

RUCHHITS

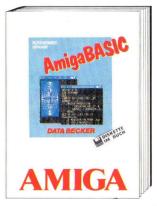


Wählen Sie gleich den richtigen Einstieg zu Ihrem Amiga 500. Denn das Hand-buch läßt Sie dabei völlig allein. Versuchen Sie es lieber gleich mit Amiga 500 für Einsteiger. Hier heißt es: anschließen und loslegen. Verständlich für jedermann zeigt Ihnen dieses Buch: Workbench, AmigaBASIC, CLI und AmigaDOS. Locker aufbereitet bietet es Ihnen alles Wissenswerte. Bis hin zu den beim Amiga 500 mitgelieferten

Zusatzprogrammen. Amiga 500 für Einsteiger 343 Seiten, DM 39,-

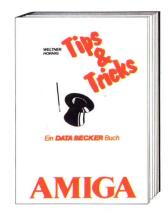


Läßt das Handbuch Sie auch in so manchen Dingen allein, das große Amiga-2000-Buch nicht. Hier finden Sie eine umfassende Einführung in die Arbeit mit Ihrem neuen Rechner - und mehr als das. Sind Sie erst einmal mit dem Amiga 2000 "per Du", zeigen Ihnen die Autoren, was einen Amiga-Profi ausmacht: Kickstart im RAM, PC-Audioausgänge, erste Hilfe bei Harddisk-Abstürzen, Laufwerkeinbau in den Amiga 2000 und, und, und. Sollten Sie also noch Fragen zu Ihrem Rechner haben, hier finden Sie die Antworten. Das große Amiga-2000-Buch Hardcover, 684 Seiten, DM 59,-



Das erfolgreiche Buch zu Amiga-BASIC. Erweitert um Kickstart 1.2. neuer Workbench und Amiga 500 & 2000. Alles, was BASIC-Programmierern Spaß macht: Grafik und Sound, Laden und Speichern von Graficraft-Bildern in BASIC-Programme, sequentielle und relative Dateien, Business-Grafik, Computeranimation, Windows, Umgang mit IFF-Bildern, Sprachausgabe und, und, und. Das Buch für Einsteiger, Aufsteiger und Profis

AmigaBASIC Hardcover, 774 Seiten, DM 59,-



Amiga Tips & Tricks. Ein Buch, das voller Überraschungen steckt: Gestaltung eigener Programme, Tips & Tricks zum AmigaBASIC, Einbinden von Maschinenprogrammen in AmigaBASIC, Einsatz von DOS-Routinen, optimierende Hilfsprogramme für AmigaBASIC-Programme, Tips zur Arbeit mit der Workbench, Aufbau der Icons. Mit vielen Anregungen, aber auch fertigen Lösungen. Ein Griff in die Trickkiste, und ungeahnte Möglichkeiten tun sich auf. **Amiga Tips & Tricks** Hardcover, 473 Seiten, DM 49,-



Hier erfahren Sie alles, was Sie bei Ihrer praktischen Arbeit mit dem AmigaDOS wissen sollten: Umlenken der Ein- und Ausgabe, sinnvoller Einsatz des Jokers, Arbeiten mit RAM-Disk und CLI, Batch-Dateien, STARTUP-Sequenz, Multitasking mit dem CLI, Aufbau der CLI-Befehle, Programmierung eigener CLI-Befehle, neue CLI-Befehle in BASIC und C. Dazu ein ausführlicher, gut strukturierter Nachschlageteil. Wer also mit dem AmigaDOS arbeitet, sollte dieses

Buch in greifbarer Nähe haben.

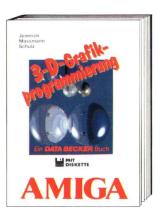
Das große Buch zu AmigaDOS

Hardcover, 310 Seiten, DM 49,—



Alles zum Amiga: In einem Band und absolut auf dem neuesten Stand: 68000-Prozessor, CIA, Blitter, Customchips, die wichtigsten Strukturen von EXE, I/O-Handhabung, Verwaltung der Resources, Multitasking, EXEC-Base, resetfeste Programme, DOS-Funktionen, IFF-Format, Programmstart von CLI und Workbench, Programmierung der EXEC- und DOS-Routinen und, und, und. Eben ein typischer Intern-Band, in dem wieder einmal nichts Wissens wertes fehlt.

Amiga Intern Hardcover, 639 Seiten, DM 69,-



3-D-Grafikprogrammierung - hier finden Sie Grafikalgorithmen für absolut realistisch gestaltete Bilder. Die einzelnen Vorlagen lassen sich dabei mit einem Editor problemlos eingeben und solange durch die Wahl verschiedener Materialien, Farben, Lichtquellen und Spiegelungen verfeinern, bis Sie eine absolut naturgetreue Darstellung erreicht haben.

Amiga 3-D-Grafikprogrammierung Hardcover, 283 Seiten inkl. Diskette, DM 59,-

Regelmäßig in der DATA WELT: Amiga Window – das Forum für den engagierten Amiga-Anwender. Mit kreativen Projekten, Interviews, Software Tests und wichtigen News. Und: Auch was sich sonst so in der Computerszene tut, erfährt der Amiga-Anwender. Die DATA WELT - das aktuelle Computermagazin. Monat für Monat überall da, wo es Zeitschriften gibt.

Lating the fire the season of the season of

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 31 0010

Einesenden an: Data Bet Oktive senden see Iniv.

CEBIT 88-EIN JAHR DANACH

Auch diesmal wurde die CEBIT ihrer Bedeutung als größte Computermesse der Welt gerecht. Viel Neues an Softund Hardware gab es an den Ständen zu sehen und entspechend groß war auch der Andrang des Publikums.

Auch unser Stand wurde rege besucht und das obwohl er etwas abseits von den Haupthallen lag. Vorgeführt wurden das neue Hardware-Projekt 'Juniorprommer'.

Was es sonst noch zu sehen und erfahren gab können Sie in unserem ausführlichen Messebricht nachlesen.

Für uns sind mit der CeBIT 10 Tage Streß vorbei - 8 Tage Messe und je ein Tag für An- und Abreise, davon wurde das gesamte Redaktionsteam ausgelaugt.



In dieser Zeit wurden einige Kilometer abgelaufen, denn schließlich wollte man alles sehen was an Neuem geboten wurde. Besonders für uns, die wir sonst vorwiegend hinter dem Schreibtisch sitzen, war das eine enorme Leistung, die sich zeitweise in einer gewissen Müdigkeit niederschlug. Doch alles ist irgendwann vorbei, und so auch diese Messe. Danach blieb aber keine Zeit zum Ausruhen, denn das Heft, das sie momentan in Ihren Händen halten, mußte fertiggestellt werden. Doch genug davon, denn schließlich gibt es auch noch erfreulichere Sachen. So haben wir z.B. einen Grund zum Feiern. Nicht nur, daß die Messe überstanden ist - es gibt noch ein viel wichtiges Ereignis: Vor genau einem Jahr kam der Gedanke auf, eine Zeitschrift für den AMIGA zu gestalten: Die KICKSTART wurde geboren.

COMMODORE hatte auf der CeBIT 87 kräftig die Werbetrommeln gerührt, und die Startschwierigkeiten des AMI-GA schienen überwunden zu sein. Zwar war der Softwaremarkt noch etwas mager, doch die Zeit schien geeignet, eine Fachzeitschrift ins Leben zu rufen. Schon nach gut einem Monat war die Premierenausgabe fertig, und die KICKSTART erschien als erste deutsche monatliche AMIGA-Zeitschrift. Doch genug davon, denn dieses Ereignis wird erst in der nächsten Ausgabe gefeiert.

Ihr

Markus Nerding

JUNIORPROMMER

Für Engagierte mit kleinem Geldbeutel

GRUNDLAGEN

Mit BASIC ans System Teil 3: Wichtige Strukturen

Seite 29

Mit dem Rechner auf DU und DU Assembler-Kurs Teil 7

Seite 40

Vom Trackdiskdevice zur singenden Floppy

Seite 77

SOFTWARE

Dynamic Studio

Das digitale 16-Spur-Heimstudio

Seite 36

The Graphic Studio

Seite 43

Source Level Debugger

Die Fortsetzung der Aztec-DB Legende?

Seite 52

Videoscape 3D PAL

3D-Animation neu geformt

Seite 48

TV-Show

Werbedemos selbstgebaut

Seite 45

HARDWARE

Der KICKSTARTER

Dem AMIGA Beine gemacht

Seite 62

Neue CPU für AMIGA

Seite 71

MIDI-Gold

Die Brücke zum computergesteuerten Studio

Seite 65

Electronic-Design

Genlock-Faszination Video und AMIGA

Seite 67

Transfile

PC an der langen Leine

Seite 42

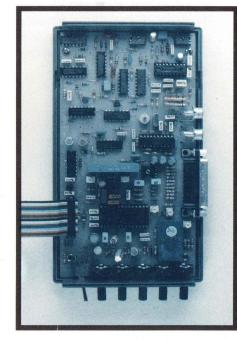
Der JUNIORPROMMER

- Ein Eprombrenner als Selbstbauprojekt

Nachdem sich viele Leser mehr Hardware-Schaltungen in der KICKSTART gewünscht haben, bringen wir in dieser Ausgabe ein Projekt. Es handelt sich dabei um einen Eprom-Brenner, den wir feierlich 'JUNIOR PROMMER' getauft haben, weil er klein, handlich und kostengünstig und somit auch für den kleinen Geldbeutel gedacht ist. Er ist in der Lage, alle Eproms der 27XXX-Serie zu brennen. In diesem ersten Teil stellen wir die komplette Hardware mit Schaltbild, Bestückungsplan und Stückliste vor. In der nächsten Ausgabe folgt dann die nötige Software.

Der AMIGA Speeder!

Wir haben dem AMIGA ein wenig auf Trab gebracht, denn mit geringem Hardwareaufwand ist es möglich, alle AMIGA-Modelle schneller zu takten. Es versteht sich von selbst, daß die Programme ihre Arbeit schneller verrichten. Einige Probleme mußten zwar überwunden werden, aber schließlich und endlich haben wir sie überwunden. Wie die ganze Sache funktioniert, erfahren Sie auf Seite



MS-DOS

PC-Intim

Teil 3: Sitzordnung

Seite 74

Turbo-Pascal 40

Neues aus Borland

Seite 87

Kleine Basteleien am Bridgeboard

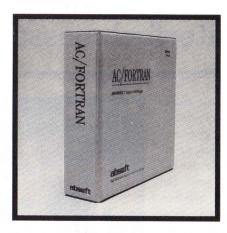
Seite 85

Low Cost-Genlock

Vor kurzer Zeit erhielt die Redaktion ein Genlock, das für nur DM 500.- im Handel erhältlich ist. In einem ausführlichen Test mußte es zeigen, ob es mit seinen weit teureren Konkurrenten mithalten kann. Ob dies dem Gerät gelungen ist, erfahren Sie auf

Fortran

Ein Fortran-Compiler für den AMIGA wurde auf Herz und Nieren getestet. Ist diese Sprache auch auf dem AMIGA einzusetzen, oder gibt es Komplikationen? Die naturwissenschaftliche Hochsprache auf dem AMIGA. Ab Seite 56



KICKS für Insider

Diesmal haben wir einige hochinteressante KICKS für Sie ausgewählt. Ein komfortabler 'DIR'-Befehl, ein einfacher Bildlader oder eine Cursorpositionierungsroutine für C-Programmierer sind nur einige KICKS. Mehr ab Seite 99

Spiele

Die neuesten Spiele standen wieder einmal der KICKSTART-Redaktion zur Verfügung: Jet - der mit vielen Vorschußlorbeeren bedachte Flugsimulator. Pink Panther - der Zeichentrickheld auf dem AMIGA. Powerstix - der alte Spielhallen-Klassiker. Und noch viele mehr. Natürlich erfahren Sie wieder interessante NEWS aus der Spielescene. Mehr ab Seite 115



KICKS FÜR INSIDER

Bist Du aber groß geworden DIR-Befehl mit Längenangabe

Seite 100

Cursorpositionierung

Komfortables für C-Programmierer

Seite 104

Aufgerissen

Großes CLI-Fenster

Seite 113

Gut gepackt

IFF-Format, na und ?

Seite 108

SPIELE

Beyond Zork

Der 'vierte Teil' der Trilogie

Seite 119

JET

Flugsimulator

Seite 116

Sidewinder

Ein Luxusballerspiel

Seite 115

Spinworld

Action ohne Satisfaction

Seite 120

Pink Panther

Macht AMIGA unsicher

Seite 121

Powerstyx

Die Quix-Variante

Seite 120

Strip Poker II Plus

Knisternde Erotik beim Pokern

Seite 118

SPIELE-NEWS

Seite 122

TOP 12

unsere monatliche Hitliste

Seite 124

RUBRIKEN

Editorial		
	Seite	3
NEWS		
	Seite	16
CeBIT '88		
Weltgrößte Computerfachmesse		
	Seite	6
HARDWARE-WETTBEWERB		
	Seite	106
TIPS & TRICKS		

Seite 92

Leserforum

Seite 94

PUBLIC DOMAIN SERVICE

Seite 128

Einkaufsführer

Seite 125

Inserentenverzeichnis

Seite 105

Vorschau

Seite 130

Impressum

Seite 130

CeBIT 88

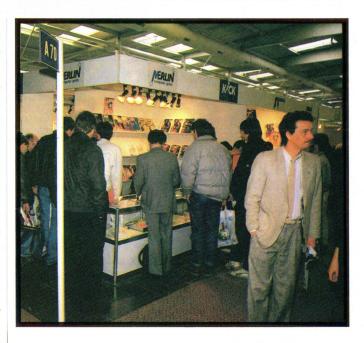
Weltgrößte Computerfachmesse in Hannover

Von 16.-23. März 1988 fand auf dem Messegelände in Hannover die weltgrößte Messe für Informations- und Kommunikationstechnik statt, die CeBIT'88 (Centrum Büro, Information, Telekomunikation).

Hubert-H. Lange, Messe-Vorstandsmitglied, beschreibt die CeBIT'88 mit einem einzigen Satz: "Die höchsten Erwartungen wurden weit übertroffen". Diese Stellungnahme läßt wohl keine Fragen mehr offen. Mit 480000 Besuchern übertraf die CeBIT'88 die Vorjahresmesse um ein Fünftel; damals zählte die Messegesellschaft knapp über 400000 Besucher. Erfreulich war auch die hohe Zahl der ausländischen Besucher; mit 97000 lag diese Zahl weit höher als im Jahr davor, als man 69000 Besucher aus dem Ausland zählte. 17500 kamen allein aus Übersee.

Die 2730 Aussteller, darunter 980 Ausländische aus 35 Ländern, waren besonders erfreut über die hohe Zahl der Fachbesucher. Die Messe AG gibt deren Zahl mit rund 300 000 an, sicherlich einer Gründe dafür, daß fast alle Aussteller von guten Geschäftsabschlüssen berichten oder noch im Nachfeld damit rechnen. Kein Wunder, daß sich 90% der Aussteller für die CeBIT'89 bereits angekündigt haben.

Einen Wermutstropfen gibt es leider auch von der Messe der Rekorde zu berichten: Die Ce-BIT 88 endete mit einem Rekorddiebstahl. Der verlustreichste Diebstahl fand in Halle 13 statt. Einem italienischen Aussteller wurde ein komplettes Videokonferenzsystem im Wert von 200.000 DM entwendet. Auch unser Stand blieb von den Langfingern nicht verschont, einige Mäuse, Disketten und ein Rechner wurden entwendet.



Immer dicht umlagert!



Die Merlin Computer GmbH war mit einem eigenen Stand auf der Messe vertreten und kann die Angaben der Messe AG nur bestätigen. Der Stand in Halle 17 war nach Öffnen der Messepforten bis zum Schließen stets von Interessenten dicht umlagert, so daß die anwesenden Redakteure der KICKSTART kaum Zeit zum Entspannen hatten. Dort konnten die Neuerungen der Firma 'live' begutachtet werden. Ob es die 512 KByte-Speichererweiterung im Selbstbau, der in dieser Aus-

gabe vorgestellte Speeder oder der Juniorprommer zum Brennen von Eproms war, alles fand regen Anklang. Des weiteren waren wir für Sie ständig auf dem Messegelände unterwegs, um die Neuigkeiten der CeBIT 88 zusammenzutragen. Interessantes gab es in Hülle und Fülle.



Alles gut überstanden!

Am letzten Tag, pünktlich um 18 Uhr, wurden die Tore der CeBIT geschlossen. Grund genug, einige Flaschen Sekt zu köpfen. So fand die Messe noch einen feucht-fröhlichen Ausklang. Unser Bild zeigt die Merlin Computer-Belegschaft in aufgelockerter Atmosphäre.

Doch für die Redaktion fängt die Arbeit nach der Messe erst an. Unterlagen müssen bearbeitet, Informationen ausgewertet und gewonnene Erfahrungen in die Tat umgesetzt werden. Der Nutznießer all unserer Bemühungen

sind natürlich in erster Linie Sie, liebe Leser. Wir sind darum bemüht, die erhaltenen Informationen auf direktem Wege an Sie weiterzugeben, denn dazu ist eine Fachzeitschrift wie die KICKSTART verpflichtet.

Das Messe-Umfeld

Im Umfeld der CeBIT fanden natürlich auch anderweitige Aktivitäten statt. Unter anderem hatte der NDR einen Messesender eingerichtet, der tagtäglich über das Geschehen auf der CeBIT berichtete. Der private Sender RTL Plus war sogar mit einem eigenen Studio in Halle 17 vertreten, von wo jeden Tag eine Stunde Reportagen und Interviews live gesendet wurden.

Des weiteren war wie immer ein Computer-Camp eingerichtet, das traditiongemäß Themen aufgriff, die im allgemeinen Messetrubel untergehen. Hauptthema war diesmal Frauen und Computer. Aber auch andere Gebiete wurden aufgegriffen.

Was gibt's neues von COMMODORE?

Die traditionelle Pressekonferenz

Wie seit je her fand am ersten Tag der CeBIT die Pressekonferenz von COMMODORE statt. Neben Pressesprecher Gerold Hahn, dem Europa-Manager Harald Speyer und Deutschland-Geschäftsführer Heinz Wiening war auch der Präsident von COMMODORE International, Irwin Gould, anwesend.

Wie schon in den letzten Quartalen zeigte man sich zufrieden. Die Auftragslage in Deutschland und Europa hat sich trotz zeitwei-



ser Produktions-Engpässe stark nach oben entwickelt. Laut Aussage von COMMODORE beträgt der Anteil der AMIGA-Familie einschließlich deren Pheripherie gut 40%, was die Bedeutung des AMIGA für COMMODORE deutlich macht.

Auch die Geschäfte mit den anderen Computersystemen (C64, MS-DOS-Rechner) laufen sehr gut. Rund 290000 Systeme in Deutschland und fast 450000 verkaufte Computer in Europa bilden ein starkes Standbein von COMMODORE International. Die 12 europäischen COMMODORE-Gesellschaften und -Distributoren haben damit zu zwei Dritteln am weltweiten Gesamtumsatz des Unternehmens beigetragen.

Die Postion als Marktführer im Micro-Computer-Bereich konnte damit behauptet werden.

Nach diesen Zahlen kam der für den AMIGA-Fan interessantere Teil der Pressekonferenz: Der Geschäftsführer von Deutschland, Heinz Wiening, stellte die neuen Produkte von COMMO-DORE vor. Besonders interessant war dabei die Produktpalette auf der Basis des AMIGA 2000. Vorgestellt wurden der AMIGA 2500 AT und der AMI-GA 2500 UX. Gemeinsam ist die Basis der Rechner, die auf dem Prozessor 68020 beruht, doch während der eine mit der 80286 AT-Karte bestückt ist, läuft der zweite unter dem Betriebssystem UNIX V. Weitere Einzelheiten zu diesen Rechnertypen folgen weiter unten in diesem Bericht, so daß hier darauf verzichtet wird. Wichtig ist dabei noch, daß alle Neuerungen auch als Nachrüstsatz für den AMIGA 2000 erhältlich sein werden, der Dank seiner offenen Sysemstruktur OSA immer wieder erweitert werden kann. Gleiches gilt prinzipiell auch für den AMIGA 500, für den es von verschiedenen Anbietern bald Erweiterungsboxen geben wird, um ihn aufzurüsten.

Supercomputer AMIGA 3000 ???

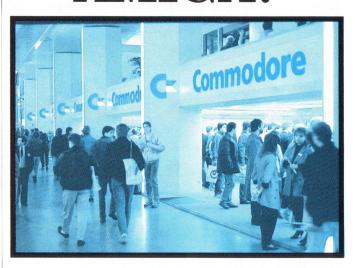
Auch ein neuer Supercomputer wurde angekündigt: Der AMIGA 3000. Zwar wird er mit Sicherheit nicht mehr in diesem Jahr erscheinen, aber dennoch erhitzt er schon jetzt die Gemüter. Großer Speicher mit 32 Bit-Rams, 2.2 GigaByte Festplatte und ein serienmäßiges Netzwerk.

Außerdem ließ man verlauten, daß auch an einem Transputer-Projekt gearbeitet wird. Der Transputer wird ebenfalls als Erweiterungskarte erhältlich sein. Ebenfalls neu ist der High-Resolution-Monitor A 2024, der für alle AMIGA-Modelle erhältlich sein wird. Er zeigt gestochen scharfe Monochrom-Grafiken in drei Auflösungen bis 1008 x

1024 Punkten, und das ohne das lästige Interlace-Flimmern.

Ankündigungen gibt es demnach genug, doch nun muß sich zeigen, in welchem Zeitraum sie realisiert werden und wann sie letztlich für den Kunden verfügbar sind. COMMODORE nannte als Termin für die A 2500-Serie den September, und auch der 3000er soll noch in diesem Jahr erscheinen. Die Erfahrung hat jedoch gelehrt, daß man diese Daten nicht so ernst nehmen sollte. Das Transputer-Projekt und vor allem der A 3000 sind noch vage Zukunftsmusik. Dies sieht man auch daran, daß so gut wie keine Daten darüber existieren.

QUO VADIS, AMIGA?



Welche Eindrücke sammelt ein AMIGA-Anwender auf der Ce-BIT? Einige Neuheiten, viele Ankündigungen und noch mehr Gedränge am COMMODORE-Stand, so könnte man es kurz umschreiben. Aber Spaß einmal beiseite: Wer sich nach den Gerüchten vor der CeBIT über neue AMIGA-Modelle, bessere Custom-Chips und noch vieles andere mehr auf irgendwelche mehr oder weniger sensationellen Neuvorstellungen gefreut hat,

der dürfte doch etwas enttäuscht diese größte Computer-Schau der Welt verlassen haben, denn von COMMODORE selbst gab es auf den ersten Blick eigentlich nichts neues zu sehen. Doch zwei Dinge fielen dann doch noch auf: Das erste war ein unscheinbarer AMIGA 2000, in dessen Innern jedoch der Prototyp einer Transputerkarte von COMMODORE seine Arbeit verrichtete.

Zauberwort TRANSPUTER

Diese beschränkte sich zum gegenwärtigen Zeitpunkt allerdings auf die Abarbeitung einer Vorabversion des wohl in naher Zukunft allgemein verbreiteten Transputer-Betriebssystems HELIOS (auf dem Bildschirm war die Bezeichnung "Helios V 0.2" zu lesen, die auf ein nicht allzu weit fortgeschrittenes Entwicklungsstadium schließen läßt), und auf die Verrichtung

einiger Benchmarks, wobei sich das System allerdings darauf beschränkte, lediglich die benötigten Zeiten auszugeben. All dies war recht wenig informativ, und es steht zu erwarten, daß noch einige Zeit vergehen wird, bevor in Sachen AMIGA und Transputer etwas aktuelles ansteht (zumindest, was eine COMMODO-RE-Transputer-Erweiterung betrifft).

Flimmerfrei und hochauflösend - der neue MONITOR



Interessanter war da schon der neue COMMODORE-Monitor A 2024, der sich offensichtlich in einem fortgeschrittenen Entwicklungsstadium befindet. Um es aber vorweg zu nehmen: Für Freunde vielfarbiger und bewegter Grafik dürfte dieses Modell recht uninteressant sein, denn es handelt sich um einen Schwarz-Weiß-Monitor. Aufhorchen läßt allerdings die Auflösung dieses Bildschirms: 1024 mal 1008 Pixel, und das nicht in zwei Farben, sondern gleich in vier (weiß, hellgrau, dunkelgrau schwarz). Damit prädestiniert sich der nach seinem Entwickler Headly Davis benannte Headly-Scan vor allem für Anwendungen wie Desktop-Publishing, Textverarbeitung und natürlich CAD-Programme.

Betrieben werden kann der Headly-Scan mit einem normalen AMIGA 2000, wobei allerdings eine spezielle Workbench vonnöten ist. Außerdem arbeitet der Headly-Scan mit einem speziellen Trick: Da der AMIGA bei einer solchen Auflösung und Farbtiefe von der Geschwindigkeit der Hardware her gar nicht in der Lage wäre, das Bild in der

geforderten Zeit (1/50 Sekunde) an den Monitor zu übermitteln, erhielt der A 2024 einen eigenen Bildschirmspeicher. Der AMI-GA übermittelt dem Monitor nun jede 50stel Sekunde nur jeweils ein Viertel des Gesamtbildes, während der Monitor ordnungsgemäß aus dem eigenen Bildschirmspeicher heraus fünfzigmal in der Sekunde ein neues Bild aufbaut. Dieses Verfahren bewirkt einen teilweise versetzten Bildaufbau, der animierte Grafik unmöglich macht, aber bei den oben erwähnten Anwendungsgebieten nicht weiter stört. Da sich die über eine Million Pixel auf einen normalen 15-Zoll-Bildschirm verteilen, liefert der A 2024 ein unglaublich scharfes, differenziertes Bild, das mit seiner Genauigkeit die fehlenden Farben schnell vergessen

Interessant dürfte auch der Preis des Gerätes sein: COMMODO-RE peilt einen Verkaufspreis von ca. 1500.- DM an, was bei der gebotenen Bildqualität als überaus preiswert anzuschen ist. Allerdings soll der Headly-Scan nicht vor Herbst '88 im Handel erhältlich sein.

Fragen an den Boss:



Interview mit Irwin Gould



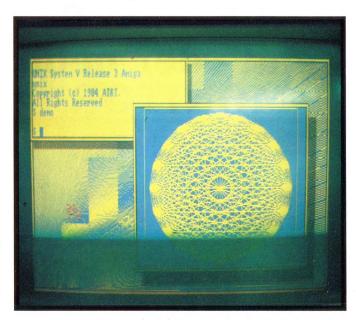
Da reine AMIGA-Neuigkeiten nicht vorgestellt wurden, galt es, sich bei COMMODORE etwas genauer umzuhören. Zu einem Gespräch durften wir Irwin Gould, den Präsident von COM-MODORE International, und Winfried Hoffmann, seines Zeichens Vizepräsident von Europa, an unserem Messestand begrüßen. Im Verlauf dieses Gespräch erhielten wir einige Informationen, was die Zukunft des AMI-GA betrifft. So sind von COM-MODORE für den Sommer einige neue Modelle geplant, die den AMIGA vor allem für professionelle Anwender interessanter machen sollen. Der AMIGA 2500, wie die Bezeichnung des neuen Modells lautet, wartet serienmäßig mit einem 68020-Prozessor und einem 68881 (mathematischer Coprozessor) auf. Auch das Problem des Interlace-Modus, das von Vielen (wohl auch von COMMODORE) als

das größte Manko des AMIGA angesehen wird, scheint nun gelöst worden zu sein: In Verbindung mit einem neuen Blitter-Chip, der jetzt ein Megabyte Chip-RAM ansprechen soll, wird der neue AMIGA in der Lage sein, 640 mal 512 Pixel ohne Interlace darzustellen. Dabei hat, wie Mr. Gould uns mitteilte, COMMODORE auch an die Anwender der 2000er-AMIGAs gedacht, denn der neue Blitterchip soll als Nachrüstsatz für diese Serie erhältlich sein, wobei COMMODORE auf einfache Nachrüstbarkeit und vor allem einen moderaten Preis Wert legt. Dies dürfte all diejenigen erfreuen, die sich erst kürzlich einen AMIGA 2000 zugelegt haben. Leider ist von COMMODORE keine Erweiterung der Anzahl der gleichzeitig darstellbaren Farben vorgesehen, wofür das Betriebssystem bereits von Anfang an ausgelegt wurde.

NEUE AMIGAS: A 2500 AT/UX

In einem Gespräch mit Gail P. Wellington, General Manager für Software- und Produktsupport, erhielten wir noch einige weitergehende Informationen über das neue AMIGA-Modell A 2500. Es wird in zwei Versionen auf dem Markt erhältlich sein, nämlich als A 2500 AT oder auch als A 2500 UX. Ersteres Modell arbeitet mit normalen AMIGA-Betriebssystem, letzteres wird serienmäßig zusätzlich mit dem Betriebssystem UNIX ausgeliefert werden, was den AMIGA der Welt der professionellen Anwender näherbringen soll. Für den A 2500 AT wird UNIX getrennt angeboten. Da die neuen AMI-GAs mit dem 32-Bit-Prozessor 68020 und dem dazugehörigen mathematischen Coprozessor arbeiten, werden sie auch serienmäßig mit 32-Bit-Speicher ausgerüstet sein, wobei man in der Grundausstattung zwischen 2 und 4 Megabyte wird wählen können. Dies wird gegenüber den bisherigen Modellen deutliche Geschwindigkeitssteigerungen mit sich bringen. Natürlich wird das Betriebssystem all diese neuen Gegebenheiten optimal ausnutzen, soll es doch in einer neuen Version, momentan als V 1.3 (die Benennung während der Entwicklungsphase) bezeichnet, verfügbar sein.

Über die Preise der neuen Modelle der Serie 2500 war noch nichts Definitives in Erfahrung zu bringen, doch man kann davon ausgehen, daß für einen Amiga der Modellreihe A 2500 je nach Ausführung zwischen 7000.- und 9000.- DM investiert werden müssen. Auch dies macht klar, daß COMMODORE den AMIGA, der sich zu ihrem absoluten Flagschiff gemausert hat, ver-



stärkt den ernsthaften Anwendern nahebringen und vom Image eines Spielecomputers, das ihm doch recht stark anhaftet, fortführen möchte. Man kann

aber in jedem Falle, ob als ernsthafter oder als Hobby-Anwender, gespannt sein auf die Präsentation der neuen AMIGAs.

KICKSTART/ WORKBENCH 1.4

Neben der Ausnutzung der Hardwareeigenschaften der neuen Rechner-Familien versprechen Kickstart und Workbench V 1.4, wie sie bei ihrer Auslieferung heißen sollen, einige Features, die das Arbeiten mit dem AMI-GA deutlich angenehmer gestalten sollen: Das Booten der Workbench von einer Harddisk wird möglich sein, und die Zugriffe auf die Harddisk, vor allem aber das Laden einer Directory, sollen deutlich beschleunigt werden. Auch in Sachen Druckertreiber hat man sich bei COMMODORE einige Gedanken gemacht, und das Ergebnis dieser Bemühungen

verspricht eine deutliche Geschwindigkeitssteigerung bei der Ansteuerung von Druckern über die Workbench. Diese Neuerungen werden wohl all diejenigen erfreuen, die mit dem AMIGA arbeiten und sich schon öfters über langsame Directories und Ausgaben auf dem Drucker geärgert haben. Unklar blieb allerdings, ob sich die neue Workbench auch auf den 2000er-Modellen mit Kickstart 1.2 betreiben läßt, allerdings gab es die erfreuliche Information, daß auch Kickstart 1.4 als Nachrüstsatz für 2000er-Anwender erhältlich sein wird.

68020-Prozessorkarte A 2620

Nachdem es in letzter Zeit vermehrt Anbieter für 68020-Boards gibt, will auch COMMODORE nicht zurückstehen und bietet mit dem A 2620 eine Prozessorkarte für den AMIGA 2000 an. Die Taktfrequenz liegt bei stolzen 14 MHz. Passend dazu gibt es auch den Coprozessor 68881 mit 14, 20 oder 25 MHz und die Speicherverwaltungseinheit 68851. Der Speicher ist als 32 Bit-RAM angelegt und in einer 2- oder 4 MB-Version erhältlich. Erscheinungsdatum und Preis wurden noch nicht genannt.

Farbdrucker MPS 1224 C

Speziell für den AMIGA und die ATs gibt es einen neuen Farbdrucker, der mit 220 Z/Sec in Entwurfsqualität und 72 Z/Sec in Schönschrift ganz schön abfegt. Besondere Merkmale sind: serienmäßiger Schub-Traktor, geringe Geräuschentwicklung, schneller Farbwechsel, komfortabler Papiereinzug und geringer Stromverbrauch.

Der neue Drucker wird die Baureihe 2000 und 2010 ablösen, woraus man ersehen kann, daß sich COMMODORE und NEC getrennt haben. Diese Entwicklung wird sicherlich einige traurig stimmen, nicht zuletzt, weil der Neue optisch nicht sehr gelungen ist und auch sonst keinen besonders soliden Eindruck macht. Der Test in einer der nächsten Ausgaben wird zeigen, was daran wahr ist.

WAS GAB'S SONST NOCH AM STAND VON COMMODORE?

Professionelle HARDWARE bei GIT-Wiese

Auf der Messe fanden wir Gelegenheit, uns mit dem Geschäftsführer der Gesellschaft für Innovative Technologie (GIT), Herrn Wiese, zu unterhalten. Die Firma hat sich speziell der Anwendung des AMIGA in Forschung, Technik und Industrie verschrieben. Über Neuerungen für diesen Einsatzbereich konnten wir einige hochinteressante Informationen sammeln. Zunächst sei hier das Turbo 2000-Board von CSA erwähnt, welches von uns schon in der 'langsamen' Version getestet wurde (KICKSTART Ausgabe 2/88). Die Karte ist in vielfältigen Konfigurationen erhältlich, so daß der Kunde sich die Geeignetste selbst zusammenstellen kann: angefangen vom einfachen Board mit 68020-CPU und 68881-FPU mit einer Taktfrequenz von 14 MHz über eine 68882-FPU mit 16 MHz bis hin zur 68030-CPU und 68881-FPU mit 14 und MHz oder 25 MHz Taktfrequenz. Natürlich können die verschiedenen Karten auch mit einem 32 Bit-RAM ausgerüstet werden. Die Preise für die verschiedenen Zusammenstellungen liegen zwischen 1550.-

DM und 36475.- DM. Des weiteren konnten wir die Arbeitsweise des sogenannten Noninterlaced-Coloradapters auf der Messe begutachten. Diese Karte läßt den vielkritisierten Interlace-Modus des AMIGA der Vergangenheit angehören. Der Einbau ist denkbar einfach, aber leider nur beim AMIGA 2000 mit der Fat-Agnus und 1 MByte auf der Hauptplatine möglich, da dieser AMIGA über einen erweiterten Videoslot verfügt. Damit die Karte einwandfrei funktioniert, ist ein Multisync-Monitor mit höherer Bandbreite und Bildfrequenz vonnöten. Die Karte ist laut Herrn Wiese speziell für Anwendungen im Bereich CAD gedacht. Die technischen Daten des Noninterlaced-Coloradap-

- Maximal darstellbare Auflösung: 702 * 512 Pixel
- Horizontalfrequenz: 31.5 KHz
- Vertikalfrequenz: 50 Hz
- Videosignal: RGB analog 0.7 Vpp, 75 Ohm
- Videoausgang: 9-polige D-Sub-Buchse

Der Preis der Karte liegt bei 1675.- DM.



