISKETTEN

3.5" 2DD		5.25" 2D	
	10 St. 50 St.		10 St. 100 St.
Blau	DM 10.90 50.00	Schwarz	DM 5.90 54.90
Grau	DM 10.90 50.00	Weiss	DM 9.90 94.90
Gelb	DM 17.90 85.90	Rot	DM 9.90 94.90
Rot	DM 17.90 85.00	Gelb	DM 9.90 94.90
Grün	DM 17.90 85.90	Grün	DM 9.90 94.90
Schwarz	DM 17.90 85.90	Blau	DM 9.90 94.90
3.5 HD		5.25" HD	
Schwarz	DM 29.90 140.00	Schwarz	DM 12.80 124.00

Diskettenboxen

DD80 für 80 Disketten 3.5"	DM 14.90
DD100 für 100 Disketten 5.25"	DM 14.90

Speichererweiterung für Amiga 500

512 KB mit Uhr DM 179.-

Versandkosten:
 Nachnahme: DM 6.00 Vorkasse: DM 3.00

lenbeginn. In diesem Fall ist die statische Größe der Library allerdings auf 64 kByte begrenzt, zur Identifikation dieses Adressierungsmodus enthält die Tabelle als erstes Wort "-1". Mit dieser Tabelle legt die Exec-Funktion »MakeFunctions()« (siehe [2], A-30) die Funktionstabelle im negativen Bereich der Basisstruktur an, sie wird im Verlauf von »MakeLibrary()« implizit aufgerufen.

– Einen Zeiger auf eine Initialisierungstabelle für die Basisstruktur.

Diese enthält Makros nach Definition im Headerfile »exec/initializers.i«.

Normalerweise werden die Einträge »In_Type«, »In_Name«, »lib_Flags«, »lib_Version«, »lib_Revision« und »lib_IdString« auf diese Weise nach Erstellung der Library-Struktur mit konstanten Werten gefüllt. Dies übernimmt die Exec-Funktion »InitStruct()« (siehe [3], Kap. 1, Seite 100 ff.), die im Verlauf von »MakeLibrary()« implizit aufgerufen wird.

– Einen Zeiger auf die Initialisierungsroutine »InitLib«. In dieser werden alle Vorbereitungen getroffen, etwa um weitere Libraries oder Devices zu öffnen, Speicher bereitzustellen und so weiter. Diese wird als letztes, nach Erstellung aller Strukturen von »MakeLibrary()«, implizit aufgerufen.

Nach erfolgreichem Abschluß aller Vorbereitungsarbeiten und Rückkehr aus der Initialisierungsroutine mit der Basisadresse als Ergebnis wird die komplette Library über die Exec-Funktion »AddLibrary()« (siehe [3], Kap. 1, Seite 26 ff.) nach impliziter Berechnung der Prüfsumme über »SumLibrary()« (siehe [3], Kap. 1, Seite 151 ff.) in die Library-Liste von »Exec« eingefügt und steht fortan zur Verfügung. Ist das Flag dagegen gelöscht, so erfolgt nur ein Einsprung in die Funktion, auf die »rt_Init« zeigt, alle fälligen Vorbereitungen müssen dann dort von Hand erledigt werden.

»RTB_COLDSTART« (0) ist nur für die Einbindung residenter Libraries beim Systemstart von Belang und soll hier nicht weiter interessieren.

»rt_Type« definiert den Typ des Moduls und ist für uns »NT_LIBRARY« (9). Dadurch weiß die Funktion »MakeLibrary()«, daß sie anschließend

»AddLibrary()« aufrufen soll. »rt_Pri« kennzeichnet die Priorität des Resident-Moduls und ist für unseren Anwendungsfall ebenfalls belanglos. »rt_Name«: Name des Resident-Moduls. »rt_IdString«: Identifikations-String des Resident-Moduls. »rt_Init«: Zeiger auf die Initialisierungsroutine (siehe oben).

Library-Rumpf und Zusammenbau

Alle benötigten Strukturen und Funktionen für den Library-Rumpf haben wir im Listing 2

Hier noch Hinweise zur verwendeten und weiterführenden Literatur:

[1] Commodore-Amiga, Inc.: Amiga ROM Kernel Reference Manual: Libraries & Devices, Revised and Updated; 1. Auflage, Addison-Wesley 1989, ISBN 0-201-18187-8

[2] Commodore-Amiga, Inc.: Amiga ROM Kernel Reference Manual: Includes & Autodocs, Revised and Updated; 1. Auflage, Addison-Wesley 1989, ISBN 0-201-18177-0

[3] Mortimore, Eugene P.: Amiga Programmers Handbook: Volume I; 2. Auflage, Sybex 1987, ISBN 0-89588-367-8

zur Definition unserer Library-Basisstruktur sowie einiger nützlicher Konstanten. Die Compile »buildfilerrequester.o«, »filerrequesterimages.o«, »global.o«, »buildboolrequester.o« und »buildcolorpalette.o« stehen uns ja noch aus den beiden letzten Teilen zur Verfügung. Jetzt können wir unsere Library über folgende Aufrufe zusammenbauen (Zeilen sind zum besseren Verständnis nummeriert):

```
1 ASM -iINCLUDE: requester_lib
  .asm
2 BLINK FROM requester_lib.o
  buildfilerrequester.o buildb
  oolrequester.o buildcolorpa
  lette.o filerrequesterimages
```

formulieren (zum Beispiel »#pragma-Feature« bei Lattice-C seit Version 4 und beim AmigaBASIC-Interpreter über die *.bmap-Dateien).

Andere legen die Parameter voreinstellungsgemäß auf dem Stack ab und überlassen die Auflösung der Referenz zur Library-Funktion dem Linker (siehe [4], Kap. 7, Seite 155 ff.). Dieser benötigt dazu sogenannte Stub-Routinen, welche die Registerwerte vom Stack laden und anschließend wieder restaurieren. Wir haben Ihnen diese als Programm »requester_lib.asm« für unsere Library zusammengestellt, sie können mit dem Aufruf

```
ASM -iINCLUDE: requester_stub
  s.asm
```

zu einem Objektmodul assembliert werden, das Sie beim Linken mit angeben müssen. AmigaBASIC-Programmierer müssen aus der Datei »requester_lib.fd« über das Hilfsprogramm »ConvertFD« (auf der Extras-Diskette) erst noch die Datei »requester.bmap« erzeugen und diese im aktuellen oder Directory »libs« ihrer Startdiskette aufbewahren, um unsere Library benutzen zu können. Alle noch offenen Fragen zum Aufruf von Library-Funktionen können sicherlich durch einen Blick in die Handbücher der verwendeten Entwicklungssysteme geklärt werden.

Für die nächste Folge haben wir außerdem noch einige Beispiele vorgesehen. Wir zeigen Ihnen anhand von Listings in den Programmiersprachen C und AmigaBASIC, wie Sie unsere Library am effektivsten nutzen können.

In Zukunft werden Ihre Programm Listings damit auch viel kürzer und übersichtlicher, da ja für die oft benötigten Programmteile File-Requester, Farbeinstellpalette und Dialogbox kein zusätzlicher Platz mehr beansprucht wird.

Wir sind jetzt schon gespannt auf interessante Leser-Listings, die regen Gebrauch von unserer »requester.library« machen und vom zusätzlichen Komfort profitieren. Vielleicht wird Ihr Programm ja sogar zum Listing des Monats gekürt?

Wir wünschen Ihnen schon heute viel Erfolg und gute Ideen!

(jb)

zusammengestellt (siehe [2], G-14; [5] und [6]). Nach Konvention wird eine Library-Funktion mit allen Parametern in festgelegten Registern aufgerufen, der Rückgabewert wird in »D0« an das aufrufende Programm übergeben. Die Registerdefinitionen für die Funktionen unserer »requester.library« finden sich in der Datei »requester_lib.fd« nach gültigen Amiga-Konventionen für .fd-Files. Da unsere Library-Routinen jedoch als C-Funktion vorliegen und diese ihre Parameter (normalerweise) auf dem Stack erwarten, müssen wir sie vor dem Einsprung aus den Prozessorregistern nach Vorschrift auf den Stack befördern. Dazu dienen drei kleine Interface-Routinen am Ende des Library-Rumpfes. Weiterhin benötigen wir noch das Include-File »requester.i«

```
.o global.o TO LIBS:request
er.library LIB LIB:les.lib
LIB:amiga.lib SMALLCODE NOD
EBUG
```

Damit ist unsere »requester.library« fertig zum Gebrauch. Zur Benutzung der Funktionen genügt es, die Library wie oben beschrieben zu öffnen und die entsprechende Funktion mit den zugehörigen Parametern aufzurufen.

Der Funktionsaufruf unserer Requester-Library

Einige Entwicklungssysteme unterstützen die Möglichkeit, die Parameter für Libraryfunktionen direkt in die zugehörigen Prozessorregister zu laden. Sie können auch den Sprung über die Library-Basis automatisch als »inline-code«

Listings

```

1: *****
2: *==
3: *== requester lib.asm
4: *== by Franz-Josef Reichert
5: *==
6: *****/
7:
8:
9:
10:
11: INCLUDE "exec/types.i"
12: INCLUDE "exec/initializers.i"
13: INCLUDE "exec/libraries.i"
14: INCLUDE "exec/io.i"
15: INCLUDE "exec/resident.i"
16: INCLUDE "exec/funcdef.i"
17: INCLUDE "exec/exec lib.i"
18: INCLUDE "libraries/dos.i"
19: INCLUDE "requester.i"
20:
21: ; --- Die Versionsnummer unserer Library sollte stets
22: ; --- die gleiche sein wie die Nummer der System-
23: ; --- Libraries, unter denen sie lauffaehig ist.
24: ; --- Dies ist V33.12, taugt also fuer alle Kickstart-
25: ; --- Versionen groesser gleich 1.2.
26:
27: VERSION EQU LIBRARY_VERSION
28: REVISION EQU 12
29:
30: ; --- Import: Globale Datenbasis und die Funktions-
31: ; --- einspruege unserer Library-Routinen.
32:
33: XREF LinkerDB
34: XREF BuildFileRequester
35: XREF BuildColorPalette
36: XREF BuildBoolRequester
37:
38: ; --- Export: Globale Daten fuer unsere Library-
39: ; --- Routinen.
40:
41: XDEF SysBase
42: XDEF GfxBase
43: XDEF IntuitionBase
44: XDEF DOSBase

```

Listing: requester_lib.asm



```

45:
46: ; --- Ein paar nuetzliche Makros: OPENLIB und CLOSELIB
47:
48: ; --- Library \1 oeffnen und Basis in rb xyz ein-
49: ; --- tragen. Bei Misserfolg nach \2 springen.
50:
51: OPENLIB MACRO
52: moveq #LIBRARY_VERSION,d0
53: lea \1Name(pc),a1
54: CALLLIB LVOOpenLibrary
55: move.l d0,\1Base(a4)
56: beq.s \2
57: move.l d0,rb_\1Base(a5)
58: ENDM
59:
60: ; --- Library \1 wieder schliessen.
61:
62: CLOSELIB MACRO
63: move.l rb_\1Base(a5),a1
64: CALLLIB LVOCloseLibrary
65: ENDM
66:
67: ; --- Hier beginnt die Code-Section!
68:
69: SECTION requester_lib, CODE
70:
71: ; --- Erster Befehl:
72: ; --- Falls jemand die Library vom CLI aus starten
73: ; --- will, Abbruch und Rueckkehr.
74:
75: moveq #0,d0
76: rts
77:
78: ; --- Hier beginnt die ROMTag-Struktur, die fuer alle
79: ; --- Libraries, Devices und Resources die gleiche
80: ; --- ist. Das Betriebssystem sucht nach dieser im
81: ; --- Lademodul und initialisiert die Library nach
82: ; --- den gefundenen Werten.
83:
84: ROMTag dc.w RTC_MATCHWORD ; immer ILLEGAL ($4afc)
85: dc.l ROMTag ; Zeiger auf sich selbst
86: dc.l EndCode ; Ende des Schutzbereiches
87: dc.b RTF_AUTOINIT ; fuehrt MakeLibrary() aus
88: dc.b VERSION ; Versionsnummer
89: dc.b NT_LIBRARY ; Type = LIBRARY
90: dc.b 0 ; Prioritaet (unbenutzt)
91: dc.l libraryname ; Name der Library

```

Listing: requester_lib.asm

TOOLBOX EDITION 68000

Wer seinen Computer oder Anwendungen programmieren will, findet hier Tips, Anregungen und vor allem Anwendungen erster Güte.

Das Projekt Fraktal wartet mit einem Weltrekord auf: Das "Apfelmännchen" wird in 7 Sekunden auf den Bildschirm eines Amiga 2000 gezeichnet!! Da waren Hexenmeister am Werk.

Viele Reviews informieren über neue Programmierer-Software und verhelfen dem Leser zum Überblick über interessante Neuerungen.

Natürlich gibt es die Programme des Heftes auch in der DATABOX. DATABOX ist der Software-Service, der bei DMV Standard ist. Wem das Tippen zu fehlerträchtig ist, der bestellt die DATABOX zum Heft.

Aus dem Inhalt:

PROJEKT FRAKTAL – Portierung in M2Amiga
 WELTREKORD – Apfelmännchen in 7 Sekunden
 ACU – Mausgesteuerter Compiler
 C-Compiler-Oberfläche
 Universeller Multitasking-FileRequester in M2Amiga
 Undercover Viruskiller mit eigenem Task
 Do-Request – Trickreicher AutoRequester
 AmigaDOS – Fenster mit Zwittereigenschaften
 M2Amiga-Inline-Code per Programm



DATABOX Amiga 29,- DM*

TOOLBOX EDITION 68000 9,- DM*

Sonderpreis für AMIGA-DOS-Leser 29,- DM*

Paketpreis Edition 68000 + DATABOX Amiga 29,- DM*

* Unabhängig von der Anzahl der bestellten Produkte berechnen wir für das Inland 4,- DM bzw. für das Ausland 6,- DM Porto und Verpackung.

DMV-Verlag · Postfach 250 · 3440 Eschwege

Nr. 1/88-89

18,- DM 145 ÖS 18 sfr
 DMV Verlag

EDITION



· Amiga · Atari · 68000 · Macintosh ·

AMIGA:

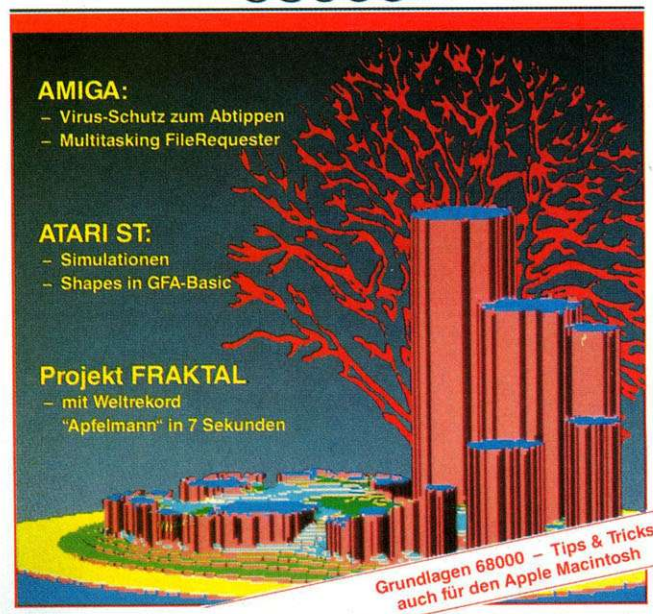
– Virus-Schutz zum Abtippen
 – Multitasking FileRequester

ATARI ST:

– Simulationen
 – Shapes in GFA-Basic

Projekt FRAKTAL

– mit Weltrekord
 "Apfelmännchen" in 7 Sekunden



Grundlagen 68000 – Tips & Tricks
 auch für den Apple Macintosh

WERKSTATT

```

92:      dc.l idstring      ; Identifikationsstring
93:      dc.l Init         ; Initialisierungstabelle
94:                               ; fuer MakeLibrary()
95:
96: ; --- Namen und Identifikationsstrings
97:
98: libraryname REQUESTERNAME
99: idstring      dc.b 'requester 33.12'
100:                dc.b ' (22 Jun 1990) by FJR', $0D, $0A, 0
101: DOSName       DOSNAME
102: IntuitionName dc.b 'intuition.library', 0
103: GfxName       dc.b 'graphics.library', 0
104:
105: ; --- Initialisierungstabelle fuer die Exec-Funktion
106: ; --- MakeLibrary(). Alle Daten und Zeiger muessen in
107: ; --- dieser Reihenfolge erscheinen.
108:
109: Init      dc.l rb SIZEOF ; Groesse der Library-Base
110:           dc.l FuncTab   ; Zeiger auf Funktionstabelle
111:           dc.l DataTab   ; Zeiger auf Init-Tabelle
112:           dc.l .InitLib  ; Initialisierungsroutine
113:
114: ; --- Funktionstabelle der Library fuer die Exec-
115: ; --- Funktion MakeFunctions(). Sie wird durch die
116: ; --- Kennung -1 angefuehrt und abgeschlossen. Dies
117: ; --- sind Word-Offsets relativ zum Tabellenbeginn.
118: ; --- Weitere Funktionen koennen am Ende (vor -1)
119: ; --- eingefuegt werden.
120:
121: FuncTab dc.w -1
122:         dc.w .OpenLib-FuncTab
123:         dc.w .CloseLib-FuncTab
124:         dc.w .ExpungeLib-FuncTab
125:         dc.w .ExtFuncLib-FuncTab
126:         dc.w .BuildFileRequester-FuncTab
127:         dc.w .BuildColorPalette-FuncTab
128:         dc.w .BuildBoolRequester-FuncTab
129:         dc.l -1
130:
131: ; --- Befehlstabelle zur Initialisierung ueber die
132: ; --- Exec-Funktion InitStruct(). Die Makros finden
133: ; --- sich im Include-File exec/initializers.i
134: ; --- Die entsprechenden Datenfelder der Library-Base
135: ; --- werden bei der Einbindung mit den angegebenen
136: ; --- Werten gefuehrt.
137:
138: DataTab INITBYTE LN TYPE, NT LIBRARY
139:         INITLONG LN NAME, libraryname
140:         INITBYTE LIB FLAGS, LIBF SUMUSED!LIBF_CHANGED
141:         INITWORD LIB VERSION, VERSION
142:         INITWORD LIB REVISION, REVISION
143:         INITLONG LIB_IDSTRING, idstring
144:         dc.l 0
145:
146: ; --- InitLib()
147: ; --- =====
148: ; ---
149: ; --- d0: Library-Base
150: ; --- a0: SegList
151: ; --- a6: SysBase
152: ; ---
153: ; --- Hier wird von der Exec-Funktion MakeLibrary()
154: ; --- nach erfolgreicher Erstellung der Basis-Struktur
155: ; --- eingesprungen. Wir nehmen die Initialisierung
156: ; --- des privaten Teils der Library-Base vor.
157: ; --- Bei Erfolg geben wir die Library-Base fuer den
158: ; --- folgenden Aufruf der Exec-Funktion AddLibrary()
159: ; --- in d0 zurueck, bei Misserfolg Null.
160:
161: .InitLib
162:     movem.l a4-a5, -(sp)
163:
164: ; --- Zeiger auf globales Datensegment setzen,
165: ; --- Library-Basis nach a5
166:
167:     lea.l LinkerDB, a4
168:     move.l d0, a5
169:
170: ; --- SysBase und Segment-Liste eintragen
171:
172:     move.l a6, SysBase(a4)
173:     move.l a6, rb_SysBase(a5)
174:     move.l a0, rb_SegList(a5)
175:
176: ; --- DOS, Graphics und Intuition oeffnen
177:
178:     OPENLIB DOS, 3$
179:     OPENLIB Gfx, 2$
180:     OPENLIB Intuition, 1$
181:
182: ; --- erfolgreich, Library-Base nach d0 und
183: ; --- Ruecksprung
184:
185:     move.l a5, d0
186:     bra.s 4$
187:
188: ; --- ??? Fehler beim Oeffnen einer Library ???
189: ; --- Fehlerstatus in d0 vermerken.
190:
191: 1$ CLOSELIB Gfx
192: 2$ CLOSELIB DOS
193: 3$ moveq #0, d0
194:
195: 4$ movem.l (sp)+, a4-a5
196:     rts

```

Listing: requester_lib.asm



```

197:
198: ; --- OpenLib()
199: ; --- =====
200: ; ---
201: ; --- d0: Version
202: ; --- a6: Library
203: ; ---
204: ; --- Hier wird aus der Exec-Funktion OpenLibrary()
205: ; --- eingesprungen. Bei Erfolg (warum nicht, schliess-
206: ; --- lich sind wir eine shared-library!) wird die
207: ; --- Library-Base in d0 zurueckgegeben, ansonsten
208: ; --- eine Null.
209:
210: .OpenLib
211:
212: ; --- Zaehler fuer Benutzertasks erhoehen.
213: ; --- Falls das delayed-expunge Bit gesetzt war, wird
214: ; --- es bei dieser Gelegenheit wieder geloescht. Wir
215: ; --- brauchen diese Library jetzt!
216:
217:     addq.w #1, LIB_OPENCNT(a6)
218:     bclr   #LIBB_DELEXP, LIB_FLAGS(a6)
219:     move.l a6, d0
220:     rts
221:
222: ; --- CloseLib()
223: ; --- =====
224: ; ---
225: ; --- a6: Library
226: ; ---
227: ; --- Hier wird aus der Exec-Funktion CloseLibrary()
228: ; --- eingesprungen. Falls die komplette Library dabei
229: ; --- entfernt wird, geben wir die Segment-Liste ueber
230: ; --- d0 zurueck. AmigaDOS kann dann den Speicherbe-
231: ; --- reich der Library freigeben.
232:
233: .CloseLib
234:     moveq #0, d0
235:
236: ; --- Wir haben einen Benutzer weniger.
237: ; --- War es etwa der letzte?
238:
239:     subq.w #1, LIB_OPENCNT(a6)
240:     bne.s 1$
241:
242: ; --- Ja, kein Benutzer mehr, wir koennen die Library
243: ; --- bedenkenlos entfernen. Wir tun dies allerdings
244: ; --- nur, wenn Bedarf dafuer besteht (delayed-expunge
245: ; --- Bit wurde zwischenzeitlich gesetzt).
246:
247:     btst   #LIBB_DELEXP, LIB_FLAGS(a6)
248:     beq.s 1$
249:     bsr.s .ExpungeLib
250:
251: ; --- Rueckkehr. d0 ist entweder Null (Library nicht
252: ; --- entfernt) oder enthaelt einen Zeiger auf die
253: ; --- Segmentliste des Library-Lademoduls.
254:
255: 1$ rts
256:
257: ; --- ExpungeLib()
258: ; --- =====
259: ; ---
260: ; --- a6: Library
261: ; ---
262: ; --- Die Library soll entfernt werden. Diese Funktion
263: ; --- kann auch aus der ramlib.library aufgerufen
264: ; --- werden, wenn der freie Speicher mal knapp wird.
265: ; --- Daher muessen wir immer pruefen, ob wirklich keine
266: ; --- Benutzer mehr eingetragen sind.
267:
268: .ExpungeLib
269:     movem.l d2/a5/a6, -(sp)
270:
271: ; --- Haben wir noch Benutzer?
272:
273:     tst.w LIB_OPENCNT(a6)
274:     beq.s 1$
275:
276: ; --- Ja, wir koennen also die Library jetzt noch nicht
277: ; --- entfernen. Wir tun dies aber bei naechster Gele-
278: ; --- genheit, sobald der letzte Benutzertask sich
279: ; --- abgemeldet hat. Delayed-expunge wird gesetzt.
280: ; --- Der Wert fuer die Segmentliste ist Null.
281:
282:     bset   #LIBB_DELEXP, LIB_FLAGS(a6)
283:     moveq #0, d0
284:     bra.s 2$
285:
286: ; --- Wir haben definitiv keine Benutzer mehr. Alle
287: ; --- in der .InitLib-Routine angeforderten Ressourcen
288: ; --- werden wieder freigegeben.
289:
290: 1$ move.l a6, a5
291:     move.l rb_SysBase(a5), a6
292:     CLOSELIB DOS
293:     CLOSELIB Gfx
294:     CLOSELIB Intuition
295:
296: ; --- Die Library-Base wird aus der Library-Liste von
297: ; --- Exec entfernt. Der Segmentlistenpointer wird in
298: ; --- d2 gesichert.
299:
300:     move.l a5, a1
301:     CALLLIB LVORemove

```

Listing: requester_lib.asm


```

302: move.l   rb SegList(a5),d2
303: move.l   a5,a1
304:
305: ; --- Groesse des Speicherbereiches ermitteln und
306: ; --- Speicher der Library-Base freigeben.
307:
308: moveq     #0,d0
309: move.w    LIB_NEGSIZE(a5),d0
310: suba.l    d0,a1
311: add.w     LIB_POSSIZE(a5),d0
312: CALLLIB   _LVÖFreeMem
313:
314: ; --- Segmentliste nach d0 und Ruecksprung.
315:
316: move.l    d2,d0
317: 2$ movem.l (sp)+,d2/a5/a6
318: rts
319:
320: ; --- ExtFuncLib()
321: ; --- =====
322: ; ---
323: ; --- Fuer Erweiterungen, zur Zeit unbenutzt. Sollte
324: ; --- Null zurueckliefern.
325:
326: .ExtFuncLib
327: moveq     #0,d0
328: rts
329:
330: ; --- Hier beginnen die Interfacerroutinen zu den drei
331: ; --- Library-Funktionen. Alle Parameter werden in
332: ; --- Registern uebergeben. Eine C-Funktion erwartet
333: ; --- ihre Argumente jedoch auf dem Stack, wohin sie
334: ; --- hierdurch erst befoerdert werden.
335:
336: .BuildFileRequester
337: movem.l   d0-d2/a0-a1,-(sp)
338: jsr       BuildFileRequester(pc)
339: lea.l     20(sp),sp
340: rts
341:
342: .BuildColorPalette
343: movem.l   d0/a0-a1,-(sp)
344: jsr       BuildColorPalette(pc)
345: lea.l     12(sp),sp
346: rts
347:
348: .BuildBoolRequester
349: movem.l   d0-d3/a0-a3,-(sp)
350: jsr       BuildBoolRequester(pc)
351: lea.l     32(sp),sp
352: rts
353:
354: ; --- EndCode markiert das Ende dieses Resident-
355: ; --- Bereiches. Es muss in der gleichen Code-Section
356: ; --- wie die Resident-Struktur selbst sein und darf
357: ; --- nicht vor dieser liegen. Ab hier wird eine Suche
358: ; --- nach neuen Resident-Strukturen (wenn ueberhaupt!)
359: ; --- fortgesetzt.
360:
361: EndCode EQU *
362:
363: ; --- Globale Daten fuer unsere C-Funktionen.
364: ; --- Dies ist ein eigenes BSS-Segment, das basis-
365: ; --- registerrelativ (small-data-model) angesprochen
366: ; --- werden kann.