Der AMIGA 2000 mit Noninterlaced- und 68030-Karte von GIT.

Weiterhin konnten wir einige Informationen über eine Netzwerkkarte für den AMIGA 2000 in Erfahrung bringen, die beispielsweise die Kommunikation und das Zugreifen auf gemeinsame Massenspeicher mit einer SUN, DEC-VAX oder Apollo Domain ermöglicht. Die Ameristar Ethernet bedient sich dabei eines bereits vorhandenen Standards, um die Kommunikation zu ermöglichen. Der Einbau ist denkbar einfach, und durch die Window-Technik des AMIGA ist das Einloggen des Rechners ebenfalls kein Problem.

Auf der Messe führte GIT das CAD-Programm X-CAD Designer auf einem ausgebauten AMI-GA 2000 mit Noninterlaced- und 68030- Karte vor. Wir konnten uns aus diesem Grunde direkt von der Leistungsfähigkeit der Neuerungen überzeugen.

GIT-Wiese Maassenstr. 10 4235 Schermbeck Tel. 02853-4099 und 4129

Der Nachfolger: NEC P6 PLUS



Bei NEC setzt man mit dem Hinweis auf die hohen Betriebskosten weniger auf Lasertechnologie als auf 24-Nadeldrucker. Neben dem P2200, an dem einige Modifikationen vorgenommen wurden, waren die echten Neuigkeiten die P6/7 Modelle, die jetzt den Zusatz "plus" erhalten. Dahinter verbirgt sich ein völlig neuer Drucker mit NEC V50 Prozessor, immerhin ein Chip, der auch in IBM-AT kompatiblen Rechnern seinen Dienst tut. Sein Hauptspeicher von 80kB dürfte so manchen Spooler überflüssig machen. Neu ist eine Druckwegoptimierung auch bei der Graphikausgabe. Es werden für diese Modelle Fontmodule erhältlich sein, die Textdruck mit einer Auflösung von 360 x 360 DPI erlauben. Im Unterschied zu Epsons 48-Nadler wird diese durch Druck einer Zeile in zwei Durchgängen erreicht.

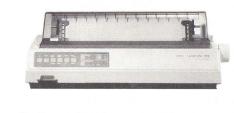
Weitere Ausstattungsmerkmale sind sieben feste Zeichensätze, eingebauter Schubtraktor mit Parkfunktion für Endlospapier, von außen zugänglicher Papierstärkenhebel sowie als Option die Aufrüstung auf Farbbetrieb. Seine Geschwindigkeit ist gegenüber dem P6 auf 265 CPS in Draft erhöht worden. Ein besonderer Gag ermöglicht bidirektionales Drucken von Graphiken: der Versatz zwischen Graphik, die von links nach rechts und solcher, die in anderer Richtung gedruckt wird, ist einstellbar und läßt sich so justieren, daß er völlig verschwindet. Auf dem deutschen Markt wird die Plus-Serie nach Auskunft von NEC Mitte des Jahres erhältlich sein. Der P6 Plus soll zu einem Preis unter 2000 DM angeboten werden.

Neues von OKI

OKI präsentierte gleich zwei neue 24-Nadler, den Microline 390 und 391. Der 391 unterscheidet sich allerdings nur durch seine größere Wagenbreite (DIN A3) vom 390. Auch bei den kleinen OKIs gibt es nun Papierparkfunktion und zusätzlich steckbare Fontmodule. Außerdem besteht



die bereits gewohnte Möglichkeit, das Endlospapier wahlweise von der Rück- oder Unterseite des Druckers zuzuführen. Die Geschwindigkeit der beiden Drucker beträgt 225/75 CPS in



Draft/Letter Quality, serienmäßig bieten sie nur einen eingebauten Font, sind aber relativ leise und machen einen stabilen Eindruck. Der Preis für die DIN A4-Version beträgt ca. 1600

DM. Weiterhin gab es auch bei OKI überwiegend Bekanntes und Bewährtes wie den Microline 393, einen 24-Nadler für höchste Ansprüche und den OKI Laserline 6 zu sehen.

Was gab's bei SEIKOSHA?

Seikosha hatte die oben erwähnte 1000 DM-Grenze bereits mit seinem SL-80 AI unterschritten. Dieser 24-Nadler heißt jetzt SL-80 IP und ist kompatibel zum NEC P6, da er nun auch die Auflösung 360 x 360 DPI im Graphikmodus beherrscht. Weitere Details: 12 eingebaute Schriftarten, 135/54 CPS in Draft/Letter Quality. Sein Verkaufspreis von 899 DM macht ihn zum momentan preiswertesten P6-Kompatiblen. Seikosha stellte auf der CeBit seinen ersten Laserdrukker, den OP-105 A, vor. Die Druckgeschwindigkeit

recht kompakten Druckers beträgt fünf Seiten pro Minute bei einer Auflösung von 300 x 300 DPI. Er ist kompatibel zu den vier gängigsten Lasern und hat 30 eingebaute Zeichensätze. Für 4999 DM bekommt der Kunde die Grundausstattung mit 512 kB Ram, die jedoch auf 1,5 MB ausgebaut werden kann.

Auf die Lasertechnik setzt man genauso bei C.Itoh, Fujitsu und Citizen. C.Itoh will mit seiner Produktpalette ausschließlich professionelle und damit zahlungskräftige Kunden ansprechen. So wird es im Bereich preiswerter Nadeldrucker in Zu-kunft keine Neuerungen geben. Den auf der Messe zum ersten Mal gezeigten CI-5, einen Laserdrucker für 5000 DM, betrachtet man bei C.Itoh als Abrundung nach unten. Er ist serienmäßig HP-Laserjet kompatibel, druckt fünf Seiten pro Minute und hat zwei eingebaute Fonts. Weitere Emulationen und Zeichensätze sind über austauschbare Kassetten möglich. Sein Hauptspeicher von 512 kB ist auf 2 MB ausbaufähig.

Trotz des unübersehbaren Vormarschs der Lasertechnologie -

auch bei gehobener Heimanwendung- bleiben die Nadeldrucker nach wie vor das wichtigste Ausgabemedium. Hier ist ein deutlicher Trend zu mehr Bedienerfreundlichkeit und Ausstattung, besonders was Zeichensätze und Emulationen auf Steckkarten und -modulen betrifft, festzustellen. Trotzdem werden die Geräte, wie fast alles auf dem Computersektor, immer preiswerter. Wir werden in den nächsten Monaten einige der auf der CeBit vorgestellten Drucker für Sie testen, um Ihnen weiterhin den Überblick über diesen vielfältigen und für den Laien nahezu undurchschaubaren Markt zu erleichtern.

Ingo Brümmer

Professional Video bei VIDEOCOMP



Die Firma VIDEOCOMP machte ihrem Namen alle Ehre und zeigt dem staunenden Publikum eine Video-Anlage, bei dem so manchem Amateur und auch dem Profi das Herz höher schlägt. Und das konnte man sehen: vor dem Hintergrund einer Filmszene schwebte eine Garfield-Figur über den Bildschirm und zusätzlich wurde eine Laufschrift eingeblendet, die mit dem PROFES-SIONAL VIDEO TITLER erstellt wurde. Diese Effekte wurden dadurch erreicht, daß über

ein spezielles Video-Mischpult Signale einer Kamera, eines Videorecorders und eines AMIGA zusammengemischt und auf einem Monitor dargestellt wurden. Besonders in der Werbebranche können solche Anlagen gut eingesetzt werden. Nähere Informationen erhält bei:

VIDEOCOMP Dipl.-Kfm. K.J. Leuze Berger Straße 193 6000 Frankfurt/M. 60 Tel.: 069- 46 71 01

Der AMIGA als Friseur ?!

Durch die guten Hardwareeigenschaften des AMIGA sind die Weichen gestellt worden: Brandneu präsentierten Bernd und Torsten Keller ein Frisurenberatungssystem namens comhair. Zugegeben, es hört sich zunächst nach einem schlechten Scherz an, aber weit gefehlt, comhair ist eine durchaus ernstzunehmende Anwendung. Experten kennen die Problematik. comhair kommt dem Kunden zugute, denn dieser kann sich individuell beraten lassen. Zu diesem Zweck wird er digitalisiert und auf dem AMI-GA-Monitor ausgegeben, auf dem dann wiederum die verschiedenen Frisuren eingeblendet werden. Der Kunde kann sich von Punk- bis hin zur New-Wave-Frisur alles einblenden lassen, der AMIGA übernimmt das Zusammenfügen beider Teile zu einem natürlichen Kopf. Aber auch beliebige Manipulationen der Frisuren lassen sich mit dem System leicht durchführen: Haare färben. Strähnchen kreieren oder die Schnittanleitung ausgeben. Die Vorführung auf dem Messegelände ließ uns alle Zweifel vergessen. Bisher waren ähnliche Systeme enorm teuer, denn



Bild: Die Dame kann sich in beliebig vielen Frisuren betrachten.

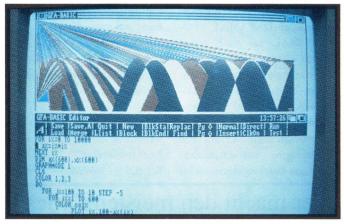
an die 100000.- DM mußten über den Ladentisch wandern. Das comhair-System ist vergleichsweise billig. Der interessierte Friseur muß etwa 20000.- DM für die Anlage ausgeben, ein Leasing ist allerdings auch möglich. Für Hobbyfriseure ist die Anschaffung aber zweifelhaft, denn wer will schon 20000.- DM für das Haareschneiden berappen. Wenn Sie das nächste mal zum Friseur gehen, können Sie sich vielleicht auch in einen AMIGA-Monitor mit einer Punk-Frisur begutachten. Anbieter:

Bernd und Torsten Keller GbR Kleiststraße 11 3180 Wolfsburg 1 Tel. 05361-21680

Erste Demo von GFA-BASIC 3.0

Nachdem GFA-Basic nachhaltig die ATARI ST-Welt begeistert, kommt die neue Version 3.0 auch für den AMIGA. Am COMMO-**DORE-Stand** führte Georg Zweschper, der für die AMIGA-Umsetzung verantwortlich ist, eine in Teilen bereits lauffähige Version vor. Wie man sehr schnell erkennen konnte, wurde dabei ein besonderes Augenmerk auf leichte Bedienbarkeit und hohe Ausführungsgeschwindigkeit gelegt. Der Editor erinnert sehr stark an die ATARI ST-Version (was jedoch kein Nachteil ist!). Er ist sehr schnell (geradezu

rasend im Gegensatz zu Amiga-Basic) und läßt sich mit der Maus und den Funktionstasten einfach bedienen. Der Befehlsumfang wird sehr groß werden. Neben den mittlerweile gängigen Strukturbefehlen werden noch einige besonders leistungsfähig Befehle implementiert. Auch die Benutzung der Betriebssystem-Funktionen wird vereinfacht werden. Mitgeliefert wird zudem das 'Requester Construction Tool', kurz C.R.T. genannt, das es ermöglicht, mit einigen Mausaktionen Menuleisten und Requester zu erstellen, die mit speziellen



Befehlen vom Basic aus verwaltet werden können. Damit bietet GFA-Basic einen beispielhaften Komfort für Programmierer, die nicht so tief in das Betriebssystem des AMIGA eindringen wollen. Diesen ist es damit möglich, eine komfortable Benutzerführung in ihre Programme zu

integrieren, ohne aufwendige Systemaufrufe durchführen zu müssen. Man darf also gespannt dem Erscheinungsdatum von GFA-Basic entgegensehen, das jedoch vorläufig auf September dieses Jahres verschoben wurde. Anbieter: GFA-Systemtechnik, Düsseldorf

DATAMAT PROFESSIONAL



Mit deutschen Menüs präsentiert sich DATAMAT PROFESSIONAL.

Das deutsche Verlagshaus Data Becker, bekannt durch viele Softwareprodukte, bringt eine neue Datenbank für den AMIGA heraus - DATAMAT PROFESSIO-NAL. Die Redaktion konnte sich auf dem Messestand von Data Becker von den Leistungen des Programms überzeugen. Das Produkt ist zunächst nur für die AMIGA-Modelle vorgesehen, wird aber wohl auch auf andere Rechner umgesetzt werden. Die einfache Steuerung des Programms über Symbole und die enorm flexible Bearbeitung der Dateien zeichnen das Programm aus. Dem Interessierten gebe ich an dieser Stelle einige Informationen über DATAMAT PRO-FESSIONAL: bis zu 8 offene Dateien können bearbeitet werden, maximale Dateigröße 2 Milliarden Zeichen, maximale Datensatzgröße 64000 Zeichen, maximal 2 Milliarden Datensätze, unbeschränkte Anzahl von Datenfeldern, massenspeicherorientierte Dateiorganisation, bis

zu 80 Indexfelder pro Datei, unbeschränkte Anzahl von komfortablen Such- und Selektierkriterien, integrierter Listeneditor und Ganzseiten-Druckmaskeneditor, komfortable Druckeranpassung sowie die Unterstützung aller Texteffekte, mehrere Dateimasken gleichzeitig auf dem Bildschirm, integrierte Basic-ähnliche Programmiersprache mit über 100 Befehlen, Programmsteuerung über Maus und Tastatur, selbstprogrammierbare Pull-Down-Menüs, alle in diesem Programm erstellten Anwendungen sind komplett maussteuerbar, Rechenfelder, echte Relationen zwischen Dateien - one for many and many for one.

Der Preis von DATAMAT PRO-FESSIONAL liegt bei 298.- DM, die endgültige Version wird im Mai lieferbar sein.

Data Becker Merowingerstraße 30 4000 Düsseldorf Tel. 0211-3100130

Wer macht den meisten Druck?!

Hannover, Messegelände, Halle 4, 1. OG: Da standen sie nun in Reih' und Glied, kreischten, pfiffen oder säuselten über ungezählte Meter Papier. Viele Hersteller von Matrixdruckern hatten hier ihre Stände aufgebaut.

Ein Rundgang zeigte wenig Neues, sondern in erster Linie überar-

beitete, bedienungsfreundlichere Geräte. Bei nahezu allen Firmen steht die Lasertechnik im Vordergrund, nicht wenige haben mittlerweile mehrere Laserdrucker im Angebot. Wir haben die wichtigsten Neuigkeiten für Sie zusammengetragen.

WELTPREMIERE:

48 Nadler von EPSON



Unbestrittener Star der Nadeldrucker war der 48-Nadeldrucker von Epson. Er hat -wie von 24-Nadlern gewohnt- 'nur' eine Auflösung von 360 x 360 Punkten pro Zoll (Dots per Inch = DPI), nutzt diese jedoch auch beim Textdruck, wodurch ein sehr scharfes Druckbild entsteht. Bei 24-Nadeldruckern beträgt der Abstand der Nadeln voneinander in senkrechter Richtung 1/ 90 Zoll. Da zwei Reihen mit jeweils 12 Nadeln versetzt nebeneinander liegen, wird eine Auflösung von 180 Punkten pro Zoll in der Senkrechten errreicht. Wenn nun, wie beim NEC P6, die Auflösung 360 DPI beträgt, so ist das nur möglich, indem der Drucker das Papier um den halben Nadelabstand, d.h. 1/360 Zoll, vorfährt. So kann in einem zweiten Druckgang in die Zwischenräume des ersten gedruckt werden. Der 48-Nadler von Epson hat in seinem Druckkopf bereits vier Nadelreihen, bei denen der Abstand der Nadeln auch jeweils 1/ 90 Zoll beträgt. Diese sind aber jeweils um 1/360 Zoll gegeneinander versetzt, wodurch die hohe Auflösung in einem Druckgang erzielt wird.

Papierführung total heißt Epsons Devise für die nächsten Druckergenerationen. Allerdings sind Ausstattungen mit Parkmöglichkeit für das Endlospapier, um Einzelblatt und Endlospapier ohne Wechsel zu verarbeiten, oder der Möglichkeit, Zeichensätze per Modul nachzurüsten, bei vielen Herstellern Standard. Der 48-Nadler bietet außerdem zehn residente Zeichensätze bei einer Geschwindigkeit von 300 bzw. 100 CPS (Zeichen/Sek.) in Draft bzw. Letter Quality. Bei uns soll das neue Flaggschiff im Juli erhältlich sein.

Weiterhin zeigte Epson den LQ-2550, eine Weiterentwicklung des 24-Nadlers LQ-2500. Er ist serienmäßig mit Farbdruckmöglichkeit ausgerüstet, besitzt acht residente Fonts und druckt 400/133 CPS in Draft/LQ. Ausgeliefert wird er ab Juni.

Auch ein neuer HP-Laserjet kompatibler Laserdrucker mit dem Namen GQ-3500 fehlt nicht im Reigen der Neuererscheiungen.

PROGRAMMIEREN

Das richtige Buch für und Umsteiger Auf-





AUS DEM INHALT

PROGRAMMIEREN AUF DEM AMIGA

WICHTIGE MERKMALE

ERNST HEINZ

cute-Programmierung

Erstellen einer eigenen Startup-► CLI und Möglichkeiten der ExeSpezielle AmigaBASIC-Arbeitsdisvette erstellen ▶ Grafikprogrammierung (ohne Ani-

Dieses Buch weist Programmieren
den Wegg wie Sie die fanfastischen
Fähigkeiten Ihres AMIGA auch von
AmigaBASKG eas nutzen Können.
Eschließt die in den Handbuchern
entstanderen Lücken, nieden se die
riffengebiliebenen Fragen in anschau-

Bauteilesatz für Uhr ohne Akku Komplett mit 512K Superpreis mit Uhr

Leerplatine mit Stecker mit Schaltplan und Bestücku

► Maussteuerung
► Menueprogrammierung von PullJown-Menüs

Fenster und Bildschirmtechnik Grafische Animation mit Sprites bisher mit anderen Computativiste man gearbeilet haben, konzipiert.

Besonderen Wert wurde auf guten Degammerstil gelegt Anhand von iber40 sehr austümfich dokumenier ein Übungs. u. Beispelprogrammen kann der Leser tranineren, fremde Pro► Kleine Einführung zur Spezialhard-vare des AMIGA

▶ Alle offenen Fragen und Probleme in Bezug auf AMIGA-spezifische Pro-grammiertechniken und -befehle wird Ihnen dieses Buch zu beantworten ramme zu lesen und zu verstehen nd richtige und gute Programmie-ing erlernen.

► Ein Potpurri von AmigaBASIC-Pro-

iber 300 Seiten DM 49,-*

Besonders werden folgende Haupttheren in me Luch befrandelt.
Fensterrechnik Menue-Technik Unterbrechnig-Technik Animation
in Amiga-Basio Grafit-Gelehle
Normale Maussteuerurik-Befehle

Zun Buch gibt es eine ProgrammDiskette mit allen aufgeführten Ubungs-

PROGRAMMDISKETTE AUS DEM INHALT

► Zur Unterstützung der praktischen Übung und Arbeit am Computer gibt es die Programmdiskette.

► An über 40 Übungs und Beispielprogrammen können Sie guten Programmen können Sie guten Programmenstil nachvoliziehen und trainiern.

► Dam ihr auf die kätiger Tipparbeit ein Ende. Diskette laden und lös geht's.

► Die Diskette ist beim Verlag erhaltie

Sound- und Sprachprogrammierung (Erzeugung menschlicher Sprachelf-zeugung von Tonen und mehrstimm-gen Meldellenf Toneszeugung gemäß musikalischer Notemotierung)

— Unterbrechungsfähigkeiten von

licher Weise beantwortet.

➤ Damit ist dieses Buch insbesondere für lernwillige Aufsteiger, d. h. mehr oder weniger erfahrene BASIC-Programmierer, sowie für Umsteiger, die bisher mit anderen Computersyste.

Für 512k zusätzliches RAM • Alle RAM's gesok-kelt • Selbstkonligurierend • Abschaltbar • Uhrensschaftung auf Platine mit Akku- bzw. Batte-rieputferung nachrüstbar

500er Speichererweiterung

Preise sind unverbindlich empfohlene Verkaufspreise

Profilaufwerk 3,5"

St. Programmieren auf dem AMIGA á DM 49,-

Ich bestelle:

an Heim-Verlag Heidelberger Landstraße 194 6100 Darmstadt-Eberstadt

BESTELL-COUPON

St. Programmdiskette zum Buch á DM 29,-

zzgl. DM 5. Versandkosten (unabhängig von bestellter Stückzahl)

□ per Nachnahme □ Verrechnungsscheck liegt bei

Name, Vorname Straße, Hausnr.

Verlag

Heidelberger Landstraße 194 6100 Darmstadt-Eberstadt Telefon 06151-56057

PLZ, Ort

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte.

349.

TEAC FD 135 FN 3,5"

1MB • 1 Zoll (2,54 cm) hoch

namhafter Hersteller 🍩 mit schrecht 2 Jahre Garantie 🌑 fast alle IC'S gesok-Blockschaltbild • teilausgereifte Ingenieurleistung • 14 Tage Umtaukelt nur professionelle Bedienungsanleitung weise Schaltplan





3,5" Laufwerk

Für alle Amiga's © einstellbare Gerätenummer © Abschaltbar © Metallgehäuse © Superflach 1 Zoll (2,54 cm) © TEAC Lauwerk 298. Komplett anschlußfertig

Laufwerk 5,25"

339.-• 40/80 Track • Laufwerkbus durchgeschleift • abschaltbar • einstellbare Adressen

HD 1,6 MB (umschaltbar)

Gemischtes Doppel 3,5"/5,25"

Für Amiga 1000 und 500 mit Software • Type bei Bestellung hite angeben • 8-8f Denehneite • 8e trite an Parlellort (Druckerport) • Mit Vorverstäfer für Micro-Anschluß (Cinch-Buckerport) • Mit Vorverstäfer für Micro-Anschluß (Cinch-Buckerport) • Mutant Sprachtigiteliserung medich • Arbeitet mit Astellen Digitzer-Pogrammen • Formarchiness

Preis auf Anfrage Preis auf Anfrage

PLCOMP

PLCOMP

4 Kanäle einschließlich 1 Thru 🖜 Optische Datenan-zeige 🛡 Formschönes Gehäuse

MIDI - Interface

89.

Wahnsinnspreis von nur

я сошь

Bootselector

Trackanzeise

49. • für alle Laufwerke (3,5"/5,25") • Laufwerkbus-durchgeschleift • mit Gehäuse

Bauen Sie die anderen Kickstart-Versionen in Ihren Amiga 500 – Einfacher Einbau ohne Liten
 Fig Original-KickstartROM und 2 zusätzliche Versionen auf EPROM

Kickstartumschaltung

Laufwerkanschlußkabel

Zum Anschluß von Laufwerken an alle Amiga's Mit Anctauaralaktronik Mit Ansteuerelektronik

Machinahmeuersand MW-Spesen 7:50 EDM & Vyeksee 3. DM Austanders and Austanders and Stellunger Stellunger, Netchenbraeursand NyeSpesson 10. DM & Vorkasse 5. DM EWir intern Intern and Intern Rechning Cound Gefahrzuden Perkadis- und Liefer und Gefahrzuden Perkadis- und Liefer -

Bestellung und Versand

3-fach Steckplatzerweiterung

Wir suchen ständig Hardware-Entwicklungen. Wir garantieren gute Umsatzprovisionen und ehrliche

A Lanfermann Lessing Str. 46 5012 Bedburg Tel. 02272/1580

Abrechnung

Jeder Steckplatz abschaltbar und einstellbare Lauf-werknummer 🖷 Steckplatzerweiterung direkt am Amigagehäuse 👁 Dadurch keine Kabellängenpro-Anschlußfertig zum Alcompsuperpreis von 49.-

239.

NEWS

KICKSTART 5/88 !

Luxuriöses Programmierutility

Inova Tools 1 nennt sich eine Sammlung von Befehlen die sich mit der Erstellung der AMIGA-Oberfläche befaßt. Wie Sie vielleicht wissen, hat diese Aufgabe unter AmigaDOS Intuition. Die sehr zahlreichen Definitionen unterstützen die bereits vorhandenen ungemein. Wer jetzt glaubt, 'mal schnell eine Datei-Auswahl-Box erstellen zu können, der wird entäuscht werden. Zwar bietet Inova Tools 1 unterstützende Befehle, aber ein einziger Funktionsaufruf genügt bei weitem nicht. Auch Befehle, die unter Intuition schon vorhanden waren, wurden erweitert oder verändert, beispielsweise beinhaltet Inova Tools einen Befehl FlashyOpenWindow(), der ein Fenster auf einen recht interessanten Weg öffnet.

Natürlich ist das Utility hauptsächlich für C-Programmierer gedacht. Für Aztec- oder Lattice-Besitzer existieren die Routinen in linkbaren 'C'-Libraries, aber auch in Form von AMIGA-Libraries, so daß die Routinen auch von anderen Programmiersprachen genutzt werden können.

Dem Produkt liegt ein Handbuch bei, in dem alle Funktionen erklärt werden. Auf der beiliegenden Diskette sind einige Demoprogramme, die die Arbeitsweise der Routinen verdeutlichen, und natürlich auch die erwähnten Libraries.

Anbieter: GTI GmbH Zimmersmühlenweg 73 6370 Oberursel Tel. 06171-53863/3748

DOS-MAN

Die deutsche Firma TRANSMO-DUL GmbH hat ein sehr ungewöhnliches Projekt in Arbeit. Es handelt sich um eine neuartige Benutzeroberfläche fiir DOS-Rechner. Das Ungewöhnliche ist wohl, daß der Anwender in der Lage ist, seine eigene Oberfläche zu getallten. TRANSMODUL beschreibt DOS-MAN als natürlichsprachiges, intelligentes Kommandosystem für MS-/PC-DOS. Das speicherresidente Programm ermöglicht es dem Anwender, Befehle in Form von ganzen, leicht verständlichen Sätzen dem Rechner mitzuteilen. Beispielsweise kann folgender Ausdruck verwendet werden:

kopiere alle heute veränderten Symphony-Dateinen nach Laufwerk A Der Befehlsausdruck muß wohl nicht näher erläutert werden. DOS-MAN wandelt diesen Befehlsausdruck in eine vom Rechner verständliche Form um und führt ihn dann aus. Alte PC-Hasen werden vielleicht milde lächeln, für Einsteiger bietet DOS-MAN aber mit Sicherheit eine interessante Alternative.

Des weiteren bietet DOS-MAN einige nützliche Features, die auch innerhalb von Programmen angesprungen werden können.

Das Produkt wird sowohl in einer englischen als auch in einer deutschen Version angboten.

Anbieter: TRANSMODUL Am Staden 18 6600 Saarbrücken

Grossmeister

In der April-Ausgabe der KICKSTART haben wir zwei Schachprogramme auf Herz und Nieren geprüft. Leider zu spät erreichte uns ein weiteres Programm namens Grossmeister. Das komplett in deutscher Sprache gehaltene Programm besitzt die üblichen Schachprinzipien

wie Einstellung des Schwierigkeitsgrades, Problemstellungen, Notation, Schachuhr und etliches mehr. Die Spielstärke des Programms ist in der leichtesten Spielstufe (1/2 Sekunde Bedenkzeit) nicht sehr hoch. Grossmeister macht hier noch katastrophale Fehler, aber schon ein Level höher (1 Sekunde Bedenkzeit) wird das Schachprogramm zum ernstzunehmenden Gegner. Die Grafik des Programms ist nicht weltbewegend; es kann zwischen 2D- und 3D- Ansicht gewechselt werden, wobei letzteres nicht gerade der Übersichtlichkeit dient. Als Besonderheit kann die deutsche Sprachausgabe Männer- bzw. Frauenstimme betrachtet werden.

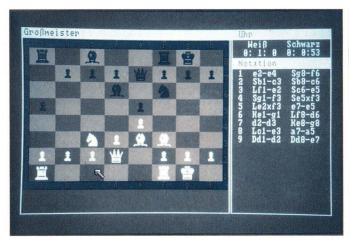
Das deutsche Handbuch erläutert knapp, aber informationsreich die Handhabung von Grossmeister und die allgemein gültigen Schachregeln. Der emfohlene Verkaufspreis liegt mit 79.- DM für ein Schachprogramm recht günstig.

Anbieter:

Heinrichson Schneider & Young oHG Herderstr. 94

D-5000 Köln 94





Der Bildschirmaufbau von Grossmeister.

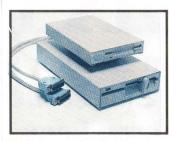
Ungewöhnliche 3 1/2 " und 5 1/4" Laufwerke

Die Firma Kupke bietet neuerdings extravagante Laufwerke für den AMIGA an. Das Besondere ist eine Trackanzeige, die dem Anwender die momentane Position des Kopfes dem Anwender mitteilt. Die Gehäusefarbe ist der des AMIGA angepaßt, ebenso ist der Bus durchgeführt, so daß bis zu vier Laufwerke problemlos angeschlossen werden können. Die NEC 1036A-Laufwerke sind über einen Schalter ab- bzw. einschaltbar. Das 3 1/2" Laufwerk ist mit Trach-Display für 379.- DM erhältlich, ohne Display 369.- DM. Das 5 1/4" Laufwerk besitzt neben den erwähnten Features

noch eine 40/80 Track-Umschaltung. Der Preis des Gerätes liegt bei 449.- DM mit Display, ohne 439.- DM.

Anbieter:

KUPKE Burgweg 52a D-4600 Dortmund 1 Tel: 0231-818325-27



Die Kurpke-Laufwerke mit Trackdisplay.

Deutsches Pagesetter

Das Desktop Publishing-Programm PageSetter ist ab sofort in einer deutsche Ausgabe erhältlich. Im Lieferumfang befindet sich ein 100 Seiten starkes Handbuch in deutscher Sprache sowie die Programmdiskette. Die Be-

dienung von Pagesetter ist ebenfalls in deutscher Sprache gehalten, so daß man nicht mehr mit der englischen Sprache zu kämpfen habt. Erhältich ist PageSetter beim Fachhändler für ca. 348.-DM

SUPERSORTER V1.3

Supersorter nennt sich ein Diskettenverwaltungsprogramm für alle AMIGA-Modelle, welches 10 Dateien zu jeweils 500 Datensätzen verwaltet. Die Bedienung erfolgt ausschließlich mit der Maus, alle Funktionen sind somit schnell und leicht zu erreichen. Das Programm ist multitaskingfähig und verfügt über Funktionen wie Listen-/Etikettendruck, Suchen, Sortieren, Löschen etc.. Das Programm besitzt keinen Kopierschutz, sondern verlangt die Eingabe eines bestimmten Kodewortes. Sicherheitskopien sind deshalb kein Problem. Das in BASIC erstellte und compilierte Programm ist recht schnell, lange Wartezeiten treten nur bei der Sortierroutine und bei langen Dateien auf.

Der Preis von Supersorter liegt mit 19.95 plus Versandkosten günstig. Inbegriffen in diesem Preis ist ein Update-Service, der es ermöglicht, kostengünstig an eine erweiterte Version zu gelangen.

Anbieter: Andreas Oesterle Software Im Ring 47 7129 Ilsfeld Preis: 19.95 DM



Second Chance...

...nennt sich ein Utilityprogramm, welches aus Gründen von Viren veröffentlicht wurde. Wie Sie vielleicht wissen, schreiben sich die Viren-Programme auf die Bootspur der Diskette. Viele Programme legen aber gerade in dieser Spur wichtige Daten ab, beispielsweise einen 'Loader', der das eigentliche Programm erst lädt. Ein Virus zerstört diese Daten, die Folge: Das Programm kann nicht mehr geladen werden. Meistens sind solche Programme gegen Vervielfältigeung gesichert, lassen sich also nicht kopieren. Second Chance soll hier Abhilfe schaffen. Das Programm liest die Bootspur der Orginal-Diskette, die in der Regel ohne Probleme gelesen werden kann, und legt diese in einem

KICKSTART- und Uhren-Modul für AMIGA 1000

Amiga 1000 Besitzer mußten bislang nach Einschalten des Rechners immer eine KICK-START-Diskette einlegen. Mit dem Modul von der Firma Kupke wird dies überflüssig. Der Amiga verlangt nach Einstecken des KICKSTART-Moduls die Workbench. Das Laden des Betriebssystems entfällt somit. Damit aber noch andere KICKSTART-Versionen geladen werden können, ist das Modul abschaltbar.

Weiterhin ist in dem Modul eine Uhr und ein Prozessorhalt-Schalter integriert, dadurch ist es möglich, zu jeder Zeit den Rechner anzuhalten. Nach erneutem Umlegen des Schalters fährt der Rechner wie gewohnt fort. Der Expansionport ist durchgeschleift, so daß Speichererweiterungen, Festplatten oder anderweitige Peripherie noch verwendet werden kann.

Die Firma bietet einen Update-Service an, dadurch können Sie die momentan aktuelle KICK-START-Version immer erhalten. Auch spezielle Versionen können verwendet werden, genannt sei nur eine Anti-Virus-Kickstart. Der Preis des Moduls liegt bei 199.- DM.

Anbieter: KUPKE Burgweg 52a

D-4600 Dortmund 1 Tel: 0231-818325-27 separaten File ab. Bei Befall der Original-Diskette mit einem Virus kann die betreffende Bootspur wieder auf die Diskette zurückgeschrieben werden. Das Programm ist somit wieder einsatzbereit. Second Chance ist für 49.90 DM erhältlich.

Anbieter: PDC GmbH Louisenstr. 115 6380 Bad Homburg Tel. 06172-24748 oder 20799



Tips & Tricks zu Jet

Der neue Flugsimulator von Sublogic - Jet, vorgestellt in dieser KICKSTART Ausgabe, ist mit Sicherheit einer der Glanzpunkte auf dem AMIGA. Das Programm bietet aber weit mehr, als man dem englischsprachigen Handbuch entnehmen kann. Die Firma PDC bietet aus diesem Grunde ein Beibuch an - Tips & Tricks zu Jet. Das Buch ist in deutscher Sprache verfaßt und kostet 29.90 DM. Für Jet und das Beibuch gibt es einen Paketpreis: 99.90 DM. Solo kostet Jet 89,90 DM.

PDC GmbH Louisenstr. 115 6380 Bad Homburg Tel. 06172-24748 oder 20799

Anbieter:

IntSwitch

Das Programm IntSwitch erlaubt es dem Anwender, Software, die nur mit 512 KByte Speicher arbeitet, auf Rechnern zu nutzen, die über ein Megabyte oder mehr verfügen. Die Softwarelösung dieses Problems schont unter anderem die Steckkontakte des Expansionports, da eine Erweiterung jetzt nicht mehr mechanisch aus dem Gerät entfernt werden muß. Auch Lösungen, die mit selbst eingelöteten Schaltern arbeiten und so die Garantie zum Erlöschen bringen, sind überflüssig. IntSwitch ist ab sofort in der neuen Version 1.1 für 29,90 DM erhältlich.