```

Listing: requester__lib.asm

```

367:
368: SECTION   __MERGED,BSS
369:
370: _SysBase      ds.l 1
371: _DOSBase      ds.l 1
372: _GfxBase      ds.l 1
373: _IntuitionBase ds.l 1
374:
375: ; --- Ende, Ende, Ende!
376:
377: END

```

Listing: requester__lib.asm

```

1: * =====
2: * =
3: * = requester.lib.fd
4: * = by Franz-Josef Reichert
5: * =
6: * =====/
7:
8:
9:
10: ##base RequesterBase
11: ##bias 30
12: ##public
13: BuildFileRequester(dirname,nodename,filename>window,titl
e)(d0,d1,d2,a0,a1)
14: BuildColorPalette(cmap,screen,title)(d0,a0,a1)
15: BuildBoolRequester(backfill,txtpen,shadow,lines,ta>wind
ow,t,g)(d0,d1,d2,d3,a0,a1,a2,a3)
16: ##end

```

Listing: requester__lib.fd

```

1: * =====
2: * ==
3: * == requester.i
4: * == by Franz-Josef Reichert
5: * ==
6: * =====/
7: IFND REQUESTER_I
8: REQUESTER_I SET 1
9:
10: INCLUDE "exec/types.i"
11: INCLUDE "exec/libraries.i"
12: INCLUDE "libraries/dos.i"
13:
14: STRUCTURE RequesterBase,0
15: STRUCT rb_LibNode,LIB_SIZE
16: BPTR rb_SegList
17: APTR rb_SysBase
18: APTR rb_IntuitionBase
19: APTR rb_GfxBase
20: APTR rb_DOSBase
21: LABEL rb_SIZEOF
22:
23: REQUESTERNAME MACRO
24: dc.b 'requester.library',0
25: ENDM
26:
27: LIBINIT
28:
29: LIBDEF _LVÖBuildFileRequester
30: LIBDEF _LVÖBuildColorPalette
31: LIBDEF _LVÖBuildBoolRequester
32: ENDC REQUESTER_I

```

Listing: requester.i



Auch in Ihnen schlummern verborgene Talente

SIE sind leidenschaftlicher Spieler.

SIE programmieren natürlich gern selbst Spiele.

Lassen **SIE** Ihre programmierten Spiele nicht in der Schublade oder in irgendeiner Ecke Spinnweben ansetzen.

Wenn **SIE** nicht wissen, wohin mit Ihrem Listing, schicken **SIE** es an den

DMV-Verlag
Redaktion AMIGA DOS
Postfach 250
Stichwort: Spieleprogrammierung
3440 Eschwege



Rund um das Thema Grafik und Animation ranken sich die Themen in unserer Bücherecke. Der »Sculpt 3D/4D Workshop« beschäftigt sich ausschließlich mit dem gleichnamigen Grafikprogramm. Die Basis dieses Buches bilden alle aktuellen Versionen, also »Sculpt 3D«, »Sculpt 4D« und »Sculpt 4D Jr«.

Sculpt 3D/4D Workshop

Im ersten Kapitel finden Sie ausschließlich Hintergrundwissen, das zur Bedienung von Grafikprogrammen für den Anwender unbedingt notwendig ist. Hier wird sowohl auf die Benutzung des Koordinatensystems als auch auf nützliche Zusatz-Hardware eingegangen. Ferner werden die Fähigkeiten der einzelnen Programmversionen erläutert. Ein potentieller Käufer findet so Entscheidungshilfen bei der Anschaffung des richtigen Programms. Allerdings wird »Sculpt« nicht anderen, gleichartigen Programmen gegenübergestellt, so daß dieser Vergleich nur eingeschränkt eine Hilfe bietet.

Das zweite Kapitel bildet zugleich den Hauptteil des Buches. Auf 160 Seiten werden die Funktionen und die Arbeitsweise von »Sculpt« erklärt. Die ansonsten rein theoretisch abgehandelten Erläuterungen werden lediglich durch zwei Übungen aufgelockert. Dieser Teil kann als der eigentliche Grundlagenteil angesehen werden, in dem der Anwender die Arbeit mit dem Programm erlernt. Wer sich allerdings nicht scheut, das englische Handbuch von »Sculpt« zu lesen, wird in diesem Teil wenig Neues finden.

Die Praxis steht im Vordergrund des dritten Kapitels. Eine kleine Infobox informiert vor jeder Übung über die jeweilige Programmversion, mit der diese nachvollzogen werden kann. Insgesamt acht Fallstudien dienen dem Anwender zum besseren Verständnis des Programms. Daneben erhält er zusätzlich noch Informationen über die additive und subtraktive Farbmischung. Der Anhang setzt sich aus zwei Teilen zusammen. Der erste Teil befaßt sich mit den »Magic Numbers«. Dies sind versteck-

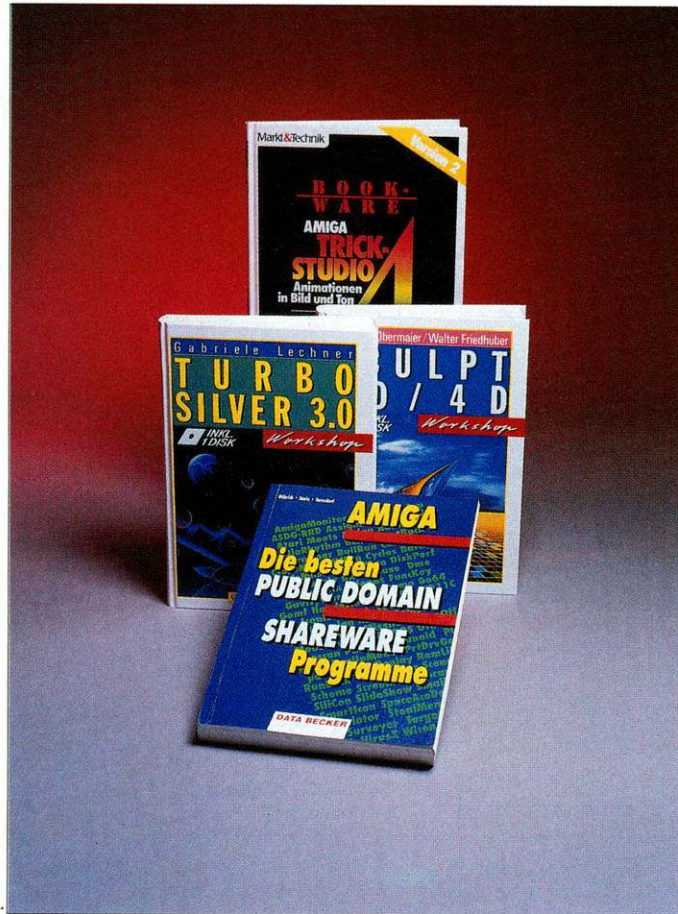
te Optionen, die der Programmierer in das Programm eingebaut hat und die mit kleinen Tricks sichtbar gemacht werden können. Im zweiten Teil des Anhangs ist eine Referenzliste zu finden, in der die wichtigsten Tastaturkommandos und Mausektionen zusammengefaßt sind.

Das Buch erläutert die einzelnen Befehle recht ausführlich. Die mitgelieferte Diskette er-

satte DM 138,- für eine deutsche Anleitung zu »Sculpt« hinblättern. Das wäre entschieden zu viel.

Name: Sculpt 3D/4D Workshop
Verlag: Gabriele Lechner
ISBN: 3-926858-10-0
Preis: 69,- DM (inklusive Diskette)

Dieser Workshop liegt in einer überarbeiteten zweiten Auflage vor. Im Ge-



leichtert den Umgang mit dem Buch, indem aufwendige Berechnungen, die bei den Fallstudien anfallen würden, wegfallen und es so zu einer erheblichen Zeiterparnis kommt. Allerdings hat der Autor vollkommen mißachtet, daß »Sculpt 4D« und »Sculpt 4D Jr.« auch einen sehr guten Animationsteil bieten. Darüber wird im Buch kein Wort verloren. Somit erhält der Anwender für 69,- DM nur die Hälfte an Informationen. Ein Buch über die Animationsmöglichkeiten ist in Planung. Spinnt man diesen Faden weiter, müßte man für diesen zweiten Teil auch noch einmal 69,- DM bezahlen, so müßte man

gensatz zum Sculpt-Workshop hat sich einiges an diesem Buch geändert. Das erste Kapitel beschäftigt sich mit den notwendigen Hintergrundwissen, ohne das »Turbo Silver« für den Anwender nicht zu bedienen ist. Es wird genauso auf die Hardware-Voraussetzungen eingegangen, wie auf die »3D-Welt« von »Turbo Silver«.

Turbo Silver 3.0 Workshop

Die Arbeit mit dem Programm wird ausführlich beschrieben, und die Funktionen werden näher erläutert. Das Buch ist

so aufgebaut, daß die einzelnen Befehle anhand eines Beispiels noch einmal verdeutlicht werden. Das zweite Kapitel befaßt sich mit grundlegenden Aspekten der Bedienung von »Turbo Silver« wie beispielsweise das Öffnen und Schließen von Dateien und ähnliches mehr.

In den Kapiteln 3 bis 7 sind die Themenbereiche sehr vielgestaltig. Das Spektrum reicht von einer Szene, die aus vordefinierten Objekten besteht, über das Entwerfen von eigenen Objekten mit entsprechenden Übungsformen. Danach lernt der Anwender anhand gezielter Übungen den Umgang mit dem Texture »Bricks«. Dreidimensionale Effekte am Beispiel der Konstruktion eines Raumes, mit unterschiedlichen Gegenständen gefüllt, schließen sich an. Wie man Animationen erzeugt, erfährt man im darauffolgenden Kapitel. Als Beispiel dient die Erstellung eines Logos.

Interessant wird es im achten Kapitel, in dem das Programm »Terrain«, als Zusatzprogramm angeboten, besprochen wird. Allerdings fallen die Erklärungen hier etwas dürrig aus. Einige Beispiele wären angebracht. Die neuen Features von »Turbo Silver 3.01 SV« sind Gegenstand von Kapitel 9. Abgerundet wird das Buch durch mehrere Anhänge, die unter anderem Zusatzprogramme auflisten oder die Datei »silver.config« erläutern. Dank der mitgelieferten Diskette müssen die Übungen nicht alle nachvollzogen werden. Das einzige Manko an diesem Buch ist die Tatsache, daß die SCR-Dateien nicht erläutert werden. So kann der Anwender, der nicht über zwei Laufwerke verfügt, schon beim Benutzen der Diskette ins Wasser fallen.

Name: Turbo Silver 3.0 Workshop
Verlag: Gabriele Lechner
ISBN: 3-926858-12-5
Preis: 69,- DM (inklusive Diskette)

Als Bookware vertreibt der Markt&Technik-Verlag das Animationsprogramm »Trickstudio A«. Mit diesem Programm ist der Anwender in der Lage, sich seinen eigenen kleinen Trickfilm zu erstellen. Das Handbuch umfaßt knapp 100 Seiten, in denen

nicht nur auf das Programm eingegangen wird, sondern auch Hintergrundwissen, wenn auch in etwas dürftiger Art, vermittelt wird. Das Handbuch reicht aber aus, um die Funktionen des Programms zu erklären. Das Programm dient der Zusammenfügung einzelner IFF-Bilder zu einer Animation. Diese Bilder können mit »DPaint« oder anderen Malprogrammen erstellt werden, die die Grafiken in dem entsprechenden Format abspeichern. Neben dieser Grundfunktion bietet »Trickstudio A« noch einige nützliche Funktionen, die kurz erläutert werden. Animationen können mit Sound hinterlegt werden. Diese gesampelten Sounds müssen entweder im IFF- oder im Future-Sound-Format vorliegen.

Trickstudio A

Eine bereits fertige Animation in ihre einzelnen Bestandteile aufzugliedern ist ein weiteres sehr nützliches Feature. Diese müssen im Anim-5-Format vorliegen (»DPaintIII«, »VideoScapes«, »TurboSilverSV«). Der Zeitaufwand, den das Programm für das Zerlegen benötigt, ist allerdings sehr hoch, da die einzelnen Dateien erst geladen und entpackt werden müssen. Dafür lassen sich die Bilder dann einzeln per Tastendruck abspeichern. Soviel zu den Möglichkeiten des Programms.

In der Praxis werden diese Funktionen zwar anstandslos ausgeführt, allerdings weist das Programm doch Mängel in der Benutzerfreundlichkeit aus. Anstelle einfacher Requester wird ein sehr umständlich zu bedienender File-Requester benutzt, der die Arbeit mit dem Programm nicht gerade erleichtert. Alle Funktionen werden über ein Menü in der unteren Bildschirmhälfte bedient. Der verdeckte Teil der Grafik kann durch Scrollen sichtbar gemacht werden. Schön ist die Möglichkeit, mit dem Editor im Interlace-Modus zu arbeiten, die Animation aber später trotzdem im Non-Interlace-Modus zu erstellen. Dies funktioniert auch umgekehrt.

Neben den Gadgets sind alle Funktionen auch über Tastatur abrufbar. Leider können Zahlen nicht direkt über die Tastatur eingegeben werden. Dar-

unter leidet natürlich die Arbeitsgeschwindigkeit. Mitgeliefert wird noch ein Programm zum Abspielen der fertigen Animationen und einige Demobilder zum Ausprobieren. Im großen und ganzen ist dieses Produkt eine gute Ergänzung zu »DPaintIII«. Benutzer von »DPaintIII« kaufen die Hälfte der Funktionen allerdings umsonst, da sie schon im Programm implementiert sind. Leider speichert »Trickstudio A« die eigenen Animationen nicht in dem Format ab, das es selbst auch wieder einlesen kann. Somit können nur Animationen anderer Programme, nicht aber die des eigenen in Einzelbilder zerlegt werden.

Diese Bookware bietet insgesamt betrachtet einen günstigen Einstieg in die Welt der Animation. Ein zusätzliches Malprogramm ist jedoch unumgänglich.

Name: Trickstudio A
Verlag: Markt&Technik
ISBN: 3-89090-886-1
Preis: 99,- DM (inklusive Diskette)

Ganz neu ist im Data-Becker-Verlag ein Buch über die PD-Programme erschienen. Dem Leser soll so der Einstieg in die Public Domain und Shareware erleichtert werden. Ein Problem, das sich Interessierten stellt, ist natürlich die unüberschaubare Menge an Programmen und Serien, die mittlerweile erhältlich sind.

Die besten Public Domain & Sharewareprogramme

Das Buch ist in sechs Kapitel und einen Anhang unterteilt. Es beginnt mit den Programmen »zur täglichen Arbeit«. Die Autoren haben im gesamten Buch die Fred-Fish-Reihe als Grundlage genommen, da dies die bekannteste und wohl auch umfangreichste Reihe ist. In diesem Kapitel werden 21 Utilities vorgestellt und besprochen. Darunter befinden sich auch allseits bekannte, wie »DMouse«, »VirusX« und »Sectorama«. Leider ist »VirusX« nicht in seiner neuesten Version 4.0, sondern in der etwas älteren Version 3.2 beschrieben. Solche Dinge las-

sen sich bei einem solchen Buch allerdings auch nicht umgehen, da fast täglich neue Programme erscheinen. Positiv ist zu bewerten, daß in dem Kapitel über »Sectorama« nicht nur die Funktionen des Programms erklärt werden, sondern die Autoren auch das nötige Hintergrundwissen vermitteln, das unbedingt notwendig ist, um Manipulationen auf einer Diskette vorzunehmen. Dieses Wissen wird durch gezielte Beispiele noch vertieft.

Die Kapitel 2 bis 4 haben Programmierhilfen, Utilities und DOS-Shells zum Gegenstand. Sie werden beschrieben, und man erhält die nötigen Hintergrundinformationen dazu. Wer allerdings auf lange Erklärungen zu den einzelnen Menüpunkten hofft, der ist mit diesem Buch nicht gut beraten. Vielmehr werden die grundlegenden Dinge kurz und bündig abgehandelt, um danach die praktische Anwendung zu zeigen. Für den Fortgeschrittenen sicherlich sinnvoll, für den Einsteiger aber vielleicht etwas zu knapp. Vor der Besprechung eines Programms wird stets der Zweck erklärt. Anschließend erfährt man, wie man es startet und wieder verläßt. Somit kann derjenige, der lieber gleich ausprobiert, auch auf seine Kosten kommen.

Im fünften Kapitel widmen sich die Autoren dann ganz den Späßen und Gags, die es in der PD gibt. Auch einige nette Animationen und Spiele werden hier vorgestellt. Compiler werden im sechsten Kapitel behandelt. Allerdings sind hier die Erklärungen zu mager ausgefallen. Durch einen nach Diskette und Namen geordneten Anhang wird das Buch abgerundet. Positiv ist, daß man direkt im Inhaltsverzeichnis erkennen kann, auf welcher Diskette sich welches Programm befindet. Somit ist das Buch für 39,- DM allen zu empfehlen, die mehr über PD wissen möchten und nicht erst lange die PD-Reihen nach guten Programmen durchwühlen wollen.

(Andreas Polk/vb)

Name: Die besten Public Domain & Sharewareprogramme
Verlag: Data Becker
ISBN: 3-89011-368-0
Preis: 39,- DM

CSV-HIGHLIGHTS

Commodore	
Speichererweiterung Commodore 1050 (256 KB) 39,-	
Commodore Farbmonitor 1084 Stereo 589,-	
Commodore Amiga 500 1799,-	
Amiga 500 + Farbmonitor 1084 S 1369,-	
Speicheraufrüstung auf 1 MB mit Uhr 149,-	
Commodore Amiga 2000 1799,-	
Amiga 2000 + Farbmonitor 1084 S 2369,-	
Amiga 3000 (16 MB) mit Festplatte 5799,-	
3000 (25 Mhz, 40 MB Festplatte) 6799,-	
3,5"-Zweitlaufwerk Amiga 2000 (Commodore) 199,-	
PC/XT-Karte mit 5 1/4"-Laufwerk 599,-	
AT-Karte mit 5 1/4"-Laufwerk 1699,-	
SCSI Controller Commodore A 2090 A 579,-	
20 MB-Festplatte für Amiga 2000 mit SCSI Controller Comm. 2090 A (autobootend) 899,-	
40 MB-Festplatte mit Controller 2090 A 1299,-	
40 MB-Filecard autobootfähig, formatiert 1199,-	
20 MB-Filecard (Seagate, 40 ms) für A 2000 mit PC-Karte oder A 1000 / Sidecar 579,-	
30 MB-Filecard (Western Digital, 65 ms) 679,-	
40 MB-Filecard (Western Digital, 29 ms) 829,-	
50 MB-Filecard (Seagate, 40 ms) 949,-	
2 MB-RAM Erweiterungskarte für A 2000 aufrüstbar bis 8 MB 699,-	
Atari	
Festplatte Vortex HD 40 Plus 899,-	
Festplatte Atari Megaflo 60 1279,-	
1040 STFM + Monochrommonitor SM 124 1099,-	
1040 STFM + SM 124 + Megaflo 30 1949,-	
Atari STE + Monochrommonitor SM 124 1399,-	
Atari Computer Mega ST 1 mit Maus + Monochrommonitor SM 124 1479,-	
Mega ST 1 + SM 124 + Megaflo 30 MB 2329,-	
Atari Mega ST 2 + Monochrommon. SM 124 2169,-	
Atari Mega ST 2 + SM 124 + Megaflo 30 2399,-	
Atari Mega ST 4 + Monochrommon. SM 124 3149,-	
Atari Mega ST 4 + SM 124 + Megaflo 30 3999,-	
Supercharger für Atari ST 649,-	
80 MB Festpl. Seagate ST 1096 N (SCSI) 999,-	
Epsondrucker (dt. Handbücher)	
LX 400 399,-	
LQ 400 (24-Nadel-Drucker) 629,-	
LQ 550 (24-Nadel-Drucker) 749,-	
Tintenstrahldrucker LX 800 (9 Düsen, NLQ, max. 240 Zeichen/Sekunde) 449,-	
Stardrucker (dt. Handbücher)	
LC-10 Color Farbdruker mit Centronics 569,-	
LC 24-10 mit Centronicsinterface 649,-	
XB 24-10 mit Centronicsinterface 1279,-	
NEC-Drucker (dt. Handbücher)	
Farboption P6 + /P7 + 249,-; EZB P 7 Plus 399,-	
NEC P 60 1499,-; NEC P 70 1899,-	
Colorit für NEC P 60 oder P 70 179,-	
NEC Farbmonitor Multisync 3 D 1399,-	
NEU: IBM-Kompatibler-AT (12 Mhz, 512 KB) 1049,-	
Druckerkabel 5 m lang für Amiga, ST 29,-	
EGA-Karte(800x600) + 14" EGA-Farbmonitor 649,-	
VGA-Karte Optima 16 bit, 512 KB 299,-	
VGA-Karte Trident 16 bit, 512 KB 269,-	
Multisync Farbmon.(0,28 mm, 1024x768) 979,-	
VGA-Farbmonitor (Aufl. 1024 x 768) 749,-	

Versandkostenpauschale: Inland DM 12,-, Ausland DM 40,- je Paket. Lieferung nur gegen NN oder Vorauskasse; Ausland nur Vorauskasse. Preise gültig ab 17.9.90.

CSV RIEGERT GmbH
Gärtnerstr. 4, 7320 Göppingen
Tel. 0 71 61 / 35 91, Fax 0 71 61 / 35 87

Eine Bitte an unsere Abonnenten

Vermerken Sie bei Schriftverkehr und Zahlungen neben der vollständigen Anschrift stets Ihre Abo-Nummer.

Sie vermeiden damit unnötige Verzögerungen bei der Bearbeitung Ihres Abonnements.

Vielen Dank

Ihre DMV-Versandabteilung

Im Laufe der Zeit leeren sich zur Ansammlung von Disketten langsam die Brieftaschen des Amiganers. Bei der Komplexität der Programme ist eine neu erworbene 3,5-Zoll-Diskette schnell einmal vollkopierte. Damit der Benutzer ein wenig mehr Programme auf seiner Diskette unterbringen kann, wurden Cruncher entwickelt, die einzelne Files bis auf 50 Prozent ihrer Länge komprimieren. Die Diskette Time Nr.7 befaßt sich unter anderem mit Packern.

Crunchen wird den Amiganern leichtgemacht. Der »Power-Packer« ist schon fast eine Legende unter den Packern im Public-Domain-Bereich. Das Programm ist nun in der Lage, Files zu crunchen und zu entcrunchen. Falls Sie Näheres zu dem Programm lesen möchten, schauen Sie doch einfach in der Werkzeugkiste der AMIGA DOS, Ausgabe 2'90 nach.

Das Programm »UDM« dient zum Archivieren von Programmnamen. Das Archivieren wird durch eine Vielzahl an Packern, die sich aus dem C-Verzeichnis als Befehle aktivieren lassen, unterstützt: »ARC«, »LHARC«, »PAC«, »ZOO«, »WARP READ«, »LHWARP READ« und »ZAP-R«. Mit den genannten Packern ist sowohl der Crunch-Vorgang als auch das Entcrunchen der Archivierungsdatei möglich. Beim Entpacken können Sie neu archivieren. Damit Sie sich beim Archivieren nicht sofort festlegen müssen, besteht die Möglichkeit, über das Unterprogramm »CDRAM« die

PD-Werkzeugkiste

Programme rund um Grafik, Crunchen und Archivieren präsentieren wir Ihnen in unserer Werkzeugkiste.

soeben erstellte Datei im RAM abzulegen. Mit »PPShow« präsentiert sich die Ergänzung zu dem allbekannten »PPMore« und zugleich ein Programm, das es ermöglicht, IFF-Grafiken auf dem Bildschirm sichtbar zu machen. Hierzu öffnet sich ein kleines Fenster, in das der betreffende File-Name einzusetzen ist.

Als zusätzliches Bonbon befindet sich ein Spiel namens »RollOn« in der Programmsammlung – eine sehr gute Umsetzung des PC-Spiels »Sokoban«. Dieses Programm ist Shareware.

Abschließend ist noch zu erwähnen, daß »Virusx 4.0« zum Aufspüren möglicher Viren zusätzlich auf der Diskette installiert wurde. Schon beim Booten prüft der Viruskiller auf Ungeziefer.

(Jürgen Seibel/vb)

Name: Time 7
Quelle: A.P.S.-Electronic

Immer wieder bietet die Public Domain Alternativen zu den "großen, kommerziellen Brüdern". Mit »Amiga-Paint« stellt sich ein Malpro-

gramm vor, das seine Verwandtschaft zum etablierten »DPaint« nicht verleugnen kann.

Warum Ölfarben? Ein Malprogramm tut's auch

Die Malfläche erstreckt sich bei diesem Programm über ei-

nen großen Teil des Bildschirms. Wenden wir uns der Bedienung von »Amiga-Paint« zu. Bevor Sie Ihren künstlerischen Ambitionen freien Lauf lassen, sollten Sie zunächst in den Präferenzen Ihre gewünschten Einstellungen vornehmen. Am unteren Bildschirmrand finden Sie eine Farbpalette, die 32 unterschiedliche Farben implementiert. Falls Ihnen dieses Angebot nicht ausreichen sollte, können Sie durch die Veränderung der Farbanteile Rot, Grün und Blau eine zusätzliche Nuancierung erreichen. Zur Wahl des Füllmusters bieten sich dem Benutzer neun verschiedene Muster an. Um Ihren Drang zur Kreativität nicht einzuschränken, öffnet sich

PD-Insider-Tips

Programmname: MachII

Autor: Brian Moats

Disk: Fish 254

Wir stellen Ihnen ab dieser Ausgabe regelmäßig Tips und Tricks zu den verschiedensten PD-Programmen vor. Das Programm »MachII« besitzt die Fähigkeit, die zehn Funktionstasten des Amiga mit diversen Zeichenketten zu belegen. Hat man jedoch die deutsche Tastaturbelegung (ist ja wohl meistens der Fall) aktiviert, so ergibt sich folgendes Problem.

Belegt man nun die Funktionstasten aus dem Programm heraus, das heißt die Eingabe der Zeichenkette über das Stringgadget von »MachII«, sieht die Zeichenkette vollkommen normal aus (Doppelpunkt, Schrägstrich und Bindestrich und so weiter). Verläßt man nun das Programm und wechselt in den CLI, um die Belegung der F-Tasten zu testen, ergibt sich ein völlig anderes Bild. Der Doppelpunkt ist kein Doppelpunkt mehr und der Schrägstrich sieht auch ganz anders aus. Diese Erscheinung hängt damit zusammen, daß »MachII« eine amerikanische Tastaturbelegung voraussetzt.

Dieses Problem kann man auf sehr einfache Art und Weise lösen. »MachII« besitzt nämlich auch die Möglichkeit, die F-Tasten durch einen Ma-

kro-Recorder zu belegen. Dazu geht man folgendermaßen vor.

1. »MachII« starten und mit »DONE« verlassen,
2. CLI starten,
3. durch [Alt-Esc] den Makro-Recorder starten. Es ertönt ein "Beep" und man wird vom Programm aufgefordert, eine Taste (Hotkey) zu wählen. Nun kann man jede Taste, die man drücken möchte (also auch die F-Tasten, wer möchte auch mit [Alt], [Ctrl] oder [Shift]), belegen.
4. Nun wird die gewünschte Zeichenkette eingetippt; dabei ist größte Sorgfalt angeraten, da »MachII« wirklich jeden Tastendruck registriert (auch [Return] und [Backspace]).
5. Um die Eingabe zu beenden, drückt man wieder [Alt-Esc], und die Belegung der Taste bringt auch mit deutscher Tastaturbelegung das gewünschte Ergebnis.

Zwei Beispiele zur Verdeutlichung:

[Alt-Esc] [F1] Diskcopy df0: to df1: [Alt-Esc]

Damit ist die Taste [F1] mit der genannten Zeichenkette belegt.

Oder

[Alt-Esc] [Shift-Alt-F10] microemacs df0:s/startup-sequence [Alt-Esc]

um die Taste [Shift-Alt-F10] zu belegen.

(Robert Münch/vb)

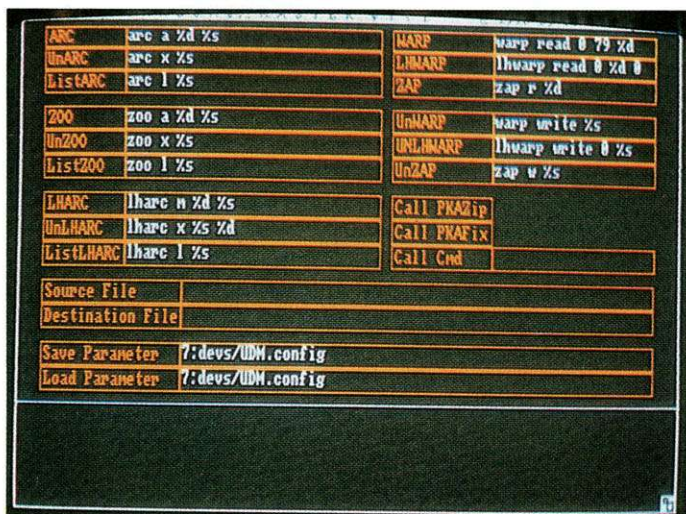


Bild 1. Das Programm »UDM« unterstützt eine Vielzahl von Packern

sich ein Editorfeld zur Gestaltung eines neuen Füllmusters. Von einem "Sprung ins kalte Wasser" kann nach den vorgenommenen Einstellungen nicht mehr die Rede sein.

Wie bei vielen anderen Malprogrammen wird auch hier mit der Maus gezeichnet. Die Art des Zeichnens wird in einem Pull-down-Menü unter dem Menüpunkt »Zeichenart« festgelegt. Für das »Freihandzeichnen« gibt es drei verschiedene Modi, die sich jedoch nur in der Stärke des Stiftes unterscheiden. Anders verhält es sich mit dem »Linienzeichnen«. Hier spielt nicht die Breite der Linien, sondern die Anzahl der Fixpunkte die entscheidende Rolle. Auch hier stehen dem Anwender drei unterschiedliche Modi zur Verfügung. Im ersten Modus legen Sie den Ausgangspunkt »A« und Zielpunkt »B« per Maus fest. Damit wird eine Linie zwischen diesen beiden

Punkten gezogen. Mit Hilfe des zweiten Modus können Polygone gebildet werden. Dabei hängt sich nach dem Zeichnen einer Linie die nächstfolgende an die gerade erstellte Linie an. Im dritten Modus drehen sich Geraden um einen definierten Drehpunkt in Bewegungsrichtung der Maus. Zur Erstellung eines Rechtecks definiert man wiederum einen Fixpunkt, von dem aus die Größe des Rechtecks per Maus variiert werden kann. Hier kommt endlich das oben bereits erwähnte Füllmuster zum Zuge. Denn die erstellten, abgeschlossenen Flächen lassen sich mit einem beliebigen Muster ausfüllen. Hinter der Option »Kreise« verbirgt sich nicht nur die Möglichkeit, Kreise zu erstellen, sondern auch alle Arten von Ellipsen.

Fertige Bilder werden im IFF-Format gespeichert. Zusätzlich ermöglicht es »Amiga-

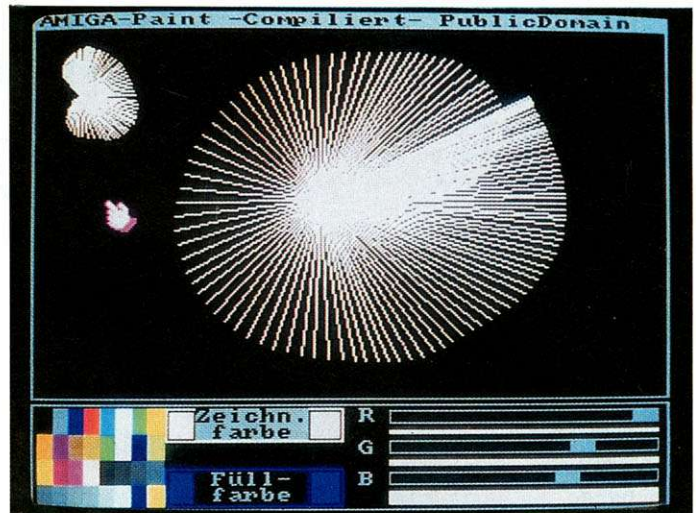


Bild 2. Einfaches und effektives Arbeiten mit »Amiga-Paint«

Paint, Ihre Meisterwerke mit einem erläuternden Text zu versehen, der beispielsweise Ihre Initialen enthält. Selbstverständlich können mit diesem Programm alle mit »DPaint« erstellten IFF-Brushes eingeladen und modifiziert werden. Ein

ansprechendes Programm zum kleinen Preis.

(Jürgen Seibel/vb)

Name: Auswahl 89
Quelle: Herrmanns & Kommelter

Eine kleine Übersicht der Vertreiber von Public Domain, Free- und Shareware sowie Prüf-vor-Kauf-Programmen (ohne Anspruch auf Vollständigkeit)

A.P.S. Electronic
Sonnenborstel 31
3071 Steimbke
Tel.: 05026/1700

Bavariansoft
Friedrich Neuper
Postfach 72
8473 Pfreimd

Bellingrath, Christian
Hans-Böckler-Str. 55
5860 Iserlohn
Tel.: 02371/24192

Digital Marketing
D. Mückter
Krefelder Str. 16
5142 Hückelhoven-Baal
Tel.: 02435/2086, 428 oder 1295

Dombrowski, Rüdiger
Postfach 71 04 62
2000 Hamburg 71
Tel.: 040/6428225

Herrmanns & Kommelter
Vom-Bruck-Platz 45
4150 Krefeld 1
Tel.: 02151/399833

Hieske, Dieter
Schillerstr. 36
6700 Ludwigshafen
Tel.: 0621/673105

Kaminski, Walter
Geilenkirchener Str. 4
5110 Alsdorf
Tel.: 02404/22963

Keim, Peter
Vogelsanger Str. 34
5000 Köln 30
Tel.: 0221/520765

Maxon Computer GmbH
Industriestr. 26
6236 Eschborn
Tel.: 06196/481811

Nordsoft
Schwenecker & Behnke
Heidelbergstr. 75
2800 Bremen 21
Tel.: 0421/611430

Pawlowski, Patrick
Software Service
Ellerbruch 19
2177 Wingst
Tel.: 04778/7294

Software Empire
Lange Str. 118
2880 Brake (Utw)
Tel.: 04401/71348

TechnikSupport Verlag GmbH
Bundesallee 36-37
1000 Berlin 31
Tel.: 030/8621314

Wolf Computertechnik
Deipe Stegge 187
4420 Coesfeld
Tel.: 02541/2874

Österreich

Küppers, Bernd
Felberstr. 7
A-5730 Mittersill
Tel.: 06562/282

M.A.R.-Computershop
A-1100 Wien
Weldengasse 41
Tel.: 0222/621535

Schweiz

Mailsoft Switzerland
Alex Brander
Feldstr. 20
CH-8330 Pfäeffikon ZH
Tel.: (0) 1 950 5611

Ihre Tips sind gefragt!!!

Wir suchen für unsere **PD-Insider-Tips** jede Menge guter Tips und Tricks zu den verschiedensten PD-Programmen. Ihr Gehirn brauchen

Sie nicht zu zermartern, ein bißchen Überlegung oder einige zündende Ideen tun's auch. Ihre Mühen sollen **nicht** unbelehrt bleiben!

Die drei besten Tips erhalten je ein **PD-Paket** nach Wahl von den Firmen:

Herrmanns&Kommelter
(Share)
und

A.P.S. Electronic

Also, ran an die Tasten, und geschrieben, was das Zeug hält.

Zu schicken ist das Ganze an:

DMV-Verlag
Redaktion AMIGA DOS
Stichwort: PD-Insider-Tips
Postfach 250
3440 Eschwege

Wer kennt sie nicht? Die kleinen Soldatenfigürchen, die fast jeder Junge während seiner Kindheit zum Spielen benutzte? Während der Lausbub mit einer Armee aus Plastikkämpen vorlieb nimmt, läßt der General a.D. detailgetreue Nachbildungen aus Zinn gegeneinander antreten, um längst verlorene Schlachten zumindest als Planspiel im Sandkasten siegreich zu vollenden. Natürlich bedarf es bei solchen Planspielen einiger präziser Regeln, die mit den Spielen Heranwachsender nichts mehr gemein haben. Es wurden Mechanismen für den Krieg im Sandkasten entwickelt, die die kämpferischen Fähigkeiten, die Verteidigungsbereitschaft und die Kampfmoral der Zinnsoldaten als absolute Zahlenwerte darstellen und das Ergebnis von Kampfhandlungen durch Zufallsoperationen mit diesen Werten ermitteln.

Vom Zinnsoldaten zum Simulationspiel

Was sich im ersten Moment ein wenig konfus anhört, wird anhand eines Beispiels weit besser deutlich: Zwei feindliche Kompanien stehen einander auf ebenem Gelände, bei bestem Wetter gegenüber. Beide sind gleich ausgerüstet und sind einander konditionell ebenbürtig. Im Spielsystem gleichen sich die Werte für Attacke, Parade, Motivation und so weiter. Führt eine Truppe nun einen Angriff aus, muß sie die Verteidigung des Gegners überwinden. Das Spielsystem ermittelt nun, basierend auf den bekannten Grundwerten, eine Zahlenskala, auf der sich ein Nullpunkt nebst positiven und negativen Zahlenbereich befindet. Ein Zufallsgenerator ermittelt für den Angreifer nun eine Zahl, die sich im Wertebereich der ermittelten Skala befindet. Befindet sich die Zufallszahl im positiven Skalenbereich, ist die Attacke geglückt. In unserem Beispiel waren die Grundvoraussetzungen beider Parteien fast gleich, also würde hier eine Skala ermittelt werden, die eine 50:50-Chance darstellen kann. Eben eine Skala mit fast gleichgroßen positiven und negativen Bereichen. Dieses Sys-

PD-Spieleshow

Diese Spieleshow präsentiert Ihnen Features wie "Simulation" und einige ansprechende Spiele.



stem könnte nun nach Belieben erweitert werden. So könnten in einem weit ausgefeilteren System eine Summe von Zufallsentscheidungen zusammengekommen werden, um die Parameter für weitere Entscheidungen zu bilden.

Der Durchbruch für den "Krieg im Sandkasten" kam mit den Computern. Hier war es auf einmal möglich, eine Vielzahl von kleineren Entscheidungen zu bündeln, da der Computer für diese sonst zeitaufwendige Arbeit nur wenige Sekunden braucht. Mit immer ausgeklügelteren Wertesystemen und feineren Definitionen von Moral und Material der Gegner zogen auch die simulierten Konflikte immer weitere Kreise. Konnte früher gerade einmal die "Schlacht bei Waterloo" nachgestellt werden, brachte der Computer den Hobbystrategen sogar den dritten Weltkrieg in erreichbare Nähe.

Komplexe Szenarien zum Thema "Was wäre wenn?" ließen sich plötzlich auf das einfachste simulieren. Kriege, die ganze Kontinente einneben, finden im Computer statt, vom Anwender durch einen Tastendruck aktiviert.

Auf jeden Fall fordern die Feldzüge der Hobbystrategen kei-

ne Menschenopfer. Wahrscheinlich ist eher, daß mit der Erkenntnis, wie schnell unser Planet durch einen Krieg unbewohnbar geworden ist, ein neues Verhältnis zum Krieg entsteht.

Der Frage nach Pro und Kontra weiter nachzugehen, ist müßig. Beide Seiten sind in der Lage, vernünftige Argumente gegeneinander ins Feld zu führen. Wahrscheinlich wird auch bei dieser Frage der goldene Mittelweg der gangbarste sein.

(hs/vb)

In das Genre "Simulationen", dessen grundlegende Züge wir gerade erläutert haben, paßt das folgende Spiel.

Als Herrscher über das römische Reich hat man es schwer. Nicht nur, daß irgendwo in Gallien ein kleines Dorf den römischen Soldaten mit unsoldatenhaften Mitteln Paroli bietet, nein, auch die Mitspieler versuchen sich den gesamten Mittelmeerraum unter die imitierten Klauen zu reißen.

So geschehen in »Imperium Romanum«, einem Shareware-Spiel, das auf der »Time 9« zu finden ist. Der Autor, Ro-

land Richter, hat ein Strategiespiel entwickelt, das sich nach geschichtlichen Tatsachen richtet, nämlich der Gründung des römischen Reiches. Bis zu vier Mitspieler können am Spiel teilnehmen, jeder von ihnen bekommt ein Land zugeordnet. Durch geschicktes Hantieren mit Armeen und Geld muß man nun versuchen, seinen eigenen Machtbereich auszudehnen. Dies geschieht durch Okkupation der anliegenden Staaten.

Imperium Romanum

Der Zufall entscheidet nun, ob man die Armee des avisierten Staates schlägt oder man sich selbst geschlagen geben muß. Durch taktische Überlegungen und richtigen Einsatz der eigenen Armee, die entweder übers Wasser (sofern Häfen vorhanden sind) oder zu Land in die gegnerische Provinz einfällt, erkämpft man sich so langsam, aber sicher sein Reich. Das eroberte Gebiet wird durch unterschiedliche Farben gekennzeichnet, die Informationen zu den Ländern (über deren Stärke) werden rechts auf den Bildschirm angezeigt. Damit es nicht ganz so martialisch wirkt, besteht noch die Möglichkeit, die eroberten Länder zu verwalten und dort für einen gewissen Lebensstandard zu sorgen.

Spiele dieses Genres haben immer einen leichten Hauch von Gewaltverherrlichung, da es anscheinend keine andere Möglichkeit gibt, um fremde Länder kennenzulernen, als diese einfach zu überfallen. Gut, man mag sagen, daß es "nur" Computerspiele sind, aber die Wirklichkeit überholt die Simulation tagtäglich, man braucht sich nur die Nachrichten anzusehen. Trotzdem hat das Spiel einen recht hohen Unterhaltungswert, weil es gilt, sich durch Taktik und Überlegung Vorteile zu schaffen, was sich zwar einfach anhört, aber nicht einfach ist. Wer das Spiel erhält, sollte dem Autor den Obulus von 10,- DM nicht vor-enthalten, die Shareware-Regeln gebieten es.

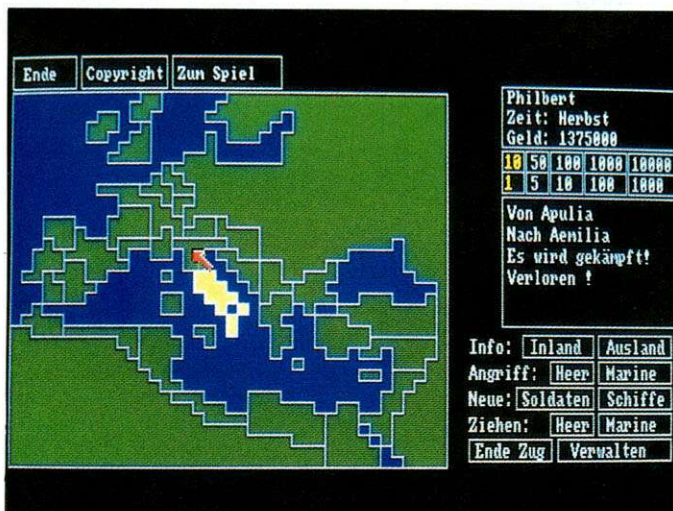
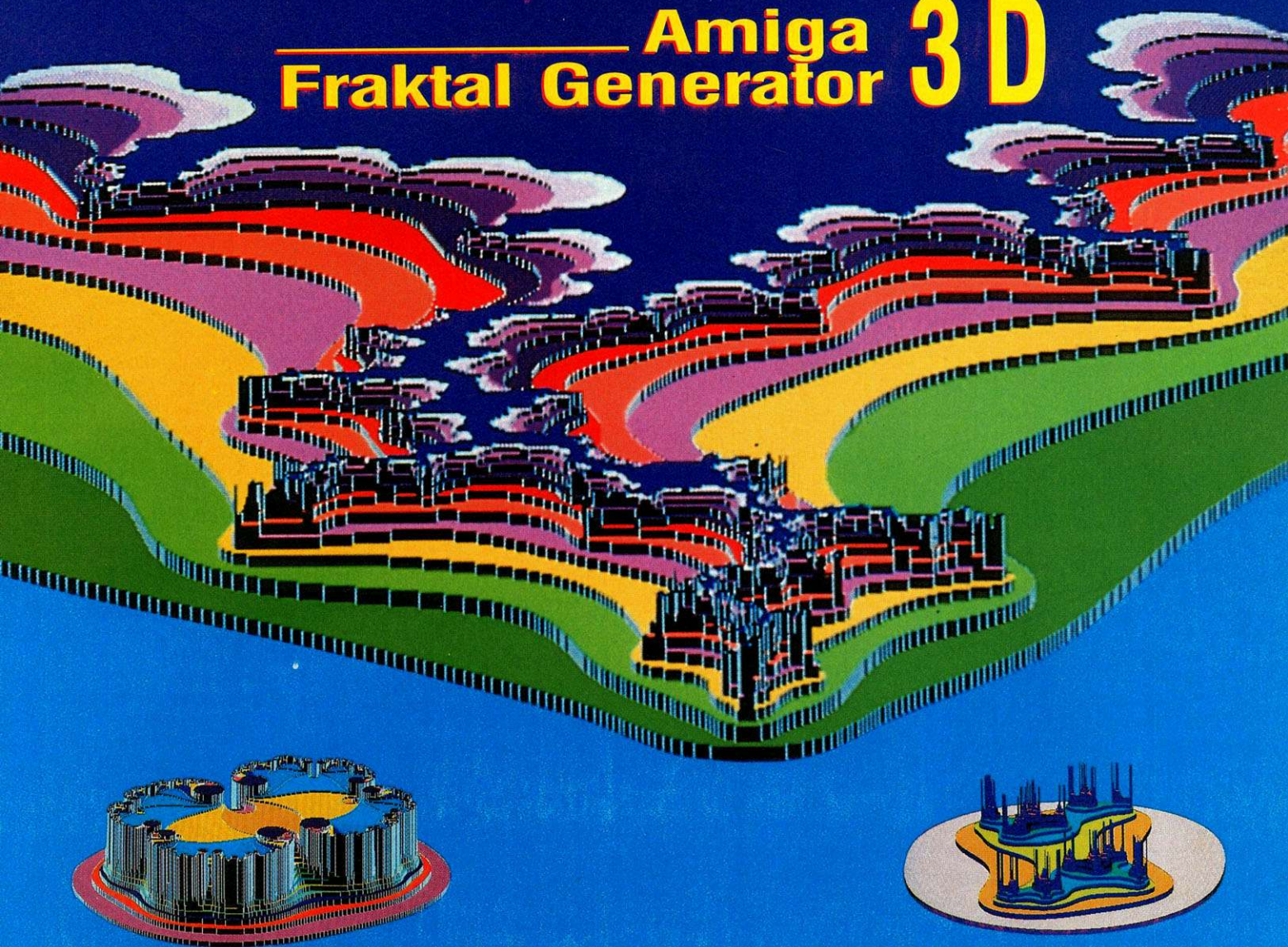


Bild 1. Um Geld und Macht geht es in »Imperium Romanum«

Name: Imperium Romanum
Enthalten auf: Time 9
Anbieter: A.P.S. Electronic

Amiga 3D Fraktal Generator



Supergrafik im Sekundentakt

Vergessen Sie alles, was Sie bisher über Fraktalgrafik-Programme gehört haben
– die unendliche Weite phantastischer Bilder erschließt sich nur über ein
superschnelles Programm: **Fraktal Generator 3D**

High-Speed

Nur noch 7 Sekunden für das Urbild!

Super-Parallel-Projektion

Frei wählbarer horizontaler Blickwinkel mit 360 Grad:
Betrachten Sie das "Fraktalobjekt" von allen Seiten
Stufenloser vertikaler Blickwinkel:
Wahlweise Sicht von oben und unten, schräg oder in der Totalen

Speicherung im IFF-Standard

Einladen der Fraktal-Bilder in Mal- und Zeichenprogramme:
Verwendung als Hintergrund, Motiv oder Vorlage

Voller Bedienungskomfort

Auswahl komplett mit Pull-Down-Menüs
Wahlweise Maus- oder Tastensteuerung

Phantastische Farbmöglichkeiten

32 Farben im Low-Resolution-PAL-Modus
Eigenes Farbrequester mit stufenloser Schieberegung

Mehrere separate Bildspeicher

Bis zu vier Bilder gleichzeitig abrufbar
Separate Farbuordnung und Animationsmöglichkeit

Farb-Animationen

Phantastische Effekte durch Amiga-Color-Cycling

AMIGA 90
Wir stellen aus
09.-11. November 1990
Halle 12, Stand 307

Amiga 3D Fraktal Generator

3 1/2"-Disk. Best.-Nr. 2901

69,- DM (unverbindliche Preisempfehlung)

Wenn Sie über den DMV-Bestellservice bestellen, gilt folgendes:

Inland:		Ausland:	
Einzelpreis	69,- DM	Einzelpreis	69,- DM
zzgl. Versandkosten	4,- DM	zzgl. Versandkosten	6,- DM
Endpreis	73,- DM	Endpreis	75,- DM

– Bitte benutzen Sie die Bestellkarte im Heft. –

DMV-Verlag · Postfach 250 · 3440 Eschwege



Zu einer Zeit, da der C64 noch der unbestrittene Star in meiner bescheidenen Computerausstattung war, fand ich auf einer der damaligen Diskettenmagazine die Simulation eines bis dato unbekannten Würfelspiels, dessen Herkunft im asiatischen Raum zu finden ist: »Yahzee«. Dieses Würfelspiel auf dem Bildschirm erfreute sich kurz darauf größter Beliebtheit, weil es insgesamt vier Personen interessante Abende versprach. Mit der Anschaffung eines Amiga und der Herausgabe des C64 als Finanzierungsobjekt für den großen Bruder geriet dieses Spiel in Vergessenheit, andere Spiele mit grafischen und musikalischen Superleistungen waren jetzt eher gefragt.

Yahzee

Nun, nach Durchsicht der Public-Domain-Disketten, kommt dieses Spiel wieder in den Vordergrund, denn es wurde auf der »Cactus #32« gefunden. Der Autor David N. Junod hat eine mausgesteuerte Version dieses Spiels herausgebracht, bei der vier Personen mitspielen können.

Worum geht es eigentlich bei »Yahzee«? Manche von Ihnen kennen dieses Spiel vielleicht unter dem Namen Kniffel. Fünf Würfel müssen auf den Tisch geworfen werden; die Summe ihrer »Augen«, also der Punkte, wird in eine spezielle Liste eingetragen. Nun wäre das ja kein Problem – wenn die Liste nur für die Eintragung der Punkte da wäre. Doch die müssen anhand von vorgegebenen Schemata in dieser Liste eingetragen werden. Zuerst jedoch, im oberen Teil der Liste, geht es nur um Augenzahlen von eins bis sechs. Würfelt man also zwei Einser, eine Drei, eine Fünf und eine Sechs, bekommt man beim Eintrag in die Einserreihe zwei Punkte, in die Dreierreihe drei, in die Fünferreihe fünf und in die Sechserreihe sechs Punkte. Ist man mit einigen Würfelzahlen nicht einverstanden, besteht die Möglichkeit, sie zweimal zu werfen und einen neuen Wurf zu tätigen, danach jedoch muß das Punktefeld gewählt werden, egal wie.