PDC GmbH Louisenstr. 115 6380 Bad Homburg Tel. 06172-24748 oder 20799

Viruskiller

Das Programm Viruskiller hilft dem geplagten Anwender gegen die unliebsamen Softwarebazillen. Viruskiller findet den Übeltäter im Speicher und auf der Diskette, so daß man auf einfache Weise den eigenen Datenbestand überprüfen und retten kann. Das Programm kann auf einfach Art bedient werden und ist somit auch für Einsteiger leicht und sicher benutzbar. Das Programm kostet 49.90 und bietet mit Second Chance vom gleichen Hersteller ein sicheres Mittel gegen Computerviren.

PDC GmbH Louisenstr. 115 6380 Bad Homburg Tel. 06172-24748 oder 20799

KORREKTUR

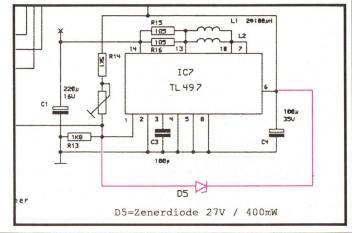
In diesem Heft hat sich im Artikel über den Junior-Prommer ein Fehler eingeschlichen, der zur Zerstörung des Gerätes führen kann.

Im unten gezeigten Schaltbildauszug ist die Zenerdiode vergessen worden. Ohne diese Diode kann die Ausgangsspannung des TL 497 bis zu 51 Volt erreichen, was in der Folge die Bauteile C4,T4,T5 und T6, sowie ein eventuell eingesetztes Eprom zerstören kann.

Die fehlende Zenerdiode ist am besten direkt an den Anschlüssen des Spannungswandlers (Pin 1 und Pin 6) anzulöten. Ohne diese Maßnahme können bei spannungsversorgtem Prommer und fehlender Steuersoftware (z.B. eingestecktem Prommer und eine Textverarbeitung im Rechner) die oben beschriebenen Probleme auftreten.

Achtung

Die von uns gelieferten Fertiggeräte enthalten die angesprochene Änderung. Die momentan gelieferte Platine enthält jedoch noch keine Lötpunkte für die angesprochene Zenerdiode, so daß Sie dies wie oben beschrieben am TL 497 (IC 7) anlöten müssen.





Wettbewerbsbedingungen siehe TOP 12 in diesem Heft	'TOP 12' Mein Lieblingsspiel		KCK START DIE FACHZEITSCHRIFT FÜR	KICKSTART-GRUßKARTE
D-6236 Eschborn	MERLIN'- Computer GmbH Redaktion KICKSTART TOP 12' Industriestraße 26	Bitte frei- machen		Bitte frei- machen
KICK DIE FACHZEITSCHRIFT FÜR START DEN AMIGA-ANWENDER		KICKSTART-GRUßKARTE	START DEN AMIGA-ANWENDER	KICKSTART-GRUßKARTE
		Bitte frei- mach		Bitt frei mach

AMMIGA ASZINAT REISERDE III WIRD nach dem gleichnamigen Werk von F. Bourgeon

Reisende im Wind 1+11

Die Geschichte ist bekannt: Die Komtessä Isa
Und der Matrose Hoel müssen kurz vor Ausund der französischen Revolution ihr Land
bruch der französischen Revolution auf dem
bruch der französischen Revolution auf dem
bruch der französischen Revolution auf dem
bruch der französischen Revolution auf den
verlassen. Tausend Gefahren Häfen zur afrikaverlassen. Tausend Gefahren Häfen zur afrikaverlassen Häfen zur afrikaverlasse

REISERDE III WIRD nach dem gleichnamigen Werk von F. Bourgeon

CHAMONIX CHALLENGE

Captain Blood

SIE sind fünf.
SIE verstecken sich, irgendwo im Sternennebel.
SIE verstecken sich, irgendwo im Grei, die vier
SIE verstecken sich, irgendwo im Sternennebel.
Die Nummer eins, die zwei, die drei, die vier
und der Bastard, Nummer fünf.
SIE erwarten Dich SIE erwarten Dich...



Chamonix Challenge
Brennt auch in Ihrem Herzen so eine Sehnsucht, und draußen herrschen die schlechtesucht, und draußen herrschen daß an eine
sten Wetterbedingungen, so daß an
sten Wetterbedingungen, so daß en ist? Paksten Wetterbedingungen, so denken ist? Paksten Wetterbedingungen, so daß en ist? Paksten Wetterbedingungen, so denken ist? Paksten Wetterbedingungen, so denken ist? Paksten Wetterbedingungen, so denken ist.

NEOGRAMES

SCHICKEN SIE HERE PHANTASIE AUF REISEN

INFOCRAMES

LIA

Ariolasoft GmbH, Postfach 1350, 4830 Gütersloh, Vertrieb Österreich: Karasoft, Vertrieb Schweiz: Thali AG

ario asoli

JUNIC PROMM

für Engagierte mit kleinem Geldbeutel.

ei diesem Hardwareprojekt handelt es sich keineswegs um einen weiteren Eprommer der absoluten Billigklasse oder der Nobelklasse, sondern vielmehr um ein extrem günstiges, aber auch sehr leistungsfähiges Gerät. Sie haben hier außerdem die Möglichkeit, die genaue Funktionsweise des Juniorprommers und damit die Funktionsweise von Eprommern schlechthin kennenzulernen.



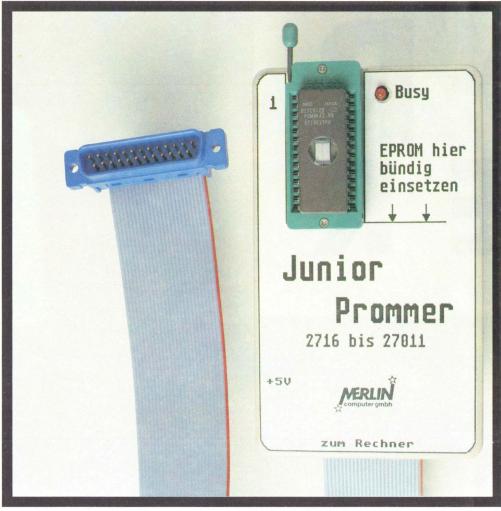


Der Juniorprommer ist weder der erste seiner Art, noch wird er der letzte sein, doch ist er wohl für viele, die eine Ausgabe von über 200.- DM scheuen, nur um mal so'n Käfer zu brennen, eine ernsthafte Alternative. Mit einem Materialaufwand von etwa 70.- DM und ein wenig Geduld können auch Sie ein Gerät besitzen, welches folgendes leistet:

- 1. Es programmiert alle Eproms vom Typ 2716 bis zum Typ 27011 und ist durch flexible Auslegung auch für zukünftige 28-polige Speicherbausteine geeignet.
- 2. Es ist vollkommen softwaregesteuert, wodurch ernsthafte Fehlbedienungen weitgehend ausgeschlossen werden.
- 3. Betrieb mit einer einzigen Spannung von 5V bei einem maximalen Strombedarf von 400 mA.
- 4. Es kann an jeder normgerechten Centronics-Schnittstelle (z.B. auch an der des Atari oder des IBM-PC) betrieben werden.
- 5. Es bietet die Möglichkeit, festzustellen, ob ein Speicherbaustein eingesetzt ist.
- 6.Es bearbeitet außerdem noch ROMs, EEPROMs und pinkompatible Speicherbausteine.

Was sind eigentlich EPROMs?

Nüchtern gesehen sind es schlicht Speicherbausteine, die zu jeder ihnen angelegten Eingangsadresse ein passendes und programmierbares Ausgangsdatenwort parat haben. Der Wertebereich der Eingangsadressen zusammen mit dem der Ausgangsdatenwörter ist ein Maß für die Speicherkapazität. Heute gebräuchliche Eproms sind die der 27XXX-Serie (Tabelle 1). XXX steht fast immer für die Speicherkapazität in Kilobit (Bit*1024). Oft eingesetzt als Programmspeicher in Microcomputern, hat das Eprom einen Nachteil und einen entscheidenden Vorteil:Programmiert werden muß ein Eprom mit einer recht aufwendigen und zeitraubenden Prozedur, wozu es natürlich Eprommer gibt. Jedoch bleiben die einmal programmierten Daten trotz fehlender Versorgungsspannung bis zum Löschen durch starkes ultraviolettes Licht erhalten.



Prototyp des Juniorprommer

Jetzt ist auch der Name verständlicher: EPROM ist eine Abkürzung für Erasable-Programmable-Read-Only-Memory. Zu Deutsch:

Löschbarer und programmierbarer Festwertspeicher.

Da ein EPROM mit UV-Licht gelöscht werden soll, ist es meist mit einem Quarzglasfenster ausgestattet, durch welches man den Halbleiterkristall mit UV-Licht bestrahlen kann, um alle Speicherzellen in einen definierten (gelöschten) Ausgangszustand zu bringen. Daher sollte es selbstverständlich sein, daß ein EPROM durch das Aufkleben eines UV-lichtdichten Aufklebers vor ungewolltem Löschen durch den im normalen Umgebungslicht enthaltenen UV-Anteil bewahrt wird.

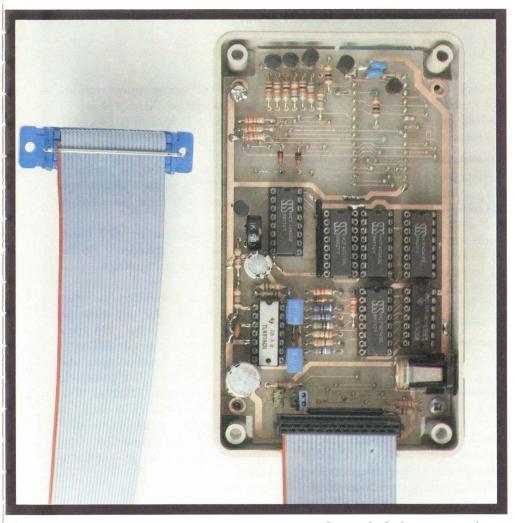
Intern werden die digitalen Informationen durch das Vorhandensein oder Nichtvorhandensein einer Ladung gespeichert. Das Aufbringen dieser Ladung geschieht auf elektrischem Wege, eben durch das Programmieren mit dem Juniorprommer, das Löschen der Ladung jedoch nur über UV-Licht.

Heute sind bereits EPROMs mit Kapazitäten bis zu einem Megabit (z.B. 27011) gängig. Da fast alle Eprom's eine Datenbreite von 8 Bit haben, muß

man nur die Zahl hinter der 27 durch 8 teilen, um die Kapazität in Kilobyte zu erhalten. Eine Ausnahme bilden hier 27011 und 27513, mit denen es jedoch etwas Besonderes auf sich hat. Beide sind sogenannte 'gebankte' EPROMs, was heißt, daß sie mangels Adreßleitungen in mehrere Abschnitte (Banks) aufgeteilt wurden, die dann mit einer speziellen Prozedur ausgewählt werden. Daher hat man zwar nicht mehr an Kapazität parallel verfügbar, jedoch kann man aus einem großen Bereich wählen. Der 27513 ist organisiert in 4 Bereiche zu je 16Kb, der 27011 in 8 Bereiche zu je 16Kb. Alle zukünftigen EPROMs höherer Kapazität im 28-poligen Gehäuse müssen zwangsläufig auf demselben Ansteuerprinzip beruhen.

Ansteuerung der EPROMs

Das Lesen der Daten aus einem EPROM ist trivial, und außerdem gleicht es hier fast allen anderen Speichern, die der Jedec-Norm folgen, wie z.B. ROMs, Statische RAMs etc. Sobald stabile Eingangssignale an den Adreßleitungen anliegen und sowohl /



Prototyp des Juniorprommer von innen

CE (/= Negation) als auch /OE aktiv sind, liegt auch, nach einer gewissen Verzögerung, genannt Zugriffszeit, das über die Adresse gewählte 8-Bit-Ausgangsdatenwort an. Die beiden Steuersignale /CE und /OE haben die Aufgabe, das EPROM zu selektieren und dessen Ausgangstreiber zu kontrollieren. /CE = 1 schaltet das ganze EPROM ab, unabhängig vom /OE Signal. /OE kontrolliert im Falle eines eingeschalteten EPROMs die Ausgangstreiber (low-Aktiv). Die restlichen Steuerleitungen haben nur in Verbindung mit der Programmierung eine Bedeutung.

Programmierung von EPROMs

Das Programmieren läuft generell bei allen EPROMs ähnlich ab. Nachdem Adressen, Daten und eventuell auch die Programmierspannung stabil anliegen, wird für eine festgelegte Zeit der Programmierimpuls gegeben. Danach legt man neue Daten und Adressen an und wiederholt den Programmierimpuls, bis alle Adressen programmiert sind. Wie das genau vor sich geht, hängt vom Typ und manchmal sogar

vom Hersteller ab. Der Programmieralgorithmus bestimmt, welcher Weg eingeschlagen wird, um die Daten dauerhaft zu programmieren. Hier gibt es grob zusammengefaßt drei Möglichkeiten:

- 1. Standardalgorithmus: Das Programmieren mit einer konstanten Zeit (50ms), nach der angenommen wird, daß die Daten sicher gespeichert sind (nicht sehr effektiv).
- 2. Bei 'intelligenten Algorithmen' arbeitet man mit wesentlich kürzeren Zeiten (1mS), nach denen jeweils geprüft wird, ob die Daten bereits akzeptiert wurden. Danach wird nochmal mit einem vielfachen (3 bis 4) der davor benötigten Zeit darüberprogrammiert, um eine sichere Programmierung aller Zellen zu gewährleisten. Die erreichbaren Zeiten liegen etwa um ein zehnfaches unter denen der Methode 1, jedoch wird diese Methode erst ab dem Typ 2764 unterstützt.
- 3. Beim sogenannten Quickpulsealgorithmus ist der Vorgang fast identisch mit dem von Punkt 2, jedoch arbeitet man mit noch kürzeren Programmierimpulsen (0.1ms) und übergeht das Nachprogrammieren. Dies ist zweifelsohne die schnellste Methode, doch

wird sie nur ab dem Typ 27512 unter speziellen Bedingungen empfohlen.

Der Standardalgorithmus kann generell auf jedes EPROM angewendet werden. In Abbildung 3 und 4 kann man sich einen Überblick über alle Speichertypen verschaffen.

Weiterhin kann man eine EPROM-Speicherzelle zwar elektrisch programmieren, jedoch kann man ein Bit elektrisch nur auf 0 bringen und nur durch UV-Licht wieder in den 'gelöschten' 1-Zustand.

die Implementierung Um EPROMs in bestehende Systeme zu vereinfachen, geben Hersteller in ihren Datenblättern meistens eine Tabelle an, aus der entnommen werden kann, welche Signale man für jeden spezifischen Betriebszustand an die Steuerleitungen des EPROMs legen muß. Falls Sie daran interessiert sind, sollten Sie auf jeden Fall die Datenblätter der Hersteller zu Rate ziehen. Deshalb soll die Programmierung hier nur an dem konkreten Beispiel des Typs 2764 erläutert werden.

Beispiel Typ 2764

Zuerst muß das EPROM gelöscht sein, d.h., alle Bits haben den 1-Zustand, bevor man sie eventuell auf 0 programmieren kann. Nachdem die Versorgungsspannung eingeschwungen ist und alle Steuersignale desaktiviert sind, wird die Programmierspannung von hier 21V an den Pin 1 (Vpp) gelegt, um das EPROM in den Programmiermodus zu bringen. Jetzt wird der Pin /CE auf 0 gesetzt, um das EPROM zu aktivieren. Danach werden die erste zu programmierende Adresse und das Datenwort angelegt, um danach den Pin /PGM für eine Millisekunde auf 0 zu pulsen. Um die Richtigkeit der programmierten Daten zu prüfen, werden die Ausgangstreiber über /OE aktiviert und mit dem programmierten Datenwort verglichen. War dieser Schritt nicht erfolgreich, so kann er bis zu 15mal wiederholt werden, bevor ein Fehler gemeldet wird. Aufgrund der verlangten Datensicherheit wird noch mit der vierfachen der bisher benötigten Zeit nachprogrammiert, bevor zur nächsten Speicherstelle übergegangen wird. Zuletzt überprüft man das vollständige EPROM noch einmal auf die Richtigkeit der Daten, bevor man es

über /CE inaktiviert und die Programmierspannung abschaltet.

Der Juniorprommer

Der Juniorprommer ist konzipiert für den Betrieb an jeder Standard-Centronics-Schnittstelle. Er stellt alle nötigen Programmierspannungen und Digitalleitungen zur Verfügung. Um die zehn Leitungen dieser Schnittstelle auf die nötige Anzahl zu erweitern, wird ein serielles Übertragungsprinzip angewandt. Ausgangsleitungen (insgesamt 24) werden mit einem Baustein 4094 realisiert, bei dem es sich um ein 8-Bit-"seriell ein / parallel aus"-synchrones Schieberegister mit nachgeschaltetem Register und schaltbaren Ausgängen handelt. Insgesamt drei dieser Schieberegister sind vorhanden, wobei alle Leitungen bis auf den TAKT parallel angesteuert werden, damit ein individueller Betrieb der Register möglich wird. Die Ausgänge von IC2 sind steuerbar, um einen bidirektionalen Datentransfer mit dem EPROM abwickeln zu können. Eingangsleitungen (insgesamt 8) werden mit dem Baustein IC1 / 4021 bedient, bei dem es sich um ein "parallel ein / seriell aus"-Schieberegister handelt, das wiederum mit separatem Takt betrieben wird. Um die Schnittstelle nicht modifizieren zu müssen, werden die seriellen Daten von IC1 über den Pin BUSY in den Rechner geleitet. Zusätzlich existiert noch ein Zähler IC6 / 4040, der die unteren zehn Adreßleitungen bedient und über die Schnittstelle inkrementiert sowie gelöscht werden kann.

Die Programmierspannungen werden an Bord mit IC7 / TL497, einem frequenzgesteuerten Sperrwandler, erzeugt und über IC5 eingestellt auf 5V,12.5V,21V bzw. 25V. Der Sperrwandler ist in der Lage, auch bei höchster Programmierspannung den nötigen Strom von etwa 30mA zu liefern. Mit T4 bis T6 kann diese Programmierspannung dann wahlweise auf die Pins 22, 23 und 1 geschaltet werden. Gleichzeitig können diese Pins auch über digitale Ausgänge der Schieberegister kontrolliert werden. T2 schaltet bei 24-poligen Speichern die Versorgungsspannung. Die Stromaufnahme des Juniorprommers ist ganz von den verwendeten EPROMs abhängig und liegt in der Regel unter 300 mA. Die

Versorgung des Juniorprommers kann der Rechner übernehmen, ist jedoch auch mit einem stabilisierten 5V-Netzteil über ST2 möglich.

Die Bedeutung der Steckbrücken

In J1 darf nur ein Adressierstecker gesteckt werden. Mit ihm wird gewählt, ob der Juniorprommer an AMIGA oder ST betrieben wird. Für J3 gilt entsprechend dasselbe. J2 ermöglicht bei einem Betrieb am Amiga die Stromversorgung von diesem aus.

Die Programmierung der Hardware

Bis auf die Pins 22, 23 und 1, auf die wahlweise eine Programmierspannung von 5V, 12.5V, 21V oder 25V geschaltet werden kann, werden alle Pins digital angesteuert. Eine Sonderstellung hat Pin 26, der zusätzlich den nötigen Strom liefert, um 24-polige EPROMs zu versorgen.

Wie Sie schon wissen, wird der Datentransport zwischen Rechner und Juniorprommer seriell über zehn Leitungen einer Centronics- Schnittstelle abgewickelt. In der Hauptsache besteht der Juniorprommer daher aus Schieberegistern und einem Zähler:

- 1. Ausgangsschieberegister SRA(IC5): Es steuert die Programmierspannung und deren Verteilung.
- 2. Ausgangsschieberegister SRB (IC4): Dessen Aufgabe ist es, die Steuerleitungen und oberen Adressen des EPROMs zu kontrollieren.
- 3. Ausgangsschieberegister SRC (IC2): In dieses Register werden die ins EPROM zu programmierenden Daten geschoben.
- Eingangsschieberegister SRD (IC1): Aus ihm werden die Zustände der Datenleitungen des EPROMs geschoben.
- Adreßzähler AZ (IC6): Er bedient die unteren Adressleitungen.
 Die benutzten Leitungen und deren Bedeutung sollen jetzt aufgezählt

/STROBE, Pin 1:

werden:

Normalerweise wird mit diesem Signal dem Drucker die Gültigkeit der Daten angezeigt. Ähnlich auch hier: Wird STROBE auf 1 gesetzt, werden alle Register im Juniorprommer geladen. Der Ruhezustand dieser Leitung ist jedoch nicht 1, sondern 0. Das heißt, der Inhalt in den Schieberegistern der 4094 wird zu den Ausgängen übertragen, und in IC1 wird das Datenwort des EPROMs parallel geladen, um danach eventuell herausgeschoben zu werden.

Bei der Amiga-Version wird dieses Signal von der Datenleitung D6 bedient (über Jumper J1 gesteckt).

D0. Pin 2:

Dieses Signal kontrolliert die Ausgangstreiber von SRC und damit die Datenrichtung zu oder von dem EPROM. Ist D0 = 1, so gibt SRC seine Daten an die Datenleitungen des EPROMs. Falls D0 = 0 ist, können die Datenleitungen des EPROMs über SRD gelesen werden. Es ist darauf zu achten, daß kein gleichzeitiges Aktivieren der Ausgangstreiber des EPROMs und der des SRC auftreten kann.

D1. Pin 3:

Dies ist die Taktleitung von IC5 (SRA), der die Programmierspannungen und deren Verteilung steuert. Bei der ansteigenden Flanke von D0 wird das Bit an D7 in das Schieberegister des SRA geschoben.

D2, Pin 4:

Dies ist ebenfalls eine Taktleitung, jedoch für IC4(SRB), der die Steuersignale des EPROMs kontrolliert.

D3, Pin 5:

Die Taktleitung für IC2(SRC), der die Datenleitungen des EPROMs kontrolliert

D4,Pin 6:

Die Taktleitung für den Adreßzähler (AZ). Die abfallende Flanke von D4 erhöht den AZ um eins. Der AZ bedient lediglich die Adressen A0 bis A9, um durch die Softwarekontrolle der übrigen Adressen eine höhere Flexibilität zu erreichen.

D5, Pin 7:

Die Taktleitung für IC1, über den die Datenleitungen des EPROMs gelesen werden können. Die jeweils ansteigende Flanke des Taktes bringt ein weiteres, invertiertes Bit aus dem Schieberegister IC1 an den Pin BUSY.

D6, Pin 8:

Rücksetzleitung des Adreßzählers IC6. Solange diese Leitung 1 ist, wird der Adreßzähler zurückgesetzt. Bei der Amiga-Version übernimmt dieser Pin die Funktion des Strobe-Signals. Die Adreßzählerrücksetzung erfolgt über Q3 von IC5 (über Jumper J3 gesteckt). **D7, Pin 9:**

Die gemeinsame Datenleitung aller Ausgangsschieberegister. Über diese Leitung wird der serielle Datenstrom zu den jeweils getakteten Ausgangsschieberegistern übertragen.

BUSY, Pin 11:

Über diesen Pin können durch IC1 die Datenleitungen des EPROMs seriell gelesen werden.

+5V, Pin 23:

Über J2 kann die Spannungsversorgung über das Centronicskabel erfolgen. Der Amiga 1000 liefert die Spannung an Pin 23. Die zur Verfügung gestellte Stromstärke erlaubt leider nur ein bedingtes Testen des Juniorprommers und ist ein Relikt aus der Entwicklungsphase. Während des normalen Betriebs sollte der Prommer über die Buchse ST2 mit Spannung versorgt werden. Die Joystickbuchse des Rechners liefert hierbei den benötigten Strom.

ACHTUNG: Der Juniorprommer darf mit gesteckter Brücke J2 nicht an einen Amiga 500 oder 2000 angeschlossen werden.

In der Praxis

Die Ansteuerung der Ausgangsschieberegister sollte mit einem Maschinenprogramm erfolgen, das den Centronics-Port direkt bedient. Das Schieben eines Datenwortes in eines der Ausgangsschieberegister folgt diesem Ablauf:

- 1. Das höchstwertigste Bit des Datenbytes auf D7 legen.
- 2. Anlegen des Taktimpulses auf die jeweilige Taktleitung.
- 3. Einmaliges Linksschieben des Datenbytes.
- 4. Wiederholen ab 1, bis das Byte vollständig herausgeschoben wurde.
 - 5. Senden des positiven STROBE-Impulses, damit die Daten des jeweiligen Registers zu den Ausgängen transferiert werden.

Die Ansteuerung des Eingangsschieberegisters SRD differiert leicht:

- Zuerst muß SRC über D0 inaktiviert werden und das EPROM aktiviert werden.
- 2. Senden des positiven Strobeimpul-

- ses, damit die Zustände der Datenleitungen in SRD übertragen werden.
- Setzen des invertierten BUSY-Signals in das niederwertigste Bit des Datenbytes.
- 4. Linksschieben des Datenbytes.
- 5. Wiederholen ab 2, bis das vollständige Byte übertragen ist.

Die Bedeutung der Schieberegisterbits SRA:

Bit 0 (Q1): Steuert den Pin 1 des EPROMs digital an. Falls an Pin 1 schon eine Programmierspannung geschaltet ist, bleibt diese Leitung irrelevant. Zu beachten ist, daß, bedingt durch C5, etwa 6 Millisekunden vergehen, bis der eindeutige Pegel am Sokkel anliegt.

Bit 1 (Q2): Hat zwei Funktionen. Zum einen steuert diese Leitung schlicht die Leuchtdiode D4 (1 = ein), und zum anderen kann man über sie die Treiberfähigkeit des EPROMs testen. Deswegen ist die Datenleitung D7 des

gestellt werden kann, ob kein EPROM eingesetzt ist oder ob das EPROM einen Defekt hat. Während des normalen Eprommerbetriebs sollte diese Leitung 1 sein.

Bit 2 (Q3): Dient in der Amiga-Version zum Rücksetzen des Adreßzählers (in Abhängigkeit von J3).

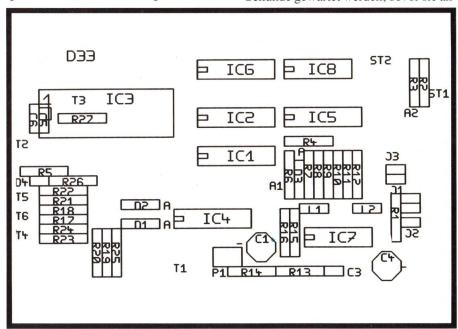
Bit 3 (Q4): Schaltet die Programmierspannung auf Pin 23 des EPROMs (1 = Spannung ein).

Bit 4 (Q5): Schaltet die Programmierspannung auf Pin 1 des EPROMs (1 = Spannung ein)

Bits 5-7: Geben die Höhe der Programmierspannung nach folgender Zuordnung an:

000 = 5V 100 = 12.5V 010 = 21V 001 = 25V

Falls die Programmierspannung umgeschaltet wird, muß danach etwa eine Sekunde gewartet werden, bevor sie an



Der Bestückungsplan erfordert eine ruhige Hand, denn es ist eng geworden.

EPROMs nicht mit einem 'Pull-up' verbunden, sondern über einen hochohmigen Wiederstand mit dieser Leitung. Das hat einen ganz bestimmten Hintergrund: Ist ein EPROM eingesetzt und nicht in seiner Treiberfähigkeit gestört, so ist der Zustand dieser Leitung nicht entscheidend für die gelesenen Daten. Ist jedoch kein EPROM eingesetzt oder dessen Treiberfähigkeit gestört, folgt das Bit 7 aus den gelesenen Daten dieser Leitung, woran fest-

einen Pin des EPROMs gelegt wird, da der Schaltregler sie sonst noch nicht ausreichend eingeregelt hat. Das gilt vor allem, wenn die Programmierspannung auf einen niedrigeren Wert geschaltet wird.

SRB:

Bit 0 (Q1): Steuert direkt Pin 20 (/CE) des EPROMs.

Bit 1 (Q2): Steuert digital Pin 22 (/ OE,Vpp) des EPROMs an. Falls an diesem Pin schon Programmierspan-

22011	27513	2.7512	27256	27128	2764	2232	2716					2716	2732	2264	27128	27256	2.7512	2/913	2.2701.1
Upp. RST	/RST	A15	Upp	Upp	Upp			d.			28	•		Vee	Vec	Vec	Uee	Uee	Uec
A12	A12	A12	A12	B12	fi12			2	28Þ	01	27			∠PGM	∠PGM	814	A14	7 bH	PGM. WE
A2	A2	A.?	fiZ	8.7	82	82	82	Q 3 1		24	26]	Vec	Uee	A13	A13	A13	A13	H1 3	H13
R6	A6	A6	A6	A6	A6	A6	A6	Q 4 2	t t	23	25	A8	A8	AB	A8	A8	A8	A8	A3
A5	A5	A5	A5	A5	A5	A5	A5	d 5	0	22	24 🕽	A9	A9	A9	A9	A9	A9	fig.	0.3
84	84	A4	A4	A4	A4	A4	84	46 4	7	21	23	Upp	ALL	A11	ALL	ALL	AH	ĤI I	1111
A3	A3	A3	A3	A3	A3	A3	A3	d2 9	,	20	22]	/OE	'DE ,Upp	∠DE	∠ OE	/DE	"OE, Upp	OE,Upp	- OE
A2	A2	A2	A2	A2	A2	A2	A2	18 6		19	21 1	A10	AIØ	A10	A10	A10	A10	H10	ATA
A1	A1	A1	A1	A1	A1	A1	A1	9 7	,	18	200	∠CE	∠/CE	ZCE .	ZCE.	∠CE	ZCE	ZCE	CE
AØ	AØ	AØ	AØ	AØ	AØ	AØ	AØ	0108	}	12	19	D.2	02	02	02	02	D2	D2	0.2
DØ	DØ	שם	DØ.	DØ	DØ	00	00	411 9)	16	18	D6	D6	D6	D6	D6	D6	D6	D6
D1	D1	DI	D1	D1	D1.	DI	D1	121	Ø	15	170	D5	05	D5	05	D5	D5	05	05
D2	D2	D2	D2	D2	D2	02	02	4 13 1	1	14	160	D4	D4	D4	D4	D4	D4	D4	DH
GND	GND	GND	GND	GND	GND	GND	GND	0141	2	13	15	D3	D3	03	D3	D3	03	D3	0.3

Alle EPROM-Typen, die der Miniprommer bearbeiten kann, sind im Vergleich dargestellt. Einen Sonderstatus besitzen die Pins 1, 22, 23, 26, und 27, welche je nach Typ eine andere Funktion erfüllen und deshalb mit einem höherem Grad an Flexibilität ausgestattet sind

nung anliegt, ist dieses Bit irrelevant. Bit 2 (Q3): Schaltet die Programmier-

spannung auf Pin 22 des EPROMs (1 = Spannung ein).

Bit 3 (Q4): Steuert Pin 21 (A10) des EPROMs an.

Bit 4 (Q5): Steuert digital Pin 23 (A11,Vpp) des EPROMs an. Wie immer hat die Programmierspannung Vorrang.

Bit 5 (Q6): Steuert Pin 2 (A12) des EPROMs an.

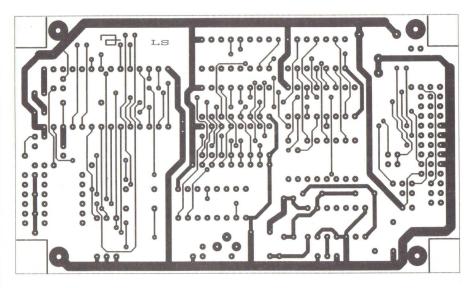
Bit 6 (Q7): Steuert invertiert Pin 26 (A13,Vcc) des EPROMs an. Diese Leitung liefert den entsprechenden Strom, um gleichzeitig die Stromversorgung von 24-poligen EPROMs zu übernehmen.

Bit 7 (Q8): Steuert den Pin 27 (A14,/ PGM) des EPROMs direkt an.

SRC und ebenfalls SRD sind direkt mit den entsprechenden Datenleitungen des EPROMs verbunden (siehe Schaltplan).

Der Aufbau

Da das Layout des Juniorprommers im Hinblick auf ein Gehäuse ausgelegt wurde, müssen der Nullkraftsockel (IC3) sowie die LED (D4) zuletzt auf die Lötseite montiert werden. Dadurch ist die Bauhöhe der anderen Bauteile, die sich ausschließlich auf der Bauteilseite befinden, unkritisch. Die Verbindung des Juniorprommers mit der jeweiligen Schnittstelle des Rechners geschieht über ein passend geschneidertes Flachbandkabel. Am einen Ende des weniger als 50 cm langen, 26-poligen Flachbandkabels wird ein 25-poliger Sub-D-Stecker quetscht, am anderen Ende ein 26-poliger Quetschverbinder. Natürlich muß Pin 1 des Sub-D-Steckers auch mit Pin 1 des Quetschverbinders zusammenliegen. Die 26. Ader des Flachbandkabels wird vor dem Sub-D-Stecker ab-

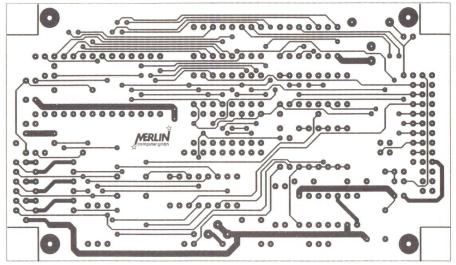


getrennt. Die Spulen L1 und L2 sind z.B. von Conrad Electronic zu beziehen. Da das Layout stellenweise recht eng geraten ist, sollte man schon ein bißchen Lötfertigkeit besitzen.

Vorschau

Im nächsten Artikel dreht sich wieder alles um die Software. Es wird ein Assemblerprogramm beschrieben, mit dem die wichtigsten Operationen mit den gängigsten Typen möglich sind. Zudem werden alle Informationen gegeben, um eigene Software zu schreiben. Außerdem ist ein fertiges Steuerprogramm für den Juniorprommer verfügbar.

Diejenigen, die nicht bis zum nächsten Heft warten möchten, können den Juniorprommer schon als Fertiggerät beziehen. Für Bastler ist auch die Platine mit Software erhältlich.



Bezugsquelle: MERLIN-Computer GmbH Industriestr. 26 6236 Eschborn

Juniorprommer Stückliste

Halbleiter

D1-D3 : 1N4148 T1,T3 : BC547B T2 : BC328-25 T4-T6 : BC557B IC1 : 4021

IC2, IC4, IC5 : 4094

IC3 : 28 poliger Nullkraftsockel

IC6 : 4040 IC7 : TL497 IC8 : 7406

Widerstände

alle 1/4 W Kohleschicht

R1,R13 : 1K8 R2 : 47 Ω R3,R5,R27 : 1K

R4,R19,R20,R25,R6 : 22K

R7 : 910 Ω 1% Metallschicht

R11 : 330 Ω 1% Metallschicht

R12: 3K9 1% Metallschicht

R14 : 5K1 R15,R16 : 1\Omega_5 R17,R21,R23 : 10K R18,R22 : 6K8

R26 : 180Ω

Kondenstoren

C1 : 220µ / 16V Elko, stehend

C3 : 100p keramisch

C4 : 100μ / 35V Elko, stehend C5,C6 : 100n vielschicht

Sonstiges

ST1 : Doppelpfostenleiste, 26 polig

ST2: Stromversorgungsbuchse, Printmontage

TR1 : Trimmer klein, stehend 1K

D4 : LED 3mm, rot

A1 : Single-Inline-Array 8*22K A2 : Single-Inline-Array 8*1K8 J1-J3 : Je nach Rechnertyp

L1, L2 : Miniaturdrossel 100 µ H

(Conrad Electronic)

Zur Not: Eine Netzentstördrossel

mit 50µH.