Im unteren Feld wird es etwas schwieriger; hier muß eine Zahl drei- oder viermal vorhan-

PLAYERS	One		
ONES	4		
TWOES	4		
THREES	12		
FOURS	8		
FIVES	5		
SIXES	12		
UPPER TOTAL	48		
BONUS			
3 of a kind	27		
4 of a kind	8		
Full House	25		
5n Straight	30		
6n Straight	8		
YAHZEE	8		
Chance	15		
LOWER TOTAL	97		
GRAND TOTAL	137		

Yahzee!
COPYRIGHT 1983 BY David N. Junod

Send Donations To:
David N. Junod
4950 Westbrook Hills
Apt. 27-5
Syracuse, NY 13215

Fantastic BBS!
Chiphead's Delight
(315) 492-9871

Largest Amiga-only
BBS in CNY

Bild 2: Viel Glück braucht man bei »Yahzee«

den sein, ein »Full House« wird benötigt (dies sind zwei gleiche und drei gleiche Augenzahlen, zum Beispiel dreimal die Zwei und zweimal die Eins), des weiteren werden eine »kleine« und eine »große Straße« benötigt, bei der vier oder fünf Zahlen in auf- oder absteigender Reihe vorhanden sein müssen (1,2,3,4 und 2,3,4,5 beziehungsweise 3,4,5,6 für die »kleine Straße« und 1,2,3,4,5 oder 2,3,4,5,6 für die »große Straße«) und schließlich das Yahzee-Feld auf dem ein Fünfer-Pasch, also fünf gleiche Augen, seinen Platz findet. Für den Fall, daß sich keine dieser Kombinationen zeigt, gibt es das Feld »Chance«, in dem jede beliebige Augenzahlenkombination eingetragen werden kann. Dieses Feld existiert allerdings nur ein einziges Mal. Sieger ist derjenige,

der die meisten Punkte einheimsen konnte.

Das Spiel auf der Cactus-Diskette entspricht den Regeln des »Yahzee« voll und ganz, kann aber in einigen Punkten nicht überzeugen. Da wäre zuerst einmal das Fehlen des Namens eintrags, die Spieler werden nur als »One« bis »Four« betitelt, die Ergebnisse müssen also auf einem Zettel notiert werden. Der Amiga wird als Gegner nicht zugelassen; eigentlich schade, denn alleine macht das Spiel nur halb soviel Spaß. Ansonsten aber garantiert »Yahzee« die Rettung so mancher aktionslosen Abendstunden.

Name: Yahzee
Enthalten auf: Cactus #32
Anbieter: siehe Liste



Bild 3: Auch Wenden will gelernt sein – zumindest in »The Turn«

Daß man durch eine Wende viel erreichen kann, dürfte wohl seit einiger Zeit bekannt sein.

Weniger bekannt ist wohl, daß man durch Wenden einiger Spielsteine seine Freizeit ganz schön verkürzen kann. »The Turn« von Peter Händel auf der »Time 9« zeigt dies überdeutlich.

Auf einem Spielfeld mit 28 Spielsteinen, angeordnet in einer 7*4-Kombination, werden aus schwarzen Spielsteinen durch Wenden weiße und umgekehrt. Der Spieler kann entweder sogenannte Level-Formationen durchspielen, die vorgegeben sind, oder zufällige Kombinationen vom Amiga auswählen lassen.

Wird ein Stein per Maus angeklickt, so werden alle um ihn herum liegenden Steine gewendet; liegt der angeklickte Stein in der Mitte, werden also die acht um ihn herum liegenden Steine umgedreht; liegt der angeklickte Stein in der Ecke, bleibt es bei drei gewendeten. Durch diese vorgegebene Drehung ist es nicht leicht, die zu erreichende Kombination schnell »herbeizudrehen«, man braucht schon eine Menge Köpfechen, um die Muster durch Drehung abzubilden.

Als kleine Hilfe findet man rechts unten im Bild das durch Drehung zu bildende Muster. Man spielt gegen die Zeit; die verbliebene Restzeit wird dann in Punkte umgewandelt.

The Turn

Wer schon immer ein Freund kniffliger Spielen war, wird bei »The Turn« bestimmt nicht nein sagen. Das Spiel ist grafisch gut gelungen und verliert durch die Möglichkeit der Zufallsmuster auch nicht so schnell an Reiz. »The Turn« ist auf jeden Fall zu empfehlen. Das Spiel befindet sich allerdings auf einer Diskette, die nur auf Amigas mit mindestens einem MByte lauffähig ist. Hier heißt es also aufpassen.

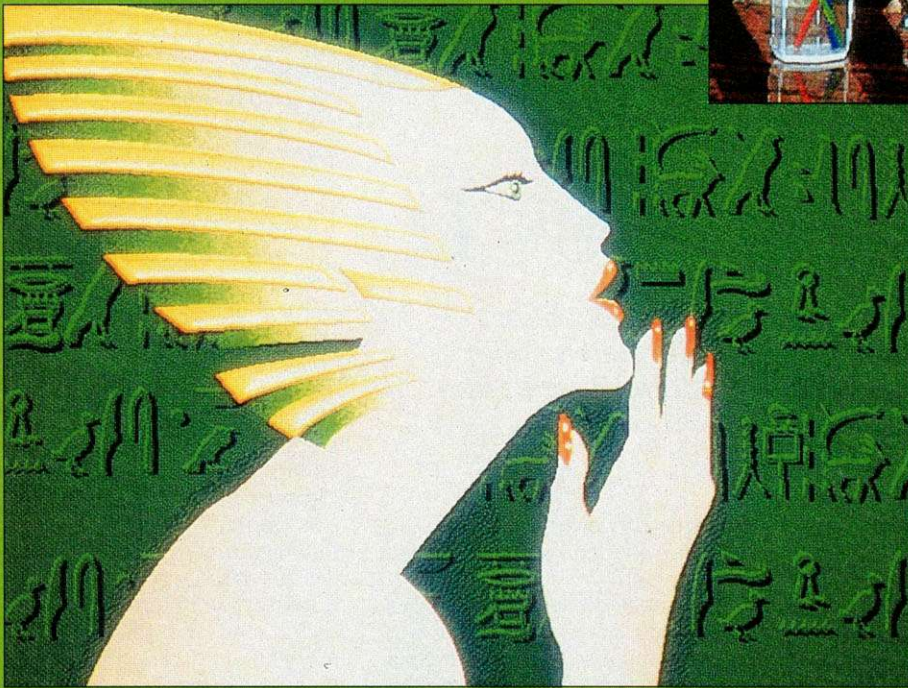
(jb/vb)

Name: The Turn
Enthalten auf: Time 9
Anbieter: A.P.S. Electronic

AMIGA
DOS

Pixel-Panorama

Der Amiga-Benutzer hält nicht nur ein grafisches Kraftpaket in Händen, er will auch sehen, was sich damit machen läßt.



Ein Frauenkopf von H. Magnussen

Auch in diesem Monat erreichten uns wieder viele Einsendungen mit Bildern unserer Leser, aus denen wir einige besonders gelungene Werke herausgesucht haben.

Unser AMIGA-DOS-Logo scheint zu gefallen. M. Nutz aus Schwalbach setzte das Motiv in einem Raytracing-Bild ein.

Das Bild »AMIGA-DOS-Logo mit Lampe« stammt von ihm. Einen ganz anderen Weg bei der Erstellung seines Bildes hat U. Drücke eingeschlagen. Die tollen Effekte, die auch auf seinem hier abgebildeten Motiv »Burgruine« zu sehen sind, erzielt er mit »Digi Paint«, das er vornehmlich im Schatten-Modus verwendet. Frauenköpfe bewegen den Diplom-Designer H. Magnussen. Er schickte uns eine ganze Diskette voller Frauengesichter. Insbesondere die Lady vor den



AMIGA-DOS-Logo mit Lampe – von M. Nutz

Hieroglyphen war so gelungen, daß wir sie Ihnen nicht vorenthalten wollten.

Haben Sie ein wenig Lust bekommen, beim Pixel-Panorama mitzumachen? Nichts leichter als das: Eine Diskette formatieren, Bilder draufkopieren, vermerken, welches Grafikformat und Programm verwendet wurde, und an folgende Adresse schicken:

**DMV-Verlag
Redaktion AMIGA DOS
Kennwort Pixel
Postfach 250
3440 Eschwege**

Für Bilder die wir im Rahmen des Pixel-Panoramas veröffentlichen, winkt ein Dankeschön in Form eines Software-Paketes im Wert von zirka 100 DM.

(hs)



Burgruine – von U. Drücke

Cron, so lautet der Name jener Welt, die es hier zu entdecken gilt. Wer die Packung dieses Spiels zum ersten Mal öffnet, wird zunächst eine Landkarte entdecken, auf der die Welt, die es zu erforschen gilt, dargestellt ist. Im Stile historischer Weltkarten gestaltet, erhält der Spieler einen Überblick über Cron und seine Landschaften, Städte und Burgen. Das Abenteuer »Might and Magic« nimmt in der Stadt Middelgate seinen Anfang. Hier wird eine schlagkräftige Truppe zusammengestellt, die mit dem Abenteurer Seite an Seite gegen Tod und Teufel angehen muß. Schon bald stellt sich heraus, daß der friedfertige Schein trügt. Middelgate wird immer wieder von Bösewichtern heimgesucht, die den armen Bewohnern das Leben schwer machen. Da gibt es zum Beispiel einen alten Zauberer, dem ein magischer Krug gestohlen wurde, oder eine alte Dame, der von Kobolden die ganze Familie entführt wurde. Aus Hilfsbereitschaft und wegen eines leeren Geldbeutels macht sich die Truppe auf den Weg, um die verschleppten Schätze wieder zu gewinnen und die armen Entführten zu befreien.

Schnell ist eine Spur gefunden, die zu einer Grotte im südlichen Teil der Stadt führt. Tief unter der Stadt haben sich Trolle und Kobolde ein regelrechtes kleines Reich eingerichtet. Schnell müssen die Abenteurer feststellen, daß hier nur Waffengewalt etwas auszurichten vermag. Nur wenige Genres können sich auf eine so treue Fange-
meinde stützen wie die Rollenspiele. Und je besser es ein solches Spiel versteht, den Benutzer mit in die fantastischen Länder auf einer fernen Welt zu entführen, desto größer ist die begeisterte Spielerschar. Dementsprechend ist die Aufmachung des Spiels darauf ausgelegt, den Spieler so weit irgend möglich in das Geschehen mit einzubeziehen.

Der Bildschirm teilt sich in verschiedene Fenster, durch die dem Spieler Einblick in die Welt Cron gewährt wird. Hier findet er einen Ausblick in das Szenario ebenso wie statistische Anzeigen, die Aufschluß über den Zustand der Helden geben. Die Spielfiguren ent-



In den Verliesen geht es heiß her

Might and Magic II

Wieder lockt eine Märchenwelt voller Wunder und Gefahren. Ein Reich voller malerischer Städte, düsterer Burgen und dampfender Verliese.

wirft der Spieler zu Beginn des Spiels selbst. Vom Ninja bis zum Barbaren reicht die Palette der möglichen Figuren. Wie bei vielen Rollenspielen entscheidet auch hier die Zusammensetzung der Party mit über Erfolg oder Scheitern der Unternehmungen. So sollte mindestens ein Priester zum Heilen der unvermeidlichen Wunden und ein Zauberer Mitglied der Party sein. Zauberer sind zwar in den niedrigen Erfahrungsstufen eher eine Bela-

stung, in fortgeschrittenen Stadien jedoch wahre Kampfmaschinen. Kämpfe, die im Verlauf der Erkundungen ausgefochten werden müssen, steuert der Spieler über die Tastatur. Eine Option, die Maus oder einen Joystick zu benutzen, gibt es leider nicht. Auch wenn es sich hier um einen echten Mangel handelt, ist »Might and Magic« kein schlechtes Spiel. Vielmehr orientiert sich das Spielkonzept an Evergreens wie

zum Beispiel »Bards Tale« und legt großen Wert auf flüssigen Spielablauf. In der Welt von Cron erwarten den Spieler über 150 verschiedene Monster, Bösewichter und Ungeheuer. Für jede Sorte Bösewicht hält das Programm ein kleines Bild bereit, das eingeblendet wird, sobald die Helden es mit der betreffenden Kreatur zu tun bekommen. Für Variationsreichtum sorgt auch ein gut durchdachtes Magiekonzept, das insgesamt 96 Zaubersprüche mit verschiedenen Wirkungen beinhaltet. Neben den ausgedehnten Ländereien, die es zu erforschen gibt, können die Abenteurer sogar durch die Zeit in andere Epochen dieser Welt reisen. Dies allerdings nicht etwa, wann und wo der Spieler es möchte. Vielmehr kann es passieren, daß die Truppe im Verlauf einer Aufgabe in die Vergangenheit der Welt Cron versetzt wird.

»Might and Magic II« setzt die Tradition wohldurchdachter Rollenspiele fort, die spätestens mit dem ersten Teil von »Bards Tale« begonnen wurde. Dem Spieler wurde auf den Spielverlauf größtmögliche Einflußnahme gestattet. Dabei wurde darauf geachtet, daß nur sinnvolle Kommandos enthalten sind, die den Spielfluß erhalten und den Benutzer nicht verwirren. Sicherlich ist mancher Rollenspielfreund von anderen Spielen her mehr gewöhnt, »Might and Magic« beschränkt sich auf das Wesentliche und bringt dadurch tatsächlich mehr. Wir sind schon auf den dritten Teil gespannt.

(hs)



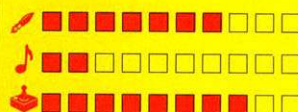
Auch Helden wollen mal was trinken

AMIGA DOS

Blitz★licht

Name: Might and Magic II
Hersteller: New World Computing
Quelle: Fachhandel
Preis: 99,95 DM

GUT





Die Schlacht der Roboter. Können Sie gewinnen?

Robot Commander

Das Los ist auf Sie gefallen. Zerstören Sie das feindliche HQ oder alle Roboter!

Daß sich der Gegner wehrt, dürfte klar sein. Zunächst stellen Sie Ihre Roboter zusammen. Bis zu hundert Stück können Sie aus zwanzig verschiedenen Typen auswählen, die Sie auch noch verändern können. Haben Sie einen Freund zur Hand, können Sie gegen ihn antreten, wenn nicht, übernimmt der Computer diesen Part ganz gut, zumal Sie seine Eigenschaften modifizieren können. Haben Sie sich erst einmal durch die unzähligen Menüs gekämpft und den Computer endlich zufriedengestellt, kann es losgehen. Das erste was auffällt ist, daß das Spiel ziemlich langsam abläuft. Gibt man einem Roboter einen Befehl, der ausgewählt ist aus der Vogelperspektive zu sehen, dauert es eine ganze Weile, bis dieser sich gemächlich in Bewegung setzt. Bei vielen Robotern hat dies der Übersicht wegen seine Vorteile, zu Beginn des Spiels nervt es ganz schön. Ist das Spiel erst einmal am Laufen, kann es doch noch recht interessant werden. Taktiken können ausgeklügelt werden, um den Gegner in einen Hinterhalt zu locken und vieles mehr. Was am Anfang nervt, stellt sich bald als ein Detail heraus, das zusätzlichen Spaß ins Spiel bringt: Nicht alle Roboter können gleich weit schauen, so daß

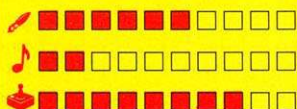
das ohnehin schon stark eingeschränkte Sichtfeld noch kleiner wird. Optisch unterscheiden sich eigene und gegnerische Roboter kaum, so daß es angebracht ist, ein mögliches Ziel zunächst zu identifizieren, bevor man den Schußbefehl gibt. Ansonsten könnte man eine böse Überraschung erleben. »Robot Commander« ist ein Spiel, das, wenn die anfänglichen Schwierigkeiten überwunden sind, vor allem den Taktikern und denen, die sich gerne mit Freunden Gefechte liefern, viel Spaß bereiten kann. (Robert Marz/mm)

AMIGA DOS

Blitz ★ licht

Name: Robot Commander
Hersteller: Ariola
Quelle: Fachhandel
Preis: 69,95 DM

OK



Das Ergebnis des Kampfes ist meist Langeweile

DOMINATION

Acht Kampfroboter, aufgeteilt in zwei Teams, stehen sich in einer Arena gegenüber. Ihr Ziel ist es, das Hauptquartier des Gegners zu vernichten.

Sie und einer Ihrer Freunde übernehmen jeweils ein Team. Sollten Sie niemanden zu einem Spiel überreden können, übernimmt der Computer den Part des zweiten Spielers recht gut. Jeder Spieler bekommt seinen eigenen Teil des Bildschirms, wo er die Taten seines aktuellen Roboters verfolgen kann. Da man nicht alle vier Roboter gleichzeitig steuern kann, können den einzelnen Robotern Aufgaben zugewiesen werden. Bei Suchen schwärmt der jeweilige Roboter aus und sucht nach Kisten mit neuen Ausrüstungsgegenständen, wie Waffen, Panzerung etc. Im Angriffsmodus attackiert er das gegnerische Hauptquartier, während er im Verteidigungsmodus das eigene verteidigt. Mit der Zerstörung eines der HQs ist eine Runde zu Ende, und die Punkte werden zusammengezählt. Dabei gibt es vier Punkte für den Sieg und jeweils einen pro vernichteten Roboter. Nach sieben Runden wird der endgültige Gewinner festgestellt. In der Zeit, die zwischen zwei Runden vergeht, können die einzelnen Roboter mit den gefundenen Gegenständen aufgerü-

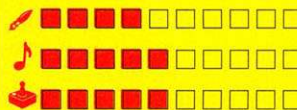
stet werden. Da hier nur begrenzt Zeit zur Verfügung steht, ist es ärgerlich, daß der Pointer auch im Ein-Spieler-Modus nur mit dem Joystick bewegt werden kann. Bei »Domination« dominiert über weite Strecken die Langeweile, selbst der Zwei-Spieler-Modus kann da nur wenig Pluspunkte verschaffen, obwohl Fans dieses Genres sicherlich die vielen vorhandenen Extras zu schätzen wissen werden. (Robert Marz/mm)

AMIGA DOS

Blitz ★ licht

Name: Domination
Hersteller: Micro-partner
Quelle: Fachhandel
Preis: zirka 70 DM

OK



AMIGA DOS -Spieletips

Eine recht kurze und knackige Lösung zu »King's Quest II« sandte uns Daniel Henning aus Dachau zu. Im Zusammenhang mit der Karte sollte auch dieses Sierra-Adventure den Spieler vor keine ernststen Probleme mehr stellen.

Die zweite Aufgabe für den König

Wir starten bei Großmutter's Haus. Hier öffnen wir zunächst den Briefkasten und nehmen den Korb an uns. Wenn wir bei der nächsten Gelegenheit den Korb an Rotkäppchen weitergeben, gibt sie uns dafür Blumen.

In der Kirche (A6) beten wir. Steht der Mönch auf, tippen wir "graham" ein und warten vor der Kirche, bis die Fee auftaucht und einen Schutzzauber niederläßt.

Auf dem Weg über die Brücke (G6) zur Tür (F7) entnehmen wir die "Brooch" aus dem kleinen Steinloch am See (B6). An der Tür lesen wir die Inschrift. Wir gehen zur Gnomenbehauung im unteren Teil des Baums (G4) und nehmen die Suppe mit. Aus der Höhle der Hexe Hagathe (G2) nehmen wir den Vogelkäfig an uns.

Aus dem Bild (D1) die "Clamshell/Bracelett" und aus Bild (D2) nehmen wir den "Stake" mit. Mit dem Dreizack (F1) schwimmen wir vom Ufer aus zur Nixe (C1), der wir den Blumenstrauß überreichen. Auf dem Seepferd reiten wir nach Westen und übergeben Neptun den Dreizack. Aus der nun geöffneten Muschel nehmen wir den Schlüssel und reiten wieder ostwärts zum Ufer zurück.

Aus dem hohlen Baum (F3) nehmen wir das Necklace mit und öffnen die erste Tür mit dem Schlüssel. Jetzt studieren wir die zweite Inschrift und eilen zum Antiquitätenladen (C6/7). Der Frau antworten wir mit "yes" und geben ihr den Vogelkäfig, worauf wir eine Lampe von ihr erhalten.

Mit dem Stoff aus Neptuns Flasche polieren wir die Lampe. "ride carpet" läßt uns auf den fliegenden Teppich steigen und zur Schlange fliegen. Hier reiben wir ein weiteres Mal an der Flasche und töten mit dem

Manche Spiele können einen gar zur Verzweiflung bringen, nur weil der entscheidende Tip fehlt. Gerade Adventure- und Rollenspiele fallen in diese Sparte, aber auch so manches Actionspiel bekommt mit dem passenden Cheatmode erst die rechte Würze...

soeben erhaltenen Schwert die Schlange. Jetzt können wir den zweiten Schlüssel aus der Höhle holen und mit dem Teppich wieder ins Tal fliegen. Wir können nun die nächste Tür öffnen und finden eine weitere Inschrift.

Wir betreten Omis Haus (A3) und geben ihr die Hühnersuppe. Unter dem Bett liegen ein Ring und ein Mantel, beides legen wir an. So gewandt begehen wir uns zum toten See an die Stelle (E4), von wo aus wir das Schloß sehen können. Mit dem Boot überqueren wir den See "enter Boat"; "enter island" und bahnen uns einen Weg durch das Dornengestrüpp. Wir betreten das Schloß und steigen links den Turm hinauf. Aus der Schublade nehmen wir die Kerzen und zünden sie auf dem Rückweg an der Fackel an. Jetzt gehen wir nach rechts, nehmen den Schinken mit und steigen die Treppe hinab. Im Keller gehen wir nach links in den Raum und öffnen den Sarg. Sollte Dracula gerade sein Mittagschläfchen halten, verziehen wir uns schleunigst und zeigen ihm das Kreuz des Mönchs. Ist der Sarg leer, nehmen wir erst das Kissen und dann den dritten Schlüssel. Mit ihm können wir die letzte Tür öffnen und das andere Land betreten. Hier fischen wir zunächst mit dem Netz, bis wir einen Fisch fangen. Dann springen wir ins Wasser. Auf der neuen Insel gehen wir nach Osten und nehmen das Amulett an uns. Im Turm erklimmen wir die Wendeltreppe und besänftigen den wachhabenden Löwen mit dem Fleisch. Jetzt öffnen wir gnußvoll die Tür und geben der Maid einen dicken Kuß. Der letzte Befehl für dieses Spiel lautet "home". War doch gar nicht so schwer,

oder? Wie schwer es aber ein Bürgermeister hat, haben alle die festgestellt, die sich mit »SimCity« ihre eigene Stadt aufbauen. Da man dort recht schnell in Geldnöte kommen kann, haben wir für Sie einige Tips zusammengefaßt, die unter anderem von Thorsten Schulz aus Hattingen stammen.

Simulierte Städte

Wenn man einen kleinen Programmfehler ausnützt, kann man immer, wenn man knapp bei Kasse ist, seine Finanzen auf \$20000 auffrischen. Das ist relativ einfach. Man muß dazu sein Geld nur regulieren, zum Beispiel mit dem Bulldozer auf \$9. Das einzige, was man jetzt noch tun kann, ist Stromleitungen legen und den Bulldozer benutzen. Sprengt man jetzt mit dem Bulldozer ein Haus, was ja \$10 kostet, aber ohne weiteres geht, hat man einen Kontostand von \$1. Speichert man jetzt ab und lädt erneut ein, kommt das Programm durcheinander, da es normalerweise ja keine negativen Beträge in der Kasse geben kann, und gibt uns \$20000. Die Sache hat noch einen anderen positiven Effekt: Lädt man nämlich einen gespeicherten Spielstand wieder neu ein, werden die Mittel für Straße, Polizei und Feuerwehr auf 100 Prozent gesetzt, so daß man, wenn man für diese Einrichtungen nicht genügend Geld übrig hat, nur zu speichern und erneut einzuladen braucht.

Eine andere Möglichkeit, an \$10000 zu kommen, ist die, bei gedrückter Shift-Taste das Wort »FUND« einzugeben. Man erhält das Geld dann sofort. Benutzt man diesen Cheat aber, bekommt man bei der Endwertung Punkte abgezogen. Außerdem sollte man die-

se Möglichkeit sowieso nicht allzusehr strapazieren, da sonst "aus heiterem Himmel" ein Erdbeben die Stadt heimsucht.

Bei der eigenen Stadt sollte man auf folgendes achten: Die einzelnen Gebiete müssen getrennt voneinander, auf einem Szenarium mit möglichst viel Küste angelegt werden, da Gebiete am Wasser am meisten Wert besitzen.

Die Industrie benötigt die Eisenbahn und später auch den Hafen, deshalb sollte in Industriegebieten auf den Ausbau der Eisenbahn besonders geachtet werden. An Kraftwerken legt man sich am besten nur Atomkraftwerke zu. Die sind zwar etwas teurer, leisten aber mehr und verursachen weniger Luftverschmutzung; welche Gefahren sie allerdings bergen, kann man im Szenarium Boston kennenlernen.

Da die Bürger in sauberer Luft leben wollen, sollte man Industrie und Wohngebiete durch den Einzelhandel trennen, ohne dabei aber zu lange Wege zu den Arbeitsplätzen zu schaffen. Für die Bürger machen ein Park und vor allem niedrige Steuern das Wohnen in der Stadt attraktiv. Da ein Stadion hohes Verkehrsaufkommen zur Folge hat, sollte es etwas außerhalb, aber in Reichweite der Wohngebiete angelegt werden. Der Flughafen, der früher oder später vom Einzelhandel gefordert wird, verursacht ebenfalls Schmutz. Spätestens mit dem Flughafen sollte das Feuerwehernetz stark ausgebaut werden, da Flugzeuge meistens über dicht besiedeltem Gebiet abstürzen. Brennt ein Industriegebäude, und der Brand kann nicht rechtzeitig gelöscht werden, so muß das Grundstück abgerissen werden, da es sonst zu Explosionen kommen kann. Für jedes abgerissene Grundstück sollte aber auf jeden Fall Ersatz geschaffen werden. Für den Fall, daß Katastrophen eintreten, ein Flugzeugabsturz reicht schon, sollte man größere Gebiete auf verschiedenen Wegen an das Stromnetz anschließen, um so die Gefahr eines großflächigen Stromausfalls möglichst einzudämmen.

Da neue Mission-Disketten für »Fighter Bomber« angekündigt wurden, ist der folgende Cheat vielleicht doch wieder interessant, obwohl noch keine Tests mit den neuen Disketten möglich waren:

Fighter Bomber

Bei der Originalversion von »Fighter Bomber« kann man nämlich unter dem Namen »Buckaroo« sämtliche Missionen fliegen.

Von Hans Ehmke aus Heidelberg erhielten wir eine Karte und ein paar Tricks zu »Drakken«, einem sehr schweren, aber gute Rollenspiel:

Drakken

Aller Anfang ist leicht, zumindest wenn man sich an den folgenden Tips orientiert:

Wir suchen den Palast mit dem Hai auf. Im Vorraum können wir das Magnetfeld ausschalten, indem wir das zweite Zeichen von links betätigen. Wir

nehmen den Schild von der Wand und gehen durch die rechte Tür. Hier nehmen wir alle Waffen und Schilder auf, betreten den nächsten Raum und verfahren dort ebenso. Wenn wir jetzt den Palast verlassen und ihn erneut betreten, können wir erfreut feststellen, daß die Waffen ersetzt wurden und den Vorgang wiederholen. Beim Waffenschmied verkaufen wir alle nicht benötigten Gegenstände und legen das Geld in Bögen und Rüstungen (Kleidung) an.

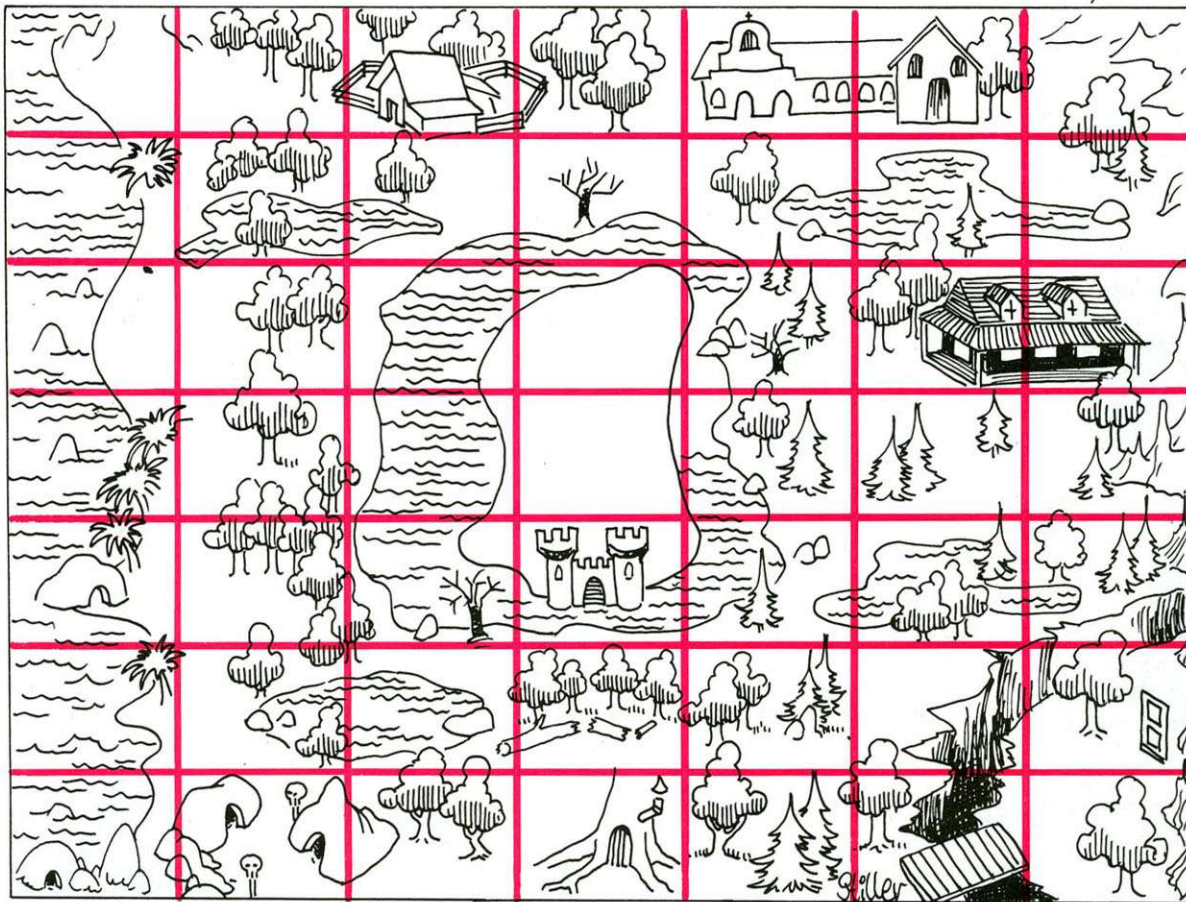
Jetzt kehren wir in den Palast zurück und können ihn komplett durchsuchen. Dabei benutzen wir aber jeweils nur eine, mit einem Bogen bewaffnete Person, der Rest sollte in der Vorhalle warten. Der beste Schutz zu dieser Zeit ist die Unsichtbarkeit. Bei Kämpfen mit den Bewohnern des Palastes erbeutet man unter anderem auch Bögen. Gelegentli-

ches Austauschen der aktiven Figur ermöglicht den wichtigen, gleichmäßigen Kräftezuwachs der Gruppe. Da der Kampf mit Prinz Hordkhen zu dieser Zeit ein zu hohes Risiko darstellt, sollte man ihn nach Möglichkeit vermeiden. Wenn alle mit einem Bogen ausgerüstet sind, können wir zum nächsten Wegkreuz gehen und es im eingeschalteten Kampfmodus berühren. Hier können gefahrlos Punkte gesammelt werden, wenn man auf die Lebenspunkte achtet und sich gelegentlich regeneriert. So lange nicht alle Figuren Level 5 überschritten haben, verbringen wir die Nächte im gefahrlosen Umfeld von Gebäuden.

Für später rät Hans folgendes: Da der Transfer von Gegenständen zwischen den Figuren nur dann funktioniert, wenn sie sich im selben Raum befinden, sollten diese über einen toten

Charakter ausgetauscht werden. Dieser ist nämlich jederzeit für alle erreichbar. Dieser Trick sollte natürlich nur ausgeführt werden, wenn der Zauberer bereits Tote erwecken kann. Bei Hans kommt es ab Level 12 aus unersichtlichen Gründen zu einer Passivität der Figur (kein Kampf, keine Zaubersprüche, Figur ist nicht mehr animiert). Die passende Medizin dazu hat er allerdings gleich mitgeliefert: Die entsprechende Figur sterben lassen, zwei Tage und Nächte ruhen lassen und erneut zum Leben erwecken, danach ist alles wieder in Ordnung. Zum Schluß bleibt noch folgendes zu sagen: Obwohl die Zeit ständig verstreicht, ist keine Eile vonnöten. In den Palästen läuft die Zeit nur in den Räumen, in denen wir uns befinden, alle verwüsteten Paläste sind schon seit Spielbeginn in diesem Zustand.

KINGS QUEST II



Karte zu King's Quest II

In jedem Abschnitt der Insel gibt es zwei Paläste, den des Prinzen und den der Prinzessin. Jedem Element (Feuer, Erde, Wasser, Luft) ist ein Paar zugeteilt. Nur im Norden wurde trotz eifriger Suche nur ein Palast entdeckt.

Zutritt zu den Palästen:

Palast mit Hai:

Aktive Figur muß unsichtbar sein.

Palast mit Zugbrücke:

Zauberspruch (U) Minarett:

Aktive Figur muß Führer der Truppe sein. Linken Turm anklicken, nach Stillstand der Figur den rechten Turm anklicken. Wurde die richtige Figur gewählt, betreten wir den Palast, wenn nicht, unser Pech.

Palast im Norden:

Zum richtigen Zeitpunkt wird uns Einlaß gewährt.

Die restlichen Paläste:

Ein bißchen Spielspaß muß ja auch noch bleiben!

Zur Karte: Die Wege zwischen zwei Punkten berücksichtigen nicht den tatsächlichen Verlauf der Straße, unwichtige Kurven und Schleifen wurden ausgelassen. Einige Zelte und Iglus, die abseits der Wege liegen, sind ebenfalls nicht eingezeichnet worden.

Ebenfalls zu »Drakken« erreichte uns ein Brief von Stefan Goebels, der einen Trick zum Verändern der Charaktere auf Diskette parat hat.

Drakken-Schummeldisk

Das Verändern der Charaktere ist gar nicht so schwer. Man erstellt eine normale Charakterdisk und schreibt sich die Namen in der richtigen Reihenfolge auf.

Mit Hilfe eines Diskettenmonitors, den man sich preiswert in der PD besorgen kann, findet man den ersten Charakter in Block 1420 (Track 64, Sector 1, Head 1). Der zweite Charakter liegt dann in Block 1420-21, der dritte in 1421-22 und der letzte in 1422-23. Jetzt braucht man nur noch den Namen des Charakters zu suchen und es kann losgehen, der Datenaufbau darf aber nur in den eingeklammerten Bereichen verändert werden!

Block 1420, Track 64, Sector 1, Header 1:

00: xxxxxxxx (xxxxxxx)
xxxxxxx xxxxxx)xx

Hier kann der Name verändert werden.

10: xxxxxx(mw) xxxxxxxx
(xpxpxpx) xxxxxxxx

mw:12:männlich, 16:weiblich;
xp:Experience Points.

20: xxxxxxxx xxxxxxxx
xxxxxxx

30: xxxxxxxx xxxxxxxx
xxxxxxx

40: xxxxxxxx xxxx(kkkg
iwwbbgl)

50: xxxxxx(II) xx(llnnnn)
xxxxxxx xxxxxx(mm)

60: (mm)xxxxxx xxxxxxxx
(g1g1g1g1 g1g1g2g2)

70: (g2g2g2g2 g3g3g3g3
g3g3g4g4 g4g4g4g4)

80: (g5g5g5g5g5 g5g5g6g6g6
g6g6g6g6 g7g7g7g7)

90: (g7g7g8g8 g8g8g8g8
w1w1w1w1 w1w1w2w2)

a0: (w2w2w2w2 w3w3w3w3
w3w3w4w4 w4w4w4w4)

b0: (w5w5w5w5 w5w5w6w6
w6w6w6w6 w7w7w7w7)

c0: (w7w7w8w8 w8w8w8w8)
xxxxxxxx xxxxxxxx

Dabei bedeuten:

kk: Kraft, 00-ff

gg: Geschicklichkeit, 00-ff

ii: Intelligenz, 00-ff

bb: Bildung, 00-ff

gl: Glück

l: Lebenspunkte

m: Magiepunkte

Liste zu g1-g8:

Waffen:

Die letzten beiden Ziffern bestimmen einen Aufschlag auf die Schlagkraft der Waffe, normal ist hier 0a.

Schwert: 3b 02 0d 3a 07 xx

Degen: 3f 02 08 38 01 xx

Dolch: 44 04 1f 1a 01 xx

Keule: 3f 02 06 3f 01 xx

Bogen: 3f 01 06 41 08 xx

Stock: 3f 02 08 3e 01 xx

Kleidung und Rüstung:

Hier können jeweils die zweiten und die letzten beiden Ziffern verändert werden, was sich auf den Schutz des Charakters auswirkt. Normalerweise sind die in Klammern angegebenen Werte voreingestellt.

Schuhe: 5f (01) 00 08 01 (00)

Jacke: 5f (0a) 00 0e 01 (00)

Schild: 5f (32) 14 2d 01 (00)

Hemd: 5f (0a) 00 0d 01 (00)

Stiefel: 5f (0a) 00 0a 02 (00)

Kleid: 58 (08) 3c 18 0c (00) nur für Priester

Schild II: 5f (64) 32 30 12 (0a)

Harnisch: 5f (64) 3c 11 0b (0a)

Schützer: 5b (64) 32 1d 07 (0a)

Kleid: 5a (64) 1e 1a 01 (32) nur für Zauberer

Liste zu w1-w8:

Zepter: 9f 01 23 05 00 00
(macht unsichtbar)

Ring: 1f 01 67 07 00 00

Zepter: 9f 00 2d 05 00 00
(Schutz)

Ring: 1f 01 6c 07 00 00 (+2
Glück)

Fackel: 1f e2 80 42 00 00

Schlüssel: 1f 00 a9 43 00 00

Zepter: 9f 02 22 05 00 00 (un-
antastbar)

Ring: 1f 00 28 05 00 00 (macht
unsichtbar)

Auch wenn das hier etwas chaotisch aussieht, sollte man sich schnell hineinfinden. Auf eines muß jedoch geachtet werden: Die Checksumme des Blockes darf bei dem Abspeichern nicht verändert werden. Gute Diskettenmonitore fragen vorher nach.

Für das Suchtspiel »Klax« hat Karsten Lotz aus Esslingen einen Cheat herausgefunden, den wir Ihnen natürlich nicht vorenthalten wollen:

Klax

Drückt man während dem Spiel die Taste "3", kommt man einen Klax-Wave weiter, drückt man die "4", kommt man sogar in den letzten (das ist der hundertste) Level und kann dort eine Million Bonuspunkte absehen, wenn auch dieser gemeistert wird. Notfalls nimmt man Taste "3" zur Hilfe.

Auch in den ersten Levels kann man kräftig Punkte sammeln, indem man die Klötzchen so stapelt, daß fünf Zweierhäufchen einer Farbe entstehen. Auf die vier äußeren läßt man jetzt jeweils einen Klotz der Farbe des mittleren Haufens fallen. Darüber bildet man dann wieder Zweierhäufchen der gleichen Farbe wie unten. Läßt man jetzt einen Klotz der richtigen Farbe auf den mittleren Haufen fallen, verschwinden alle Steine und man erhält zirka 250000 Punkte.

Jochen Rutschlein aus Radolfzell sandte uns eine Komplettlösung zu »Kult«. Da das Spiel nicht unbedingt leicht ist, haben wir uns entschlossen, sie auch Ihnen zugänglich zu machen.

Kult

Ziel des Spiels ist es, unsere Freundin zu finden, sie zu befreien und mit ihr diese Welt zu verlassen. Vorher müssen jedoch fünf Prüfungen abgelegt werden, für die man vier Gegenstände benötigt. Den er-

sten Gegenstand erhält man vom Meister persönlich, die anderen muß man durch Tausch oder beim Changer besorgen. Dazu bringen wir ihm die Totenschädel und können im nächsten Zug einen Gegenstand tauschen. Bei dem Becherspiel sollte man aber vorher abspeichern, da es auf Zufall hinausläuft. Die erste Prüfung: "De Profundis"

Dazu brauchen wir das Seil. Zunächst inspizieren wir die Plattform und warten, bis der Haken aus der Decke kommt. Wir benutzen das Seil als Lasso und warten, bis das Granitmonster erscheint. Jetzt springen wir auf das Monster. Bevor wir es zum Ausgang ziehen, nehmen wir das Seil wieder mit. Der Schädel, der nun vor dem Ausgang liegt, wird automatisch aufgenommen.

Die zweite Prüfung: "In Gegenwart des Skorpions"

Hierzu brauchen wir die Steinfliege. Nachdem wir die Statue angebetet haben, gehen wir durch die Spinnwebentür. Hier kriechen wir durch die Spinnweben am Boden. Wir schenken der Herrin die Fliege, umarmen sie, lehnen ihr Liebesangebot aber ab; wer ein wenig über das Liebesleben der Spinnen Bescheid weiß, ahnt warum. Nun geben wir der blauen Spinne die Fliege und verlassen den Raum. Wir stecken uns die rote Spinne in den Mund und passieren so die Falltür. Hier nehmen wir den Schädel in Empfang, denn die Prüfung ist schon bestanden. Da wir den Gegenstand nicht zurückbekommen, müssen wir auf einen tauschwilligen Aspiranten warten. Wollen wir nicht tauschen, können wir ihm den Gegenstand auch mit Telekinese entreißen.

Die dritte Prüfung: "Die Wand" Hierzu benötigen wir den Dolch. Es ist wichtig, daß die Wände halb geschlossen sind. Sollte das nicht der Fall sein, betreten wir die zweite Zone, verlassen den Raum und kommen sofort wieder. Nun gehen wir solange zwischen den Zonen hin und her, bis die Türen ganz geschlossen sind und wir in der zweiten Zone stehen. Wir gehen nun durch die linke Seitentür, stellen uns auf die Stufe und stecken den Dolch in den Schlitz. Wir nehmen nach der Türdrehung den Dolch von den Dingen im Raum und ge-

hen durch die linke Passage. Dort legen wir die Hand auf das Loch im Boden und der Schlüssel wird gefunden. Nun nichts wie zurück durch die hintere Passage, die Prüfung ist gelöst.

Die vierte Prüfung: "Der Gehängte"

Hier brauchen wir die Laterne. Haben wir sie nicht, können wir im dunklen Raum mittels "Solar" Licht machen. Mittels "Fliege" gehen wir auf die

Plattform und betätigen den Hebel. Etwa in der Mitte der Plattform liegt der Schädel in einer Öffnung. Steigen wir jetzt von der Plattform hinunter, ist die Prüfung gelöst.

Die fünfte Prüfung: "Die Zwillinge"

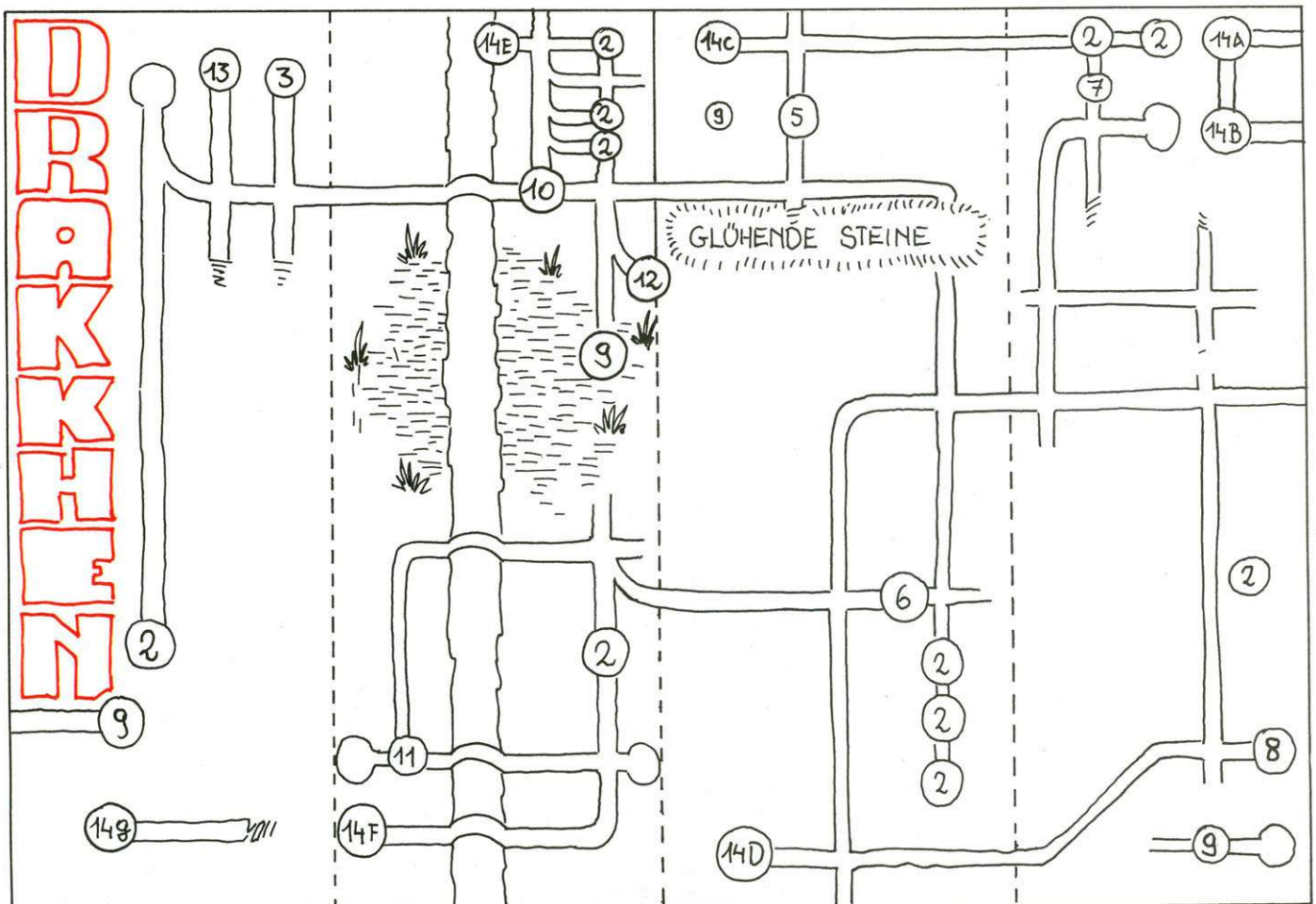
Benötigt wird der Becher. Wir öffnen beide Schlangen, gehen zur Quelle und inspizieren den Schlangenkopf. Wenn wir auf das Auge drücken, können wir den Becher füllen. Wir leeren den Becher in die linke Schlange, nehmen den Würfel von den Dingen im Raum und stecken ihn in die rechte Schlange. Jetzt begeben wir uns in den Raum "Wer wird errettet" und heben die beiden Hände ganz rechts und links außen. Nach einem Druck auf den Schlangenkopf merken wir uns die Gravur auf der nun erscheinenden Platte. Im Quellenraum inspizieren wir die Plattformwürfel. Der Würfel mit der entsprechenden Gravur wird aufgehoben, in ihm liegt der Schädel. Der Raum kann nur noch in den Ring ver-

lassen werden, da die beiden anderen Türen verschlossen sind.

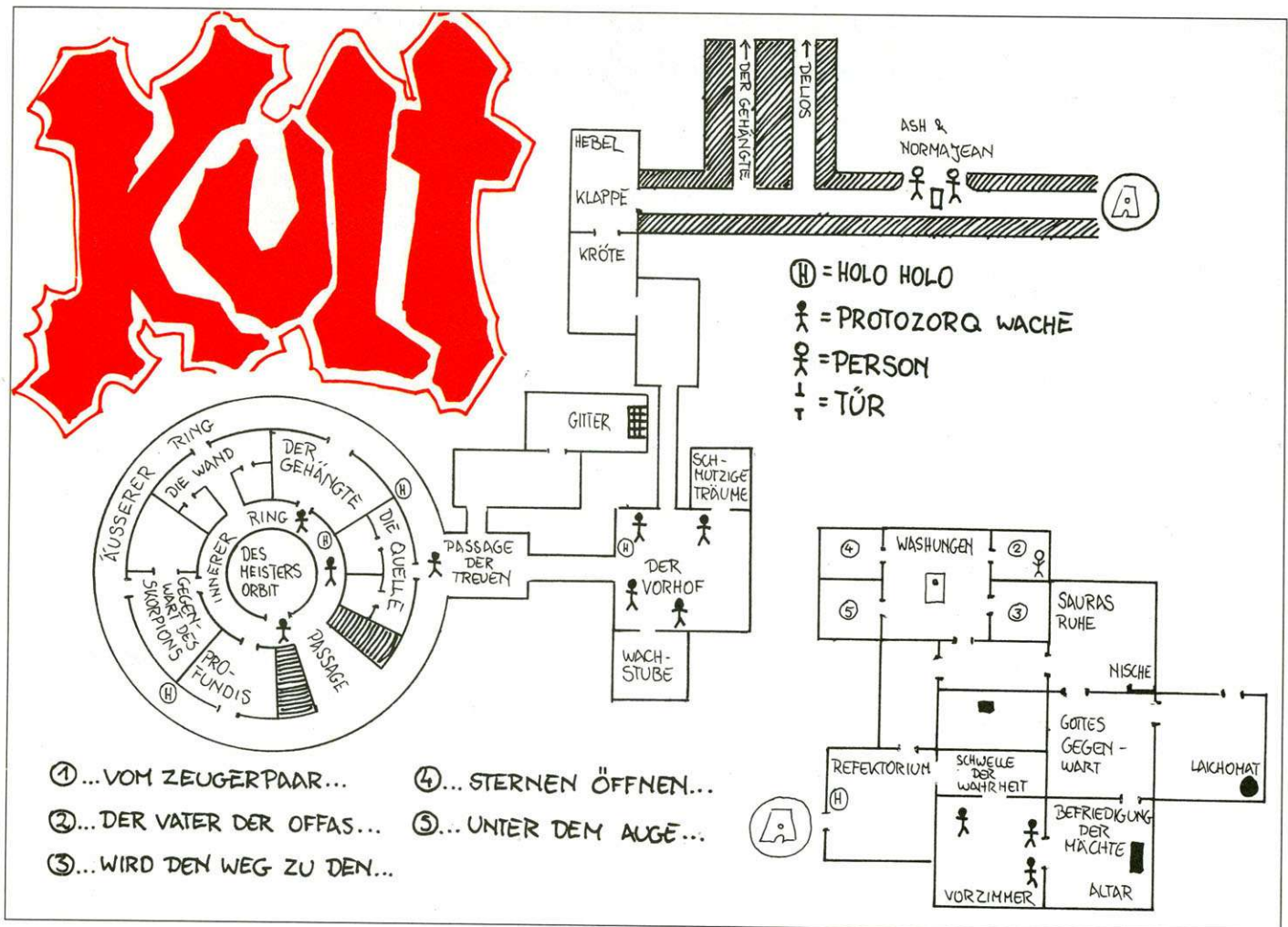
Nachdem wir den fünften Schädel beim Changer abgegeben haben, erscheint der Meister der Prüfungen mit zwei Wachen. Wir erhalten vom Meister ein "Ey", welches sich noch als sehr wichtig erweist. Im Ring passieren wir die Wache vor der Passage der Treuen, indem wir das "Ey" vorzeigen und gehen in den Vorhof. Hier gehen wir aber nicht in den Traumraum, sondern betäuben die Wache vor dem oberen Ausgang mit "Hirnwischer" und gehen dort hinaus. Jetzt ist Eile erforderlich, denn sonst schnappen uns die Wachen, die uns verfolgen. Wir gehen links durch die Passage und dann geradeaus. Wir untersuchen die Kröte und drücken dann in ihr Auge. Nachdem sich die Wand gedreht hat, machen wir mit "Solar" Licht und bewegen den Hebel mit Hilfe von Telekinese. Den Raum selbst dürfen wir auf keinen Fall betreten. Wir

Legende zur Drakken-Karte:

- 1: Position zu Spielbeginn
- 2: Wohnungen (Häuser, Zelte, Iglus)
- 3: Waffenschmiede
- 4: Palast mit Hai (Prinz der Erde)
- 5: Palast (Prinzessin der Erde)
- 6: Gasthof
- 7: Pyramide
- 8: Minarett
- 9: Tempel
- 10: Palast (Prinzessin des Wassers)
- 11: Palast mit Zugbrücke (Prinz des Wassers)
- 12: Friedhof
- 13: Palast im Norden
- 14a: Teleportfeld zu 14g
- 14b: Teleportfeld zu 14g
- 14c: Teleportfeld zu 14d
- 14d: Teleportfeld zu 14c
- 14e: Teleportfeld zu 14f
- 14f: Teleportfeld zu 14e
- 14g: Teleportfeld zu 14a



Karte zu Drakken



Karte zu Kult

gehen nun an dem Gitter auf der linken Seite vorbei und immer weiter geradeaus durch die Galerie. Schließlich erreichen wir eine große Höhle, in der wir die an uns gestellte Forderung akzeptieren. Danach erzählen wir die Wahrheit.