Evtl. IC-Fassungen: 5*16pol, 4*14pol

Verbindungskabel:

1 25 poliger Sub-D Quetschstecker

1 26 poliger, doppelreihiger

Pfostenverbinder, Quetschausführung max. 50cm 26 adriges Flachbandkabel

1 Gehäuse 123L*79B*27T

.Gängige Speichertypen der Jedec-Norm.

Тур Art Kapazi. P.Span. P.Algo. EPROM 2Kb EPROM 4Kb 25V 25V 2716 2732 50mS 2732A EPROM 4Kb 21V 50mS EPROM 8Kb 21V 1ms, 4x 2764 2764A EPROM 12.5V 1ms, 3x 8Kb 27128 EPROM 16Kb 21V 1ms, 4x 27128A EPROM 16Kb 12.5V 1ms, 3x 27256 EPROM 32Kb 12.5V lms, 3x 27512 EPROM 64Kb 12.5V 1ms, 3x/ 0.1ms,0x 27513 EPROM 16Kb*4 12.5V lms, 3x/ 0.1ms,0x 27011 EPROM 16Kb*8 12.5V lms, 3x/ 0.1ms,0x 2804 EEPROM 0.5Kb 5V 10mS 2816 EEPROM 2Kb 5V 10mS 5V 10mS 2864 EEPROM 8Kb 28256 EEPROM 32Kb 4716 ROM 2Kb 4732 ROM 4Kb 4764 ROM 8Kb 47128 ROM 16Kb

ROM 32Kb

RAM 32Kb

RAM

RAM

2Kb

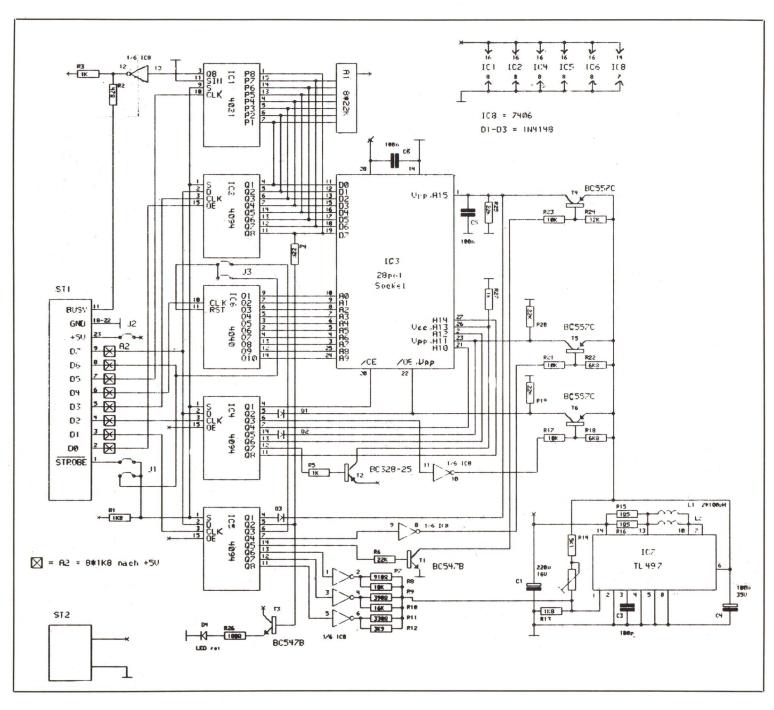
8Kb

47256

6116

62256

6264



Achtung ★ Achtung ★ Achtung Amiga-Programmierer

Wir suchen Amiga-Programme aus allen Bereichen (Grafik, Anwendung, Spiele usw.).

Assembler- und C-Programme bevorzugt!

Umsatzbeteiligung bis zu 50 % möglich!

Sollten Sie Interesse haben, wenden Sie sich noch

heute an:



Software 2000 Lange Straße 51, 2320 Plön / Holstein Telefon 0 45 22 / 13 79





In der letzten Ausgabe haben Sie einige Routinen aus der Graphics Library kennengelernt. Dabei wurde oft als Parameter ein Zeiger auf die Rastportstruktur angegeben. Was ist denn nun diese Rastportstruktur? Nun, sie ist die wichtigste Struktur in der Graphics Library. Hier werden Daten gespeichert, die für die Zeichenroutinen wichtig sind (z.b Zeichenmodus, der Zeichenfont, u.s.w).

n Tabelle 1 ist die Rastportstruktur beschrieben. Die Basisadresse dieser Struktur erhalten Sie mit dem Amiga Basic-Befehl WINDOW (8). Der Offset beschreibt den Wert, den Sie zu dieser Basisadresse dazuaddieren müssen, um die gewünschte Adresse zu erreichen. Byte beschreibt die Länge des entsprechenden Wertes, also B für 1-Byte Länge, W für 2-Byte Länge und L für 4-Byte Länge. Wenn Sie zum Beispiel erreichen wollen, daß der Abstand zwischen den Textzeichen größer werden soll, so schreiben Sie:

POKEW WINDOW (8)+64, Wert

Für 'Wert' setzen Sie den gewünschten Abstand. Weitere wichtige Strukturen sind die View-, die Viewport- und die Rasinfo-Struktur. In Tabelle 2 sind diese Strukturen beschrieben. Wir werden einige Parameter später noch genauer erklären. In der Graphics Library gibt es Routinen, die es uns erlauben, diese Strukturen einzurichten und dem System zugänglich zu machen. Meistens

werden zum Erstellen von Screens und Windows die entsprechenden Routinen in der Intuition Library benutzt. In Amiga Basic geschieht dies mit den Befehlen SCREEN und WINDOW. Dabei werden automatisch die entsprechenden Strukturen (Rastport, Viewport, Bitmap u.s.w) eingerichtet. Wir möchten Ihnen nun die Routinen zum Anlegen dieser Strukturen näher erklären, weil Sie damit einige Möglichkeiten zur Grafikdarstellung in der Hand haben. Beschreiben wollen wir die einzelnen Routinen anhand des Programms Viewdemo (siehe Listing). Zuerst legen wir uns mit der Routine AllocMem aus der Exec Library (siehe Teil 2) genug Speicher für die Strukturen an. Dann werden die Variablen für die einzelnen Strukturen gesetzt. Nun wird die Bitmap initialisiert:

CALL

InitBitmap(bitmap, tiefe, breite,
höhe)

Den Befehl CALL habe ich in dem Programm nicht benutzt.

bitmap = der Zeiger auf die Bitmap Struktur

tiefe = die Anzahl der Bitplanes (bestimmt die Anzahl der Farben)

breite = die Breite der einzelnen Bitplanes

höhe = die Höhe der einzelnen Bitplanes

Sie können eine Bitmap in der Größe von maximal 1024 x 1024 Pixeln einrichten. Von dieser riesigen Bitmap können Sie nun den sichtbaren Bereich in alle Richtungen scrollen (zu dieser fantastischen Möglichkeit aber später mehr). Bedenken Sie aber, daß eine solche Bitmap eine ganze Menge an Speicherplatz benötigt. Mit 'tiefe' bestimmen Sie die Anzahl der Farben. Wenn Sie nur zwei Farben brauchen, so reicht eine Bitplane. Mit zwei Bitplanes können Sie 4 Farben, mit 3 Bitplanes 8 und mit 4 Bitplanes 16 Farben darstellen. 5 Bitplanes können Sie nur anlegen, wenn Sie die niedrige Auflösung von 320 Pixeln Breite benutzen. Dann sind 32 Farben möglich. 6 Bitplanes müssen Sie nehmen, wenn Sie den HALFBRITE- oder HAM-Modus benutzen wollen.

Als nächstes werden nun soviele Bitplanes eingerichtet, wie in der Variablen 'tiefe&' angegeben ist. Dazu dient die Routine:

bitplane&=AllocRaster&
(breite, hoehe)

'breite' und 'hoehe' kennen Sie von der Routine InitBitmap.

Nach Aufruf der Routine steht in der Variablen bitplane& der Zeiger auf den Beginn der Bitplane. Diesen Zeiger setzen wir in die Bitmapstruktur ein. Am Schluß des Programms wird der Speicherbereich, den diese Bitplane benötigt, dem System wieder zur Verfügung gestellt mit:

CALL FreeRaster (bitplane, breite, hoehe)

Als nächstes geht es an das Anlegen einer Rastportstruktur.

CALL InitRastport (rastport)

In 'rastport' steht der Zeiger auf den angelegten Speicherbereich.

Diese Routine belegt die Rastportstruktur mit Standardwerten. Sie brauchen dann nur noch die Werte einzusetzen, die Sie haben wollen. In unserem Programm lassen wir die Standardwerte und setzen nur den Zeiger auf die Bitmapstruktur ein. Diesen Zeiger müssen wir ebenfalls in die Rasinfostruktur einsetzen. Gleichzeitig schreiben wir die Werte 'rxoffset' und 'ryoffset' in die Rasinfostruktur. Diese Werte beschreiben den Beginn des auf dem Bildschirm sichtbaren Bereichs einer großen Bitmap. Nun wird die Viewportstruktur initialisiert:

CALL InitVPort (viewport)

'viewport' ist der Zeiger auf den angelegten Speicherbereich.

Dann werden die entsprechenden Werte in die Viewportstruktur mittels des Befehls POKE eingesetzt. Diese Werte sind: die Breite des Viewports, die Höhe des Viewports, der Zeiger auf die Rasinfostruktur, und der darzustellende Modus (in Tab. 2 sehen Sie die möglichen Darstellungsarten).

In 'DxOffset' und 'DyOffset' steht der Beginn des Viewports auf dem Screen. Wenn Sie mehrere Viewports darstellen wollen, so müssen Sie außerdem noch den Zeiger auf den nächsten Viewport einsetzen. In Bild 1 sehen Sie noch einmal die Bedeutung der Variablen Rx- und RyOffset aus der Rasinfo Struktur und Rx- und RyOffset, Breite und Höhe aus der Viewportstruktur am Beispiel von zwei Viewports.

CALL Initview (view)

Damit wird die Viewstruktur initialisiert. In 'view' steht der Zeiger auf die Viewstruktur.

Anschließend müssen Sie noch den Zeiger auf die Viewportstruktur in die Viewstruktur einsetzen.

colormap& = GetColorMap& (anzahl)

Damit wird Speicher für eine Colormap angelegt. In 'anzahl' schreiben Sie die Anzahl der gewünschten Farben.

Zum Setzen der Farben gibt es die Routine

CALL LoadView (view)

Damit wird die erzeugte Copperlist aktiviert .Das Programm benutzt übrigens den Dual-Playfield-Modus. Aus diesem Grunde werden jeweils zwei Rastport-, Rasinfo- und Bitmapstrukturen eingerichtet. Bei diesem Modus sind die Farbnummern 0-7 für Playfield 1 und die Nummern 8-15 für Playfield 2 vorgesehen. Die Farbnummern 0 und 8 entsprechen der Hintergrundfarbe und sind transparent. Ist an der jeweiligen Bitplane diese Farbnummer gesetzt, so kann das andere Playfield dort gesehen werden. Wenn Sie in der Viewportstruktur bei Modus den Wert \$40 einsetzen, so liegt Play-

Call SetRGB4CM (colormap, n, rot, grün, blau)

colormap = der Zeiger auf die Colormap

n = die gewünschte Farbnummer aus der Colormap

rot/grün/blau = der gewünschte Farbanteil der jeweiligen Farbe (0-15)

Zum Löschen der Color Map dient folgende Routine:

CALL FreeColorMap(colormap)

Jetzt sind alle Strukturen angelegt. Machen wir uns also an die Arbeit, unsere Strukturen darzustellen. Dazu müssen wir eine sogenannte Copperlist einrichten. Eine Copperlist ist eine Reihe von Maschinenroutinen, die 60 x in der Sekunde durchlaufen wird. In diesen Maschinenroutinen stehen Befehle, die die Hardware-Register des Amiga beschreiben. Aber keine Angst, sie brauchen nicht in Maschinensprache zu programmieren. In der Graphics Library gibt es Befehle, die uns das Anlegen dieser Copperlist abnehmen.

CALL MakeVPort (view, viewport)

legt die Copperlist des Viewports an. In 'view' steht der Zeiger auf die Viewstruktur, und in viewport der Zeiger auf die Viewportstruktur.

CALL MrgCop (view)

Diese Routine bindet verschiedene Copperlists zusammen.

field 2 vor Playfield 1, ansonsten ist es umgekehrt. Mit diesem Modus läßt sich leicht ein 3dimensionaler Effekt erzeugen, der vor allem für Spiele genutzt wird. Jedes Playfield kann nun unabhängig vom anderen benutzt werden. Wenn Sie in Playfield 1 etwas zeichnen wollen, so benutzen Sie die Rastportstruktur für dieses Playfield, für Playfield 2 die entsprechende Rastportstruktur für Playfield 2. Jedes Playfield kann auch unabhängig vom anderen gescrollt werden. Diese Möglichkeit wird im Programm ebenfalls demonstriert. Zum Scrollen brauchen Sie nur die Werte RxOffset und RvOffset in der Rasinfo-Struktur zu ändern und dann eine neue Copperlist zu erstellen (mit MakeVPort, MrgCop und LoadView). Weitere Möglichkeiten haben Sie dadurch, daß Sie mehrere Viewports darstellen können. Jeder Viewport kann eine andere Colormap benutzen und einen anderen Modus haben. Diese Viewports können aber nur untereinander dargestellt werden und dürfen sich nicht überlappen. Setzen Sie den Beginn des Viewports auf dem Bildschirm in die Variable Dy (siehe auch Tabelle 2). Sie können natürlich auch die vorhandene Viewportstruktur ändern. Den Beginn der Viewportstruktur ermitteln Sie mit Amiga Basic folgendermaßen:

viewport&=PEEKL (WIN-DOW(7)+46)+44

Wie wäre es zum Beispiel mit einer Workbench mit 16 Farben? Zum Ermitteln der Viewstruktur gibt es eine Routine in der Intuition Library:

view=ViewAdress&

Neben den Zeichenroutinen befinden sich in der Graphics Library die Routinen zum Darstellen und Bewegen von Sprites und Bobs. Diese Routinen werden von Amiga Basic weitgehend unterstützt, so daß wir hier nicht näher darauf eingehen wollen. Wir werden aber in der nächsten Folge noch ein paar Befehle zur Sprite-Animation kennenlernen.

Weiter geht es mit der Intuition Library. In dieser Bibliothek befinden sich Routinen, die die grafische Benutzeroberfläche des Amiga (eben Intuition) unterstützen. Vieles funktioniert bei Intuition ganz automatisch. Wenn Sie zum Beispiel ein Fenster vergrößern oder verkleinern, so paßt Intuition sofort die neue Fenstergröße an. Auch vorher verdeckter Speicherinhalt eines anderen Fensters wird wieder dargestellt. Wie am Anfang dieser Folge erwähnt, wird auch das Öffnen von Screens und Windows mit Intuitionroutinen erledigt. Wie Sie diese Routinen mit Amiga Basic benutzen können, ist in KICKSTART 12/1987 nachzulesen.

Zum Öffnen eines Screens gibt es die Routine:

screen= OpenScreen& (newScreen)

'newScreen' ist ein Zeiger auf eine NewScreen-Struktur, die das Aussehen des Screens beschreibt

Zum Öffnen eines Fensters gibt es den Befehl:

window=OpenWindow& (newWindow)

'newWindow' ist ein Zeiger auf eine New-Window-Struktur, die das Aussehen dieses Fensters beschreibt.

Welche Werte Sie in die NewWindow- und NewScreen-Struktur setzen müssen, können Sie wiederum in KICKSTART 12/1987 nachlesen. In

Die Rastportstruktur : Offset Byte Bezeichnung 0 T. Zeiger auf Layer Struktur Zeiger auf Bitmap Struktur 8 L Zeiger auf Area Pattern Struktur 12 Zeiger auf Tmp Ras Struktur L 16 Zeiger auf Area Info Struktur L L 20 Zeiger auf Gels Info Struktur 24 В Mask (gibt die Bitplanes an, in denen gez. wird) В Farbnummer für Vordergrundfarbe 25 26 В Farbnummer für Hintergrundfarbe 27 В Farbnummer für Area Outline Pen 28 В Zeichenmodus В Anzahl der Farbmusterelemente (f. PATTERN Befehl) 30 В nicht benutzt 31 B Linienmuster (wird vom System benutzt) 32 W Flags 34 W Linienmuster (f. PATTERN Befehl) 36 W momentane Cursorposition f. X-Koordinate 38 W momentane Cursorposition f. Y-Koordinate 8xB für Blitter Funktionen 48 W Pen Breite 50 Pen Höhe W Zeiger auf Font Struktur 52 L 56 В Style Bits 57 B Flags für Textstruktur 58 W Texthöhe Textbreite 62 W Text Basislinie 64 W Textzwischenraum 66 Zeiger auf Task der diese Rastportstruktur benutzt 70 30 Bytes reserviert für zukünftige Erweiterungen

Offset	Byte	Bezeichnung
0	L	Zeiger auf Viewportstruktur
4	L	Zeiger auf Long Form Copper List
8	L	Zeiger auf Short Form Copper List
12	W	Dy Offset
14	W	Dx Offset
16	W	Modus
		ruktur:
Offset	Byte	Bezeichnung
Offset 0		
Offset	Byte	Bezeichnung
Offset 0 keine)	Byte	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn
Offset 0 keine)	Byte L	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn Zeiger auf ColorMap Zeiger auf DspIns Struktur Zeiger auf SprIns Struktur
Offset 0 keine) 4 8 12 16	Byte L L L	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn Zeiger auf ColorMap Zeiger auf DspIns Struktur
Offset 0 keine) 4 8 12 16 20	Byte L L L L	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn Zeiger auf ColorMap Zeiger auf DspIns Struktur Zeiger auf SprIns Struktur
Offset 0 keine) 4 8 12 16 20 24	Byte L L L L L	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn Zeiger auf ColorMap Zeiger auf DspIns Struktur Zeiger auf SprIns Struktur Zeiger auf ClrIns Struktur Zeiger auf UCopIns Struktur Breite des Views
Offset 0 keine) 4 8 12 16 20 24 26	Byte L L L L L W W	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn Zeiger auf ColorMap Zeiger auf DspIns Struktur Zeiger auf SprIns Struktur Zeiger auf ClrIns Struktur Zeiger auf UCopIns Struktur Breite des Views Hoehe des Views
Offset 0 keine) 4 8 12 16 20 24 26 28	Byte L L L L L W W W	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn Zeiger auf ColorMap Zeiger auf DspIns Struktur Zeiger auf SprIns Struktur Zeiger auf ClrIns Struktur Zeiger auf UCopIns Struktur Breite des Views Hoehe des Views Dx Offset
Offset 0 keine) 4 8 112 16 20 24 26 28 30	Byte L L L L L W W W W	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn Zeiger auf ColorMap Zeiger auf DspIns Struktur Zeiger auf SprIns Struktur Zeiger auf ClrIns Struktur Zeiger auf UCopIns Struktur Breite des Views Hoehe des Views Dx Offset Dy Offset
Offset 0 keine) 4 8 12 16 20 24 26 28 30 32	Byte L L L L L W W W W W	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn Zeiger auf ColorMap Zeiger auf DspIns Struktur Zeiger auf SprIns Struktur Zeiger auf ClrIns Struktur Zeiger auf UCopIns Struktur Breite des Views Hoehe des Views Dx Offset Dy Offset Modus
Offset 0 keine) 4 8 12 16 20 24 26 28 30	Byte L L L L L W W W W	Bezeichnung Zeiger auf nächste Viewportstruktur (0 wenn Zeiger auf ColorMap Zeiger auf DspIns Struktur Zeiger auf SprIns Struktur Zeiger auf ClrIns Struktur Zeiger auf UCopIns Struktur Breite des Views Hoehe des Views Dx Offset Dy Offset

Offset	Byte	Bezeichnung
0	L	Zeiger auf nächste RasInfo Struktur
4	L	Zeiger auf Bitmap Struktur
8	W	Rx Offset
10	W	Ry Offset
Die We	rte für l	
	rte für	Modus :
4	rte für i —— = LACE	Modus : Interlace Modus
4 \$40	rte für l = LACE = PFBA	Modus :
4 \$40	rte für = LACE = PFBA = HALFB	Modus : Interlace Modus setzt Playfield 2 nach vorne
4 \$40 \$80 \$400	= LACE = PFBA = HALFBI = DUALP	Modus : Interlace Modus setzt Playfield 2 nach vorne RITE (64 Farben)
4 \$40 \$80 \$400 \$800	TTE für	Modus : Interlace Modus setzt Playfield 2 nach vorne RITE (64 Farben) F Dual Playfield Modus

Offset	Byte	Bezeichnung
0	L	Zeiger auf nächstes Window
4	W	Beginn der linken Ecke des Fensters
6	W	Beginn der oberen Ecke des Fensters
8	W	Breite des Fensters
10	W	Höhe des Fensters
12	W	Y-Koordinate des Mauszeigers
14	W	X-Koordinate des Mauszeigers
16	W	Minimale Breite des Fensters
18	W	Minimale Höhe des Fensters
20	W	Maximale Breite des Fensters
22	W	Maximale Höhe des Fensters
24	L	Flags
28	L	Zeiger auf erste Menuestruktur
32	L	Zeiger auf Fenstertitel
36	L	Zeiger auf erstes Requester
40	L	Zeiger auf erstes Double Klick Requester
44	W	Anzahl der aktiven Requester
46	L	Zeiger auf Screen Struktur
50	L	Zeiger auf Rastport Struktur
54	4xB	Beschreibt Aussehen des Rahmens
58	L	Zeiger auf die Rastportstruktur des Rahmens
62	L	Zeiger auf erstes Gadget
66	L	Zeiger auf 'Parent' Window Struktur
70	L	Zeiger auf 'Descendant' Window Struktur
74	L	Zeiger auf Sprite Datas des Mauszeigers
78	В	Höhe des Mauszeigers
79	В	Breite des Mauszeigers
80	В	X-Offset des 'Hot Spot' des Mauszeigers
81	В	Y-Offset des 'Hot Spot' des Mauszeigers
82	L	IDCMP Flags
86	L	Zeiger auf User Port
90	L	Zeiger auf Window Port
94	L	Zeiger auf Message Key
96	В	Farbnummer für den Text in der Titelzeile
97	В	Farbnummer für Hintergrund Titeltext
98	L	Zeiger auf Imagestruktur f. abgehakte Menuepunk
102	L W	Zeiger auf Screentitel dieses Windows
108		X-Koord, des Mauszeigers bei Gimmezerozerowinde
	W	Y-Koord, des Mauszeigers bei Gimmezerozerowindo
110	W	Breite des Fensters bei Gimmezerozerowindow Höhe des Fensters bei Gimmezerozerowindow
116	W L	
		Zeiger auf User Daten
120	L L	Zeiger auf User Daten Zeiger auf Layer Struktur

den Variablen 'window' und 'screen' stehen nach Aufruf dieser Routinen jeweils ein Zeiger auf eine Windowbzw. Screen-Struktur. Wie diese Strukturen aussehen, können Sie den Tabellen 3 und 4 entnehmen. Amiga Basic stellt uns nach dem Öffnen eines Fensters mit dem Befehl WINDOW den Zeiger auf diese Window-Struktur mit dem Befehl WINDOW(7) zur Verfügung. Die Screen-Struktur ermitteln Sie mit:

PEEKL (WINDOW (7) +46)

Wenn Intuition einen Screen und Windows dargestellt hat, dann können Sie diese mit der Maus verschieben, vergrößern, verkleinern, öffnen, schließen, nach vorn oder nach hinten legen. All diese Aktionen können Sie aber auch per Programm durchführen. Dafür stellt uns Intuition entsprechende Routinen bereit.

CALL MoveScreen (screen, dx, dy)

Damit läßt sich der Screen verschieben. screen = der Zeiger auf die

Screen Struktur.

dx,dy = die Verschiebung in Pixeln.

Da die Verschiebung in X-Richtung in der derzeitigen Version des Amiga noch nicht implementiert ist, setzen Sie den Wert für dx auf 0. Sie können den Screen also nur herauf- oder herunterschieben. Ein negativer Wert verschiebt den Screen nach unten und ein positiver nach oben.

CALL MoveScreen (screen, dx, dy)

Das ist das Gegenstück zum Move-Screen Befehl für ein Window.

window = ein Zeiger auf die Window Struktur. dx,dy = die Verschiebung des Windows in

Pixeln.

Negative Werte verschieben das Fenster nach links, bzw. oben, ein positiver Wert verschiebt das Fenster nach rechts, bzw. unten.

Call SizeWindow(window,dx,dy)

Damit läßt sich ein Fenster vergrößern oder verkleinern.

window = der Zeiger auf die Window-Struktur.

dx,dy = die Anzahl der Pixel, um die das Fenster vergrößert oder verkleinert wird. Zum Schließen eines Screens oder Windows gibt es auch Routinen:

CALL CloseScreen (screen)
schließt einen Screen.

CALL CloseWindow (window)
schließt ein Fenster

Um ein Fenster oder einen Screen nach vorne oder nach hinten zu legen, stellt uns Intuition folgende Routinen zur Verfügung:

CALL ScreenToFront (screen)
bringt den Screen in den Vordergrund.

CALL ScreenToBack (screen)
legt diesen Screen hinter alle anderen.

CALL WindowToFront (window)
bringt dieses Fenster in den Vordergrund.

CALL WindowToBack (window)
legt dieses Fenster unter alle anderen.

Für die Workbench gibt es extra Routinen:

CALL WBenchToFront
und
CALL WBenchToBack

Amiga Basic gibt uns bei Aufruf des Befehls WINDOW (7) immer nur den Zeiger auf das Amiga Basic-Ausgabefenster zurück, selbst wenn gerade ein anderes Fenster aktiv ist (z.B. das List-Fenster oder das Fenster eines anderen Tasks). Wie Sie die Window-Struktur eines anderen Fensters auslesen, zeige ich Ihnen jetzt. Dazu benötigen wir zuerst einen Zeiger auf die Intuitionbase-Struktur. Den Zeiger auf diese Struktur bekommen Sie, wenn Sie eine Library mit der Routine OpenLibrary aus der Exec Library aufrufen.

intbase= OpenLibrary& (libname, version)

libname = der Zeiger auf den Namen der zu öffnenden Library.

version = die Versionsnummer. Diese Zahl muß größer oder gleich der gesuchten Versionsnummer sein.

Die Win	dowstru	ıktur :
Offset	Byte	Bezeichnung
0	L	Zeiger auf nächstes Window
4	W	Beginn der linken Ecke des Fensters
6	W	Beginn der oberen Ecke des Fensters
8	W	Breite des Fensters
10	W	Höhe des Fensters
12	W	Y-Koordinate des Mauszeigers
14	W	X-Koordinate des Mauszeigers
16	W	Minimale Breite des Fensters
18	W	Minimale Höhe des Fensters
20	W	Maximale Breite des Fensters
22	W	Maximale Höhe des Fensters
24	L	Flags
28	L	Zeiger auf erste Menuestruktur
32	L	Zeiger auf Fenstertitel
36	L	Zeiger auf erstes Requester
40	L	Zeiger auf erstes Double Klick Requester
44	W	Anzahl der aktiven Requester
46	L	Zeiger auf Screen Struktur
50	L	Zeiger auf Rastport Struktur
54	4xB	Beschreibt Aussehen des Rahmens
58	L	Zeiger auf die Rastportstruktur des Rahmens
62	L	Zeiger auf erstes Gadget
66	L	Zeiger auf 'Parent' Window Struktur
70	L	Zeiger auf 'Descendant' Window Struktur
74	L	Zeiger auf Sprite Datas des Mauszeigers
78	В	Höhe des Mauszeigers
79	В	Breite des Mauszeigers
80	В	X-Offset des 'Hot Spot' des Mauszeigers
81	В	Y-Offset des 'Hot Spot' des Mauszeigers
82	L	IDCMP Flags
86	L	Zeiger auf User Port
90	L	Zeiger auf Window Port
94	L	Zeiger auf Message Key
96	В	Farbnummer für den Text in der Titelzeile
97	В	Farbnummer für Hintergrund Titeltext
98	L	Zeiger auf Imagestruktur f. abgehakte Menuepunkte
102	L	Zeiger auf Screentitel dieses Windows
106	W	X-Koord. des Mauszeigers bei Gimmezerozerowindow
108	W	Y-Koord. des Mauszeigers bei Gimmezerozerowindow
110	W	Breite des Fensters bei Gimmezerozerowindow
112	W	Höhe des Fensters bei Gimmezerozerowindow
116	L	Zeiger auf User Daten
120	L	Zeiger auf User Daten
124	L	Zeiger auf Layer Struktur

In intbase steht nach erfolgreicher Öffnung die Basisadresse dieser Library. Der Wert 0 bedeutet, daß das Öffnen nicht geklappt hat. Aus dieser Struktur können Sie nun den Zeiger auf den momentan aktiven Screen, das aktive Window und den Zeiger auf den ersten Screen ermitteln, (siehe Programm OpenLib).

Damit wollen wir es für diese Folge genug sein lassen. Ich hoffe, daß es Ihnen wieder etwas Spaß gemacht hat und Sie etwas Neues über Ihren Amiga kennengelernt haben. In der nächsten Ausgabe geht es weiter mit der Intuition Library. Wir werden unter anderem lernen, wie der Mauszeiger ein anderes Aussehen bekommen kann und wie wir diesen Zeiger per Programm bewegen können.

```
1: 'Programm: Viewdemo mit Dual Playfield Modus
 3: DECLARE FUNCTION AllocMem&
                                   LIBRARY
 4: DECLARE FUNCTION GetColorMap& LIBRARY
 5: DECLARE FUNCTION AllocRaster& LIBRARY
 7: LIBRARY "exec.library"
 8: LIBRARY "graphics.library"
10: vbreite&=320
                        'Breite View
11:
     vhoehe&=220
                         'Hoehe View
12: modes&=&H400+&H40 'Dual Playfield / Playfield 2 vorne
13:
14: flag&=2^1+2^16
15: speicher&=AllocMem& (362,flag&) 'Speicher anlegen
16: IF speicher&=0 THEN ERROR 7
17.
18: rastport1&=speicher&
19: viewport& =speicher&+100
20: view& =speicher&+140
21: rasinfol& =speicher&+158
22: bitmapl& =speicher&+170
23: rastport2&=speicher&+210
24: rasinfo2& =speicher&+310
25: bitmap2& =speicher&+322
26:
27: breite&=640:hoehe&=400:tiefe&=1
28: Initbitmap bitmap1&,tiefe&,breite&,hoehe&
29: Initbitmap bitmap2&,tiefe&,breite&,hoehe&
                                                      ' Zwei Bitmaps und
30: FOR i=1 TO tiefe&
                                                       ' Bitplanes
31: bitplane1&(i)=AllocRaster& (breite&,hoehe&)
                                                      ' initialisieren
       IF bitplanel&(i)=0 THEN ERROR 7
 32:
33: POKEL bitmap1&+8+(i-1)*4, bitplane1&(i)
34: NEXT
 35: FOR i=1 TO tiefe&
     bitplane2&(i)=AllocRaster& (breite&, hoehe&)
36.
      IF bitplane2&(i)=0 THEN ERROR 7
37:
38:
     POKEL bitmap2&+8+(i-1)*4,bitplane2&(i)
39: NEXT
40:
41: rxoffset1&=0:ryoffset1&=0
42: InitRastport rastport1&43: InitRastport rastport2&
                                                      ` Zwei Rastportstrukturen
                                                       ' initialisieren
                                                      ' Werte in Rastport und
44: POKEL rastport1&+4, bitmap1&
45: POKEL rasinfol& ,rasinfo2&
                                                      'Rasinfostruktur setzen
     POKEL rasinfol&+ 4, bitmapl&
46:
47: POKEW rasinfol&+ 8, rxoffsetl&
48: POKEW rasinfol&+10, ryoffset1&
50: rxoffset2&=0:ryoffset2&=0
51: POKEL rastport2&+4, bitmap2&
                                                      ' Werte fuer zweite
52: POKEL rasinfo2&+ 4,bitmap2&
                                                      ' Rastportstruktur setzen
53: POKEW rasinfo2&+ 8, rxoffset2&
54: POKEW rasinfo2&+10, ryoffset2&
55:
56: InitVPort viewport&
57: POKEW viewport&+24, vbreite&
                                                   'Viewport initialisieren
58: POKEW viewport&+26, vhoehe&
                                                      ' und Werte setzen
59: POKEW viewport&+32,modes&
60: POKEL viewport&+36,rasinfol&
61:
                                                       ' View initialisieren
 62: InitView view&
 63: POKEL view&, viewport&
                                                       ' und Werte setzen
64:
 65: colormap&= GetColorMap& (12)
                                                      'Farbtabelle einrichten
66: SetRGB4CM colormap&, 0, 0, 2, 1
67: SetRGB4CM colormap&, 1, 15, 15, 15
                                                      ' Farben setzen
                                                    ' und
68: SetRGB4CM colormap&, 8, 0, 2, 1
                                                      'Farbtabelle in
 69: SetRGB4CM colormap&, 9, 2, 12, 1
 70: POKEL viewport&+4, colormap&
                                                      ' Viewportstruktur setzen
 71 .
72: MakeVPort view&, viewport&
                                                      ' Copperlist f. Viewport
 73: MrgCop view&
                                                       'Copperlist zusammenbinden
```

```
74: LoadView view&
                                                        ' Copperlist aktivieren
75:
76: setrast rastport1&,0
 77: setrast rastport2&,8
 78: setapen rastport1&,1
 79: y&=0:x1&=300:y1&=400
 80: FOR x&=1 TO 600 STEP 30
                                                        ' In Bitmap 1 zeichnen
 81: move rastport1&,x&,y&82: draw rastport1&,x1&,y1&
 83: NEXT
 84:
 85: setapen rastport2&,9
 86: FOR i=1 TO 100
 87: RANDOMIZE TIMER
88: x&=INT (1+RND(TIMER)*640)
                                                        ' In Bitmap 2 zeichnen
 89: y&=INT (1+RND(TIMER)*400)
 90: xr&=INT (1+RND(TIMER)*20)+x&
 91: yr&=INT (1+RND(TIMER)*20)+y&
92: rectfill rastport2&,x&,y&,xr&,yr&
 93: NEXT
 94:
 95: FOR k&=1 TO 300
 96: POKEW rasinfo2&+8,k&
                                                        ' Playfield 2 scrollen
       IF k&>200 THEN
 97:
        POKEW rasinfo2&+10, PEEKW (rasinfo2&+10)+1
 98:
     END IF
99:
100: MakeVPort view&, viewport&
101: MrgCop view&
102: LoadView view&
103:
      FOR 1=1 TO 80:NEXT
                                                        ' Verzoegern
104: NEXT
105.
106: FOR k&=1 TO 300
107: POKEW rasinfol&+8,k&
                                                        ' Playfield 1 scrollen
108:
       MakeVPort view&, viewport&
109: MrgCop view&
110: LoadView view&
111: NEXT
112.
113: FOR k=1 TO 5000: NEXT
                                                        ' Einen Moment warten
114:
115: aus:
116: FreeColorMap colormap&
117: FOR i=1 TO tiefe&
118: FreeRaster bitplane1&(i),breite&,hoehe&
119: FreeRaster bitplane2&(i),breite&,hoehe&
                                                        'Speicher wieder
                                                        'freigeben
120: NEXT
121: FreeMem speicher&, 362
122:
123: LIBRARY CLOSE
                                                        'Library schliessen
124:
125: STOP
                                                         ' zum Workbenchmodus
```

Listing 1: Das Programm Viewdemo

```
1: DECLARE FUNCTION OpenLibrary& LIBRARY
2: LIBRARY "exec.library"
3: LIBRARY "intuition.library"
4: lib$="intuition.library"+CHR$(0)
5: intbase&=OpenLibrary& (SADD(lib$),0)
6: PRINT "Aktive Window ="PEEKL(intbase&+52)
7: PRINT "Aktive Screen ="PEEKL(intbase&+56)
8: PRINT "First Screen ="PEEKL(intbase&+60)
9: CALL CloseLibrary (intbase&)
10: LIBRARY CLOSE
```

Listing 2: Gibt Infos über aktuelle Fenster und Screens

DYNAMIC STUDIO

Das digitale 16-Spur-Heimstudio

Begriffe wie MIDI, Homerecording oder Sequenzer werden in letzter Zeit, auch im Mikrocomputerbereich, immer populärer. Durch die ständig wachsende Leistungsfähigkeit der Mikrocomputer lassen sich im Zusammenspiel zwischen Computer und elektronischen Musikinstrumenten neue Wege beschreiten. Das Zauberwort heißt hier MIDI (Musical Instrument Digital Interface). Mittels dieser Schnittstelle und der entsprechenden Software lassen sich zu Hause Ergebnisse erzielen, die bislang nur professionellen Aufnahmestudios vorbehalten waren.

ew Wave Software, bekannt durch ihr Drummachine-Programm Dynamic Drums für den Amiga, bieten jetzt eine Komplettlösung für den MIDI-Anwender an. Dynamic Studio heißt dieses Softwarepaket. Voraussetzung für die optimale Nutzung des Programms ist das Vorhandensein einer MIDI-Schnittstelle (z.B MIDI-Gold) und natürlich eines oder mehrerer MIDI-fähiger Gerätschaften, wie Synthesizer etc. Der Lieferumfang enthält ein gut geschriebenes englisches Handbuch und zwei Programmdisketten, wobei die zweite Disk mit Daten für die Drummachine belegt ist. Kompatibel erwies sich das Programm im Test zu allen drei Amigasystemen (A1000,A500 A2000). Für den professionellen bzw. semiprofessionellen Anwender ist jedoch eine Speichererweiterung vonnöten, um auch wirklich lange Sequenzen aufzunehmen. DS startet nach dem Einlegen der Programmdisk (Disk Nr.1) automatisch und meldet sich nach kurzer Zeit mit dem MIDI-Sequenzerteil des Programms (siehe Bild 2). DS untergliedert sich in drei Teile, nämlich den MIDI-Sequenzer, die Drummachine und den sogenannten Sequenzer-Editor. Alle Teile des Programms stehen sofort nach dem Booten zur Verfügung und können per Mausklick oder per Tastendruck abgerufen werden.

1. Die Drummachine

Hat man nach dem Laden die "ESC"-Taste gedrückt oder im Menü "Drum Machine" angewählt, erscheint ohne störende Nachladezeit der Steuerbildschirm der Drummachine (siehe Bild 1). Gespielt werden die einzelnen Drums (Trommeln) über den Zehnerblock der Tastatur. Um dies zu erreichen, muß jedoch vorher ein Drumkit geladen worden sein. Auf der Disk Nr.2 sind hier einige schon sehr beeindruckende Drumkits gespeichert. Wurde ein Drumkit geladen, werden im rechts unten im Bildschirm die einzelnen Drums mit Namen (z.b. Bassdrum, TomTom usw.) entsprechend ihrer Zuordnung zu den Tasten des Zehnerblocks angezeigt. Will man ein eigenes Schlagzeug kreieren, kann man einzelne Drums laden (hierzu kann jeder IFF-Sample-Sound verwandt werden!) und frei ihre Zuordnung zu den Tasten bestimmen. Natürlich lassen sich auf diese Weise erzeugte Drumkits auch abspeichern und somit jederzeit wieder aufrufen. Im linken Teil des Bildschirms befinden sich 16 Schieberegler, mit denen die einzelnen Drums gestimmt (in ihrer Tonhöhe verändert) und in der Lautstärke beeinflußt werden können. Mittels eines Zufallsgenerators kann man jedem Drum einen realistischen Effekt zuordnen. Hierbei wird die Tonhöhe des angewählten Drums bei jedem Schlag minimal verändert (der Stock eines Schlagzeugers trifft ja auch nicht immer die gleiche Stelle des Schlagfells), was den Sound realistischer erscheinen läßt.

Achtung Aufnahme!

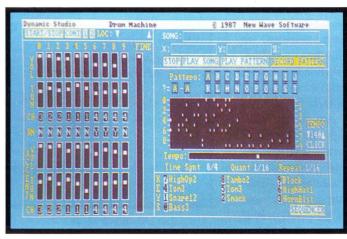
Will man einen eigenen Song aufnehmen, legt man erst einmal den gewünschten Takt, in dem der Rhythmus laufen soll, fest. Hier sind alle Kombinationen vom 4/4 bis hin zu sehr exotischen Takten (z.b 7/4) möglich. Um den eigenen Song aufzunehmen, wählt man "Record Pattern" an. Ein symbolisiertes Metronom beginnt nun im vorher eingestellten Takt zu ticken, und der Rechner ist bereit zur Aufnahme. Alle jetzt über den Zehnerblock gespielten Rhythmen werden im Speicher als sogenanntes "Pattern" übernommen. Die Aufnahme wiederholt sich je nach eingestelltem Takt ständig, so daß man jedes Drum einzeln exakt einspielen kann.

lenden Schlagzeugers!). Da sämtliche Sounds der einzelnen Drums gesampelt sind und sich, wie schon angesprochen, eigene Drumkits zusammenstellen lassen, sind die Ergebnisse, die sich mit der Drummachine erzeugen lassen, qualitativ sehr hoch und werden auch gehobenen Ansprüchen gerecht.

2. Der Sequenzer

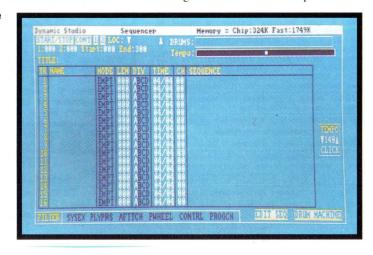
Der Sequenzer bildet das eigentliche Kernstück von Dynamic Studio. Mit ihm wird die Verbindung zur Außenwelt mittels MIDI geknüpft. Der Sequenzer wird wie auch die Drummachine entweder mit der Maus oder mit der "ESC"-Taste aktiviert. Er bietet die Möglichkeit, 16 verschiedene MIDI-Kanäle gleichzeitig anzusteuern, und,

Die übersichtliche Drummachine von Dynamic Studio.

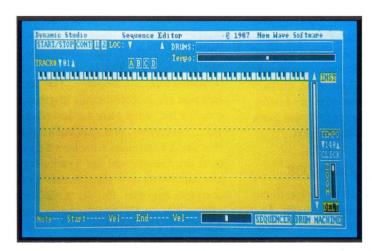


Die Auflösung bei der Aufnahme beträgt eine 192stel Note und wird somit professionellen Geräten gerecht, da damit auch sehr schwierige Synkopen und Rhythmen zu realisieren sind. Alle gespielten Noten werden grafisch angezeigt und können per Maus ediert werden. Diese Art der Kontrolle macht Änderungen von Rhythmen zu einem Kinderspiel. Ist man mit seinem Pattern zufrieden, kann man dieses im sogenannten "Patterfield" ablegen. Hier können 20 verschiedene Patterns gespeichert werden. Durch Kombinationen dieser Patterns lassen sich dann beliebige Begleit-Rhythmen, also ein "Drum-Song", erstellen. Sowohl einzelne Patterns als auch fertige Drum-Songs können jederzeit auf Disk gesichert werden. Um einen realistischen Rhythmuseffekt zu erzielen, hat man ferner die Möglichkeit, einzelne Beats in der Zeit und in ihrer Tonhöhe zu beeinflussen (Effekt eines richtig spiein Verbindung mit der Drummachine, diese Kanäle auch zu synchronisieren. So kann man sich zuerst mit der Drummachine einen Rhythmus kreieren und dann zu diesem die Keyboardparts einspielen. Bei der Aufnahme kann man wählen, ob ein eventuell vorher aufgenommener Rhythmus als Taktgeber fungieren soll oder ob man lieber frei

Das Herzstück von Dynamic Studio der 16-Spur-Sequenzer.



spielen möchte. Die einzelnen Aufnahmespuren eines Stückes werden hier als "Tracks" bezeichnet, Kombinationen aus solchen Tracks wie bei der Drummachine als Sequenz. Hat man sich nun zur Aufnahme mit Rhythmus entschieden, wird brav im jeweiligen Takt "vorgezählt" und man kann loslegen. Die Länge eines einzelnen Tracks bzw. der Gesamtzahl der Tracks wird hier nur durch den verfügbaren Speicherplatz begrenzt. Bei der Aufnahme werden alle am Keyboard ausgeführten Aktionen, wie natürlich das Spielen der Melodie und der Akkorde, das Umschalten von Sounds, Pitchwheelaktionen etc. exakt in den Speicher des Amiga übernommen. Der Sequenzer bietet hier eine ganze Reihe von Funktionen, um z.B. Soundbankänderungen am Synthesizer zu ignorieren oder Änderungen am Pitchwheel nicht in den Speicher einzutragen. Hat man seine Sounds aufgenommen, kann man die einzelnen Kanäle auf vielfältige Weise edieren. So ist es möglich, einzelne Kanäle mit Hall zu versehen, zu kopieren, zu transponieren (also in eine andere Tonart zu versetzen), zu "stretchen", d.h., die Länge zu variieren, und und und... Die Beschreibung aller Möglichkeiten würde leider den Rahmen des Tests sprengen. Im Test arbeitete der Sequenzer mit einem Yamaha DX 7- und einem Oberheimkeyboard problemlos zusammen. Sowohl externe wie auch interne Synchronisation wurden ohne Abstürze verkraftet. Als einziger Nachteil erwies sich, daß mit der Drummachine vorher erstellte Rhythmen in Verbindung mit dem Sequenzer etwa 10 Beats pro Minute langsamer liefen. Dieses Problem lies sich aber durch entsprechendes Timetuning der Drummachine problemlos



Änderungen? Kein Problem mit dem Sequenz-Editor.

ausgleichen. Meiner Meinung nach ein ehr kleiner Makel, der die sonst hervorragenden Möglichkeiten des Sequenzers nicht schmälern kann.

3. Der Sequenz-Editor

Der Sequenz-Editor (siehe Bild 3) ist der letzte der drei Teilbereiche von Dynamic Studio. Mit ihm ist es möglich, einzelne Noten von vorher mit dem Sequenzer eingespielten Tracks grafisch, also mit der Maus, zu edieren. Diese Art des Edierens macht Änderungen ähnlich wie schon bei der Drummachine zu einem Kinderspiel. Mit einem Schieberegler kann man die gewünschte Sequenz in den zu bearbeitenden Bereich bewegen. Jetzt wählt man den zu edierenden Track aus und kann nun einzelnen Noten mit der

Maus verändern. Mittels eines Playback-Modus läßt sich die edierte Sequenz abspielen. Hierbei läuft ein Balken durch den Editorscreen, mit dessen Hilfe es einfach ist, "falsche" Noten zu erkennen. Man hat die Möglichkeit, wie bei Textverarbeitungssystemen, einzelne Noten auszuschneiden, zu kopieren, Noten zu löschen und einzufügen. Da diese Funktionen allesamt mit der Maus kontrolliert werden, ist die Handhabung einfach zu erlernen und zu beherrschen. Am unteren Rand des Bildschirms stellt der Editor die jeweils aktuellen Werte des zu edierenden Tracks dar. "Note" zeigt zum Beispiel die jeweilige Tonhöhe der gerade durchlaufenden Note an, "Start" und "End" die Start- und Endposition in 1/ 192stel und "Vel" den eigentlichen Notenwert. Den Editor kann man praktisch als Zugabe des Herstellers verstehen, er ist aber trotzdem nicht zu unterschätzen. Einmal eingesetzt, kann man sich schwerlich wieder von ihm trennen, denn er stellt ein sehr nützliches Tool dar, das die Arbeit gerade an sehr schwierigen Passagen ungemein erleichtert.

Fazit:

Mit Dynamic Studio ist ein sehr leistungsfähiges Programm auf den Markt gekommen, das durch seine einfache Bedienung und Erlernbarkeit für jeden MIDI-Freak interessant sein sollte. Schon der alleinige Einsatz der Drummachine würde reichen, um so manchen Musiker die Ohren spitzen zu lassen, und sicherlich einige personelle Probleme in manchen Bands lösen. Let's rock !!!

Bewertungskasten:

- + sehr gute Drummachine
- + einfache Bedienung
- + Alle Änderungen direkt mit der Maus
- + Komfortabler Editor
- englisches Handbuch
- Drummachine in Verbindung mit MIDI etwas langsamer

Bezugsquelle: PDC, Bad Homburg Preis: DM 375.-

ENDE

Verbinden Sie Ihren SHARP Pocketcomputer mit Commodore AMIGA.

Übertragen von Daten und Programmen des SHARP-Rechners in beide Richtungen, Editieren und Drucken auf allen AMIGA Modellen möglich. Alle Daten und Programme können schnell und sicher auf Diskette abgespeichert werden. TRANSFILE AMIGA ist vollständig mit der Maus zu steuern und ohne Kopierschutz! Es ist für die SHARP-Pockets PC 1260/61/62/80, PC 1350/60, PC 1401/02/03/21/25/30/45/50/60/75 geeignet.

Weitere SHARP-Rechnertypen sind in Vorbereitung!
TRANSFILE AMIGA anschlußtertig und komplett mit Interface, Diskette und Anleitung nurDM 129,00

* Bei Bestellung unbedingt Rechnertypen angeben *

Händleranfragen erwünscht!

Als TRANSFILE 64 auch für C-64 bzw. C-128 erhältlich

Als TRANSFILE 64 auch für C-64 bzw. C-128 erhältlich!

Ausführliche. Info gegen adressierten Freiumschlag anfordern. Versand ins Ausland nur gegen Vorkasse!

YELLOW-COMPUTING Wolfram Herzog & Joachim Kieser

Im Weingarten 21 D-7101 Hardthausen 3
Telefon 07139 / 8355

WSW Public Domain Angebot wie "Kick-Start" Einzeldisk ab 10 Stück 5.00 DM 6.00 bis 9 Stück Paket 1-10, 11-20, 21-30 u.s.w. DM 45.00 Paket Spiele PD 2,3,8,10,34,44,45, Paket Anw. PD 5,6,14,21,28,36 Paket Bilder PD 12,13,19,23,24,39 Paket Anim. PD 15,20,26,27,33 33,00 28.00 24.00 DM DM 50.00 Paket Grafik und Sound 11 Disk Paket Utilities 10 Disketten Paket Programmier PD 1,16,17,22,50,51 DM 28,00 Paket "The very best of PD" 5 Disketten mit 5 Disketten mit Preis inld. Disk zzgl. Porto und Verp. DM 5.00 (Ausl. DM 10.00) NN plus 1.70, besser V.-Scheck

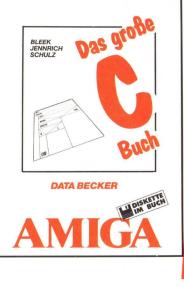
Schulstraße 25 7516 Karlsbad 4 ab 17 Uhr 07202/5713

C. V. S. - Versand

Laufwerke extern 3,5"
GÜLEM, abschaltbar 370,00 DM
TEAC , abschaltbar 320,00 DM
Speichererweiterungen
A 500 extern 2 MByte PROFEX 948,00 DM
A 500 intern 500 KByte ... auf Anfrage
Drucker/Scanner
MEC 2200 Pinwriter (24 Nadel) 998,00 DM
Olivetti DM 105 Farbdrucker 698,00 DM
Handy-Scanner mit 16 Graustufen , incl.
Grafikpaket , von Cameron 858,00 DM
Disketten 3,5"
No Name 2D 100% Errorfree ab 2,40 DM
Marken 2D 100% Errorfree ab 2,95 DM
Software
AMIGA TULP neue Utility-Disk mit Copy,
Viruskiller, RAM-Deleter etc. 49,95 DM
SOftware
AMIGA TULP neue Utility-Disk mit Copy,
Viruskiller, RAM-Deleter etc. 49,95 DM
CV.S. , Nauher Berg 1, 2306 Schönberg
tel. Bestellannahme: (0431) 55 15 15
Aktuelle Preisliste auf Anforderung
Preisänderungen vorbehalten

Top aktuell:









Das Buch, das zur Amiga-Floppy keine Frage offenläßt. Hier finden Sie Dinge, die Sie in Ihrem Handbuch vergeblich suchen werden: Floppy-Operationen unter Workbench und unter AmigaDOS im CLI, relative und sequentielle Dateien, Aufbau der Diskette. Zugriff über Trackdisk-Device, Track lesen und schreiben, Kodierund Dekodierroutinen des Betriebssystems - alles, was Sie zur Amiga-Floppy wissen müssen, wird hier detailliert und ausführlich beschrieben. Dazu eine Reihe nützlicher, hilfreicher Programme, die natürlich gleich auf Diskette mitgeliefert werden: z.B. ein Superkodierprogramm oder ein Floppyspeeder. Das Amiga Floppybuch - umfassendere Informationen werden Sie kaum finden.

Amiga Floppybuch Hardcover, 398 Seiten inkl. Diskette, DM 59,-

Das große C-Buch zum Amiga ein Buch für alle, die Spaß an C gefunden haben und nun darauf brennen, eigene, professionelle Programme zu entwickeln. Ausführlich erfahren Sie hier, wie ein C-Compiler arbeitet und wie Sie selbst die schwierigsten Probleme in C lösen - die beste Garantie, ein wirklich fehlerfreies Programm zu schreiben. Natürlich sollten Sie hierfür auch eine optimale Benutzeroberfläche entwickeln. Daher der zweite Schwerpunkt im großen C-Buch: die Programmierung von Screens, Windows, Pull-Down-Menüs und Gadgets. Gemeinsam mit den Autoren können Sie nun Ihr erstes großes Programmprojekt verwirklichen: einen C-Editor, der die Folding-Funktion ("Wegfalten" von Textteilen) beherrscht. Wer mit diesem Buch arbeitet, wird in Zukunft höchstens noch eine Fehlerquelle haben: Tipfehler. Das große C-Buch zum Amiga Hardcover, ca. 680 Seiten inkl. Diskette, DM 69,-

Der DATA BECKER Führer zu AmigaDOS und AmigaBASIC ein nahezu unverzichtbares Nachschlagewerk für jeden Amiga-Anwender. Schnell und zuverlässig finden Sie hier jederzeit die Informationen, die Sie für Ihre tägliche Arbeit brauchen. Ob zu Amiga-DOS oder zum AmigaBASIC dieser DATA BECKER Führer läßt Sie nicht im Stich. Alle Befehle und Kommandos sind übersichtlich nach Sachgruppen, alphabetisch mit Kurzsyntax und nach Stichworten geordnet. Eben alles auf einem Blick. Ein Buch, das einfach zu jedem Amiga gehört.

Der DATA BECKER Führer zu AmigaDOS & AmigaBASIC 269 Seiten, DM 24,80

Still herdinge für

DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

Mit dem Rechner auf Du und Du

Ein Assemblerkurs für Einsteiger (Teil 7)

ies ist der letzte Teil unseres Assemblerkurses für Einsteiger. In der nächsten Ausgabe der KICKSTART beginnt ein brandneuer Kurs, der sich nun nicht mehr mit den Assemblerbefehlen, sondern intensiv mit der Programmierung der Hardware des AMIGA auseinandersetzt. Ein Hauptteil des Kurses wird die Copper-, Blitter- und Grafikprogrammierung (Bitmaps, (Dual) Playfields, Sprites) ausmachen. Doch nun zum letzen Einsteigerteil. In diesem letzten Part gehen wir noch einmal auf die Rotationsbefehle von Bits ein. Diese Befehle heißen ASx, ROx, ROXx, wobei sich das x auf die Richtung der Bewegung bezieht (L = links, R = rechts).

Wir verschieben

Das mit dem arithmetischen Verschieben ist ganz einfach und funktioniert so: beinhaltet das Datenregister D0 zum Beispiel den Wert 1, was als Bytewert binär 00000001 ergeben würde, hat es nach dem einmaligen Verschieben nach links den Wert 2 (00000010). Und so ist ja nun auch schon klar, was die Verschiebebefehle machen - sie verschieben die Bits innerhalb eines Bytes, Wortes oder Langwortes um eine vorgegebene Stellenzahl in eine vorgegebene Richtung. Viele Worte, einfache Bedeutung. Die Syntax dieser Befehle lautet:

ASx #Stellenzahl, Datenregister

Hierzu ein kleines Beispielsprogramm:

1 CLR.L d0 2 MOVE.B #01,D0 3 ASL.B #04,D0 4 RTS

Bei diesem Beispielsprogramm hätte das Register D0 am Ende den Wert dezimal 16 (%00010000), denn der ASL-Befehl hat das Bit um vier Stellen nach links verschoben. Hätten wir den Wert 1 um acht Stellen nach links verschoben, hätte dies die Bytegrenzen gesprengt, die wir ja durch das Kürzel ".b" hinter dem ASL-Befehl gesetzt haben. Da hat man aber vorgesorgt. In einem solchen Fall setzt der ASL-Befehl einfach das X- und zusätzlich das C-Flag. Dieses Flag simuliert dann einfach das "fehlende" 8te Bit (natürlich nur innerhalb eines Bytes, in einem Word das 16te und in einem Longword das 32te Bit):

%00000001 verschoben um acht Stellen nach links ergibt

X-Flag Byte 1 00000000 = dez. 256

Genauso funktioniert logischerweise der ASR-Befehl. Zu dieser (arithmetischen) Verschiebung ist noch zu sagen, daß eine Verschiebung nach links eine Multiplikation mit 2 und eine Verschiebung nach rechts eine Division durch 2 darstellt. Wird ein Bit aus einer festen Grenze (b,w,l) herausgeschoben, so setzt der ASx Befehl zusätzlich das N(egativ)-Flag.

Der AMIGA-Assemblerbefehlssatz beinhaltet noch eine logische Verschiebung innerhalb eines Bytes, Words, oder Longwords. Deren Syntax ist:

LSx #Stellenzahl, Datenregister.

Analog zum Befehl LSL n,<ea> gibt es noch sein Gegenstück, den LSR-Befehl, was "Logical Shift Right", zu deutsch "logische Verschiebung nach rechts", bedeutet. Er hat die gleiche Syntax wie LSL, wobei n nun die Anzahl der bitweisen Verschiebungen nach rechts enthalten muß. Die folgende Skizze soll noch einmal die Schieberichtung verdeutlichen.

Ursprüngliches Byte:0010 0000
<--Schieberichtung links:0100 0000
--->Schieberichtung rechts:0001 0000

Wie bei ASx/LSL wird auch bei LSR eine Null in das freiwerdende Bit geschoben. Das Bit, das auf der anderen Seite hinausgeschoben wird, kommt in das X (Extented)- und zugleich in das C(Carry)-Flag. Ein Beispiel hierzu:

```
1 clr.1 dl
2 move.b #%01011110,dl
3 lsr.b #1,dl
4 rts
```

In Zeile 1 wird das Datenregister D1 gelöscht. Dann wird es mit bin. #01011110 geladen. Dies entspricht

dez. 94 und hex. 5e. In Zeile 3 wird D1 um eine binäre Stelle nach rechts geschoben, was einer Division durch zwei entspricht. Dabei verschieben sich die "Einsen" im Byte wie folgt:

```
0101 1110
Schieberichtung: ->
Nachgeschoben 0-> _010 1111 -> 0 in X-
und C-Flag.
```

Nach Ablaufen des Programms befindet sich der Wert dez. 47 bzw. hex. 2F im Datenregister D1. Rechnen Sie doch selbst einmal nach!

"Wir rotieren"

Bei den vorhergehenden Schiebe-Befehlen war es so, daß eine Null nachgeschoben wurde und das herausgeschobene Bit im X- bzw. C-Flag landete. Es gibt aber noch eine andere Art von Verschiebe-Befehlen, nämlich solche, die das Byte/Word/Longword rotieren lassen. Sie brauchen keine Angst zu haben, daß Ihr Computer ins Strudeln gerät, aber die Tastatur sollten Sie trotzdem festhalten. Die Befehle heißen ROR und ROL. Sie haben die Syntax ROR n,<ea> bzw. ROL n,<ea>. Dabei gibt n wieder Anzahl der Stellen an, um die das Byte rotiert werden soll. Der Unterschied zu den vorhergehenden Befehlen ist nun, daß das herausgeschobene Bit auf der anderen Seite in die freiwerdende Stelle eingepaßt wird. Hier wieder ein Beispiel:

```
1 clr.1 d2
2 clr.1 d3
3 move.b #%10001111,d2
4 rol.b #2,d2
5 move.b #%00111100,d3
6 ror.b #3,d3
7 rts
```

In Zeile 3 wird D2 mit dem Wert dez. 143 geladen. In der nächsten Zeile wird D2 um 2 Stellen nach links rotiert. Dies sieht folgendermaßen aus:

```
1000 1111
Schieberichtung: <---
10 <-- 0011 11__ <-- 10
0011 1110
```

Aus der dez. 143 wird schließlich eine dez. 62. Die bin. 10 wird vorne heraus-

geschoben und hinten wieder angehängt. Das Ergebnis steht in D2. In Zeile 5 wird D3 mit dez. 60 gelanden und in Zeile 6 um 3 binäre Stellen nach rechts rotiert. Auch hier ein Schaubild:

```
0011 1100
Schieberichtung: ->
100 -> ___0 0111 ->100

1000 0111
```

Das Datenregister D3 enthält nun den Wert dez. 135. Dabei ist zu beachten, daß die Bits blockweise und nicht einzeln rotiert werden.

Es gibt noch eine Erweiterung zu den ROx-Befehlen, die ROXx-Befehle. Diese haben die Syntax:

ROXx #Stellenanzahl, Datenregister.

Der einzige Unterschied zu den ROx-Befehlen besteht darin, daß hier nicht nur innerhalb einer Byte-, Word- oder Longwordgrenze rotiert wird, sondern es wird auch das X-Flag in die Rotation mit einbezogen.

THE SUMMARY

Nun als kleines Schmankerl alle Assemblerbefehle, die bis jetzt in diesem Kurs behandelt wurden:

Befehlss	yntax mögliche Kürz	zel
MOVE.x	Konstante, Adressregister	A,Q
MOVE . x	Konstante, Datenregister	Q
MOVE . x	Adresse, Adresse	-
MOVE.x	Register, Register	A
MOVE . x	(Adressregister), Register	-
MOVE . x	(Adressregister)+, Register	
MOVE . x	-(Adressregister), Register	-
MOVEM.x	mehrere Register, Adresse oder Stack	M
ADD.x	Quelle, Ziel	A, X, I
SUB.x	Quelle, Ziel	A,X,I
MUL.x	Quelle, Ziel	S,U
DIV.x	Quelle, Ziel	S,U
EXT.x	Register	-
RTS		
CMP.x	Quelle, Ziel	-
Вхх	Label	
DBxx	Datenregister, Label	-
JSR	Label	45 July -
JMP	Label	-
BSET	Bit, Register / Adresse	-
BCLR	Bit, Register / Adresse	
BCHG	Bit, Register / Adresse	-
CLR.x	Register	-
ASL.x	Stellenzahl, Register / Adresse	-
ASR.x	Stellenzahl, Register / Adresse	
LSL.x	Stellenzahl, Register / Adresse	-
LSR.x	Stellenzahl, Register / Adresse	-
ROR.x	Stellenzahl, Register / Adresse	-
ROL.x	Stellenzahl, Register / Adresse	=
ROXR.x	Stellenzahl, Register / Adresse	-
ROXL.x	Stellenzahl, Register / Adresse	_

So, damit ist alles vorbei). Aber bald, geht's weiter mit dem Advanced As-

semblerkurs... Also, bis dann - und bleibt uns treu.

PC AN DER LANGEN LEINE



Daß der AMIGA mit dem IBM-PC (respektive seinen Derivaten Sidecar und Bridgeboard) ausgiebig und gut kommunizieren kann, dürfte hinreichend bekannt sein. Neu hingegen ist die Möglichkeit, mit einer anderen Spezies von PC Daten auszutauschen. Gemeint sind die kleinen, sehr beliebten und weit verbreiteten Sharp-PCs. Transfile von Yellow-Computing ermöglicht diese Ankopplung und soll daher einmal unter die Lupe genommen werden.

Bei Sharp steht hier das Kürzel PC für PocketComputer. Eine Vorstellung dieser Rechner ist wohl nicht notwendig, ein jeder wird sie kennen. Ob Schüler, Student oder Ingenieur, jeder wird seinen PC als Hilfsmittel schätzen. Bisher standen diese Rechner aber ziemlich isoliert im Raum. Von Sharp gibt es zwar ein Kassetten- Interface, mit dem man seine Programme auf einem normalen Kassettenrecorder abspeichern kann, doch jeder wird die Umständlichkeit dieser Prozedur kennen oder sie sich vorstellen können. Ebenso sieht es mit dem Drucker aus, der nur auf schmale Kassenstreifen druckt und dazu noch teures Thermopapier benötigt.

Die Firma Yellow-Computing bietet

seit neuestem aber ein Interface an, mit dem man seinen Sharp an den Amiga ankoppeln kann. Das Paket besteht aus einem Verbindungskabel, einer Programmdiskette und einem Handbuch.

Zur Installation wird das mitgelieferte Kabel bei ausgeschalteten Rechnern einfach an den zweiten Mausport des Amiga und den Interface-Stecker des Sharp angesteckt. Erwähnenswert ist, daß nur der jeweils an das Kabel anzusteckende Rechner ausgeschaltet werden muß. Läßt man das Kabel immmer im AMIGA eingesteckt, kann man ohne Ausschalten desselben jederzeit den (ausgeschalteten) Sharp anschließen.

Die Diskette ist nicht kopiergeschützt, was die Erstellung einer Sicherheitskopie erleichtert. Im Handbuch wird auch ausdrücklich darauf hingewiesen, nur mit einer Kopie der Originaldiskette zu arbeiten. Auch sonst ist die (deutsche) Dokumentation gut gelungen. Alle Features von Transfile sind gut erläutert, und es werden Beispiele zu allen Aktionen gegeben.

Die Bedienung ist ebenfalls benutzerfreundlich ausgefallen. Das fängt schon bei der Installation an. Auf der mitgelieferten Diskette befindet sich eine Batch-Datei, welche die Installation von Transfile auf einer Workbench-Diskette selbständig vornimmt.

Hier ergibt sich aber ein kleiner Nachteil: Der Programmierer von Transfile war wohl der Meinung, daß man nur einen PC braucht und läßt deswegen alle Dateien, die zum Ansprechen des Bridgeboard bzw. Sidecars notwendig sind, von der Workbench löschen. Ebensowenig hat man an eine Installation auf Festplatte gedacht.

Transfile selbst kann über ein Icon auf der Workbench gestartet werden. Es erscheint ein Menübalken, der es erlaubt, den Sharp-Typ auszuwählen und der die verschiedenen Möglichkeiten zu Datentransfer, Datenspeicherung und Bearbeitung bereithält. Die Bedienung ist sehr einfach, selbst der Neuling findet sich sofort zurecht. Bei der Datenspeicherung kann man zwischen den Typen Binaer und ASCII wählen. Bei ASCII-Speicherung werden die vom Sharp verwendeten Token in Klartext verwandelt, was die Edierung von Programmen auf dem AMIGA ermöglicht. Neben dem Edieren mit dem systemeigenen ED können die Dateien auch ausgedruckt werden.

Wer kann mit wem?

Herausragend sind die verschiedenen Kombinationen der verwendbaren Rechner. Alle AMIGA-Modelle arbeiten mit Transfile zusammen, was bei der Kompatibilität der AMIGAs nicht verwundert. Bei den verschiedenen Sharp-Modellen sieht dies jedoch anders aus.

Untereinander nicht kompatibel, vermag Transfile es trotzdem, mit einer ganzen Reihe verschiedener Sharps zusammenzuarbeiten. Dies wären: PC 1260/61/62, PC 1350/60, PC 1401/02/03/21/25/30/50/60/75. Die Anpassung an den jeweiligen Sharp-Typ erfolgt über verschiedene Tokentabellen, die von einem Menü aus angewählt werden können. Ebenfalls möglich ist eine Voreinstellung des Sharp-Typs, was eine Auswahl per Hand nach dem Programmstart unnötig macht.

Fazit

Transfile stellt eine überaus nützliche Erweiterung zum Sharp PC dar. Bequemes Edieren, schnelles Abspeichern von Daten sowie gute große Ausdrucke stellen die Hauptpluspunkte von Transfile dar. Führt man sich noch vor Augen, daß ein Drucker/Kassetteninterface von Sharp rund das 1,5-fache kostet, steht die Entscheidung für Transfile fest.