Jetzt führen wir einen Zonenscanner aus und entdecken eine Phiole zwischen den Dingen im Raum. Der Mumie übergeben wir die Bohne, ansonsten lassen wir sie aber in Ruhe, da sie sonst zerfällt. Wir gehen durch den linken Gang bis zu einer Klappe, passieren sie und finden uns im Refektorium wieder. Mit dem Hackbeilchen gehen wir durch den unteren Ausgang zur Schwelle der Wahrheit. Wir besiegen die Priesterin und stecken das "Ey" in den geöffneten Mund der Statue links. Untersuchen wir jetzt das Pult, gelangen wir in den Raum hinter das nun gehobene Gitter. Auf einem Schrank steht ganz oben eine Statue, die wir mittels Telekinese herunterholen. Wir verlas-

sen den Raum wieder und treten durch die rechte Tür in den Raum "In Gottes Gegenwart". Hier verweilen wir nur kurz und gehen wieder rechts in den Raum "Befriedigung der Mächte". Eine weitere Priesterin muß besiegt werden und der Hohepriesterin Saura wischen wir das Hirn und geben ihr anschließend aus der Phiole zu trinken. Überrascht und erfreut stellen wir fest, daß Saura unsere Freundin ist, und trinken zur Regeneration unserer PSI-Punkte selbst aus der Phiole. Wir gehen jetzt zweimal nach links zu "Sauras Ruhe" und stellen dort die Statue in das Fach neben die andere. In den sich öffnenden Tunnel stecken wir den Affen, gehen nach oben zu "In Gegenwart Gottes" und verwenden den Zonenscanner ein weiteres Mal, worauf sich die Sternenvand hebt. Wir treten hindurch und warten einen Zug lang. Jetzt müssen wir den Gott Zorq mit "extremer Gewalt" besiegen und den

Hirnwischer auf Harrks anwenden. Wir schließen die Klappe mittels Telekinese und warten einen weiteren Zug. Zum Schluß jagen wir Harrks noch das Beil in den Rücken, und wir haben es geschafft.

Lars Knutzen sandte uns einen Tip zu »Starglider II«. Wie man hier relative Unverwundbarkeit erlangen kann, ist gar nicht schwer:

Starglider II

Wir halten den Icarus mit F an und richten ihn in Richtung Boden. Jetzt gehen wir mit der Backspace-Taste in den Pausenmodus und tippen "were on a mission from god1" ein. Die daraufhin erscheinende Versionsnummer beantworten wir mit "K" und können von nun an nur noch an den Egrogen-Silos kaputtgehen.

Ebenfalls von Lars Knutzen stammt dieser Tip zu »Ghouls'n'Ghosts«:

Ghouls'n'Ghosts

Wenn wir während des Spiels »KARENBROADHURST« ein-

geben und der Cheat vom Spiel bestätigt wurde, können wir ohne Gefahr die Gegner berühren. Die Hintergrundgrafik bleibt allerdings weiterhin gefährlich.

Auch die Tips zu »Forgotten Worlds« und zu »Silkworm« stammen von Lars Knutzen, die wir Ihnen, auch wenn die beiden Ballergames etwas älter sind, trotzdem nicht vorenthalten wollen.

Vergessene Welten

Im Titelscreen tippen wir »ARC« ein und drücken die Help-Taste, worauf wir im Spiel mit »L« sofort in den nächsten Level und mit »S« in den nächsten Shop gelangen.

Seidenwürmer

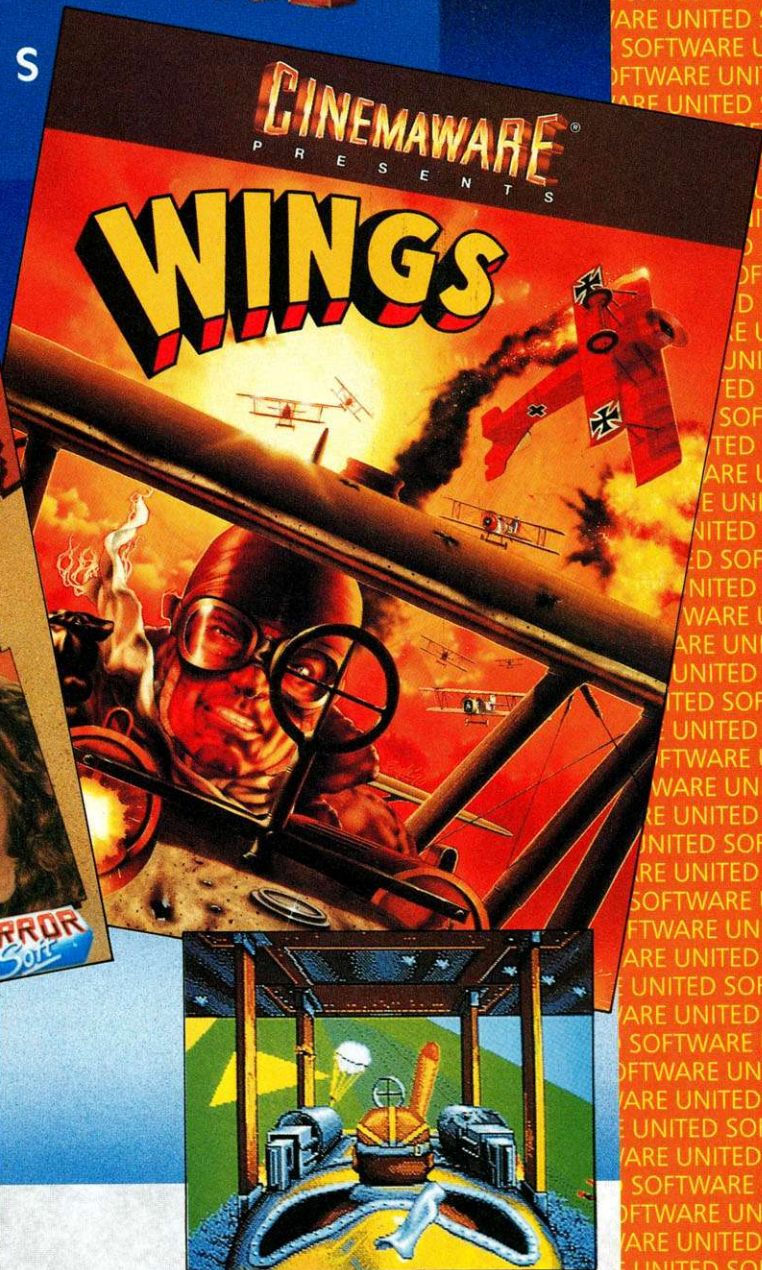
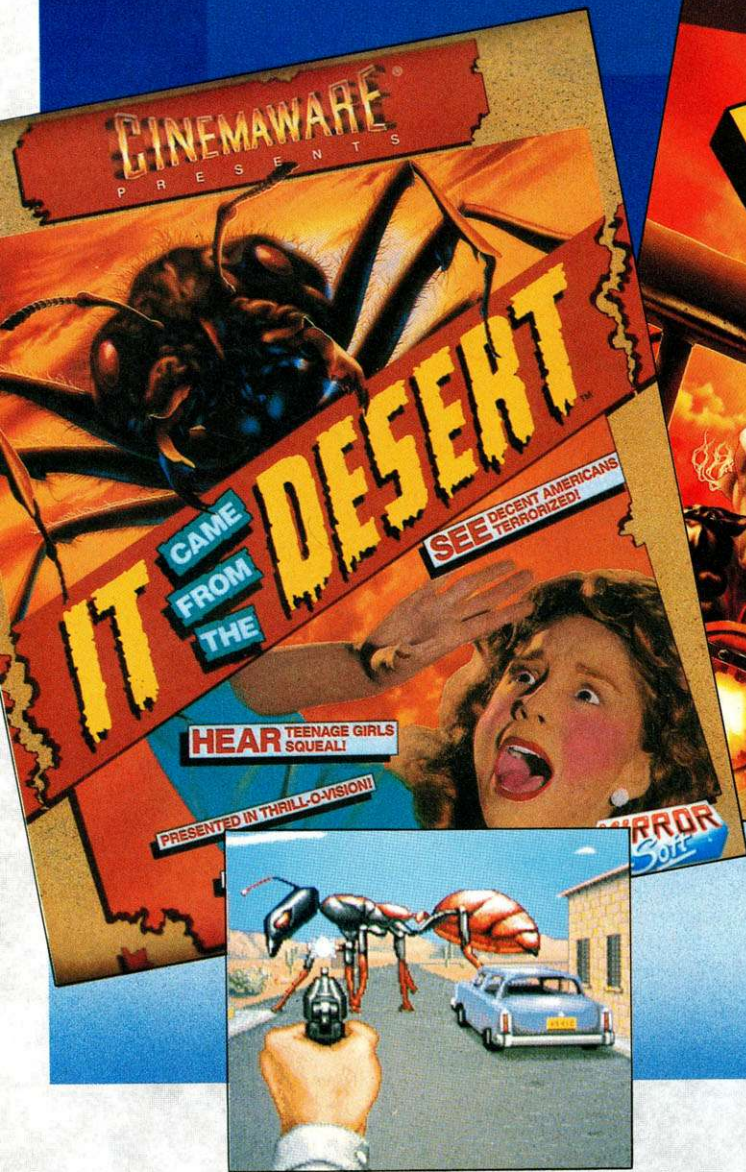
Bei »Silkworm« brauchen wir nur im Titelscreen [F10] zu drücken und »scrap 28« einzugeben, worauf wir unendlich viele Leben erhalten und mit den Tasten [1] bis [8] zwischen den Levels eins bis zwölf wählen können.

(Robert Marz/mm)

Erfolgreich!

CINEMAWARE

P R E S E N T S



CINEMAWARE hat sich einen führenden Namen mit seinen erstklassigen Arcade-Adventures wie ROCKET RANGER und IT CAME FROM THE DESERT, sowie den beiden Sportspielen TV SPORTS BASKETBALL und FOOTBALL geschaffen. Jetzt wenden sich die Filmspezialisten aus den Vereinigten Staaten dem populären Bereich der Flugsimulation zu. Mit WINGS kommt der neue Streifen in Ihre Computerkinos. Natürlich im typischen CINEMAWARE-Stil mit Edelgraphik- und Sound, verbunden mit einem historischen Hintergrund.

WINGS für Amiga. Schon jetzt erfolgreich!

CINEMAWARE
Mirrorsoft

United Software
Das relevante Programm

Obwohl diese Tatsache schon seit langem bekannt ist, hat man bis jetzt nur wenig dagegen unter-
nommen. Zum Schluß sind Sie als Fachmann hinzugezogen worden...

Das Gamma-System verfügt über neun Planeten und fünfzehn Monde, auf denen Sie sich mit Fahrzeugen oder zu Fuß frei bewegen können. Für die Gefährte benötigen Sie allerdings Zündschlüssel. Gleiter können sich mit jeweils zehn Geschwindigkeitsstufen nach vorne oder hinten bewegen. In der schnellsten Stufe kann man meistens nicht mehr exakt steuern, da die Geschwindigkeit viel zu groß ist. So ist es sinnvoll, die gewünschte Richtung in einer geringen Geschwindigkeit einzustellen und erst dann zu beschleunigen.

Gleich nach Ihrer Ankunft auf Eris erwartet Sie ein Gleiter, dessen Schlüssel im Flughafen liegen.

Sie bekommen während des ganzen Spiels Ihre Umgebung in 3D-Vektorgrafik aus Ihrem Blickwinkel zu sehen. Die Planeten sind riesig groß und normalerweise in der anfänglich zur Verfügung stehenden Zeit keinesfalls vollständig zu erforschen. Das soll man auch gar nicht, sondern man soll sich auf sein Gefühl verlassen und intuitiv handeln. Laut Anleitung gibt es eine Vielzahl von Lösungswegen, und auch wenn der Planet zerstört werden sollte, geht das Spiel noch weiter. Alles zu durchsuchen und zu erforschen ist beinahe unmöglich. So gibt es auf einem Planeten Millionen von fast völlig gleich aussehenden Pyramiden. Klar, daß nicht in jeder ein wichtiges Teil stecken kann. Man kann und soll nicht alle Gebäude durchsuchen. Bedeutsame Gebäude unterscheiden sich meist in irgendeiner Form von sonst gleichen Nachbarn. Trifft man zum Beispiel auf ein rosa Häuschen inmitten einer Gruppe von grünen oder auf ein ein wenig abseits stehendes, könnte das etwas zu bedeuten haben.

Gefundene Gegenstände kann man benutzen, verwahren oder verkaufen, um sich das nötige Kleingeld für andere Gegenstände zu besorgen. Viele der Gegenstände kann man auch benutzen und untersuchen. Die Schlüssel für die



Retten Sie Eris, die Hauptwelt des Gamma-Systems

DAMOCLES

Ein Komet droht Eris, den Hauptplaneten des Gamma-Systems, zu vernichten. Nur noch wenige Stunden Zeit bleiben, etwas gegen den Kometen zu unternehmen.

Fahrzeuge wurden bereits erwähnt, aber auch Akten erweisen sich als nützliche Hilfsmittel zur Informationsbeschaffung. Um von einem Planeten zum anderen zu kommen, können Sie Raumschiffe benutzen. Da sie sehr schnell fliegen, vergeht für Sie die Zeit aber schneller als auf den Planeten, so daß ein Flug wertvolle Zeit kostet. Schaut man sich auf den Planeten und Monden um, findet man viele interessante und hilfreiche Dinge. So

liegt zum Beispiel im Gebäude des Außenministeriums ein Ordner, in dem Sie sich über den Kometen und Dinge, die mit ihm zusammenhängen, informieren können.

Sie können mitnehmen, was immer Sie finden. Ob es jedoch sinnvoll ist, Tische und Bänke an sich zu nehmen? Zumal Sie höchstens zehn Gegenstände gleichzeitig tragen können und ein zu hohes Gewicht Sie langsamer macht. Und für Diebesgut solcher Art

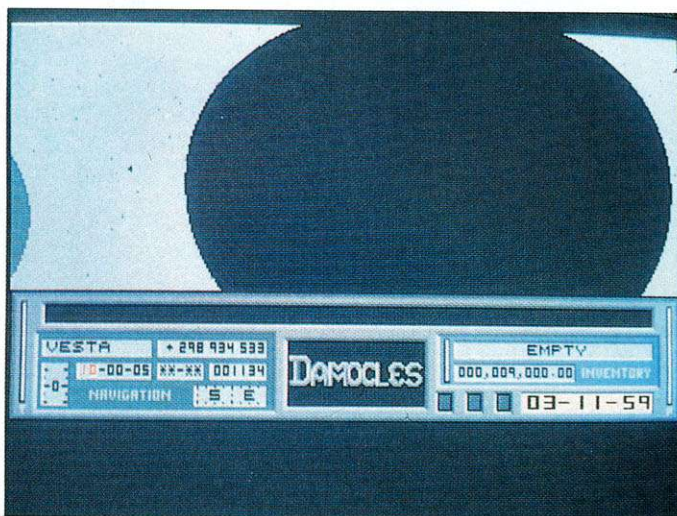
bekommt man beim Händler im allgemeinen auch nicht viel. »Damocles« ist ein Spiel mit praktisch uneingeschränkten Handlungsmöglichkeiten für den Spieler. Was zählt, ist Kombination und schnelles Handeln, um die Aufgabe zu erfüllen.

Ein besseres und vor allem wesentlich schnelleres interplanetares Transportsystem als die Raumschiffe ist ein Teleporter, der nicht schwer zu finden ist und in der vierten Etage eines Hauses steht. Hat man erst einmal seine Funktion verstanden, kann man blitzschnell von einem Planeten zum nächsten wechseln.

Die einzelnen Häuser und Städte sind durch Straßen für die Gleiter beziehungsweise Flugzeuge verbunden. An die Straßen muß man sich allerdings nicht halten. Es ist auch durchaus möglich, nach Koordinaten, die man ständig eingeblendet sieht, zu fliegen und so sein Ziel wesentlich schneller zu erreichen.

Tüftlern und Freunden von Action-Adventures mit 3D-Grafik kann der Damocles als Nachfolger von »Mercenary« ruhigen Gewissens empfohlen werden. Wer aber eine kräftige Ballerei erwartet, wird enttäuscht sein; obwohl fast alle Fahrzeuge mit Waffen ausgerüstet sind, gibt es laut Anleitung keinen Feind, der sich aggressiv verhält. Provozieren Sie jedoch einen Angriff, werden fremde Schiffe sich natürlich verteidigen und versuchen, Sie zu vernichten.

(Robert Marz/mm)



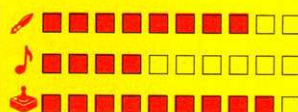
Es gibt viel zu entdecken

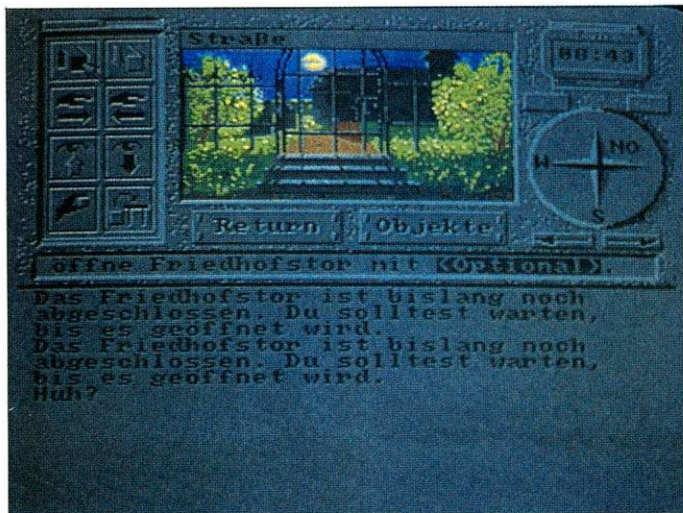
AMIGA DOS

Blitz ★ licht

Name: Damocles
Hersteller: Novagen
Quelle: Fachhandel
Preis: 84,95 DM

GUT





Inspektor Griffu tappt im Dunkeln

Inspektor Griffu

Irritierte Inspektoren, böse Blutsauger, panische Priester und bibbernde Bauern. Jeder Vampirfilm ist voll davon!

Das Grauen kommt per Mausklick. Im neusten Abenteuerspiel vor Ariola Soft übernimmt der Spieler die Rolle des Inspektor Griffu, um einem meuchelnden Blutsauger das greuliche Handwerk zu legen. Neu ist dieses Thema nicht gerade. Der Held des Spiels beginnt seine Nachforschungen zuerst im örtlichen Gasthof. Gesteuert wird Inspektor Griffu mit der Maus, was die lästige Tipperei erspart. Sämtliche möglichen Aktionen und sichtbaren Objekte sind per Mausklick erreichbar, was den bekannten "Abenteuerspiel-Effekt" des zweifelten Wortsuchens umgeht. Vorhandene Gegenstände und Personen werden auf zwei verschiedenen Bildschirmen dargestellt, der erste für die im Raum vorhandenen Objekte, der zweite für das Inventar. Zwischen diesen beiden Bildschirmen kann über einen Schalter umgeschaltet werden. Die Aktionseingabe erfolgt durch Anklicken des entsprechenden Symbols und (sofern erforderlich) eines oder mehrerer Objekte. So entstehen komplette Sätze, die der Parser (der Übersetzungsteil des Programms) verarbeiten kann.

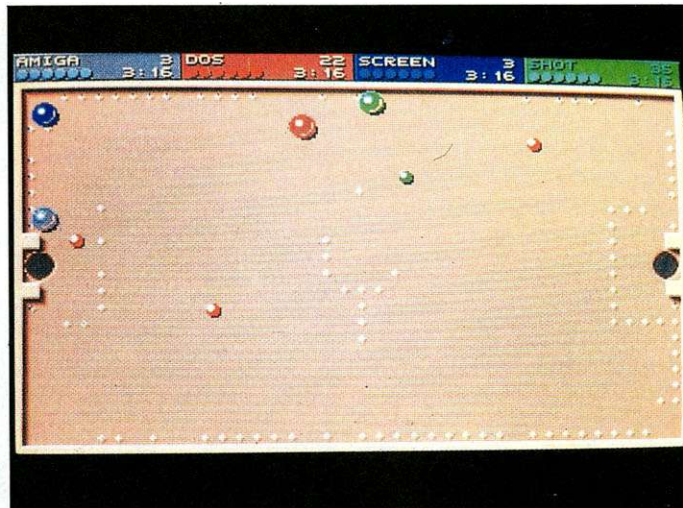
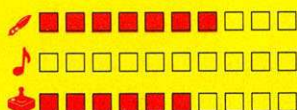
Technisch gesehen ist Inspektor Griffu ein gutes Abenteuerspiel. Hätten die Programmierer des Spiels etwas mehr Phantasie investiert oder sich eines ganz anderen Themas bedient, wäre ein sehr gutes Abenteuer entstanden. Dennoch, Fans dieses Genres finden hier sicher eine Bereicherung ihrer Programmsammlung. Denn spannender als ein Krimi im Fernsehen ist Inspektor Griffu allemal.

(tb)

AMIGA DOS
Blitz ★ licht

Name: Inspektor Griffu
Hersteller: Ariola Soft
Quelle: Fachhandel
Preis: 69,95 DM

OK



Mit Geschick und Glück kommt man ans Ziel

Qatbol

Wer kennt sie nicht, die Labyrinthspiele aus Holz, bei denen eine Kugel aus Metall durch Kippen der Platte manövriert werden muß?

Ein findiges Programmierer-Team hat sich drangesetzt und eine Umsetzung auf Heimcomputer programmiert. Und da man bei Computern die Naturgesetze so richtig auf den Kopf stellen kann, wurden einige Dinge integriert, die in normalen Holzspielen gar nicht realisierbar sind. So gibt es Löcher, die keine sind, versteckte Löcher, Sprenglöcher, die nach Berührung eines Zünders explodieren, Magnete, Teleportfelder und vieles mehr. Und damit nicht immer nur einer spielt und der Rest zuschaut, können sich bis zu vier Spieler gleichzeitig an einem Level versuchen.

Damit bei mehreren Spielern das Chaos nicht komplett wird, bezieht sich eine durch Joystick oder Maus hervorgerufene Kippung des Brettes immer nur auf die eigene Kugel. Auch das Ziel des Spiels wurde ein wenig abgewandelt: Es müssen nämlich zunächst eine Anzahl von Bonuskugeln in die Löcher geschubst werden, bevor man sich selbst versenken darf. Für die Punkte, die man dafür und für das Aufsammeln von Bonuspunkten erhält, kann man sich später

neue Kugeln kaufen. Die Kugeln gibt es in drei Ausführungen, aus Glas, Metall und Plastik, die natürlich unterschiedliche Eigenschaften haben. Glas zerspringt sogar, wenn es zu fest gegen Wände prallt.

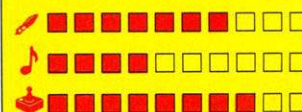
Alles in allem ist »Qatbol« empfehlenswert, denn wenn man mit Freunden oder auch nur mit dem Computer gleichzeitig in der Arena ist, kommt fast immer Stimmung auf.

(Robert Marz/mm)

AMIGA DOS
Blitz ★ licht

Name: Qatbol
Hersteller: Ariola Soft
Quelle: Fachhandel
Preis: 69,95 DM

GUT



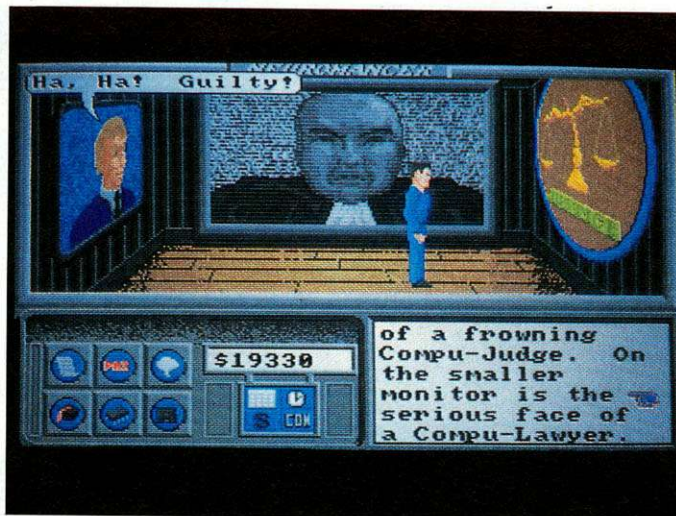
Die Grafik ist sehr stark verbessert worden; wer Neuromancer schon auf dem C64 gespielt hat, wird sie kaum wiedererkennen. Auch der Sound wurde Amiga-Verhältnissen angepaßt, so daß einem ungetrübten Spielchen eigentlich nichts mehr im Wege steht. Um es gleich vorwegzunehmen: An der Handlung hat sich nichts verändert. Die Texte wurden eins zu eins übernommen, und auch die Link-Codes und Paßwörter sind gleichgeblieben.