Hersteller: Yellow-Computing & ECPS Im Weingarten 21, D-7101 Hardthausen 3 Tel. 07139/8355 Preis: DM 99.-

THE GRAPHICS STUDIO



Wenn man ein Grafikprogramm testet, kommt man nicht drum herum, es mit dem ja nun fast schon klassischen Deluxe Paint II von Electronic Arts zu vergleichen. Um es gleich vorweg zu sagen: The Graphics Studio von Accolade kann DPaint nicht das Wasser reichen, obwohl manche zeichentechnische Funktionen besser gelöst wurden als bei DPaint.

m das insgesamt sehr ansprechende Programm nicht vorzuverurteilen, möchte ich zuerst auf die positiven Seiten des Graphics Studios eingehen. Zu den herausragenden Merkmalen des Programms gehören die Funktionen zum Zeichnen geometrischer Grundfunktionen (Linien, Rechtecke, Kreise). Es

können sowohl einfache, aber auch aneinandergereihte Linien gezeichnet werden. Auch Linien, die alle einer Koordinate entspringen, sind vorgesehen. Kreise und Rechtecke können wahlweise auch konzentrisch gezogen werden, eine Funktion, die sehr nützlich sein kann. Insgesamt bekommt man den Eindruck, daß dieses Programm mehr für Zeichner denn für Grafiker oder Designer gedacht ist. Ein besonderes Merkmal des Graphics Studios ist das Füllen von Flächen mit vordefinierten Mustern, die frei edierbar sind. Dies geschieht mit einer großartigen Geschwindigkeit, die bis jetzt ihres gleichen sucht. Natürlich besitzt THE GRAPHICS STUDIO alle Grundfunktionen von Zeichenprogrammen. Ein besonderes Feature, das diesem Malprogramm mitgegeben worden ist, ist das Erstellen von eigenen Slideshows. Das Programm hierzu ist bereits auf der Diskette enthalten. Apropos Diskette, der Kopierschutz des Programms ist schlichtweg genial zu nennen. Der Anwender hat die Möglichkeit, von seinem Original mit einem einfachen Kopierprogramm (Workbench) eine Sicherheitskopie anzulegen. Jedoch wird vor dem Starten des Programms von der Kopie die Originaldiskette verlangt, was es einem illegalen Benutzer unmöglich macht, dieses Programm zu benutzen. Ein weiterer Pluspunkt des Programms ist das gut gemachte Handbuch, das auch dem Anfänger die Funktionen des Programms näherbringt. Es ist zwar in Englisch geschrieben, aber auch für Anwender mit Schulenglisch leicht verständlich.

Nun zu den nicht so gut gelösten Seiten des Programmpakets: Die größten Negativpunkte des Graphics Studios ist die fehlende Ausnutzung der PAL-Auflösung sowie die nicht implementierte Tastaturanwahl der Funktionen. Es ist - gelinde gesagt - etwas mühselig, alle gewünschten Tools per Maus auszuwählen. Auch ist keine freie Wahl der Auflösung und der zu verwendenden Farben (sprich Bitplanes)



vorgesehen, nur zwischen LORES und HIRES ist wählbar. Das Programm konvertiert eigenständig beim Laden LORES-Bilder ins HIRES-Format (positiv), was jedoch die Ladezeit extrem verlangsamt (weniger positiv). Will man jedoch ohne Laden eines neuen Bildes die Auflösung verändern, bleibt das Bild nicht erhalten. Was die Programmierer sich dabei gedacht haben, wissen die Götter (ich nicht). Auch die Handhabung der Brushes ist nicht überzeugend gelöst worden. Es

ist zum Beispiel zwar möglich, Brushes aus dem Bild zu lösen und auf einen anderen Screen zu bringen, jedoch ist es nicht möglich, diese in der Größe zu verändern oder zu rotieren. Diese Funktionen sind zwar implementiert, jedoch funktionieren sie nur direkt auf dem aktuellen Bild. Daß damit auch keine perspektivische (oder dreidimensionale) Darstellung oder das BENDEN von Brushes zu erreichen ist, ist ja somit klar.

Ein Tip: Wer sich also dieses Programm zulegen will, sollte es sich vorher genau ansehen, um keine Enttäuschung zu erleben. Vor allem sollte er sich überlegen, für welchen Zweck er ein Malprogramm benötigt. The Graphics Studio läuft auf allen Amigaversionen mit optionalem externen Laufwerk und Drucker.

THE GRAPHICS STUDIO

Bewertung:

- + komfortable geom. Grundfunktionen
- + schnelles Füllen (mit Patterns)
- + Slideshow
- + benutzerfreundlicher Kopierschutz
- keine PAL-Auflösung
- keine Tastaturfunktionen
- beschränkte Auflösungswahl
- unkomfortable Brushhandhabung

Anbieter: GTI, Oberursel Preis: DM 105.-

ENDE

Pix-Mate: Grafik-Prozessor... der Vielseitige

DM 129.-



- Neue »Hyper Slice«-Blitter-Technologie beschleunigt Grafik-Manipulation um den Faktor 10
- Arbeitet mit allen Bildformaten, konvertiert Bildformate in Sekundenschnelle
- Pix-Mate konvertiert HAM-Modus perfekt in Grautöne.
 Ideal für Desktop
- Der Histogramm-Equalizer justiert Farbkontraste in allen Variationen
- Über 3000 Bildeffekte können generiert werden!
- Die Spectra Plus Farb-Palette kontrolliert die Amiga Farbvielfalt in noch nie dagewesener Brillianz und Präzision
- Pix-Mate unterstützt alle Amiga-Grafik Modi (auch Halfbright 64)
- Pix Mate wird kpl. mit deutschem Handbuch geliefert



Borsigallee 18 6000 Frankfurt/M. & 069-410071/72 Schweiz: MICROTRON Bahnhofstraße 2 CH-2542 Pieterlen Tel. 032872429



- Software zur Konfiguration von verschiedenen SCSI-Hard-Disk Laufwerken.
- Hyperschnell durch DMA. (Direkter Speicherzugriff)
- Mit 3.5" HD-Laufwerken kann eine File-card aufgebaut werden!!
- Ab Lager lieferbar, versandkostenfrei!! natürlich von:



Borsigallee 18 6000 Frankfurt/M. & 069-410071/72 Schweiz: MICROTRON Bahnhofstraße 2 CH-2542 Pieterlen Tel. 032872429

WERBEDEMOS MIT TV SHOW

Für den AMIGA gibt es ja jetzt schon viele Demonstrationsprogramme, die man oft in Geschäften sieht. Diese Demos kann man billig als Public Domain erhalten. Jetzt hat die Softwarefirma Zuma Group "TV SHOW" auf den Markt gebracht. Inwieweit dieses Programm als Demoprogramm genutzt werden kann oder wofür es sonst zu gebrauchen ist, werden Sie in diesem Test erfahren. Erster Eindruck: Geliefert wird die Diskette und das dicke Handbuch in einer sehr robusten Videocassettenschachtel.

ieser erste gute Eindruck hielt aber nicht lange an. Ich wollte die Diskette gleich starten, ohne das Handbuch studiert zu haben. Selbststartend war das Programm nicht, also probierte ich es von der Workbench aus. Dort erschien auch gleich ein Icon zum Anklicken. Ich tat dies, und es öffnete sich ein Fenster, in dem mehrere Funktionen zu sehen waren. Da ich gleich etwas sehen wollte, klickte ich ersteinmal "TV SHOW Play" an. Nach kurzer Ladezeit wurde die Workbench verlangt (ich arbeitete mit einem Laufwerk), die ich dann auch einlegte. Doch kurz darauf kam eine Requestermeldung "File not found" und ich mußte zum Workbenchscreen zurückkehren. Da dieser Versuch fehlgeschlagen war, klickte ich das Icon "Learn" an. Nun folgte eine 10-minütige Erklärung des Programms. Sie zeigt die wichtigsten Funktionen und ist gleichzeitig eine Demonstration für die Leistung des Programms, weil diese Erklärung mit ihm geschrieben wurde. Hierdurch begriff ich die meisten Funktionen und legte los.

Kein Kopierschutz

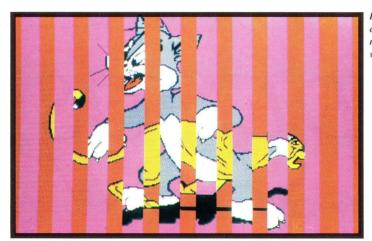
Als erstes legte ich mir jetzt eine Sicherheitskopie der Diskette an, was nicht schwer war, da das Programm keinen Kopierschutz hat. Das ist sehr angenehm, weil man es dadurch auch auf eine Harddisk kopieren und von dieser aus starten kann.

Was kann TV SHOW?

TV SHOW ermöglicht es, verschiede-

ne Bilder (IFF-Format) nacheinander auf dem Bildschirm erscheinen zu lassen. Für die Übergänge von einem Bild zum nächsten gibt es viele verschiedene Möglichkeiten. Man kann die Bilder überblenden, aus verschiedenen Richtungen auf den Bildschirm scrollen lassen, zeilenweise auf den Bildschirm bringen oder sich durch per Zufall gesetzte kleine Bildabschnitte aufbauen lassen. Es gibt natürlich noch mehr Möglichkeiten, die ich hier jetzt aber nicht alle aufzählen möchte. Bei den Bildern kann man auch die Farben cyclen lassen, wie man es von anderen Bildern, die man zum Beispiel mit Deluxe Paint II erstellt hat, kennt. Auch Brushes (kleine Grafiken) finden bei diesem Programm eine Verwendung. Sie können nach Belieben auf dem Bildschirm positioniert werden. Die Position eines Brushes wird mit der Maus festgelegt. Zusätzlich kann man noch bestimmen, wie ein Brush auf dem Bildschirm erscheinen soll. Hierbei stehen die gleichen Möglichkeiten wie bei den Bildern zur Verfügung, wobei man einen Brush aus mehr Richtungen zu seinem Platz auf dem Bildschirm scrollen lassen kann.

Leider ruckt die Bewegung der Brushes meistens (außer bei der horizontalen und vertikalen Bewegung). Zu den Bildern kann man noch durch den eingebauten Sprachsynthesizer einen Text sprechen lassen. Wird jedoch nachgeladen, während der Computer "spricht", kommt er ins stocken. Damit derjenige, der sich einen erstellten Ablauf ansieht, auch tätig werden kann, ist es möglich, Tasten abzufragen, die dann den weiteren Ablauf bestimmen.



Die Erstellung eines Demos wird mit der Maus vorgenommen.

Der Editor

Das Erstellen eines Ablaufes ist sehr einfach, da man alles mit der Maus macht. Man muß überall nur irgendwelche Icons anklicken, die zum Beispiel die Art des Erscheinens eines Bildes auf dem Bildschirm festlegt (Bild 1). Per Tastatur muß man nur den Text, der gesprochen werden soll, eingeben. Hat man einen Ablauf fertig, so gibt es noch die Möglichkeiten, etwas anzufügen, einzufügen oder zu löschen. Auch das geschieht alles mit Hilfe der Maus.

Das Handbuch

In dem mitgelieferten Handbuch, das leider in Englisch geschrieben ist, findet man zu jeder Funktion eine sehr ausführliche Erklärung. Da das Handbuch leicht verständlich geschrieben ist, versteht man es auch, wenn man keine großen Englischkenntnisse besitzt.

Drucker

Das Programm verfügt auch über eine Funktion mit der man sich seinen Ab-

lauf ausdrucken lassen kann. Man erhält dann eine Tabelle, in der der ganze Ablauf aufgelistet wird (in welcher Reihenfolge, welches Bild, wie lang es auf dem Bildschirm bleibt, welcher Text dazugesprochen wird u.s.w). Man erhält also eine Art Drehbuch.

Fazit

Für den Heimgebrauch ist dieses Programm wohl nicht zu gebrauchen, auch für Videoamateure nicht, die Computerbilder oder gar Animationen in einen Videofilm einbinden wollen. Das Programm ist zwar genlockkompatibel, da aber das Scrollen der Brushes oft ruckt, sollte man sich für das Geld lieber ein Animationsprogramm kaufen. Als Demonstrationsprogramm für Geschäfte ist es nur bedingt geeignet. Zwar kann ein Geschäft mit ihm Reklame für sich selbst machen, was mit anderen Demonstrationsprogrammen nicht möglich ist, da diese einen festen Ablauf haben. Dafür demonstrieren diese die Leistung des Computers oft besser. Mich würde zum Beispiel die JUGGLER-Demonstration (siehe KICKSTART PD 10) viel mehr faszinieren als eine ruckende Darstellung von Bildern. Außerdem ist der Preis von ca 200.- DM viel zu hoch.

Der Sprachsynthesizer ist nur wenig zu gebrauchen, da er nur die englische Aussprache beherrscht.

Positiv ist, daß alle Bilder im IFF-Format sein müssen und deshalb kompatibel zu den gängigen Malprogrammen sind. Es werden sogar HAM- Bilder und Bilder mit einer Auflösung bis zu 700 * 480 Punkten akzeptiert. TV SHOW läuft auf allen drei AMIGA mit mindestens 512 KB. Vom Hersteller wird empfohlen, über 1 MB zu verfügen, weil es sonst zu Problemen kommen kann, wenn man mit Bildern arbeitet, die viel Speicherplatz verbrauchen (Bilder im H.A.M.- Modus oder in der höchsten Auflösung), ansonsten läuft aber alles auch mit 512 KB. Zu bemerken wäre noch, daß das Programm kein einziges Mal abstürzte, während ich es testete.

TV SHOW

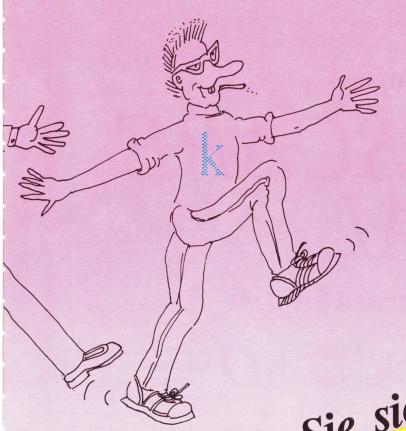
- + leichte Bedienung durch die Maus
- + übersichtlich aufgebaut
- + leicht erlernbar
- + kein Kopierschutz
- + ausführliches Handbuch
- + akzeptiert alle Auflösungen
- + Bilder sind im IFF-Format
- Handbuch in Englisch
- Preis zu hoch
- Darstellung ruckt oft

Hersteller: Zuma Group inc.

Anbieter : Intelligent Memory, Frankfurt

Preis : ca. 200.- DM

ENDE





K. Schneider / O. Steinmeier

GRUNDLEHRGANG

Der richtige Einstieg

Heim Verlag

Holen Sie sich auch AMIGA-GRUNDLEHRGANG AUS DEM INHALT:

1. Die Hardware des AMIGA
1. die versch AMIGA-Modelle Druckers des versch AMIGA-Modelle Druckers des verschens des verschens

DM 59,-Buch und Diskette unverbindlich empfohlener Verkaufspreis WICHTIGE MERKMALE:

WICHTIGE MERKMALE:

WICHTIGE MERKMALE:

Das Buch für den richtigen Einstieg mit

* Das Buch für den richtigen Einstieg mit

* Auf ür den Kest ein den Kest ein den Gemandagen der Computertechnik und

Seiten werden der Leser leicht verlächtigen der Computertechnik und

Seiten werden der Leser leichtechnik und

Seiten werden der Leser leicht der Keiten der Kapten der Kapten der kan einste der Mater die Mausstems gewidmet. Hier erlächt die Maustern sein der Kent dem Kapten der Kapten d

- führliche Befehlsübersicht mit detaillierreit Erklärungen 4. Zum Training 4. Programm-Diskette mit allen abgedruck-* Programm-Sachworterklärung (Fachwör-ten Listings * Ausführlicher ter-Lexikon) * Ausführlicher ter-Lexikonis mit entspr. Verweisen)

BESTELL-COUPON

an Heim-Verlag Heidelberger Landstraße 194 6100 Darmstadt-Eberstadt

PLZ, Ort_

zzgl. DM 5,- Versandkosten (unabhängig von der bestellten Stückzahl)

per Nachnahme

Name, Vorname

Straße. Hausnummer

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte

Heim Verlag

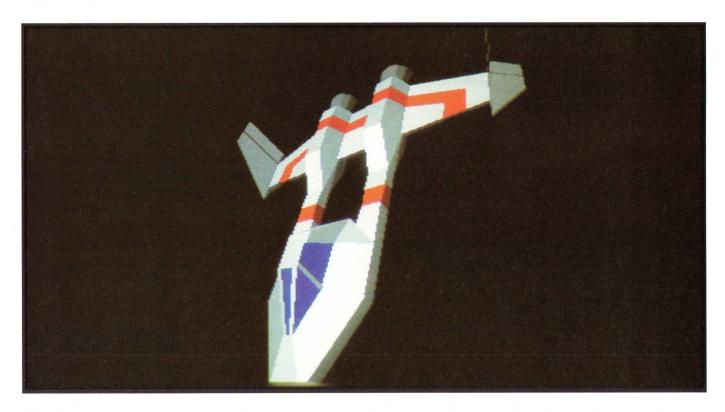
Heidelberger Landstraße 194 6100 Darmstadt-Eberstadt Telefon 06151-56057

VIDEOSCAPE PAL

3D-Animation neu geformt

Als letztes Jahr das Programm "Aegis Videoscape 3D" auf den Markt kam, war das Staunen groß. Schnelle Animationen in der dritten Dimension flitzten mit atemberaubender Geschwindigkeit über so manchen Amiga-Screen. Ob Kugeln, Raumschiffe, Autos oder sogar fraktale Gebilde, mit Videoscape kam Leben und Bewegung in den Amiga. Jetzt liegt ein Update des Programms vor, und wir wollen vorstellen, was sich alles geändert und verbessert hat.

as erste, was sofort nach dem Öffnen der Packung auffällt, ist das ins Deutsche übersetzte Handbuch. Wer den witzigen und lockeren Stil des englischen Handbuchs gekannt hat, dem wird vielleicht die etwas nüchterne Übersetzung nicht so liegen, aber dafür entfällt das lästige Nachschlagen in Langenscheidts gesammelten englischen Übersetungsversuchen. Nach dem Booten meldet sich das Programm in fast altbekannter Form. Alle Menüs, Gadgets und Requester erscheinen leider nach wie vor in englischer Sprache. Änderungen im Arbeitsscreen fallen zwar auf den ersten Blick kaum auf, sind aber da! So ist das "Window-



Drag-Gadget" in die obere rechte Ecke gewandert, und der Menüpunkt "Clear" unterteilt sich jetzt in zwei Untermenüpunkte, nämlich "Objects only" und "All settings".

Hat man bei der ersten Version des Programms "Clear" angewählt, wurde leider der gesamte Speicher gelöscht. Dies hat zur sehr langen Ladezeiten, gerade bei komplizierten Animationen, geführt. Jetzt hat man die Möglichkeit, mit "Objects only" nur Objekte zu löschen, und man kann z.B. verschiedene Objekte in beliebigen Szenen ausprobieren, ohne lange Floppyzugriffszeiten ertragen zu müssen. Wählt man "All Setttings", entspricht dies explizit dem alten "Clear"-Befehl, und der gesamte Arbeitsspeicher wird gelöscht. Lädt man Objekte oder Szenen, wird jetzt während dem Laden die Anzahl der Polygone und Punkte angezeigt.

Die wichtigsten Neuerungen, denen das Programm auch seinen Namen verdankt, sind die neuen PAL-gerechten Grafikmodi. Videoscape arbeitet ja im sogenannten "Overscan"-Modus, d.h., es wird der gesamte darstellbare Bildschirmbereich ausgenutzt (keine Ränder oder Bidbegrenzungen an oberen oder untereren bzw. linken oder rechten Bildschirmrand). Aegis verwandte diesen Darstellungsmodus, um die damit erstellten Animationen attraktiver für Video zu machen. Leider erschien bislang auf PAL-Amigas immer ein schwarzer Streifen, erfüllt vom absoluten Nichts, am unteren Bildrand, egal ob auf dem Screen oder auf den hergestellten Video. Doch damit ist

jetzt endgültig Schluß. Die neue PAL-Version erkennt automatisch (wie auch unzählige andere Programme), ob ein PAL- oder NTSC-Amiga angeschlossen ist. Hier kann man nur sagen: Glücklich ist der Europäer, denn er kann ab sofort auf folgende Darstellungsmodi zurückgreifen: 352x280 in 32 Farben, 704x280 in 16 Farben noninterlaced und 352x560 in 32 Farben bzw. 704x560 in 16 Farben interlaced. Aber nicht nur die Auflösung wurde verbessert, sondern noch eine ganze Anzahl von weiteren Punkten. So wurde z.B die Länge der Animfiles drastisch reduziert: Sie sind im Durchschnitt nur noch halb so lang wie unter der ersten Version von Videoscape! Daß so erstellte Animationen auch ca. 10-20% schneller laufen, ergibt sich von selbst.

Neue Farben braucht das Land!

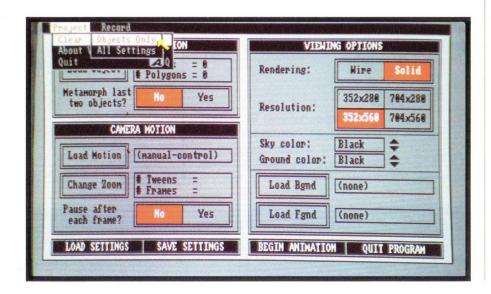
Eine ganze Anzahl von neuen Farben wurde in die Version 1.1 implementiert. So kann man jetzt auch transparente Farben zum Einsatz bringen, um so z.B Glasflächen oder ähnliches darzustellen. Doch was wären Farben ohne Licht? Jetzt hat man die Möglichkeit, bis zu 4 Lichtquellen in frei wählbarer Intensität zu setzen und so seine Animationen perfekter auszuleuchten. Die Hilfsprogramme "EGG" und "OCT" liegen auch in einer überarbeiteten Version (V 1.10) vor. So kann man jetzt "EGG" sofort mit der Einga-

be einer "0" beenden, ohne erst das ganze Programm durchgehen zu müssen. "OCT" erlaubt es in der neuen Version, die Objektdaten in Binärform abzuspeichern. Vorteil hier: Die Objektfiles werden kürzer und können so schneller geladen werden. Möchte man die Objektdaten jedoch im ASCII-Format speichern, kann man diese mit "OCT" beliebig hin- und herkonvertieren.

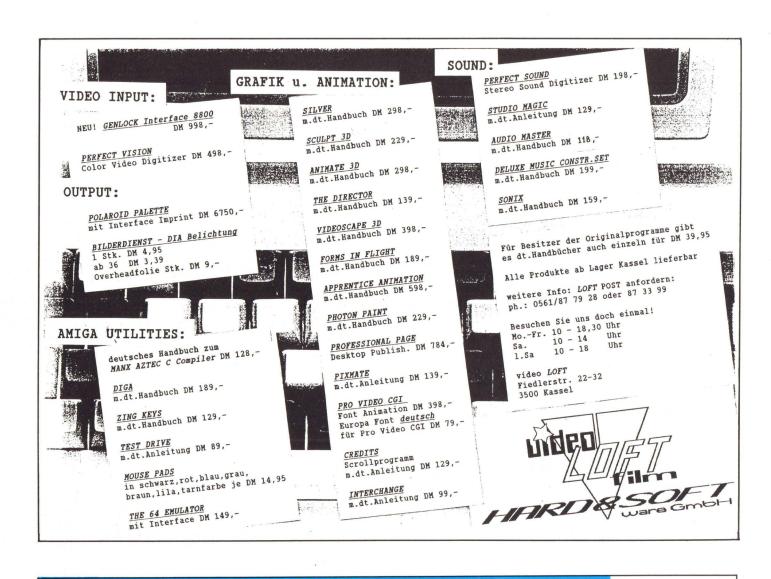
Des weiteren können jetzt Kameraund Objektmotionfiles im direkten Austausch verarbeitet werden. So kann man z.B. die Bewegung eines Objekts ohne Probleme als Kamerafahrt einsetzen und umgekehrt.

Bugs go Home

Fehlerfreie Dinge gibt es wenig, schon gar nicht fehlerfreie Programme. So waren auch in der ersten Version von Videoscape leider ein paar "Bugs" versteckt, von denen aber die Version 1.1 befreit ist. So gab es in der Version 1.0 stellenweise Probleme mit dem Multi-tasking, oder auch Probleme beim Einsatz überstarker Lichtquellen. Diese sind jetzt behoben, und uns ist es trotz intensiver Anwendung des Programms nicht gelungen, dieses in das Nirwana einer Guru-Mediation zu versetzen. Bleibt zu sagen: Wer Besitzer einer "alten" Version von Videoscape ist, der sollte schnellstens den UpDate-Service in Anspruch nehmen, und wer noch kein Videoscapebesitzer ist (und sich für bewegte 3D-Grafik begeistert), der sollte es schleunigst werden, es lohnt sich!



+ kürzere Animfiles + schnellere Animationen + Hilfsprogramme "EGG" und "OCT" in neuer Version - Menüs in Englisch



* * AUTOREN GESUCHT

Sie

- ... haben eine gute Programmidee
- ... wollen ein Buch schreiben
- ... kennen eine Menge Tips u. Tricks
- ... möchten Ihre Erfahrungen weitergeben

Wir

- ... bieten Ihnen unsere Erfahrung
- ... unterstützen Ihre Ideen
- ... sind ein leistungsstarker Verlag
- ... freuen uns von Ihnen zu hören

Buch



Heim Verlag

Programm



Schreiben Sie uns

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

GRAFIK SPEZIAL

PROGRAMMDISKETTE

Alle im Sonderheft veröffentlichten Programme inklusive der dazugehörigen Quelltexte sind auch auf Diskette erhältlich.

Diese bietet sich für alle Anwender an, die die Programme nutzen, aber die Listings nicht abtippen möchten, oder für diejenigen, die nicht über den jeweils verwendeten Compiler (C oder Modula 2) verfügen. Die Diskette (nebenstehend abgebildet) enthält folgende Programme:

- Programm Picswitch inkl. C-Source

Dieses Programm dient zum Erstellen von Animationen aus IFF- Bildern, wozu ein Differenzbildverfahren verwendet wird, das die Größe der Animationsdateien reduziert und die Abspielgeschwindigkeit erhöht.

- Programm Transformer inkl. C-Source Transformer arbeitet nach dem Prinzip des Texture Mapping, womit Sie beliebige IFF-Bilder um Kugeln oder Röhren gewickelt darstellen können.
- Programm Pflanze inkl. Modula 2-Source Dieses Programm simuliert mathematisch natürliche Wachstumsformen und stellt diese grafisch dar.

- Programm WBCop inkl. Modula 2-Source

Ein Programm zum Verändern der Farblisten eines Workbenchbildschirms. Es können sowohl der Hintergrund wie auch der Text oder die Umrandungsfarben in durchlaufenden Farben dargestellt werden.

- Programm ObjConvert in BASIC

Dieses Programm dient der Übertragung von Objektdateien von Sculpt 3D zu Videoscape 3D.

Desweiteren finden Sie alle Listings und die dazugehörigen Programme der Grundlagenartikel dieses Hefts auf dieser Diskette.



Bitte senden Sie mir die Original-Programmdiskette zur KICKSTART GRAFIK SPEZIAL zum Preis von DM 25.-

Ich zahle keine Versandkosten Den Betrag begleiche ich durch beigefügten Verrechnungscheck

Name:		

Straße:____

Plz/Ort:

Einsenden an:

Heim-Verlag Heidelberger Landstraße 194 6100 Darmstadt/Eberstadt

SOURCE LEVEL DEBUGGER

Die Fortsetzung der Aztec-DB Legende?

Dem Aztec C-Compiler haben seine Entwickler bisher schon einen guten Debugger mitgegeben, doch war schon in der 3.4'er Version die Rede von einem Source Level Debugger, mit dem das Debuggen zum Kinderspiel werden sollte. Nun ist er endlich da. Die Frage lautet also: Ist er so gut, wie erhofft wurde?

m Moment wird zusammen mit dem SDB lediglich eine Vorversion des Handbuchs mitgeliefert, wobei alle registrierten Benutzer nach der Fertigstellung des vollständigen Handbuchs dieses zugeschickt bekommen (laut Handbuch bis Anfang April). Da es sich bei dem SDB wie auch bei dem C-Compiler um ein amerikanisches Produkt handelt, ist die Anleitung ebenfalls in Englisch abgefaßt.

Im Lieferumfang befindet sich neben dem Handbuch (natürlich) auch noch eine Diskette, auf der sich neben dem eigentlichen Debugger noch ein Directory mit Demos befindet. Hier kann man zwischen 3 verschiedenen Files wählen, mittels derer man sich die Funktionen des SDB vorführen lassen kann. Dies ist sehr lehrreich; man sollte diese Demos keinesfalls weglassen, da sie nicht nur pure Theorie sind, sondern auch die Vorgehensweise beim Debuggen ein wenig erklären.

An dieser Stelle mag so mancher fra-

gen: Was ist eigentlich ein 'Source Level Debugger' und was ist da anders? Wenn man normalerweise einen Debugger verwendet, steht man meistens vor einer ganzen Menge von Registerangaben, einigen Flags sowie einer oder mehrerer Zeilen 68000-Assembler-Code. Wenn man diesen Assemblercode beherrscht, stellt das natürlich kein allzu großes Problem dar. Doch viele Leute programmieren ja in C, weil sie Assembler nicht oder noch nicht können. Weil jedoch C in Assembler übersetzt wird, können viele den (recht guten!) normalen DB nicht verwenden. Somit heißt es für sie nur noch: Probieren, probieren und nochmals probieren. An diese freie Stelle tritt nun der neue SDB. Das Besondere an ihm ist, daß beim Debuggen auch der Sourcetext angezeigt wird und daß beim Tracen nicht ein Assembler-Befehl, sondern ein C-Befehl ausgeführt wird. Somit werden zum Beispiel nicht mehr folgende Zeilen einzeln abgearbeitet:

move.1 GfxBase,a6 move.1 RastPort,a1 move.1 XPos,d0 move.1 YPos,d1 jsr -246(a6)

sondern lediglich die Zeile:

Draw (RastPort, XPos, YPos);

Dies erleichtert natürlich die Arbeit erheblich und gibt, durch die ausschließliche Verwendung von C, auch reinen C-Programmierern die Möglichkeit, ihr Programm rechnergestützt zu debuggen.

Natürlich bezahlt man diese Vorteile mit einigen Nachteilen, so kann man z.B. nicht mehr beliebige Programme debuggen oder untersuchen, sondern nur solche, bei denen ein zusätzliches File ([Name].dbg) sowie Source vorhanden sind. Außerdem funktioniert SDB nur mit .dbg-Files, die auch mit dem neuen Aztec C-Compiler V 3.6 erstellt wurden; die von Version 3.4 erzeugten .dbg-Files werden nicht angenommen, wodurch man gezwungen ist, sich das Update zu besorgen, falls man die Version 3.4 oder frühere besitzt. Da stellt sich allerdings die Frage, warum nicht auch an die V3.4-Besitzer gedacht wurde, denn schließlich kostet das Update auch noch mal Geld...



Der Bildschirmaufbau des SDB.

Das Fenster des SDB ist in 3 Zonen aufgeteilt (siehe Bild 1):

- Das Fenster f
 ür den C-Source, die aktuelle Zeile wird invertiert dargestellt.
- 2.Das Kommando-Fenster (eigentlich nur eine Zeile), in dem man die Debug-Befehle eingibt.
- 3.Das Fenster für alle Ausgaben des Debuggers.

Dabei kann man die Gesamtgröße des Windows mittels Size-Gadgets und die Größe der beiden Fenster 1 und 3 mittels des Gadgets rechts von der Kommandozeile verändern. Es fällt positiv auf, daß man beim Debuggen nicht auf die Maus angewiesen ist, da man das Scrollen der beiden Fenster sowie ihre Größe auch über Tastatur verändern kann. Des weiteren kann man das SDB-Window, das sich beim Programm-Lauf ständig nach vorne schaltet, mittels F10 nach hinten bzw. wieder nach vorne bringen.

Wenn man größere Programme debuggen will, sollte man die Source-Files in die Ram-Disk legen, da bei den Sprüngen ständig die neue Zeile angezeigt werden muß und deshalb auf Disk zu-

gegriffen wird. Dies empfiehlt sich auch, wenn mehrere Source-Files zu dem fertigen File geführt haben, denn hier muß ja zudem noch jeweils zwischen den Files umgeschaltet werden. Das Ausgabefenster verfügt, wie das Source-Window auch, über einen Pufferspeicher von etwa 2000 Zeichen, auf die man beliebig zurückgreifen kann. Auch das Kommando-Fenster verfügt über eine History von 512 Zeichen, in der gleiche Befehle nur einmal abgelegt werden, was recht gut für die Ausnutzung des Speichers ist. Es wäre allerdings schön gewesen, wenn man diese beiden Werte oder wenigstens die Größe des Puffers für das Ausgabe-Window frei bestimmen könnte.

Der SDB verfügt außerdem noch über die Möglichkeit, Macros zu definieren, die dann automatisch eine Reihe von Befehlen ausführen, ohne daß der Benutzer alles per Hand eingeben muß. Man kann sich also eine kleine 'Bibliothek' mit Grundstrukturen zum Debuggen anlegen.

Von den 'normalen' Befehlen möchte ich an dieser Stelle nur die beiden wohl erstaunlichsten ansprechen: den Execute- und den Print-Befehl.

Mittels des Execute-Befehls kann man normale C-Befehle von einer einfachen Addition wie 'e x++' bis hin zu Zeilen wie 'e printf ("Hallo!")' ausführen, der Print-Befehl kann ganze Strukturen mit Name, Art des Eintrags (int, Struct, ...) und Inhalt ausgeben, z.B. mittels 'p *win', wenn 'win' als Zeiger auf eine Window-Structur definiert war, und das alles, ohne eine einzige Hex-Zahl eingeben oder interpretieren zu müssen.

Alle Aus- und Eingaben kann man, genau wie im normalen DOS auch, umleiten, des weiteren kann man auch sämtliche Befehle von Disk holen, was bei den Demos angewandt wurde.

Die mir vorliegende Version 3.6a hatte allerdings noch einige kleine Probleme; so konnte man z.B. ein mit 'g' gestartetes Programm nicht abbrechen, selbst dann nicht, wenn man z.B. das Line-Tracing angestellt hatte. Des weiteren kann man pro SDB-Lauf nur ein Programm debuggen, was allerdings nicht so wichtig ist, da man meist sowieso nur ein Programm bearbeitet. Wahrscheinlich werden diese kleineren Sachen noch korrigiert. Ansonsten ist der Aztec Source Level Debugger ein Programm, das für jeden ernsten Programmierer ein unbedingtes Muß ist und die 'täglich Arbeit' am Amiga stark vereinfachen kann. Jeder, der noch den alten 3.4'er Aztec besitzt sollte sich stark überlegen, ob er sich nicht das Update besorgt, zumal der SDB mit einem Preis von DM 149.recht preiswert ist.

Bezugsquelle: Merlin-Computer GmbH Tel.: 06196/481811

ENDE



KICKSTART GRAFIK SPEZIAL

Voller Tips, Tricks und Berichte für Anwender, Programmierer, Amiga- und Grafik-Fans! Aus dem Inhalt: Grundlagen der Grafikprogrammierung: Screenund Windowprogrammierung unter Intuition: Sämtliche IDCMP-, Window- und Screen-Flags und was sie dem Programmierer ermöglichen. Die grafischen Betriebssystemroutinen des Amiga: Grafikprogrammierung auf komfortable Art und Weise. Fonterstellung: So sind eigene Zeichensätze kein Problem. Copperprogrammierung: Spielereien mit

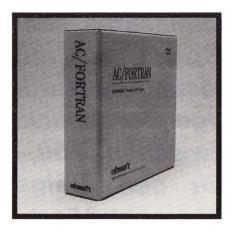
Farben und Auf ware: Der rich 6 Monitore und 3.000,- DM im Test. Druck



lösungen. Hardtige Durchblick: zwischen 750,für den Amiga gemacht: 3 Farb-

drucker stellen sich vor. Eingelesen: Digitizer für den Amiga. Listings: Texture Mapping: Wickeln Sie Ihre IFF-Bilder um Kugeln und Röhren. Picture Animation: Erzeugen Sie Trickfilme aus Ihren Bildern. Graphtale Pflanzen: Interessantes aus der Welt der mathematischen Grafik. Neue Grafik-Software: Animate 3 D, The Director, Aegis Videotitle, Butcher 2.0 deutsch, PIXmate. Fürs Auge: Eindrücke aus der professionellen Grafikszene. Außerdem: Bücher für Grafikprofis und solche, die es werden wollen. Ray-Tracing-Grundlagen.