Aber für alle, die Neuromancer nicht kennen, hier erst einmal die Story:

Sie sind Case, ein Cyberpunk, wie sich die Computerecks in der Zukunft nennen werden. Für sie zählt das Fleisch, also der Körper nichts, sondern lediglich das, was man computermäßig drauf hat. Auch die Computer haben sich ein wenig verändert. Es gibt ein allumfassendes Netzwerk, die Matrix, in die sich die Decks (die Computer) einloggen und dem Benutzer ein Bild davon direkt ins Hirn projizieren können. Das sieht dann aus wie ein dreidimensionales Gitter, in dem sich der Cyberpunk frei bewegen kann.

Die festen Datenbasen, sprich Mailboxen, werden als geometrische, sich bewegende Gebilde dargestellt, an die man andocken kann. Damit aber niemand unbefugt an die Basen heran und Daten klauen kann, sind sie mit sogenanntem Eis geschützt, das ein Einbrechen verhindern soll. Das Eis ist ein Programm, das den Cyberpunk direkt angreift und versucht, ihn hirntot zu machen, sprich auszuschalten. Der Punk seinerseits kann das Eis mit Eisbreaking-Software angreifen und versuchen, es mürbe zu machen und eine Stelle zum Durchbrechen zu finden. Einmal drin in der Basis, kann er dann beliebig an den Daten manipulieren, sich dort gespeicherte Programme auf sein eigenes Deck ziehen und alles, was ein Hacker heute auch schon kann, tun.

Das gefährlichste, was sich in der Matrix befindet, sind die künstlichen Intelligenzen, die Als. Sie sind nur extrem schwer zu schlagen und das schon gar nicht mit Software, sondern nur mit Skills, die man sich kaufen und direkt ins Gehirn einbauen kann.



Überall gibt es Probleme: In der realen Welt...

Neuromancer

Lange haben die Neuromancer-Fans auf eine Umsetzung des Kult-Spiels gewartet. Jetzt endlich, nach mehr als zwei Jahren Wartezeit ist sie da.

Und es gibt das PAX-System, in dem jeder Bürger seine eigene ID und ein dazugehöriges Konto hat. Von diesem Konto können, falls vorhanden, Credits auf den hip geladen werden, die dann wie Bargeld funktionieren.

Es gibt also eine ganze Menge Probleme, denen sich Case gegenübergestellt sieht, sowohl in der realen Welt als auch in der Matrix. Er ist am Anfang nämlich mit lächerlichen 2006 Credits ausgestattet und hat sein Deck beim

Pfandleiher versetzt. Außerdem ist sein Deck nicht Cyberspace-fähig, was offiziell ja auch verboten ist, und bislang nur mit »ComLink 1.0« ausgestattet, einer Software, die nur zu sehr wenigen Datenbasen eine telefonische Verbindung herstellen kann und den Eintritt in den Cyberspace natürlich nicht ermöglicht.

Sein primäres Problem ist es also, sich mit Geld auszustatten, ein gutes Deck zu kaufen und an »ComLink 6.0« heranzukommen, die höchste und

als einzige Cyberspace-fähige Linksoftware. Bis dahin ist allerdings ein steiniger Weg, der allzuoft im Gericht landen wird. Dort wartet ein Verteidiger auf uns, der uns gegen Geld die Todesstrafe verschaffen will, und ein Richter, der uns ständig zu höchsten Strafen verurteilt. Ist aber kein Geld auf dem Konto, kann er es auch nicht einziehen, hat man aber alles bei sich, geht es verloren, wenn man stirbt und wieder erweckt werden muß.

Der Cyberspace ist in Bereiche aufgeteilt, die nur von bestimmten Cyberjacks aus erreicht werden können. Cyberjacks sind Buchsen, über die man sich an die Matrix ankopeln kann. Hier ist ein weiteres Problem, denn an wichtige Jacks kommt man gar nicht so leicht heran. In der Matrix ist Case dann fast ganz auf sich gestellt. Aber es gibt noch sogenannte Rom Constructs, eine Digitale Nachbildung ehemaliger Cyberspace-Größen, die mit Rat und Tat zu Hilfe stehen. So können sie unter anderem begrenzt kaputte Software auch im Cyberspace debuggen.

»Neuromancer« ist ein Adventure-artiges Spiel, das mich schon auf dem C64 begeistert hat. Jetzt, auf dem Amiga, hat es mich extrem positiv überrascht. Die Schlußgrafik und der Sound waren schon früher die Krönung, auf dem Amiga sind sie einfach super. Schade nur, daß das Spiel nur mit 1 MByte Speicher läuft. »Neuromancer« ist sein Geld in jedem Falle wert.

(Robert Marz/mm)



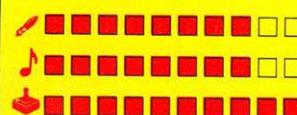
...und im Cyberspace, denn das Eis schlägt zurück!

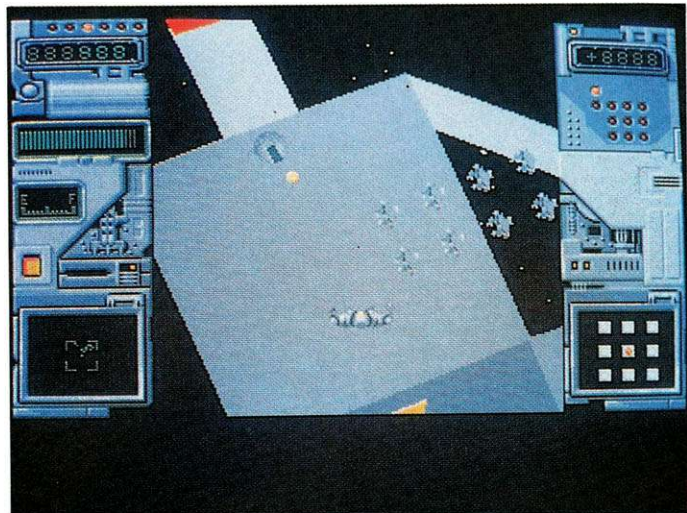
AMIGA DOS

Blitz ★ licht

Name: Neuromancer
Hersteller: Electronic Arts
Quelle: Fachhandel
Preis: 79,95 DM

SUPER





Im Kampf hat Rotox eine Menge zu tun



Im 22. Jahrhundert wird Leutnant Kowolski bei einem Spezialeinsatz schwer verwundet und in einen Cyborg verwandelt...

Im Testkomplex, der freischwebend im Welt-
raum errichtet wurde, soll er
seine Fähigkeiten unter Be-
weis stellen.

Hier hat er die Möglichkeit, die
Waffen zu sammeln, die er für
seine Flucht benötigt.

Der Spieler sieht den Cyborg,
den er lenkt, das ganze Spiel
über aus der Vogelperspekti-
ve. Beim Lenken dreht sich
nicht Rotox selbst, sondern
der Boden unter ihm, eine Me-
thode, die zunächst erfolgreich
Verwirrung stiftet. Der Test-
komplex besteht aus verschie-
denen Plattformen, die alle
gänzlich von den Finsterlingen
befreit werden müssen, die
dort ihr Unwesen treiben. Jede
Plattform ist dabei in neun Ab-
schnitte unterteilt, die mei-
stens irgendein Bild darstellen.
Extrawaffen liegen in den Ab-
schnitten verstreut auf dem
Boden herum. Sie haben nur
einen begrenzten Energievor-
rat, der allerdings wieder auf-
gefrischt werden kann. Stirbt
Rotox aber, oder fällt er von der
Plattform, sind sämtliche bis-
her gefundenen Extrawaffen
verloren. Ist ein Abschnitt ge-
säubert, wird das dem Spieler
durch ein Aufblitzen des Bild-
schirms mitgeteilt. Dann ist es
an der Zeit, den nächsten Ab-

schnitt aufzusuchen. Die ein-
zelnen Abschnitte sind durch
sich in unterschiedlicher Form
bewegende Brücken verbun-
den, so daß es gar nicht so
leicht ist, sie zu überqueren.

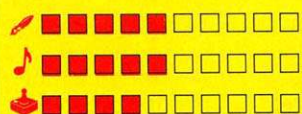
»Rotox« ist ein Spiel, das nur
ausgesprochene Fans dieses
Genres für längere Zeit zu be-
geistern vermag. Immerhin
liegt dem Spiel eine deutsche
Anleitung sowie eine Kurzge-
schichte bei.

(Robert Marz/mm)

AMIGA DOS
Blitz ★ licht

Name: Rotox
Hersteller: U.S.Gold
Quelle: Fachhandel
Preis: 84,95 DM

GEHT SO



Software-Noten und wie sie zu- standekommen

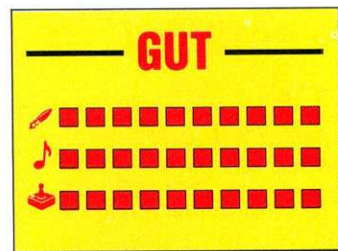
Daß Ihre AMIGA DOS bei den Spiele-Tests ein Urteil
vergibt, haben Sie selbst sicherlich schon gesehen.
Wie dieses Urteil aufgebaut ist, das soll hier noch
einmal im Detail erklärt werden.

AMIGA DOS beurteilt jedes Spiel anhand verschiedener
Kriterien: Grafik, Musik und Spielspaß. Die Noten dafür
werden in LED-ähnlichen Diagrammen, die am Ende jeder
Review zu finden sind, unter den Oberbegriffen Grafik,
Sound und Motivation dargestellt. Diese Oberbegriffe wer-
den durch einen Pinsel für Grafik, eine Note für Sound und
einen Joystick für Motivation symbolisiert.

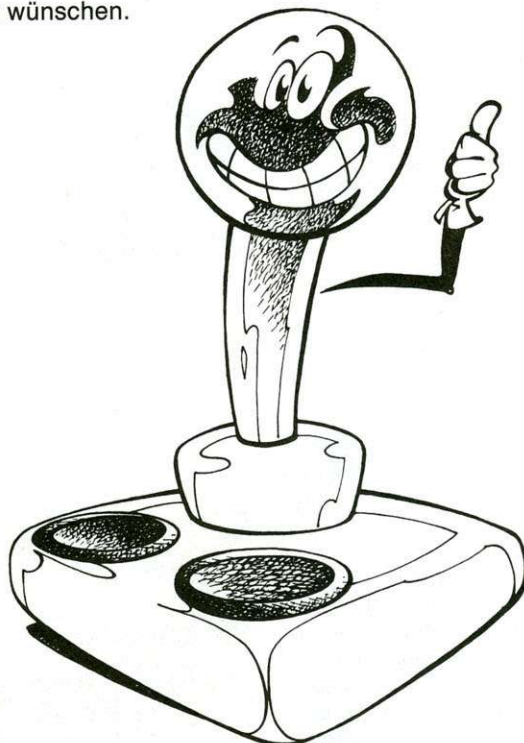
Die Note selbst richtet sich nach der Höhe des wiederge-
gebenen Wertes. Je mehr Einheiten des Diagramms aus-
gefüllt sind, desto besser ist die Beurteilung für den ent-
sprechenden Oberbegriff. Zu diesen Noten gesellt sich
ferner ein individuelles Urteil des Testers. Eine vierte Note
also, die allerdings nicht wiedergegeben wird.

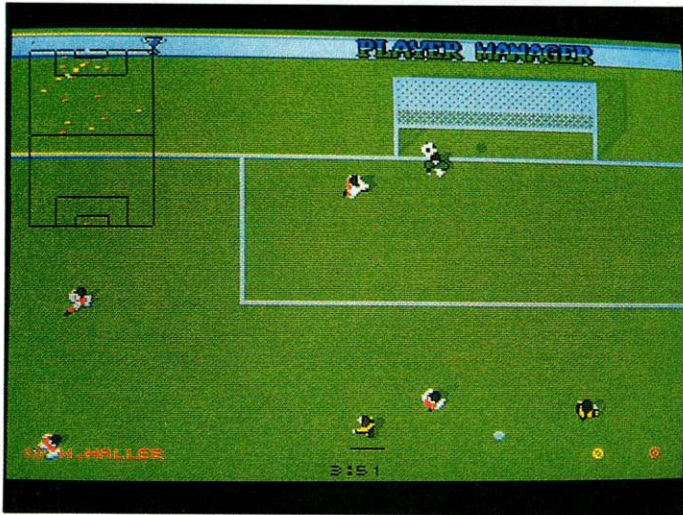
Aus diesen vier Werten bildet sich schließlich das Gesamt-
urteil:

- SUPER
- GUT
- OK
- GEHT SO
- SCHLECHT



Die Bedeutung dieser
Bewertungen erklärt sich
von selbst. Uns bleibt
noch, Ihnen viel Spaß mit
dem AMIGA DOS Spiele-
teil zu wünschen.





Auf dem Rasen entwickeln sich heiße Gefechte



»Kick Off« gehörte schon immer zu den guten Fußballsimulationen, was Spielbarkeit und Realitätsnähe betraf. Wie ist der Nachfolger?

Der Spieler hat extrem viele Handlungsmöglichkeiten; da gibt es fast nichts, was es nicht gibt. Neben der Zusammenarbeit mit Player-Manager, einem Spieler-Manager-Programm, kann man auf vier Hauptstrategien während des Spiels zurückgreifen. Außerdem ist es möglich zu foulern, und da der Ball vor dem Spieler her getrieben wird und ihm nicht am Fuß haftet, hängt es nicht vom Zufall ab, ob man dem Gegner einen Ball abnehmen kann, sondern vom spielerischen Können. Es gibt 24 verschiedene Schiedsrichter. So schaut der eine eher mal bei einem Foul weg, während die typische Handbewegung des anderen der Griff zur roten Karte ist.

Die Vielfalt des Spiels zeigt sich überall. Einen Eckstoß kann man auf neun verschiedene Arten schießen.

Das Verhältnis von Spielfeld zu Spielergröße stimmt, so daß kräftig, aber ruckelfrei gescrollt wird. Das Spielfeld selbst wird aus der Vogelperspektive dargestellt, und da es so groß ist, mußte ein Radarschirm her, der die Positionen der Spieler anzeigt.

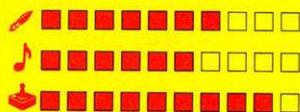
»Kick Off 2« zählt zu den besten Fußballsimulationen, die ich bislang gesehen habe. Man kann fast alles damit machen, ob Spielertransfer oder Auswechseln während des Spiels. Der uns zum Test vorgelegten Version lag noch das Programm »World Cup 90« bei, mit dem es möglich ist, die Weltmeisterschaft in den Originalgruppen daheim zu erleben.

(Robert Marz/mm)

AMIGA DOS Blitz★licht

Name: Kick Off 2
Hersteller: Anco
Quelle: Fachhandel
Preis: 69,95 DM

GUT



Flimbo im Kampf gegen Dandruffs Kreaturen



Wird es unserem Helden Flimbo gelingen, seine geliebte Pearly aus den Klauen des bösen Franz Dandruff zu befreien?

Pearly, die Schönheitskönigin von Dewdrop-land, ist entführt worden. Hinter dieser ruchlosen Tat steckt der grausame Professor Franz Dandruff. Er hat Pearly in seine Gewalt gebracht und auf seinem Gut versteckt. Da tritt unser Held Flimbo auf den Plan.

Dieses Jump'n'Run-Spiel weiß durch gute Grafik, ein softes Scrolling und nette Musikuntermalung zu überzeugen. Flimbos Aufgabe besteht darin, Pearly zu befreien. Dazu muß er sich durch die verschiedensten Levels hindurchkämpfen. Auf diesem Weg warten viele Gefahren in Form von mutierten Kreaturen auf ihn, die er mit seiner Waffe bekämpfen muß. Diese Kreaturen sind jedoch mit Liebe und Humor gezeichnet, daß das Grauen hier keinen Platz findet. Man wird eher dadurch amüsiert. Erschießt man eine Kreatur, bekommt man entweder Bares, Zeit, Leben oder eine Schriftrolle. Sinnvoll ist es, nach verborgenen Räumen wie Dazz Bazians Laden oder Dandruffs Schatzkammern zu suchen. Im ersten erhält man gegen Bares eine Zusatzwaffe, vorübergehende Unangreifbarkeit, eine Schriftrolle, Ex-

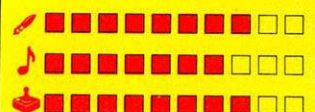
tra-Zeit oder die Superschriftrolle. Ferner bekommt man von Dazz Bazian die nötigen Informationen, um den entsprechenden Level zu komplettieren. »Flimbo's Quest« ist ein Superspiel, das langanhaltenden Spielspaß garantiert. Das Lesen des deutschen Handbuchs bereitet mir große Freude. Man benötigt schon eine gehörige Portion Humor, um die Texte als ernstzunehmende Beiträge aufzufassen: deutsches Sprache, schweres Sprache.

(vb)

AMIGA DOS Blitz★licht

Name: Flimbo's Quest
Hersteller: System 3
Quelle: Fachhandel
Preis: 84,95 DM

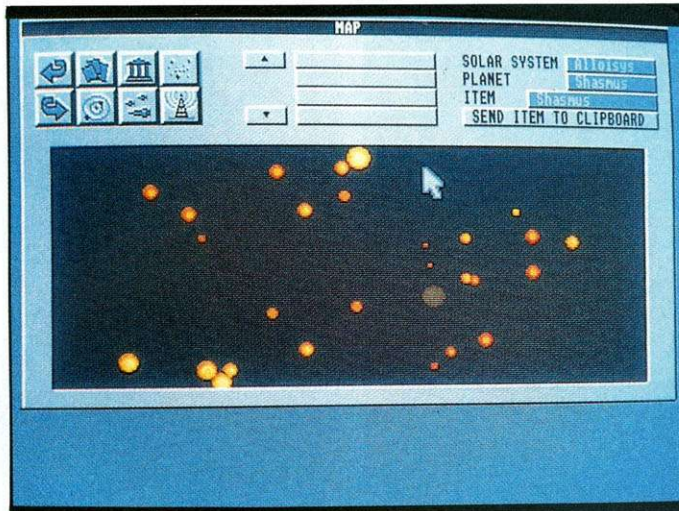
GUT



Imperator zu sein ist keine leichte Aufgabe, das wird man spätestens nach einigen Jahren (im Spiel) merken. Nach Ihrer Wahl im Jahr 2020 sieht es in Ihrem Imperium ziemlich arm aus. Sie müssen es also ausweiten, eine Flotte bauen, Handel mit anderen Imperien treiben, das eigene Volk bei Laune halten, damit es Sie wiederwählt und vieles andere mehr.

Da Sie nicht alles alleine machen können, haben Sie Untergebene, die Sie nach ihren Fähigkeiten und ihrer Loyalität zu Ihnen aussuchen sollten. An diese Untergebenen können Sie Posten verteilen, sie zu Botschaftern in anderen Ländern ernennen, zum Führer von Planeten oder auch zum Kommandanten einer Flotte auswählen. Sie müssen auch bezahlt werden und, das ist wichtig, sie altern wie Sie auch. Um diesen Alterungsprozeß zu stoppen, ist Nostrum nötig, eine Droge, die nicht synthetisiert werden kann und nur auf wenigen Planeten zu finden ist. Je nachdem, wieviel von der Droge pro Jahr zur Verfügung steht, wird der Alterungsprozeß verlangsamt oder gar ganz angehalten. Wenn Sie tausend Jahre regieren wollen, brauchen Sie eine ganze Menge Nostrum, zumal die Loyalität ihrer Untergebenen extrem steigt, wenn Sie sie mit Geld und Nostrum versorgen. Ebenso steigt die Loyalität, wenn Sie Orden vergeben. Aber Ihre Freunde sind auch gleichzeitig Ihre Feinde, denn alle fünfzig Jahre finden Wahlen zum Imperator statt, die Sie selbstredend gewinnen müssen. Ihre Untergebenen lassen sich, je nach Loyalität, die Chance nicht entgehen, selbst Kaiser zu werden. Sie können die Wahlen auch jederzeit vor Ablauf der fünfzig Jahre einberufen, wenn Sie gerade eine besonders große Tat vollbracht haben und der Meinung sind, hoch im Ansehen der Bevölkerung zu stehen.

Das gesamte Spiel wird über Fenster gesteuert. Ihre Handlungen bestimmen Sie dann aus den Optionen und Informationen, die die einzelnen Fenster Ihnen anbieten. Dabei können Sie eine Anzahl von Fenstern öffnen und beliebig auf dem Bildschirm verteilen. Auf diese Weise können Sie sich Ihren eigenen individuel-



Die Galaxis. Unendliche Weiten...



Wer möchte nicht gerne einmal unsterblich sein und ein Imperium für tausend Jahre lenken? Der Schlüssel zu diesem Traum heißt »Imperium«.

len Screen zurechtbasteln, der die Angaben enthält, die Sie persönlich für besonders wichtig halten. Fenster, die Sie gerade nicht brauchen, aber nicht immer wieder von neuem öffnen wollen, können Sie bis auf die Titelleiste schrumpfen lassen.

Außerdem können Sie sich auf einer dreidimensionalen Weltkarte über die Größe Ihres und der anderen Imperien informieren, Daten über einen bestimmten Planeten abrufen oder diesen Planeten auf das

Klemmbrett geben, welches ermöglicht, bequem zwischen Karte und den anderen Fenstern zu kommunizieren.

Ihnen steht eine Flut von Handlungsmöglichkeiten zur Verfügung. Das reicht vom Erheben von Aus- und Einfuhrsteuern über das Bauen und Entwerfen von Raumschiffen bis hin zum Erlassen von Handelssanktionen gegen andere Imperien und Allianzen. Strukturschwache Planeten wollen entwickelt werden, neue Systeme entdeckt und

besiedelt. Dabei bieten nicht alle Planeten gleiche Startbedingungen, sondern diese hängen vom Typ des Planeten, der zugehörigen Sonne und der Umlaufbahn ab. Im Falle eines Falles stehen ebenso vielfältige militärische Maßnahmen zur Wahl. Sie haben gewonnen, wenn Sie entweder tausend Jahre regiert haben oder alle Planeten der anderen Imperien erobert haben, was wesentlich einfacher ist. Es gibt verschiedene Arten von militärischen Begegnungen, von denen eine noch nicht einmal einer bestehenden Allianz abträglich ist: Treffen sich nämlich zwei Flotten zufällig im leeren Raum, können sie aufeinander schießen und sich vernichten, ohne daß dies eine Beziehung zwischen zwei Imperien beeinträchtigen würde.

Imperium bietet eine Fülle von Handlungsmöglichkeiten, die das Herz eines Strategiespielers höher schlagen lassen. Viele neue Elemente wurden in diesem Spiel unter einer sehr benutzerfreundlichen Bedienungsfläche vereinigt. Hat man sich daran erst einmal gewöhnt, und vor allem an die Tatsache, daß jede Aktion erst dann rechtskräftig wird, wenn sie mit einem »OK« quittiert wurde, läßt es sich sehr gut spielen. Wann ein Jahr vorüber ist, bestimmen ganz allein Sie; die Auswirkungen Ihrer Aktionen können Sie dann im nächsten Jahr feststellen. Wem die Lenkung des gesamten Imperiums zu viel ist, der kann Aufgabenbereiche komplett an den Computer übertragen, der seine Aufgabe sehr gut macht. (Robert Marz/mm)



1000 Jahre Herrschaft können ganz schön stressen

AMIGA DOS

Blitz ★ licht

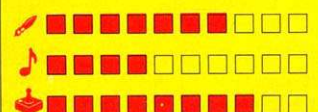
Name: Imperium

Hersteller: Electronic Arts

Quelle: Fachhandel

Preis: 89,- DM

GUT





Erleben Sie eine der blutigsten Epochen Chinas mit

DYNASTY WARS™

Das alte China zur Zeit der Han-Dynasty wird von den Aufständen rebellischer Generäle geschüttelt. Vier tapfere Krieger rüsten zum entscheidenden Kampf...

Allein oder zusammen mit einem Freund sollen Sie die Recken in den Kampf führen. Dazu müssen Sie sich aber zunächst für einen der vier Streiter entscheiden, denn alle unterscheiden sich in ihren geistigen Fähigkeiten, ihrer Schlagkraft und ihren bevorzugten Waffen.

Sollten Sie einen Freund zur Hand haben, können Sie den Bösen auch zu zweit gleichzeitig an den Kragen gehen. Schwingen wir uns also aufs Pferd und reiten in den Kampf. Massen von Rebellen erwarten Sie schon. Das Szenario scrollt langsam von links nach rechts, und Sie können beliebig auf dem Bildschirm herumreiten. Die Stärke der Attacken, die die Reiter mit ihren Waffen ausführen, hängt von der Kraft des Kämpfers ab und davon, wie lange Sie vor dem Schlag den Knopf gedrückt hatten.

In gefährlichen Situationen stehen zusätzliche Taktiken zur Verfügung, die allerdings an den Kräften des Streiter zehren. Am Ende eines jeden Levels wartet dann ein ebenfalls reitender General, der erst besiegt sein will, bevor es in den nächsten Level geht.

Das Spiel bietet nur sehr wenig Abwechslung. Die Gegner sind sich sehr ähnlich und zumindest die ersten vier Generäle gleichen sich aufs Haar. Selbst die Zwei-Spieler-Option verschafft nur wenig Pluspunkte, zumal die Steuerung etwas seltsam ist und man nur sehr langsam gegen die Scroll-Richtung reiten kann. Alles in allem ein müdes Gefecht gegen die Langeweile und Krämpfe im Feuerdaumen.