FORTRAN FÜR DEN AMIGA



Jetzt bekommt der Amiga den professionellen Schliff: Auch für diese Maschine wird nunmehr ein leistungsfähiger Fortran-Compiler angeboten. Der Kandidat stammt von der Firma Absoft und heißt mit vollem Namen 'AC/Fortran for Amiga'. Über erste Eindrücke mit diesem hoffnungsvollen Sprößling soll hier berichtet werden.

um Test lag eine amerikanische Originalversion vor. Bezüglich der Software und der Dokumentation ist dies ganz in Ordnung, selbst wer in Englisch schwach ist, wird damit immer noch besser bedient sein als mit einer schlechten Übersetzung. Hinsichtlich der vor dem Öffnen der Originalverpackung zu lesenden Lizenz- und Garantiebestimmungen wird man für die kommerziell vertriebenen Versionen zumindest einen deutschen Beipackzettel erwarten. Die Bestimmungen selbst umfassen zwei Druckspalten und bewegen sich im üblichen Rahmen. Insbesondere dürfen die von den Benutzerprogrammen benötigten Laufzeit-Routinen an Dritte weitergegeben werden. Glücklicherweise heben sich diese Formulierungen deutlich von dem von Commodore verbreiteten Schwachsinn ab, wo man als stolzer Käufer etwa eines Amiga 500 die Systemdiskette erst nach Lektüre eines dreiseitigen Vertragswerks verwenden darf, dessen erste zweieinhalb Seiten in dilettantischster Diktion und unverständlicher Typographie den Benutzer

knebeln (man darf das Handbuch anscheinend nicht einmal übersetzen?) und Commodore von fast jeglicher Haftung freistellen, und in dessen letzter halber Seite dann schlußendlich verkündet wird, '... daß die obenstehende Beschränkung möglicherweise auf Sie nicht zutrifft.' Immerhin findet sich anschließend zur Klarstellung die ebenso brillante wie präzise Formulierung: 'Diese Garantie gibt Ihnen bestimmte gesetzliche Rechte, und Sie haben möglicherweise sonstige Rechte, die von Land zu Land verschieden sind.' Alles klar?

Lieferumfang

In einer stabilen Pappschachtel bekommt man das Handbuch, dem in der ersten Seite eine Diskette mit den Programmen eingesteckt ist. Die Verpackung ist solide, die Diskette scheint hinreichend geschützt, so daß keine Transportschäden zu befürchten sind. Das Handbuch besteht aus einem stabilen Plastikringbuch im amerikanischen 3-Loch-Format mit Einlegeblättern der Größe etwa 18,3 mal 21,6 Zentimeter. Immerhin ist es nach etwas Tricksen mit dem Locher möglich, eigene Blätter, etwa im Format DIN A 5, einzuheften

Arbeitsweise

AC/Fortran wurde nicht speziell für den Amiga, sondern ganz allgemein für Rechner mit der 68000-CPU entwickelt. Einige dieser Rechner und Betriebssysteme werden an verschiedenen Stellen des Handbuchs erwähnt. Da ist es ganz natürlich, daß die Fenster-Verwaltung des Amiga nicht besonders unterstützt wird. Keines der Programme des Fortran-Systems ist durch Anklicken zu starten, das gesamte System ist im CLI zu bedienen. Dies kommt der von Großrechnern her gewohnten Arbeitsweise recht nahe, und hier ist auch die für diesen Compiler angepeilte Zielkundschaft zu suchen. Auch sonst gibt es einige Macken, die zur Bedienung eher einen gestandenen Profi als einen unbedarften Anfänger erfordern.

So ist beispielsweise gleich zu Beginn einer Sitzung mittels des Befehls

```
MAKEDIR RAM:C
COPY SYS:C TO RAM:C
ASSIGN C: RAM:C
MAKEDIR RAM:L
COPY SYS:L TO RAM:L
ASSIGN L: RAM:L
COPY DF1:L TO L:
COPY DF1:C TO C:
STACK 70000
```

Bild:2 Befehlsdatei FORTRANINI zum Einrichtn des Arbeitsplatzes auf der Ramdisk

STACK der Stapelspeicher des Amiga zu vergrößern, da hier sämtliche lokale Variablen einer Programmeinheit hineingepackt werden. Dabei wird der Stack bei Unterprogramm-Aufrufen natürlich nicht geräumt; bei geschachtelten Aufrufen summiert sich somit der Platzbedarf. Empfohlen wird ein Wert von 70000 (Byte), die Voreinstellung für den Amiga ist 4000. Das Handbuch weist nonchalant darauf hin, daß der tatsächlich benötigte Platzbedarf leicht anhand der vom Compiler für die beteiligten Programmeinheiten erzeugten Liste ausgerechnet werden kann. Für den Fachmann ist dies übliche Routine, aber wenn einer bisher in Basic programmiert hat ...

Der vorgeschlagene Wert von 70000 Byte für den Stack genügt allerdings auch nicht immer, dies macht sich dann durch einen unerklärlichen Systemabsturz oder, wenn man Glück hat, mit einer der allseits beliebten Guru-Meditationen bemerkbar. So geschah es uns beim unten beschriebenen Rechenzentrum-Benchmark, für den wir dann vorsichtshalber den Stack auf 100000 Byte hochsetzten.

Bestandteile

Wie bereits mehrfach erwähnt, handelt es sich hier um einen Compiler; die verwendeten Programmeinheiten müssen also zunächst mittels eines Editors als Quelltext eingegeben, dann übersetzt und die dabei erzeugten Maschinenprogramme anschließend mit den benötigten Bibliotheks-Routinen zu einem lauffähigen Programm zusammengebunden werden. Das Übersetzen besorgt bei AC/Fortran der Compiler F77 mit mehreren Overlays, zum Binden gibt es den

Linker F77L. Weiter hat man zur Fehlersuche das Programm DEBUG und für die Pflege von Bibliotheken aus Maschinenprogrammen übersetzten den 'Librarian' F77LIB. Ein Editor ist im Lieferumfang nicht enthalten. Dies ist aber auch ganz in Ordnung so, denn einen hierfür besser geeigneten Editor als MicroEMACS wird man für Geld wohl nicht bekommen, und den gibt es ja bekanntlich als Public-Domain-Programm. Dem Tester stand allerdings nur die schon etwas angestaubte Version 1.1 für den Amiga zur Verfügung, aber selbst damit kann man leben.

Der Einstieg

Handbuch auspacken, Diskette einlegen, loslegen ... Das wäre so ein Traum! Sieht man von den üblichen Faxen, wie Sicherheitsdiskette anlegen, Installationshinweise lesen und ähnlichen, einmal ab, so ist zunächst erst eine kleine Systemdiskette zurechtzuzimmern. Die vier oben erwähnten Programme F77, F77L, DE-BUG und F77LIB sind in der Schublade (woanders sagt man Ordner dazu) die Overlays F77001.FC bis F77004.FC nebst einigen weiteren Hilfsdateien in der Schublade L unterzubringen. Hierzu muß man sich schon überlegen, was man von seiner Systemdiskette dafür löschen will.

Zum Testen wurde ein Amiga 500 mit einem Zusatzspeicher von 2 MByte und einem zweiten Laufwerk verwendet. Hier wird man natürlich sofort die Schubladen C: und L: von der Systemdiskette in die Ramdisk kopieren und dort auch noch die benötigten Programme des Fortran-Systems unterbringen. Da ohnehin das CLI benötigt wird, werden all diese Arbeiten selbstverständlich von einer Befehlsdatei erledigt. Bild 2 zeigt die bei diesem Test verwendete Befehlsdatei, die unter

dem Namen FORTRANINI auf der Systemdiskette abgespeichert war und nach Aktivierung des CLI mittels EXECUTE FORTRANINI aufgerufen wurde. Dazu mußten die Systemdiskette im eingebauten und die Fortran-Diskette im Zusatzlaufwerk liegen.

Der Compiler

Übersetzt wird durch Eingabe des Befehls F77 im CLI, wodurch das gleichnamige Programm in der Schublade C: aufgerufen wird. Diesem Befehl werden im Unix-Stil Optionen und die Namen der zu übersetzenden Dateien angefügt. Eine typische Befehlszeile sieht dann etwa so aus:

```
1> f77 -u test.f
```

Die Option '-u' bedeutet hierbei, daß die mit einem Stern an Stelle einer Dateinummer vorgenommene Ein-/ Ausgabe über den Bildschirm respektive die Tastatur läuft.

Des weiteren unterscheidet der Compiler sorgfältig zwischen Groß- und Kleinschreibung, jedenfalls in den Variablennamen. So hat beispielsweise die Variable 'DELTA' überhaupt nicht mit der Variablen 'delta' zu tun. Auch dies ist völlig unüblich, glücklicherweise gibt es auch hier die Möglichkeit, diesen Unfug über eine Compiler-Option, nämlich '-K', abzustellen. Übersetzungen wird man wohl also grundsätzlich mit den Optionen '-UK' ausführen.

In einer Hinsicht hält sich der Compiler geradezu sklavisch an die Fortran-Norm: Nicht explizit initialisierte Größen werden nicht automatisch auf Null gesetzt, sondern erhalten irgendwelche zufälligen Werte. Dies gilt nicht nur für die Elemente von Feldern, sondern bereits für einfache Variablen. Leider gibt es hier keine Compiler-Option, die

übersetzung des RZ-Benchmarks, 521 Zeilen					
		Sekunden	Zeilen/min	Zeilen/sec	Beschl.Faktor
Diskette		120	260	4.3	1
Diskette, Opt	ion -*	27	1157	19.3	4.4
Ramdisk		14	2232	37.2	8.6

Bild 3: Übersetzungszeiten des Rechenzentrums-Benchmarks.

```
LAPILACE-BENCHMARK
WEITERE MESSERGEBNISSE BITTE AN...
  J.W.GOETHE-UNIVERSITAET FRANKFURT
  INSTITUT FUER ANGEWANDTE PHYSIK
  PROF. DR. R. BECKER
  POSTFACH 11 19 32
D-6000 FRANKFURT 11
    DIMENSION U(0:100,0:100)
FUER AMIGA-AC/FORTRAN IST EINE DATA-ANWEISUNG ERFORDERLICH...
    DATA U/10201*0./
    DO 1 I=1,100
    U(I,100)=1
    U(I-1,0)=0
    U(0, I-1)=0
    U(100,I)=1
 1 CONTINUE
    OMEGA=1.8
    DO 2 I=1.300
    DO 3 IY=1.99
    DO 3 IX=1,99
    DU=(U(IX, IY+1)+U(IX, IY-1)+U(IX-1, IY)+U(IX+1, IY))/4-U(IX, IY)
    U(IX, IY) = U(IX, IY) + OMEGA*DU
  3 CONTINUE
    FEHLER=U(50,50)-0.5
  2 IF (MOD(I,10).EQ.0) PRINT*, I, FEHLER
    STOP
```

Bild 4: Quellprogramm zum Laplace-Benchmark

Linderung bringen könnte. Übernimmt man Programme von anderen Rechnern, so werden wohl des öfteren DATA-Anweisungen einzubauen sein.

Sprachumfang

Der Werbetext auf der Verpackung verspricht einerseits den vollen Sprachumfang von Fortran 77, andererseits aber auch vorweggenommene Erweiterungen gemäß dem kommenden und in Umrissen bereits erkennbaren neuen Standard, der zur Zeit mit 'Fortran 8X' umschrieben wird. In der zur Verfügung stehenden Zeit konnten wir natürlich nicht jede Anweisung auf Verträglichkeit mit der Norm nachprüfen, wesentliche Einschränkungen waren jedoch nicht zu erkennen.

Insbesondere sind die geforderten Datentypen INTEGER (4 Byte), REAL (4 Byte), DOUBLE PRECISION (8 Byte), COMPLEX (2*4 Byte), LOGICAL (4 Byte) und CHARACTER (1 Byte je Zeichen) implementiert. Weitere Datentypen gibt es nicht, wie zum Beispiel BYTE, INTEGER*2, LOGICAL*1, DOUBLE PRECISION COMPLEX oder ähnliches. Dies ist zwar im Fortran-Standard nicht gefor-

dert, wird aber in manchen Compilern als Zugabe eingebaut.

Durch die Compiler-Option '-W' können zwar für Variablen und Feldelemente der Typen INTEGER und LOGICAL 16 statt 32 Bit vereinbart werden, vor Verwendung dieser Option kann man jedoch gar nicht eindringlich genug warnen: Sie wirkt nämlich nicht auf Konstanten, die nach wie vor mit 32 Bit bestimmt werden, und ist damit eine schöne Fehlerquelle.

Die in Fortran übliche Einteilung der Quellzeilen in Felder ist auch bei AC/Fortran strikt einzuhalten. Zusätzlich sind jetzt aber auch Inline-Kommentare möglich, die, wie für Fortran 8X vorgesehen, durch Ausrufezeichen von der eigentlichen Fortran-Anweisung abgetrennt werden.

Weiter sind mehrere ausführbare Anweisungen in einer Zeile ab Spalte 7 erlaubt, diese sind dann jeweils durch ein Semikolon (;) voneinander zu trennen

Die Anweisung VIRTUAL erlaubt es, Felder auf dem Massenspeicher statt im Hauptspeicher anzulegen. Dieses Vorgehen scheint allerdings erst bei Verwendung einer Festplatte sinnvoll. Im Hinblick auf eine optimale Steuerung der Ein-/Ausgabe ist hier aber auf jeden Fall Vorsicht geboten: Der Zugriff auf Daten des Massenspeichers sollte soweit wie möglich sequentiell, bei Feldern heißt das etwa spaltenweise, erfolgen, dabei werden einmal geladene Daten, am besten solange es geht, im Hauptspeicher gepuffert. Sonst kann nämlich die benötigte Rechenzeit des Programms zur unliebsamen Überraschungen werden.

Ausbau vorbereitet

Das Studium der beim Compiler möglichen Optionen ergibt einige überraschende Nettigkeiten: So ist eine Option vorgesehen, mittels der Code für den Prozessor 68020 erzeugt werden kann. Für Glückliche mit einer 68020-CPU und einem Arithmetik-Koprozessor 68881 soll sogar Inline-Code erzeugt werden können. Durch diese besondere Art würde der Koprozessor überaus wirkungsvoll eingebunden. Beim Inline-Code wird beispielsweise zur Addition zweier Gleitkomma-Zahlen kein Unterprogramm mehr aufgerufen, sondern es werden gleich die nötigen Befehle zur Verwendung des Koprozessors generiert. Dieser erledigt die Aufgabe zum einen schneller als ein entsprechendes Programmstück, zum anderen entfällt auch der zum Aufruf eines Unterprogramms nötige Verwaltungsaufwand. Die Inline-Lösung ist also wesentlich intelligenter und wirkungsvoller als lediglich der Austausch der Laufzeit-Bibliothek durch neue Arithmetik-Routinen, die dann ihrerseits lediglich den Koprozessor in Gang setzen. Andererseits muß dies aber auch schon im Compiler berücksichtigt werden, was für dessen Entwickler natürlich einen zusätzlichen Aufwand bedeutet.

Der von uns getestete Compiler ignorierte jedoch jegliches an ihn gestelltes Ansinnen in dieser Richtung; der Schlingel hat offensichtlich den Schwindel mit der falschen CPU und dem nicht vorhandenen Koprozessor gemerkt und die entsprechende Option ignoriert. Was passiert, wenn man einmal seinen Amiga mit einer solchen CPU und dem Koprozessor aufrüstet, wird so natürlich nicht klar. Die Dokumentation breitet hierüber den barmherzigen Mantel des Schweigens aus.

Immerhin bleibt die Hoffnung auf das, was in einer zukünftigen Version des Compilers vielleicht einmal möglich sein wird.

Handhabung

Der Compiler laßt sich zwar ohne Ramdisk und mit nur einem Diskettenlaufwerk betreiben, besonders optimal ist das aber nicht. Für den weiter unten Rechenzentrums-Bencherläuterten mark benötigten wir so genau zwei Minuten. Eine beträchliche Beschleunigung läßt sich schon allein dadurch erreichen, daß vom Compiler erzeugte temporäre Hilfsdateien auf die Ramdisk gelegt werden, dies wird bewirkt durch die Option '-*'. Bei diesem Programm reduzierte sich die Übersetzungszeit auf weniger als ein Viertel. Spielte sich gar alles auf der Ramdisk ab, so konnte die benötigte Zeit nochmals etwa halbiert werden, insgesamt weniger als ein Achtel der ursprünglichen Wartezeit. Die genauen Ergebnisse haben wir in Bild 3 zusammengestellt. Es ist schon ein Unterschied, ob man volle zwei Minuten oder gerade 14 Sekunden auf ein Ergebnis warten muß.

Für die Namen der Fortran-Quellprogramme gibt es leider keine voreingestellte Endung, wie etwa '.FOR' oder '.F'. Verzichtet man daher konsequenterweise auf eine solche Endung, so wird man eine bitterböse Überraschung erleben: Beim Übersetzen wird der erzeugte Code in eine Datei geschrieben, deren Name aus dem der Quelldatei durch Weglassen der Endung gewonnen wird. Raten Sie mal, was dabei mit einer Quelldatei ohne Endung passiert! Selbstverständlich ganz ohne Rückfragen!

Neue Strukturen

Die Möglichkeiten der Schleifenbildung gehen erheblich über den Standard von Fortran 77 hinaus. Neben der gewöhnlichen DO-Schleife gibt es noch ein DO WHILE, sowie ein Block-DO, die beide mit END DO oder RE-PEAT abgeschlossen werden. Mittels der Anweisung EXIT kann man ein beliebiges DO-Konstrukt verlassen; die Anweisung CYCLE leitet unmittelbar den nächsten Schleifendurchlauf ein.

Laplace-Benchmark,	alle Zeiten in	Sekunden		
Maschine	Compiler	REAL*4	REAL*8	
MacIntosh	Basic	54000		
Commodore C64	Assembler	18000		
Atari ST	ProFortran 2.1	3322	5786	
Amiga	AC/Fortran	2411	3587	
IBM AT02 m. 80287	Fortran (?)	1280	?	
MicroVax II	VMS-Fortran	80	114	

Bild 5: Rechenzeiten zum Laplace-Benchmark

Fallunterscheidungen werden in Fortran meist als Block-IF-Struktur realisiert, hierzu sieht der Standard die Anweisungen IF...THEN / ELSE IF / ELSE / ENDIF vor. Die Auswahl wird hierbei an Hand logischer Ausdrücke vorgenommen. In AC/Fortran kommt hier noch eine Block-CASE-Struktur hinzu. Diese verfügt über die Anweisungen SELECT CASE / CASE / CASE DEFAULT / END SELECT und kann mit allen anderen als dem logischen Datentyp als Auswahlkriterium arbeiten. Da diese Datentypen alle enumerativ sind (CHARACTER wie gewohnt über ASCII-Code), sind in der CASE-Anweisung neben Aufzählungen auch Bereichsangaben möglich.

Der Linker

Wie bereits auf manchen Großrechnern, so gibt es auch unter AC/Fortran zwei Arten des Bindens: Die dynamische und die statische. Dynamisches Binden heißt: Es wird gar nichts gebunden; das übersetzte Programm wird gleich gestartet, und zur Laufzeit werden alle benötigten Routinen bei Bedarf in Form von Overlays nachgeladen. Dies wird dadurch ermöglicht, daß der Compiler beim Übersetzen einer Datei für alle Aufrufe, die nicht aus Unterprogrammen in dieser Datei befriedigt werden, einen entsprechenden Code generiert.

Sollen bei diesem dynamischen Vorgang auch eigene Routinen nachgeladen werden, so sind diese mit der Option 'R' zu übersetzen. Ferner darf dann jede Datei nur eine Programmeinheit enthalten, der Dateiname muß mit dem Namen des Unterprogramms übereinstimmen. Durch die Option 'R' bekommt die Datei mit der übersetzten Programmeinheit den Typ '.SUB',

nach dem beim Nachladen zur Laufzeit gesucht wird. Wird bei einem Overlay-Programm die END- oder RETURN-Anweisung erreicht, dann kann dessen Platz bei Bedarf von einem weiteren Overlay beansprucht werden.

So kann man Programme laufen lassen, die 'am Stück' nicht mehr in den Speicher passen würden.

Die herkömmliche Arbeitsweise ist das statische Binden, hier wird mit Hilfe von F77L das Hauptprogramm mit allen Unterprogrammen und Bibliotheksroutinen zu einem lauffähigen Paket verschnürt. Zur Ausführung wird dann aber immer noch das Laufzeit-System L:F77.RL benötigt. Für Stand-Alone Versionen kann dieses aber auch explizit hinzugebunden werden. Genauer gesagt: Im Handbuch wird dies behauptet.

Tatsächlich bekommt man aber bei dem Versuch, die Laufzeitbibliothek hinzuzubinden, die Fehlermeldung:

?file is not a FORTRAN library: 1:f77.rl

Übrigens: Wird der Aufruf von F77L mit einer Guru-Meditation belohnt, dann hat man vergessen, zuvor den Stack auf mindestens 50000 zu setzen! Da der Linker inkrementell arbeitet, sind auch Zwischenstufen beim Binden möglich. Man kann etwa alle eigenen Programme statisch, also mit dem Programm F77L, und die Laufzeit-Bibliothek dynamisch, also zur Laufzeit, hinzubinden. Selbstverständlich wandelt der Linker die vom Compiler erzeugten 'dynamischen' Aufrufe für hinzugebundene Unterprogramme in 'normale' um.

Das ständige Hinzuladen der Laufzeit-Bibliothek ist allerdings ziemlicher Unsinn. Die Datei F77.RL umfaßt gerade 17408 Byte und könnte, etwa beim Systemstart, einmal in den Speicher geladen werden und für den Rest der Sitzung dort verbleiben. Dies war bei den älteren Versionen von Prospero-Fortran auf dem Atari ST längst Standard.

Verbindung zum Betriebssystem

Zur Nutzung der Betriebssystem-Routinen des Amiga gibt es ein Interface-Programm namens AMIGA.SUB nebst einigen INCLUDE-Dateien zur vereinfachten Parameterdefinition. Zur Dokumentation dienen ein 55-seitiger Teil des Handbuchs, sowie der Quelltext der Routine AMIGA, der ebenfalls auf der Diskette enthalten ist.

Fehlersuche

Zur Fehlersuche, neuerdings heißt das ja 'debuggen', gibt es zwei Möglichkeiten. Zum einen kann man Fortran-Anweisungen durch einen der Buchstaben 'X' oder 'D' in der ersten Spalte als Kommentarzeilen kennzeichnen. Diese Zeilen werden dann vom Compiler ignoriert.

Verwendet man beim Übersetzen jedoch die Option '-X', dann werden diese Zeilen wie normale Programmzeilen behandelt und übersetzt. Typisch stehen dann hier irgendwelche Abfragen oder Druckanweisungen. Eine solche Möglichkeit ist bei vielen Compilern üblich.

Zusätzlich gibt es aber noch einen besonderen 'Debugger', unter dessen Kontrolle ein übersetztes Programm ausgeführt und zur Laufzeit inspiziert werden kann. Man kann das Programm in Einzelschritten ausführen, Prüfpunkte setzen, Größen anzeigen oder verändern, kurz, es steht das übliche Befehlsspektrum zur Verfügung. Einzige Voraussetzung: Das Programm muß mit der Option '-S' übersetzt werden. Dadurch wird die Datei <Programmname>.SYM erzeugt, der der Debugger alle benötigten Adressen entnehmen kann. Freundlicherweise wird der Bildschirm beim Start des Debuggers in zwei Teile aufgespalten, so daß man gleichzeitig sowohl die Debug-Befehle und deren Resultate wie auch die vom untersuchten Programm produzierte Ausgabe ansehen kann.

Daneben kann man durch eine Compiler-Option erreichen, daß beim Zugriff auf Felder, die nicht variabel dimensioniert sind, Bereichsüberschreitungen abgeprüft werden. Diese Voraussetzung dürfte in Unterprogrammen allerdings wohl selten erfüllt sein.

Mittels einer weiteren Option wird beim Übersetzen eine Liste erstellt, die lediglich die Fehlermeldungen enthält, und in einer Datei gespeichert. Treten keine Fehler auf, so wird diese Datei selbstverständlich erst gar nicht erzeugt.

Dokumentation

Die gesamte Dokumentation ist, wie oben bereits angedeutet, in Englisch. Das Fortran-Handbuch selbst erscheint recht vollständig und ist gut lesbar geschrieben. Man wird jedoch bald ein Stichwortverzeichnis (Index) vermissen; auch hätte man sich, etwa in einem Anhang, eine übersichtliche kleine Liste all der Dinge gewünscht, die in AC/Fortran über den Standard hinaus verfügbar sind.

Rechenergebnisse

Neben dem Sprachumfang eines Compilers ist natürlich auch dessen Geschwindigkeit sowie die Geschwindigkeit des erzeugten Codes von Bedeutung. Über die Geschwindigkeit des Compilers selbst haben wir oben berichtet, für die Messung der Geschwindigkeit der erzeugten Programme wurden.

den zwei Benchmark-Programme herangezogen.

Zunächst wollen wir einmal den Laplace-Benchmark aus Bild 4 betrachten. Offensichtlich handelt es sich hier um eine reine Arithmetik-Aufgabe, wenn auch die Ausgabe einiger Zwischenergebnisse mit gemessen wird.

Physikalisch löst man die Laplace-Gleichung auf einem zweidimensionalen, quadratischen Gebiet zu vorgegebenen Dirichlet-Randbedingungen.Die numerische Lösung wird auf einem in beiden Richtungen äquidistanten Gitter mit jeweils 99 inneren Gitterpunkten berechnet, die beiden Differentialquotienten zweiter Ordnung durch die zugehörigen zentralen Differenzenquotienten approximiert. Dies führt auf den typischen Laplace-5-Punkt-Stern, wie er in den beiden verschachtelten und auf die Anweisung mit der Nummer 3 endenden Schleifen erkennbar ist. Das entstehende Gleichungssystem wird mit der SOR-Methode (Sukzessive Überrelaxation) iterativ gelöst, als Relaxationsparameter wird hier der Wert 1.8 gewählt. Insgesamt werden 300 Iterationsschritte durchgeführt.

Die exakte Lösung hat im Mittelpunkt den Wert 0.5, dieser wird alle 10 Iterationsschritte mit der berechneten Näherung U(50,50) verglichen und der Fehler ausgedruckt. Das Programm wurde sowohl in einfacher wie auch in doppelter Genauigkeit gerechnet, die Ergebnisse findet man in Bild 5.

Dort werden auch einige Ergebnisse

RZ-Benchmark, alle	2 Zeiten in Sekur	iden	
Maschine	Massenspeicher	Compiler Se	ekunden
IBM AT02 m. 80287	Festplatte	Ryan McFarland Fortran V2.01	499
dito	Festplatte	MicroSoft Fortran V4.00A	1069
dito	Festplatte	1BM Professional Fortran V1.00	1553
IBM PS2/60	Festplatte	Ryan McFarland Fortran V2.01	254
dito	Festplatte	IBM Professional Fortran V1.00	680
Atari ST	Diskette	ProFortran V1.1	1896
dita	Ramdisk	ProFortran V1.1	1194
dita	Festplatte	ProFortran V2.1 (GEM)	1236
dito	Ramdisk	ProFortran V2.1 (GEM)	1188
Amiga	Diskette	AC/Fortram	1188
dita	Ramdisk	AC/Fortran	945

Bild 6: Rechenzeiten zum Rechenzentrums-Benchmark.

für andere bekannte Maschinen und Compiler angegeben. Wie man sieht, schneidet AC/Fortran auf dem Amiga gar nicht so schlecht ab.

Beim zweiten Kandidaten handelt es sich um den 'Benchmark der Arbeitsgemeinschaften der Leiter wissenschaftlicher Rechenzentren'. Dieser Benchmark versucht einen repräsentativen Querschnitt von typischen Fortran-Anweisungen zu messen. Insbesondere kommen auch umfangreiche Dateioperationen vor. Daher bekommt man hier nicht nur einen Hinweis auf die Geschwindigkeit des Programms, sondern auch auf den Einfluß des Massenspeichers. Am schnellsten dürfte wohl immer die Version mit der Ramdisk sein, daneben haben wir noch die Diskette gemessen. Eine Festplatte stand zum Test nicht zur Verfügung. Den Ergebnissen in Bild 6 sind einige Zeiten gegenübergestellt, die mit den gleichen Parametereinstellungen auf anderen Maschinen gewonnen wurden. Auch hier schlägt sich AC/Fortran auf dem Amiga recht wacker.

Bei den Ergebnissen aus den Bildern 5 und 6 wird augenfällig demonstriert, daß es nicht nur auf die jeweilige Maschine, sondern besonders auch auf den Compiler ankommt. So dürften beispielsweise die mit AC/Fortran auf dem Atari ST erzielten Zeiten unter denen des Fortran-Compilers von Prospero und etwa in der Größenordnung der beim Amiga gemessenen Zeiten liegen, was durch einen Vergleich der Taktzeiten der Prozessoren ja plausibel wird. Leider liegen diese Messungen für den Atari noch nicht vor.

Maschinenkonfiguration

Die von AC/Fortran an die Hardware gestellten Forderungen sind recht gering: Ein Amiga 500 in der Grundstufe genügt, 256 KByte Diskettenplatz reichen für das Fortran-System ebenfalls aus. Bleibt es bei dieser Grundkonfiguration ohne Speichererweiterung und mit nur einem Laufwerk, dann kann es bei umfangreicheren Prorgammentwicklungen mitunter doch recht eng werden. Als Minimum an Arbeitserleichterung wird man die vom Compiler erzeugten Hilfsdateien in die Ramdisk legen (Option '-*'), sinnvolle

Erweiterungen für die Hardware wären ein Speicherausbau und eine Festplatte, vielleicht in dieser Reihenfolge.

Zusammenarbeit mit anderen Sprachen

Das Einbinden von in C und Assembler geschriebenen Sprachen ist möglich und wird in speziellen Anhängen des Handbuchs genau beschrieben. Durch eine spezielle Compiler-Option kann man auch das Übersetzen in Maschinensprache verhindern und stattdessen Assembler-Quellcode erzeugen lassen. Wer es schon immer mal etwas genauer wissen wollte, ist damit bestens bedient.

Was uns nicht gefiel...

Daß die Laufzeit-Bibliothek nicht bei Bedarf hinzugebunden werden kann, ist schon sehr lästig; somit sind noch keine 'stand-alone'-Programme möglich. Dieser Fehler dürfte aber hoffentlich bald behoben sein. Das andauernde Hinzuladen der Laufzeit-Bibliothek beim dynamischen Binden ist ebenfalls nicht Stand der Technik, obwohl das einmalige Hinzuladen so schnell geht, daß es unsere Meßergebnisse praktisch gar nicht beeinflußte. Genausowenig ist es einzusehen, warum man immer erst eine Compiler-Option angeben muß, um die Standard-Ein- und Ausgabe auf das Terminal zu legen. Eine Kreuzverweisliste wäre manchmal doch recht nützlich, hierfür wird aber kein Programm mitgeliefert.

Im Dauerbetrieb mußte sich der Compiler noch nicht bewähren, neben den beschriebenen offensichtlichen Fehlern und Unstimmigkeiten sind daher noch keine weiteren aufgetreten. Wie die Erfahrung lehrt, ist aber auch hier, wie bei allen anderen Programmsystemen, mit dem Auftreten versteckter Fehler immer zu rechnen.

Bewertung

Mit AC/Fortran erwirbt man einen leistungsfähigen und sehr schnellen Compiler, der auch sehr schnellen Code erzeugt. Die Bedienung ist befehlsorientiert und verurteilt die Maus zur Untätigkeit, hier orientiert man sich sehr an der Arbeit auf dem Großrechner. Im Hinblick auf die hauptsächliche Anwendung von Fortran, nämlich numerische Berechnungen, ist dies aber durchaus akzeptabel. Die Fehlersuche wird sowohl durch Debug-Zeilen als auch durch einen separaten Debugger wirkungsvoll unterstützt. Umfangreiche Erweiterungen des Sprachstandards Fortran 77 erleichtern das strukturierte Programmieren.

Diese sind jedoch nutzlos und hinderlich, wenn portierbarer Quellcode angestrebt wird. Einige Erweiterungen, wie etwa die Unterscheidung von Groß- und Kleinbuchstaben in Namen, widersprechen sogar dem erwarteten Standard Fortran 8X. Zusammenfassend hat man aber einen guten und empfehlenswerten Compiler; einige unschöne Ecken und Kanten dürften sicher im Verlauf weiterer Releases abgeschliffen werden. Die getestete Version meldete sich mit

'Absoft FORTRAN 77 Compiler Ver sion 2.3'

Hersteller: Absoft Coorperation Michigan Anbieter:

GTI Fachhändler Preis: DM 545.-

ENDE

DER KICKSTARTER

AMIGA bis zu 30% schneller.

Die Taktfrequenz eines normalen Amiga liegt bei knapp über sieben Megahertz. Durch eine einfache Schaltung kann man, je nach Mut und Gerätetoleranz, die Taktrate des Rechners um zwanzig oder mehr Prozent erhöhen. Neben der deutlich erhöhten Rechenleistung, entschärft man damit auch noch das Flimmern des Interlace-Modus.

u Beginn des Artikel sollen erst einmal einige Besonderheiten dieses Umrüstsatzes besprochen werden. Die Bausteine im Amiga, besonders die hochintegrierten Customchips von Commodore (Fat Agnus, Paula und Denise) sind einer grenzenlosen Takterhöhung nicht gewachsen. Quittieren zum Beispiel überfahrene Speicherbausteine diese Überlastung einfach mit Speicherfehlern, sieht es bei komplexeren Bausteinen schon anders aus. Viele dieser ICs zeigen als erste Folge der Takterhöhung eine gesteigerte Verlustwärme, die im Laufe des Betriebs unter Umständen zur Zerstörung der Bauteile führen kann.

Nach diesen abschreckenden Worten will ich Ihnen jedoch mitteilen, daß in der Redaktion zwei Rechner, ein Amiga 1000 und ein Amiga 2000 A, problemlos mit neun Megahertz laufen. Der Amiga 2000 B des Autors schaffte mit einem schnelleren Prozessor sogar testweise 10 Megahertz. Einzige Voraussetzung für den Betrieb der Speederschaltung ist ein Multisync-Moni-

tor, da konzeptionell sich mit der Taktrate des Rechners auch die Bildwiederholfrequenz erhöht. Der normale Commodore-Monitor ist in diesem Fall total überlastet und kann sich nicht mehr auf die Signale des Rechners synchronisieren. Bei einem Multisync hingegen wird durch die erhöhte Bildwiederholfrequenz das dargestellte Computerbild deutlich ruhiger. Das Flimmern von großen einfarbigen Flächen verschwindet vollkommen. Auch im Interlace-Modus zeigt der Monitor ein fast ruhiges Bild. Ein guter Multisync kommt in der Interlace-Darstellung an das Bild eines 'Montags'-1081 in Normal-Darstellung heran.