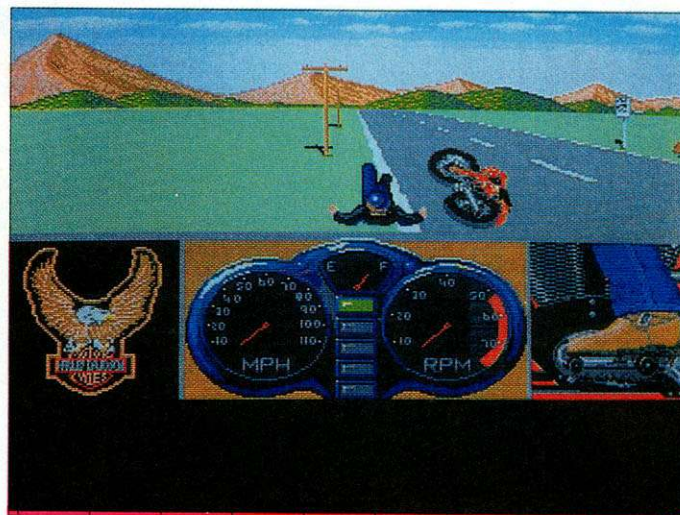
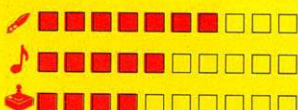
(Robert Marz/mm)

AMIGA DOS

Blitz ★ licht

Name: Dynasty Wars
Hersteller: U.S. Gold
Quelle: Fachhandel
Preis: zirka 85 DM

OK



Absteigen leichtgemacht...

Harley Davidson

Die Fransenlederjacke flattert im Wind und an jedem Baum der Allee eine Zündung. Was ist das? Eine Harley Davidson.

Die etwas "betagteren" unserer Leser werden sich bestimmt an einen Kultfilm erinnern, der einen wahren Kaufrausch von großvolumigen Zweirädern nach sich zog: Easy Rider. Eben dieser Film stand offensichtlich Pate bei der neusten Kreation von Mindscape. In diesem Spiel übernehmen Sie die Rolle eines Harley-Fahrers, der sich durch diverse Spiele bis zum "Biker-Gott" hocharbeiten soll.

Zu Spielbeginn werden Sie nach Ihrem Namen und der Anzahl ihrer Haare gefragt (das ist kein Witz!). Haben Sie sich entschieden, ob Sie lieber buschig behaart oder kahlköpfig durchs Biker-Leben fahren wollen, gelangen Sie an eine Tankstelle. Dort können Sie alles erwerben, was das Harley-Herz höher schlagen läßt: von der neuen Lederjacke bis zum Austauschmotor. Das heißt, Sie könnten es erwerben, wenn genügend Geld vorhanden wäre. Hier hilft ein schneller Griff in die Ladenkasse, wobei man sich aber vor schießwütigen Ladenbesitzern in acht nehmen sollte. Also "on the road" und nichts wie weg. Die Fahrsimulation zieht sich etwas in die Länge. Neben dem ruckelnden Scrolling des Ge-

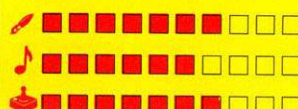
ländes kommt es häufig zu "Zusammenstößen" mit Wasserpfützen. Der Autor dieses Artikels ist noch nie mit einer Pfütze kollidiert; höchstens zu Ungunsten der Pfütze. Im Anschluß an die Fahrsequenz können Sie sich erkundigen, ob es irgendwo ein Treffen gibt. Hier können Punkte und Ansehen mit diversen Geschicklichkeitsfahrten gesammelt werden. Und wenn alles gut ging, sind Sie irgendwann der Ober-Biker. (tb)

AMIGA DOS

Blitz ★ licht

Name: Harley Davidson
Hersteller: Mindscape
Quelle: Fachhandel
Preis: 99,- DM

GUT



AMIGA DOS

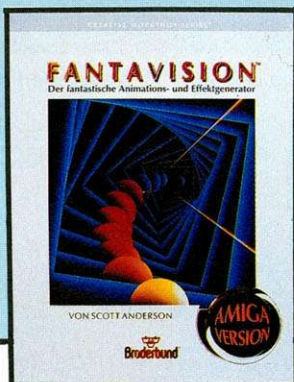
SHOP

mit phantastischen Preisen

FANTAVISION

Lassen Sie Ihrer Phantasie freien Lauf mit diesem hervorragenden Animations- und Effektgenerator und Ihr Computer wird in wenigen Augenblicken zum Filmstudio.

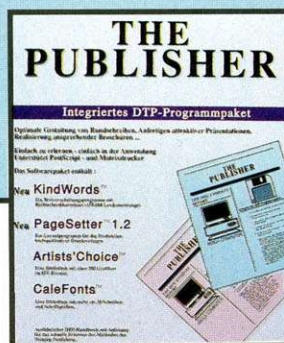
nur DM 49,95*



The Publisher AMIGA

Umfassendes Desktop-Publishing und Desktop-Presentation im Paket: Ein Textverarbeitungsprogramm, Layoutprogramm, eine Bibliothek mit mehr als 200 IFF-Grafiken und mehr als 35 Schriften.

nur DM 199,-*



AMIGA Spielebuch

Das umfassende deutsche Nachschlagewerk mit Tips & Tricks zu den beliebtesten AMIGA-Spielen. Dieses Buch zeigt Ihnen, was in Ihrem AMIGA steckt.

nur DM 29,-*



**AMIGA
Köln 90**
Wir stellen aus
09.-11. November 1990
Halle 12, Stand 307

Speichererweiterung AMIGA 500

Mit 512 KByte zusätzlichem Speicher öffnen sich Ihrem Amiga ganz neue Welten! Mit externem Ein/Aus-Schalter, integrierter Kalenderuhr und Batterie.

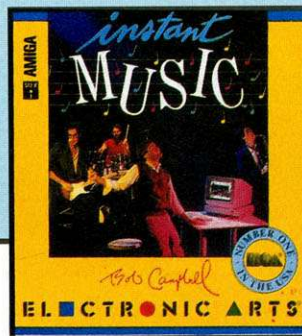
nur DM 149,-*



INSTANT MUSIC AMIGA

Musik für heiße Sohlen live aus dem Amiga! Ihre Musikideen erklingen absolut fehlerfrei, da sie automatisch an Takt und Tonart angepaßt werden.

nur DM 19,95*



* Unabhängig von der Anzahl der bestellten Produkte berechnen wir für das Inland 4.- DM bzw. für das Ausland 6.- DM Porto und Verpackung.

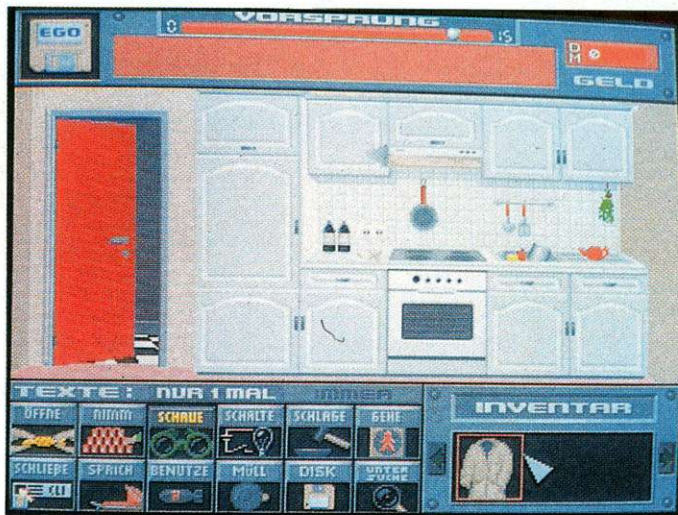
Bitte benutzen Sie die Bestellkarte.

DMV-Verlag . Postfach 250 . 3440 Eschwege

DMV
Daten- und
Medienverlag



Olympic Crater Games können Sie in Atem halten



Können Sie die Plantage gegen das Kartell halten?

Moon Blaster

Hintergrund dieses actiongeladenen Ballerspieles sind die Olympic Crater Games, die Sie als Repräsentant der Royal Air Force besuchen. Ihre Aufgabe ist, den letzten Jahr verlorenen Titel wieder zu gewinnen. Gute 3D-Vektorgrafik und eine sensible Steuerung lassen einen gleich richtig einsteigen.

Fatal Heritage

Eines schönen Tages erreicht Sie die Nachricht eines Notars. Er teilt Ihnen mit, daß Ihr Onkel gestorben ist und Ihnen eine reichhaltige Erbschaft hinterließ. Nur leider neidet Ihnen jemand die geerbte Plantage, und fortan sitzen Ihnen die gedungenen Schergen eines kriminellen Kartells im Nacken. Den Spieler erwartet ein Grafik-Adventure mit Icon-Steue-

Demnächst auf Ihrem Computer

Ballern, Adventure und Jump'n'Run bilden die Zutaten zu einem Cocktail, der im Herbst sicher helfen wird, graue Regentage mit viel Kurzweil zu verbringen.

rung, das komplett in Deutsch gehalten wurde. Kein Wunder, das Produkt stammt aus einer alemannischen Software-Schmiede.

Über 150 Bilder des Adventures gilt es zu erforschen; jedoch wird die Aufgabe durch die verschiedenen Mitstreiter, die zur Verfügung stehen, ein wenig erleichtert. Action-Sequenzen lockern das Adven-

ture auf und geben Euch Joystickartisten eine Chance.

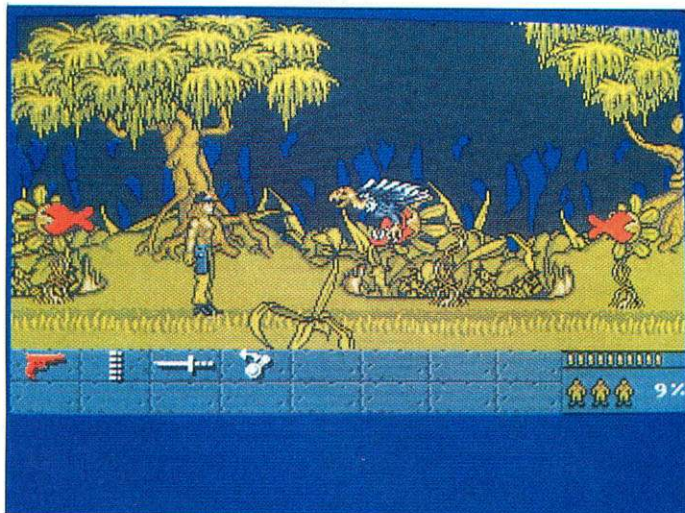
Gold of the Aztecs

Kinetika Software hat sich in Zusammenarbeit mit U.S.Gold an das auf dem Amiga nicht einfache Thema Jump'n'Run herangewagt. »Gold of the Az-

tecs« überrascht mit einer Vielzahl von zum Teil sehr humorvoll gestalteten Monstern und flüssigen Animationen, die dem bisher recht brachliegenden Genre des Jump'n'Run neues Leben einhauchen könnten.

Operation Stealth

Nach dem Erfolg von »Future Wars« wartet nun Delphine Software mit einem neuen epischen Werk auf. »Operation Stealth« versetzt Sie in die Person des Geheimagenten John Glames, der den riskanten Auftrag übernommen hat, den verschwundenen Prototyp des Stealth Fighters wieder zu beschaffen. Sowohl Story als auch Grafik und Musik versprechen ein äußerst spannendes Abenteuer. Die Bildschirmtexte sind zudem auch als Sprachausgabe verfügbar. (mm)



Der Weg zum Gold der Azteken: hart, aber herzlich



Als Geheimagent kennt man keine Probleme...

Impressum

Herausgeber

Christian Widuch

Redaktionsleitung

Stefan Ritter

Chefredaktion

Markus Matejka (mm)

Leitender Redakteur

Jürgen Borngießer (jb)

Redaktion

Heinrich Stiller (hs), Claus Daschner (cd),

Vera Brinkmann (vb), Thomas Baum (tb)

Freie Autoren dieser Ausgabe

Robert Marz, Michael Anton, Andreas Polk,
Franz-Josef Reichert, Antje Hink, Edgar Meyzies,
Andreas Vorgeitz, Robert Münch, Oliver Heggelbacher,
Stefan Heller, Matthias Kopp

Redaktionsassistenten

Anke Kerstan (ke), Susanne Eska (es)

Chef vom Dienst

Matthias Bloß

Produktionsleitung

Gerd Köberich

Bereichsleitung

Claudia Ebbrecht (Fotosatz/Lektorat),

Margarete Schenk, Helmut Skoupy

(Montage/Reprografie)

Layout

Michael Grebenstein

Fotografie

Heike Reifer

Fotosatz

Marcus Geppert

Lektorat

Heike Wiegand

Montage/Reprografie

Manuela Eska, Peter Gajewski

Werbegestaltung

Mohamed Hawa

Anzeigenleitung

Wolfgang Brill

Anzeigenverkauf

DMV-Verlagsbüro München

Zaunkönigweg 2c, 8000 München 82

Telefon (0 89) 4 39 10 87, Telefax 0 89/4 39 10 80

Leitung: Britta Fiebig

Anzeigenverkauf: Monika Schöbel, Peter Schätzle,

Hannelore Schulzki, Michael Hofmann

Anzeigenverwaltung und Disposition

Andrea Giese, Karina Ehrlich, Beate Kranz,

Christina Füllgraf

Anzeigenpreise

Es gilt die Anzeigenpreisliste Nr. 1 vom 01.05.1990

Anschrift Verlag/Redaktion:

DMV Daten & Medien-Verlag

Widuch GmbH & Co. KG

Fuldaer Straße 6

3440 Eschwege, Telefon (0 56 51) 8 09-0,

Telefax (0 56 51) 8 09-333

Vertrieb

Verlagsunion Erich Pabel-Arthur Moewig KG (VPM),

Friedrich-Bergius-Straße 20, 6200 Wiesbaden

Druck

Druckerei Jungfer, 3420 Herzberg

Bezugspreise

»AMIGA DOS« erscheint monatlich.

Einzelpreis DM 6,50/str. 6,50/6S 52,-

Abonnementpreise

Die Preise verstehen sich grundsätzlich einschließlich

Porto und Verpackung.

Inland:

12 Ausgaben: DM 70,-

6 Ausgaben: DM 35,-

Europäisches Ausland:

12 Ausgaben: DM 100,-

6 Ausgaben: DM 50,-

Außereuropäisches Ausland:

12 Ausgaben: DM 120,-

6 Ausgaben: DM 60,-

Bankverbindungen:

Postscheck Frankfurt/M: Kto.-Nr.: 23043-608

Raiffeisenbank Eschwege:

BLZ: 522 603 85, Kto.-Nr.: 245 7008

Die Abonnementbestellung kann innerhalb einer

Woche nach Auftrag beim DMV-Verlag, Postfach 250,

3440 Eschwege, schriftlich widerrufen werden. Zur

Wahrung der Frist reicht der Poststempel. Das Abonne-

ment verlängert sich automatisch um 6 bzw. 12 Monate,

wenn es nicht mindestens 6 Wochen vor Ablauf beim

Verlag schriftlich gekündigt wird.

Für unverlangt eingesandte Manuskripte und Datenträ-

ger sowie Fotos übernimmt der Verlag keine Haftung.

Die Zustimmung zum Abdruck wird vorausgesetzt.

Eine Haftung für die Richtigkeit der Veröffentlichungen

kann trotz sorgfältiger Prüfung durch die Redaktion vom

Herausgeber nicht übernommen werden. Die geltenden

gesetzlichen und postalischen Bestimmungen sind zu

beachten.

Die gewerbliche Nutzung, insbesondere der Programme,

Schaltpläne und gedruckten Schaltungen, ist nur mit

schriftlicher Genehmigung des Herausgebers zulässig.

Das Urheberrecht für veröffentlichte Manuskripte liegt

ausschließlich beim Verlag. Nachdruck sowie Vervielfäl-

tigung oder sonstige Verwertung von Texten nur mit

schriftlicher Genehmigung des Verlages.

Namentlich gekennzeichnete Fremdbeiträge geben nicht

in jedem Fall die Meinung der Redaktion wieder.

SPIELEN SIE GERNE?

Lieben Sie es, in finsternen
und unheimlichen Ge-
wölbten den letzten Geheimnis-
sen auf den Grund zu gehen?
Zeichnen Sie gerne Karten von
Gewölbten und Spiellandschaf-
ten?

**Dann sind Sie der richtige Part-
ner für uns.**

Denn für unsere AMIGA-DOS-
Spieletips suchen wir ständig
Tips, Tricks, Karten, Cheat-
modes und alles andere, was das
Spielerherz höher schlagen läßt.
Wenn Sie interessante Informa-
tionen, Karten oder Lösungsvor-
schläge für Computerspiele her-
ausgefunden haben, sind Sie un-
ser/e Mann/Frau.

Durch Ihre Information können
Sie vielen Spielern weiterhelfen,
die ohne Ihre Hilfe unter Umstän-
den monatelang an der gleichen
Problematik "festhängen". Oft
sind es nur ein Wort oder ein klei-
ner Hinweis, die dem Spielge-
schehen neues Leben einhau-
chen.

**Aus diesen Gründen möchten
wir Sie um Ihre Mitarbeit bitten.
Wenn Sie also über Spieletips
– gleich welcher Art – verfü-
gen, würden wir uns freuen,
von Ihnen Post zu bekommen.**

Und hier nun die Adresse, an die
Sie Ihre Spieletips schicken kön-
nen:

**DMV-Verlag
Redaktion AMIGA DOS
Stichwort: Spieletips
Postfach 250
3440 Eschwege**

PS: Unter allen Einsendungen
verlosen wir jeden Monat fünf
Software-Titel für die besten
Spieletips.

Die Inserenten

A + L.....	71
Alsdorfer PD Center.....	84
Ami Shows Europe.....	11,75
A.P.S. Elektronik.....	85
Auriga.....	85
B + S Computer.....	84
Bonito.....	93
CIK-Computertechnik.....	85
CLS Computerladen Schäfer....	83
Compedo.....	28
Computing.....	21
Compy Shop.....	85
CSV Riegert.....	105
DELTA SOFT.....	83
DMV.....	18,30,31,109,127,131
Dombrowski.....	83
Donau-Soft.....	85,99
EUROSYSTEMS.....	9
FSE Strauß-Elektronik.....	47
German-Soft.....	84
GNE.....	84
HAMO.....	84
HERMES Software.....	45
Herrmanns & Kommelter.....	84
HK-Computer.....	2
IC-InterComputing.....	13
Jochheim Computertuning.....	49
Kirschbaum.....	77
Kunze.....	85
Lukowiak Audio-Video.....	83
Mac-Soft.....	83
MVC.....	77
Neuroth Hardware-Design.....	84
Nürnberger PD-Studio.....	99
Omega Datentechnik.....	83
Optivision.....	83
Pawlowski.....	84
Pielago Software.....	83
PUBLIC DOMAIN CENTER.....	84
P.V. Computershop.....	85
ROSSMÖLLER.....	15,39
Soft Art + Hardware Tute.....	93
Stengel Computershop.....	83
G. v. Thienen.....	71
United Software.....	119,132
Vesalia Computer.....	59
WENNGATZ.....	39
Wolf Computertechnik.....	53

Im nächsten Heft

Sprachen – quo vadis?

Englisch oder Französisch gehören bei vielen bereits zum Sprachschatz wie das tägliche Brot. Doch wie steht's mit den Programmiersprachen? Wir liefern Informationen. Welche Sprache eignet sich für welchen Zweck am besten?

Insider – bekannt aus der PD-Szene und von vielen Software-Entwicklungen – halten Plädoyers für ihre Lieblingssprache.



Fruitmania: Gambling wie in Las Vegas

Amiga'90 in Köln

Wir haben extra für Sie weder Kosten noch Mühen gescheut, um Ihnen das Neueste vom Neuen von der weltgrößten Amiga-Messe in einem großen, topaktuellen Vorbericht zu präsentieren.

Preiswert und gut

Unser PD-Workshop zum Sammeln gibt Tips zum Umgang mit drei sehr unterschiedlichen Programmhits.

Wettbewerbe ohne Ende...

Ziehen gerade Sie das große Los bei einem unserer vielen Wettbewerbe? Wir halten Preise für Sie parat, daß einem als Redakteur(in) schon manchmal das leichte Zucken in die Glieder fährt. Preise im Gesamtwert eines Wagens der gehobenen Mittelklasse warten auf Sie in der AMIGA DOS 11'90!

Backups ohne Frust

Wer tagtäglich große Datenmengen verarbeiten muß, benötigt entsprechende Speichermedien. Streamer heißt hier das Zauberwort. Wir liefern Ihnen einen Test eines bekannten Produktes.

Was gibt's noch?

Viele Testberichte zu den aktuellsten Hard- und Software-Produkten: Digitizer, Grafik- und Animationsprogramme, Compiler, Transputerkarte und und und ... Dazu viele knackige, kleine Tips&Tricks zu den verschiedensten Programmiersprachen und brandheiße Spieletests.

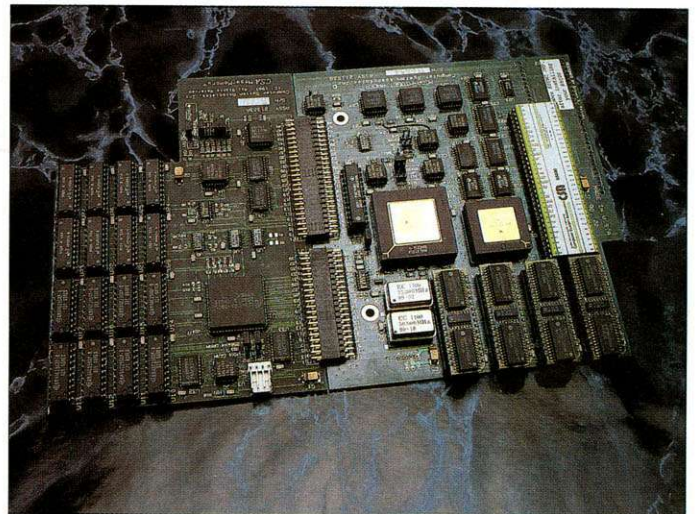
Die nächste

AMIGA DOS

finden Sie ab

10. Oktober '90

bei Ihrem Zeitschriftenhändler



Mega Midget, eine superschnelle 68030-Karte für den A500 und A2000



Was lange währt, wird endlich gut – »BECKERtext II«

AMIGA DOS

Spielboxe

mit traumhaften Preisen!

ZAK McKracken



Der neue urkomische Thriller von Lucasfilm: Können ein Sensationsreporter, Zwei Studentinnen und ein altes Brot die Erde vor Außerirdischen retten? Nicht ohne Ihre Hilfe!

nur DM 69,95*



STARFLIGHT

Erkunden Sie 270 Sternsysteme mit 800 Planeten und sieben außerirdischen Rassen. Einschließlich Ausbildung von Offizieren und Trickfilmgrafik mit Planetenrotation.

nur DM 49,95*

MANIAC MANSION

Lucasfilm präsentiert eine total verrückte Story wie im Kino. Machen Sie sich für die gefährlichste, spannendste und albernste Rettungsaktion Ihres Lebens bereit.

nur DM 69,95*

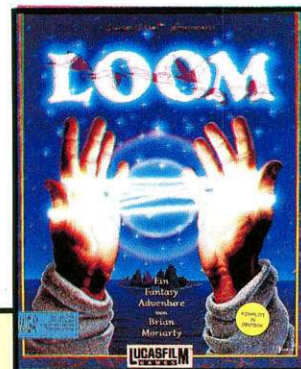


OIL IMPERIUM

Business-Spiel um Geld, Macht und fiese Tricks. Beweisen Sie Ihr Gespür bei den Aktion-Sequenzen, in denen Sie Öl suchen, Pipelines bauen und Feuer löschen.

nur DM 39,95*

**AMIGA
Köln 90**
Wir stellen aus
09. - 11. November 1990
Halle 12, Stand 307



LOOM

Das bahnbrechende Fantasy-Spektakel von Lucasfilm mit märchenhaften Landschaften in grafischer Perfektion. Komplett in Deutsch mit deutschem Hörspiel.

nur DM 69,95*

BARD'S TALE II

Das bekannte Fantasy-Abenteuer jetzt 50% größer mit sechs Städten und 25 scrollenden Labyrinthen. Genau das richtige für ausgefuchste Abenteurer.

DM 39,95*



* Unabhängig von der Anzahl der bestellten Programme berechnen wir für das Inland 4.- DM bzw. für das Ausland 6.- DM Porto und Verpackung.
Bitte benutzen Sie die Bestellkarte.

F/A 18 Interceptor

Fliegen Sie das faszinierendste Flugzeug unserer Zeit in sieben Missionen mit unterschiedlichen Schwierigkeitsgraden in realitätsnaher 3D-Grafik.

nur DM 39,95*



VON DEN
PROGRAMMIERERN VON...

FUTURE WARS

TRÄGER DES "IN DIN" - PREISES FÜR DAS BESTE ÜBERSEE-SPIEL

EIN WEITERES

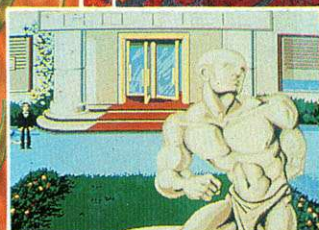
Cinématique™

SPITZEN PRODUKT...

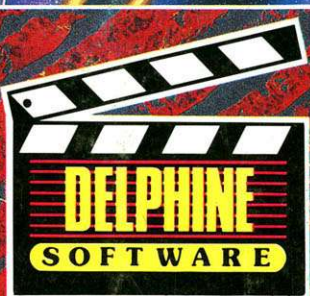
OPERATION STEALTH



Bilder der
ST Version



Bilder der Amiga-
Version



Operation Stealth stürzt Dich in die Welt internationaler Spionage, ein brandneues, interaktives Cinématique™ Adventure. Operation Stealth ist das zweite Spiel, das das einzigartige Cinématique™ Betriebssystem, seit Future Wars weiterentwickelt und verbessert, nutzt.

- Vollständig mouse-gesteuert
- Einfaches Point-and-Click-Interface
- Kompatibel mit AdLib- und Roland-Sound-Card (MS-DOS-Version)
- 256 Farben in der VGA-Version unter MS-DOS.
- Super-Soundtrack aus den Delphine-Studios

DEMNÄCHST...



U.S. Gold Ltd., Units 2/3, Holford Way,
Holford, Birmingham B6 7AX.
Telephone: 021-625-3388.

GRUISE
FOR A CORPSE

© 1990 DELPHINE SOFTWARE.
Alle Rechte vorbehalten. Cinématique
ist eingetragenes Warenzeichen der
Delphine Software.

ERHÄLTICH FÜR
ATARI ST • AMIGA
MS/PC-DOS KOMPATIBEL