Schrittmacher

Um das Prinzip des Speeders zu durchleuchten, schauen wir zuerst einmal die Takterzeugung in einem Standard-Amiga an. Der Grundtakt des Amiga wird von einem 28-Megahertz-Quarzoszillator geliefert. Beim Amiga 1000 wird dieser Takt diskret durch vier geteilt, um den Systemtakt von sieben Megahertz zu erhalten. Im Amiga 500 und Amiga 2000 übernimmt der Fat Agnus diese Arbeit.

Weitere Frequenzen, die aus dem Grundtakt generiert werden, sind zum Beispiel der Baudratentakt, die Floppysynchronisation und die Bildfrequenzen (vertikale und horizontale Bildwiederholfrequenz).

Hintertür

Die Entwickler des Amiga haben eine Möglichkeit vorgesehen, den Rechner extern zu takten. Es existieren hierzu zwei Leitungen des Takterzeugungsteils, die nach außen geführt sind und an der RGB-Buchse des Rechners zur Verfügung stehen. Über einen der beiden Anschlüsse kann dem Rechner der neue Takt zugeführt werden. Der zweite Pin wird über einen TTL-Pegel gesteuert und bewirkt die Freigabe des externen Taktes.

An dieser Stelle setzt die Grundidee des Speeders an. Durch einfaches Einspeisen eines höheren Grundtaktes sollte der Rechner beschleunigt werden. Die Grundlage dieser Überlegung konnte schon in ersten Probeaufbau bestätigt werden.

Der Rechner lief mit einem 36-Megahertz Quarz - dies entspricht einer effektiven Frequenz von 9 Megahertz - problemlos. Das dargestellte Bild wird durch die sich ergebende Bildwiederholrate von ca. 64 Hertz im Vergleich zu 50 Hertz im Originalzustand deutlich ruhiger. Das Flimmern in einfarbigen Flächen verschwindet vollkommen. Im Interlace-Modus ergibt sich eine Bildwiederholfrequenz von 32

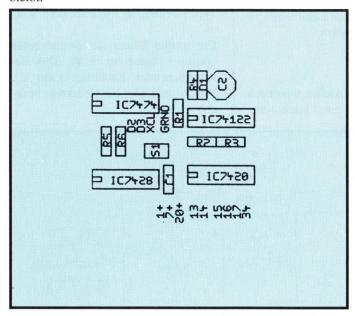
Hertz, ein Wert, der ausreicht, um diese Betriebsart auch einmal länger als zwei Sekunden zu ertragen.

Leider kann der originale Commodore-Monitor 1081 oder 1084 diese Bildfrequenzen nicht mehr synchronisieren, so daß der Genuß eines schnellen Rechners leider den Besitzern eines Multisync-Monitors vorenthalten bleibt. derartigen Takterhöhung nicht mehr. Daß auch die eingestellte Baudrate um den entsprechenden Prozentsatz abweicht, ist in diesem Fall erst einmal nebensächlich.

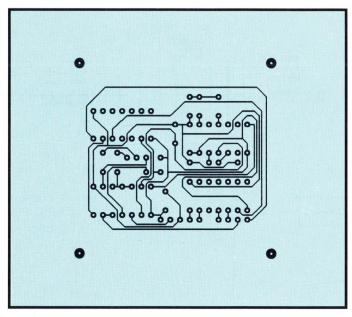
Abhilfe

Die ganze beschriebene Problematik läßt sich durch einen einfachen Schal-

Bestückungsplan



Platinenlayout (Achtung ,Schicht tauschen)



Aus dem Takt

Die bis jetzt aufgezeigte Minimallösung ist zwar einfach und billig, sie hat jedoch einen entscheidenden Nachteil. Der Rechner verweigert im 9-Megahertz-Betrieb jede Art eines Floppyzugriffs. Die Synchronisierung der eingelesen Daten mit einem intern erzeugten Takt über ein PLL arbeitet bei einer

tungstrick beheben. Man steuert die Taktfreigabeleitung des Amiga über ein TTL-Signal, das über die geschickte Auswertung der Selektsignale am Floppybus gewonnen wird. Der Rechner wird bei einem Laufwerkszugriff einfach mit der normalen Taktfrequenz betrieben und nach erfolgtem Zugriff auf neun Megahertz umgeschaltet.

Funktionsweise

Um ein sicheres Signal zu erhalten, das den Floppybetrieb anzeigt, sind einige Pegel aus dem Rechner nötig. Alle sieben benutzten Leitungen sind an den beiden Portbausteinen des Amiga erreichbar und werden einfach über einen Zwischensockel abgegriffen. Die nötige Betriebsspannung und die Masseverbindung zum Rechner werden auf die gleiche Weise entnommen, so daß keinerlei Lötverbindungen im Rechner hergestellt werden müssen.

Folgende Signale werden dem Amiga entnommen:

/Sel0,/Sel1,/Sel2,/Sel3

Diese Leitungen stellen die Selektierungsleitungen für die Laufwerke DF0: bis DF3: dar. Die Signale liegen in invertierter Form vor.

Mtr

Dieses Signal steuert die Flip-Flops der Laufwerksmotoren.

/Res

Diese Leitung dient zum Rücksetzen der Speederlogik beim Einschalten. /Rdy

Diese Leitung zeigt den Zustand des angewählten Laufwerks an.

+5V

dient zur Spannungsversorgung der Speederschaltung.

GND

ist die gemeinsame Masse mit dem Amiga.

Die Signale /Sel0 bis /Sel3 werden über ein NAND mit vier Eingängen (1/2 7420) logisch ODER verknüpft und auf den Takteingang eines Flip-Flops (1/2 7474) geschaltet. Der D-Eingang dieses ICs ist mit Mtr (negiertes /Mtr) verbunden. Diese Schaltung liefert entsprechend der Amigaspezifizierung ein 'Laufwerk in Benutzung'-Signal. Um sicher zu gehen, daß das angesprochene Laufwerk auch wirklich vorhanden und in der Lage ist beschrieben oder gelesen zu werden, wird das oben erzeugte Signal noch über ein logisches UND mit Rdy verknüpft.

Das bis jetzt gewonnene Signal ist für eine Taktumschaltung grundsätzlich schon geeignet, jedoch für den praktischen Einsatz viel zu nervös (die Software des Trackdisk-Device selektiert die Laufwerke nur für ca. 0.2 Sekunden). Der zweite Teil der Schaltung

beseitigt jetzt noch diesen Schönheitsfehler.

Das wie oben beschrieben erzeugte Signal wird jetzt auf zwei Wege aufgeteilt. Der erste Weg führt über ein retriggerbares Mono-Flop, dessen Zeitkonstante über R3 und C2 auf ca. eine Sekunde eingestellt ist. Über dieses Mono-Flop wird jeder ankommende Impuls auf mindestens diese Zeit verlängert, weitere Impulse starten die Zeit neu. Dieses Signal wird mit dem Ursprünglichen logisch ODER verknüpft, um sicherzustellen, daß das Selektsignal auch nach Abfallen des Mono-Flops den Rechner noch auf langsamer Taktrate laufen läßt. Dieses

so erzeugte Signal steuert über zwei Treiber die Kontroll-Leuchtdioden des Speeders und die Freigabeleitung für den externen Takt.

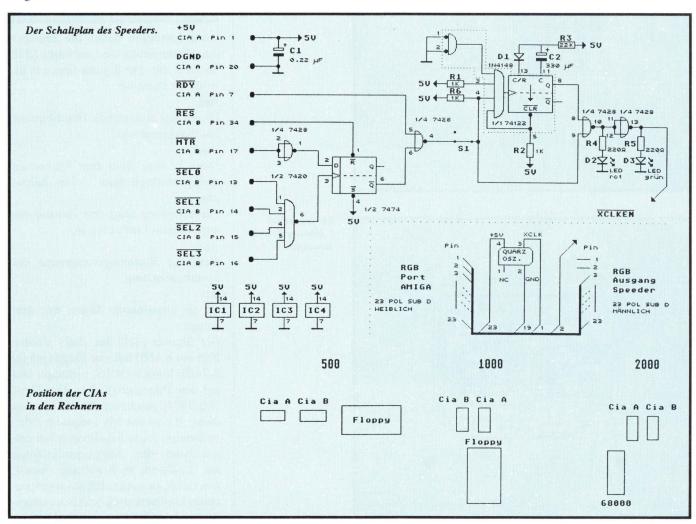
Mit dem Schalter S1 kann der Speeder wahlweise, auch während des Betriebs, ein- oder ausgeschaltet werden.

Der externe Takt wird über einen Zwischenstecker am RGB-Port des Amiga eingespeist. Die komplette Speederschaltung kann so ohne Lötarbeiten im Amiga eingebaut werden.

Aufbau

Die Schaltung ist im Aufbau unkritisch und kann leicht auf einer Lochrasterplatine untergebracht werden. Das hier abgedruckte Platinenlayout läßt die Schaltung jedoch professioneller aussehen. Überprüfen Sie bitte besonders die Anschlüße zu den beiden 8520, da diese Bausteine mit ca. 90,- DM nicht gerade billig sind. Das Foto der fertigen Platine und die Einbautabelle mit den Bauteilenummern der Port-Bausteine in den verschiedenen Amiga Typen helfen hier mehr als viele Worte.

Für Bastler liegen die Bauteilkosten inklusive Platine bei ca. 40,- DM. Bei entsprechender Nachfrage kann die Platine auch über die Redaktion bezogen werden.



Speeder Stückliste	C1 0,22 µF	36 MHz TTL	16 pol 70 cm.
	C2 330 µF		
		5x14 pol. Fassung gedreht	1 Platine ca. 7x8 cm.
IC1 7474	R1 1 k	2x40 pol. Fassung gedreht	
IC2 7420	R2 1 k		Kosten der Bauteile ca.
IC3 7428	R3 22 k	1x23 pol. SubD weiblich	40 DM
IC4 74122	R4 220	1x23 pol. SubD männlich	
	R5 220		
D1 1N4148	R6 1 k	1 Miniaturschalter 1xEin	
D2 LED rot			
D3 LED grün	OSC1 Quarz-Oszillate	or Flachbandkabel farbig ca.	

MIDI GOLD

Die Brücke zum computergesteuerten Studio

Da der Amiga, wie viele andere Computer auch, von Haus aus über keine sogenannte MIDI-Schnittstelle verfügt, bieten in letzter Zeit mehrere Hersteller Hardwarelösungen an, um diesen Mangel an Kommunikationsfreudigkeit zu beheben, und so den Amiga auch in der Welt des Homerecordings und der digitalen Studios seine Fähigkeiten beweisen zu lassen.

as ist MIDI? MIDI ist eine Abkürzung aus dem Englischen und bedeutet "Musical Instrument Digital Interface", also eine Schnittstelle für elektronische Musikinstrumente, wie Synthesizer oder Drumcomputer. MIDI hat sich seit seiner Einführung vor ein paar Jahren als die Standardschnittstelle etabliert, vergleichbar mit der Centronicsschnittstelle bei den Mikrocomputern. Mit MIDI können Daten von MIDI-fähigen Geräten mit einer hohen Übertragungsgeschwindigkeit (31,25 Kbaud) ausgetauscht werden. Dabei ist es möglich, bis zu 64 MIDI-Geräte miteinander zu vernetzen und in Echtzeit Daten auszutauschen. Um hier die Koordination nicht zu verlieren ging man mehr und mehr dazu über, die Steuerung der einzelnen Geräte und der einzelen Musiksequenzen zentral, also mit dem Computer, zu übernehmen. Viele Musikprogramme für den Amiga wie z.B "Music Studio" oder "De Luxe Music" beinhalten solche Fähigkeiten schon.

MIDI-Gold ist ein Produkt der amerikanischen Firma Golden Hawks Technology. Im Lieferumfang befindet sich das eigentliche Interface, ein 10seitiges, in Englisch gefaßtes Heftchen mit Schaltplan und wenig Information sowie ein Anschlußkabel für die RS232-Schnittstelle des Amiga.

Lieferbar ist MIDI-Gold in zwei Versionen: eine für den Amiga 1000, die andere, wegen des geänderten RS232-Steckers, für Amiga 2000 und 500. Software liegt MIDI-Gold nicht bei, dafür sind im "Handbuch" zwei C-Routinen abgedruckt, die sich der glückliche Besitzer eines C-Compilers gerne abtippen darf. Hier handelt es sich einmal um ein Programm, das dazu dient, die RS232-Schnittstelle für MIDI zu konfigurieren, und um eine Routine, die das "SYNC-OUT"-Signal von MIDI-GOLD (z.B für die Ansteuerung eines Drum-Computers) bereitstellt. Wer zu faul zum Tippen ist, kann sich diese Programme aber auch vom Hersteller gegen Aufpreis nachbestellen.

Das Interface selbst ist ein ca. 15 cm langes und ca. 3 cm hohes Metallkästchen. Auf der Rückseite befindet sich ein 25poliger Stecker, mit dem das Interface mit dem mitgelieferten Kabel an die RS232-Schnittstelle des Amiga angeschlossen wird. Dies hat den Vorteil, daß man das Interface beliebig (natürlich im Rahmen der Kabellänge) positionieren kann. Dies hat sich bei unserem Test einige Male sehr positiv ausgewirkt. Bei einigen Konkurrenzprodukten ist dies nicht möglich, weil diese meistens ohne Kabel direkt an den RS232-Port angebracht werden.

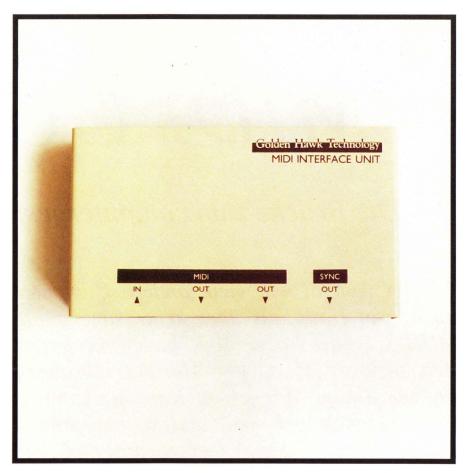
An der Vorderseite befinden sich die eigentlichen MIDI-Anschlüsse. Es sind vier, wie bei MIDI üblich, 5polige DIN-Buchsen. Zur Verfügung stehen: MIDI-In (Empfang von Daten), zweimal MIDI-Out (Senden von Daten) und ein SYNC-Out (Senden eines externen Clocksignals für Drumcomputer, Sequencer etc.). Auch das Innenleben von MIDI-Gold macht einen ordentlichen und gut durchdachten Eindruck.

Zum Testen verwendete ich ein Yamaha DX7 (Keyboard) und eine Linn-Drummachine. Beide Geräte ließen sich problemlos mit Programmen wie Dynamic Studio oder Musicstudio in Kombination mit MIDI-Gold betreiben. Besonders die Fähigkeit von MIDI-GOLD, mittels der SYNC-Out Buchse z.B. die Drummachine exakt ins Timing zu setzen, machte das Arbeiten mit komplizierten Rhythmen zu einem wahren Genuß. Auch waren keine Guru-Meditations oder andere Abstürze, selbst unter härtesten Testbedingungen (gleichzeitiger Betrieb von 6 mühsam zusammengeliehenen Synthesizern + Drummachines), festzustellen.

Somit kann ich MIDI-Gold jedem MIDI-begeisterten Anwender empfehlen, die Mängel beschränken sich auf das etwas zu dünn geratene Handbuch und die fehlenden Software. Wer einen Amiga besitzt, in die MIDI-Welt einsteigen möchte - ob als Hobby oder professionell - und jetzt vor der Kaufentscheidung steht, der sollte sich auf jeden Fall einmal die Fähigkeiten von MIDI-Gold zeigen lassen. In diesem Sinne: Let's Rock!

MIDI GOLD

- + solide Verarbeitung
- + 2 x MIDI-Out
- + externes sendbares Clocksignal (SYNC-Out)
- + frei positionierbar dank RS232-Kabel
- spärliches englisches Handbuch
- fehlende Beispielssoftware (nur gegen Aufpreis zu haben)



"MIDI-Gold von GOLDEN-HAWKS-TECHNOLOGY"

ENDE

Die Retter kommen – Eine totsichere Sache... Ab heute gibt es keine VIREN mehr!



1. Virus Finder

nur 49,90 DM

Mit dem Virus Finder können Sie schnell und problemlos alle Ihre Disketten nach einem Virus durchsuchen und "heilen". Z.B. SCA und Byte Bandit Virus gibt es danach **nicht** mehr! Er erkennt und löscht auch Viren, im Speicher! Bedienungsanleitung in deutsch auf Diskette (Virus Finder Doc). Virus Finder läßt sich auch auf der Workbench installieren!

2. Second Chance

nur 49,90 DM

Speichert, auch von kopiergeschützten Programmen, den Boot-Block auf eine seperate Diskette ab (auf einer Diskette können über 800 Boot-Blöcke gespeichert werden). Wenn der Boot-Block auf dem Original durch Viren zerstört wurde, kann der zuvor gesicherte Boot-Block auf das Original zurückgeschrieben werden. Die Software ist wieder einsatzfähig.

3. Int Switch nur 29,90 DM

Dieses Programm löst alle Ihre Probleme mit Software, die mit Speichererweiterungen nicht laufen. Int Switch schaltet beim Amiga 2000 XT-Kaarte und Speichererweiterungen an. Int Switch schaltet beim Amiga 1000 Sidecar und Speichererweiterungen ab. Int Switch schaltet beim Amiga 500 Expansionskarten und Speichererweiterungen ab.

Nachnahme 6,- DM

Vorkasse 4,- DM

Ausland: nur gegen Vorkasse 10,- DM

PDC GmbH, Louisenstraße 115, 6380 Bad Homburg Telefon (0 61 72) 2 47 48 oder 2 07 99 VON THOMAS SCHLERETH

ELECTRONIC-DESIGN

Genlock-Faszination Video und Amiga

Das Mischen von zwei Videosignalen ist mittlerweile in aller Munde. Das Computersignal sowie das einer fremden Quelle (Videorecorder, Kamera) wird zusammengemischt und somit überlagert. Die erstaunlichsten Effekte lassen sich dadurch erzielen. Ein preisgünstiges Gerät stand uns diesmal zum Test zur Verfügung.

unktionsweise eines Fernsehsystems. Ein herkömmliches Fernsehbild ist in 625 Bildschirmzeilen aufgeteilt, die sich gleichmäßig, von oben nach unten, über den Bildschirm verteilen. Dieses Gesamtbild teilt sich noch einmal in zwei gleichmäßige Teile, Halbbilder genannt, auf. Das erste Halbbild besteht aus den Zeilen 1 bis 313, das zweite reicht analog dazu von der 313. bis zur 625. Zeile.

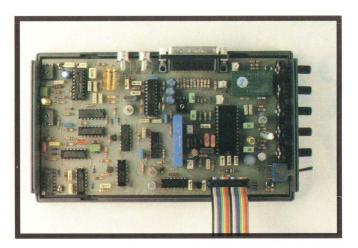
Diese beiden Bilder müssen ständig neu produziert bzw. aufgebaut werden, da sich nur so eine Bewegung auf dem Bildschirm realisieren läßt. Jedes Halbbild wird alle 25 ms (Millisekunden) erneut aufgebaut. Würde das gesamte Bild von oben nach unten in einer Etappe aufgebaut werden, statt in zwei Einzelschritten, würde sich auch die Zeit von einer Bildsequenz zur anderen verdoppeln. Hierbei wäre der Bildschirm bei einer Aufbauzeit von 50 ms statt den normalen 25 ms genau doppelt so lange dunkel. Da aber bei einem Zeitabstand von 50 ms zwi-

schen den einzelnen Bildern das menschliche Auge diesen Bildaufbau wahrnehmen könnte, entsteht der Eindruck, daß das Bild flimmert. Das Sichtbarmachen auf dem Glasschirm erfolgt mit Hilfe von Elektronen-Strahlkanonen. Mit Hilfe dieser Einrichtung werden die Punkte, aus denen das Bild schließlich besteht, auf ihre Zielpositionen auf dem Schirm projeziert. Diese Zielpositionen werden durch ein Synchronisationssignal bestimmt. Das Signal weist die Elektronenstrahlkanonen an, die richtige Positionen auf dem Bildschirm anzusteuern.

Bei einem Farbfernseher müssen zusätzlich noch Farbinformationen übertragen werden. Ein solches Farbbild setzt sich aus den Grundfarben Rot, Grün und Blau zusammen. Um diese Farben auf dem Schirm sichtbar zu machen, sind demnach auch drei Elektronenstrahlkanonen nötig.



Das Genlock ist kompakt und handlich.



Der Aufbau des Genlocks ist recht komplex.

Technische Informationen zum Computer:RGB-Bild

Bei einem Computerbild auf dem Commodore Amiga beträgt die Auflösung, im non-interlace Betrieb, 313 Zeilen. Bei diesem Bild wird der Bildschirmbereich nicht in zwei Teile aufgeteilt, sondern das Bild wird in einem Zug aufgebaut. Wie oben beschrieben, würde durch die langen Zeitdifferenzen zwischen den einzelnen Bildaufbauphasen ein Flimmern des Bildschirms entstehen. Um dies zu vermeiden, ging man hier einen anderen Weg. Das Bild wird einfach in 50 ms zweimal geschrieben, es flimmert demnach nicht. Der Unterschied besteht also in der Art, wie das Bild neu produziert wird. Bei einem Fernsehsignal wird das Bild in zwei Bilder zu je 25 ms Aufbauzeit geteilt, ein Monitorsignal hingegen teilt das Bild nicht auf, dafür wird es in 50 ms zweimal aufgebaut.

Da das Bild, wie bereits erwähnt, nur aus 313 Zeilen besteht, wäre es auch nur halb so groß. Dieser Umstand wird einfach umgangen, indem jede zweite Zeile auf dem Monitor ignoriert wird und schwarz bleibt. Wenn Sie sehr nahe an den Monitor herangehen, werden Sie leicht feststellen, daß nach jeder Zeile mit Bild- bzw. Farbinformationen eine schwarze Zeile folgt. Diese Leerzeilen werden auch nicht neu aufgebaut. Rechnerisch ergibt sich so auch eine andere Anzahl von Zeilen. Es sind, anders als bei dem Fernsehsignal, 626 Zeilen. Die drei Grundfarben werden, mit Unterstützung von entsprechenden Verstärkerstufen, direkt von den drei RGB Ausgängen an der Rückseite des Amiga abgestrahlt. Das Synchronisationssignal wird ebenfalls von diesen Ausgängen bereitgestellt.

Ein Videorekorder am Amigabildschirm

Alle Videorekorder bieten die Möglichkeit, einen RGB-Monitor anzuschließen. Dieses Signal, das der Videorekorder abgibt, wird als FBAS-(Farbe, Bild, Austast-Synchronisation) Signal bezeichnet. Bei einem FBAS-Signal werden die beiden Synchronisationssignale, die ja unterschiedlich sind, und die Bildinformationen zusammengefaßt.

Der Aufbau eines Bildes erfolgt nach einem immer wiederkehrenden Vorgang.

Das Mischen bzw. Überlagern von zwei Bildern in den beiden verschiedenen Formaten

Die Art und Weise, wie die beiden unterschiedlichen Systeme arbeiten, macht es sehr schwer, die Signale so zu mischen, daß ein sinnvolles neues Bild erzeugt werden kann.

Es kann eine Vielzahl von Schwierigkeiten auftreten, die ein solches Vorhaben vereiteln. Eine dieser Schwierigkeiten besteht in erster Linie darin, eine gewisse Gleichheit in bezug auf den Start eines Aufbauvorganges zu gewährleisten. Die beiden Bilder müssen also gleichzeitig mit der ersten Zeile bei ihrem Aufbauvorgang beginnen und auch weiterhin völlig synchron laufen. Bildüberlagerungen in ihrer feinsten Art sind immer wieder im Fernsehen zu bewundern. Mit einer sehr aufwendigen Technik werden bei einer Liveaufnahme alle Kameras mit einem gemeinsamen Mastertakt versorgt, denn nur so läßt sich gewährleisten, daß die Bilder scheinbar ineinander überlaufen.

Neben dem Prinzip eines Mastertaktgebers gibt es noch die Möglichkeit, eine der beiden Videoquellen auf die andere zu synchronisieren. Bei einem Genlock-Interface (Interface = Schnittstelle, hier zwischen Bildquelle und Bildausgabeeinheiten) wird die Computerquelle auf die Videoquelle synchronisiert.

Hier tauchen wieder einige Schwierigkeiten auf. Ein Problem besteht darin, daß die Anzahl der Zeilen unterschiedlich ist (625 bei Video-, 626 bei Computerbildern). Natürlich würde sich diese eine überflüssige Zeile vom Computer durch entsprechende Software zum Verschwinden bringen lassen. Diese Lösung hat aber zur Folge, daß das Genlock-Interface nur mit Unterstützung durch ein Programm funktionieren würde. Mit anderen Programmen könnte es dann, auch durch die Multitaskingfunktionen des Amiga, zu Konflikten kommen.

Es liegt also nahe, das Problem von seiten der Hardware, sprich dem Genlock-Interface, zu lösen. Es wird einfach dafür gesorgt, daß das Computerbild wieder an die richtige Stelle gerückt wird. So wird ein synchroner Ablauf der beiden Bilder gewährleistet. Die wichtigste Aufgabe des Genlock-Interfaces besteht in dem Wandeln der einzelnen Bildformate und dem zeitrichtigen Überlagern der Bilder. Der Amiga bietet ein RGB-Signal, die Videoquelle (Rekorder oder Kamera) meist ein FBAS-Signal. Um die Bilder nun zu verbinden, wird ein Signal umgewandelt. Wird nun das RGB-Signal des Amiga in ein FBAS-Signal umgewandelt, ist ein Qualitätsverlust die Folge. Allerdings kann dieses Signal vom Videorekorder verstanden werden, es kann also auf eine Videokassette aufgenommen, mitgeschnitten werden. Das FBAS-Videosignal liegt ja bereits in der gewünschten Form vor und braucht durch das Genlock-Interface nicht mehr bearbeitet zu werden. Ein Problem ist das

Umwandeln des RGB-Signals des Amiga. Besonders hinsichtlich der Farbinformationen muß eine absolut präzise Umwandlung erfolgen.

Ist das nicht der Fall, und es gibt solche Geräte, wird das RGB-Computerbild nur mäßig bis schlecht wiedergegeben. Ein Musterbeispiel für hervorragende Verarbeitung ist durch das PAL-Genlock Interface der Münchner Firma Electronic-Design gegeben. Steffen Kramer ging bei seinem Genlock-Interface einen Weg, der eine optimale Bildqualität gewährleistet.

In jedem Monitor muß ein RGB-Signal erzeugt werden, um die Bildröhre anzusteuern. Bei diesem Genlock-Interface wird das aus der Videoquelle kommende Signal in ein RGB-Signal gewandelt. In dieser Form kann das so entstandene RGB-Bild mit dem computereigenen RGB-Signal leicht in Verbindung gebracht werden. Das entstehende Bild wird an den Monitor des Computers weitergeleitet. Der Vorgang des Wandelns im Monitor entfällt, und das Bild wird in einer ausgezeichneten Brillianz wiedergegeben. Das heißt also, daß Sie nach Anschluß eines Genlock-Interface das in dieser Form arbeitet, wie es bei dem Electronic-Design Genlock-Interface der Fall ist, ein Computerbild ohne Qualitätsverlust erhalten, das mit einem Videobild gemischt worden ist, das ebenfalls in einer ausgezeichneten Schärfe und Klarheit vorliegt.

Nun hat das Genlock zwar ein RGB-Bild erzeugt und auf dem Computermonitor gezeigt, aber dieses Bild läßt sich nicht speichern. Auf einem Computerspeichermedium (Diskette, Platte oder andere Speichermedien) läßt sich nur die Grafik, die über dem Videobild liegt, speichern, denn das Videosignal wird nur in RGB gewandelt, aber nicht digitalisiert. Ebenfalls zeigt eine Hardcopy (Bildschirmcopy) auf einen Drucker nichts von dem Videobild. Der Grund ist Ihnen sicher schon klar. Dem Amiga liegen überhaupt keine Informationen vor, daß eine Manipulation des Bildes erfolgt ist. Sobald die RGB Signale den Amiga durch den Bildschirmausgang verlassen haben, ist die Sache für den Computer praktisch abgeschlossen. Was mit dem Signal dann passiert, hat keinerlei Auswirkung auf den Computer. Das bedeutet daß eine Störung des Prozessors

seitens des Genlock Interface nicht möglich ist. Der Amiga arbeitet wie gewohnt weiter. Auch andere Pheripheriegeräte werden nicht gestört.

Um nun die erzeugten Effekte zu speichern, muß man sich einem anderen Speichermedium zuwenden. Hier wird einfach ein zweiter Videorekorder benutzt. Sie merken vielleicht jetzt schon, daß ein Genlock-Interface alleine nicht genügt.

Nur mit mindestens zwei Videorekordern ist ein sinnvolles Arbeiten möglich. Da aber das gemischte Signal RGB ist und der Videorekorder FBAS verlangt, muß das Signal erneut gewandelt werden. Auch diese Aufgabe wird vom Electronic-Design Genlock-Interface übernommen. Die Leistungen des Wandlers im Inneren des Interface entscheiden über die Qualität der Aufnahme. Das Electronic-Design Genlock-Interface verfügt über einen sehr guten Wandler, der die Qualitätsverluste gering hält.

Nähere Beschreibung des Electronic-Design Genlock-Interface

Ich möchte nun etwas genauer auf das Genlock-Interface eingehen, denn die technischen Grundlagen, wie sie oben beschrieben wurden, sind keine Voraussetzungen, um dieses Genlock Interface einzusetzten.

Das Genlock-Interface wird zwischen den Amiga-RGB-Ausgang und den Monitor geschaltet. Hinzu kommt nun das Videosignal, wobei das Genlock einen Cinchstecker benötigt.

Falls Ihr Videorekorder andere Ausgänge hat (SCART o.ä.), ist ein ent-

Eine von vielen Möglichkeiten das Genlock einzusetzen. sprechendes Kabel nötig. Ein solches Kabel ist im Lieferumfang nicht enthalten, da es eine Vielzahl von Anschlüssen gibt und Sie praktisch alle bis auf ein Kabel nicht brauchen.

Solche Kabel von einem beliebigen Ausgang nach Cinch bekommen Sie in jedem Elektronikladen. Da sie aber nicht billig sind (zwischen 25,- und 80,- DM), können Sie sich ein entsprechendes Kabel auch einfach selber zusammenlöten.

Für unseren Test lag ein Genlock-Interface vor, bei dem die Verbindung zwischen Amiga und Genlock sehr kurz (nur ca. 10 cm) war. Dies machte die Bedienung der Regler am Genlock Interface sehr beschwerlich. Als wir Steffen Kramer darauf hinwiesen, begründete er die fehlende Länge des Kabels mit einem möglichen Qualitätsverlust. Da aber das Betreiben des Genlock-Interface an einem Amiga 1000, bei dem der Monitor obenauf steht, unmöglich macht, wird bei Electronic-Design an einer Lösung gearbeitet, die diesen Mißstand behebt. Uns wurde versichert, daß das Genlock-Interface mit einem längeren Kabel ausgestattet wird, der Qualitätsverlust wird mit einem zusätzlichen Verstärker im Innern des Genlocks ausgeglichen.

Das Electronic-Design Genlock-Interface verfügt, neben einigen Reglern, die noch im einzelnen vorgestellt werden, über drei Anschlüsse und einen Stecker. Der Stecker ist für den Amiga-Monitor, die Anschlüsse sind für die Verbindung zum Monitor (auf dem Gehäuse mit AMIGA betitelt), für die Videoquelle (Video-In) und für einen weiteren Videorekorder (Video-Out) bestimmt, der zur Aufnahme dienen soll.





... noch eine weitere Genlock-Anwendung.

Das Genlock-Interface der Firma Electronic-Design besitzt, im Gegensatz zu anderen, meist teureren Systemen, eine Vielzahl von Reglern (Poties).

Insgesammt sind es fünf Stück, die an der rechten Außenseite angebracht sind. Auf der Oberseite wurden sie mit folgenden Namen versehen: MODE, FADER, ILLUMINANCE, CONTRAST und COLOUR.

MODE und FADER ermöglichen ein Ein- bzw. Ausblenden des Videobilds und eine genaue Einstellung der Verhältnisse der beiden Bilder. Die drei anderen Regler dienen zum Verbessern des Bildes. Hier können Sie, ähnlich wie bei einem Farbfernseher, das Bild so einstellen, bis es Ihnen zusagt. Es ist darauf zu achten, daß bei der Einstellung des Monitors eine entsprechende Helligkeit bzw. Dunkelheit vorhanden ist, da sonst Differenzen zwischen dem sichtbaren Bild und der Aufnahme entstehen könnten.

Am Genlock-Interface ist weiterhin ein Kippschalter angebracht, mit dem Sie das Gerät ein- und ausschalten können. Ist der Schalter auf "aus", läuft die Bildschirmausgabe des Amiga wie gewohnt. Das heißt, daß kein Unterschied zwischen einer Anlage mit Genlock-Interface und einer Anlage ohne vorhanden ist. Ist das Interface eingeschaltet, wird statt der Hintergrundfarbe das Videobild dargestellt.

An einem Beispiel möchte ich das etwas beschreiben. Wenn Sie die Workbench booten (laden), ist in der normalen System-Configuration die Hintergrundfarbe blau, und die Fenster sind weiß. Bei einem eingeschalteten Genlock Interface sind die Fenster auch weiterhin weiß, das Blau wurde nun allerdings mit dem Videobild getauscht.

Die Fenster, in ihrer weißen Farbe, liegen nun über dem Videobild. Ein wichtiger Hinweis, der auch in der Anleitung vermerkt wurde, besteht darin, daß das Genlock bei einem eingeschalteten Amiga nicht ein- oder ausgeschaltet werden sollte. Es zerstört zwar weder Amiga noch Genlock, aber der Amiga stürzt in den allermeisten Fällen mit einer Guru-Meldung ab.

Electronic-Design bietet mit diesem Genlock-Interface eine äußerst leistungsstarke Version eines Interface. Erfreulich ist auch die Tatsache, daß einige Regler ein Steuern des GenlockInterface ermöglichen. Wir hatten die Möglichkeit, das Interface an den verschiedensten Videoanlagen zu testen und dadurch auch Feinheiten des Gerätes genau zu erkennen. Diese Eigenarten sind jedoch für einen Videofreak, der seine eigenen Videofilme mit einigen Effekten untermalen will, nicht relevant, da die Anlagen bereits in einer höheren Preis- und Leistungsklasse lagen.

In dem Beiheft, das im Lieferumfang enthalten ist, wird zwar kurz, aber klar und deutlich die Funktion des Genlocks erklärt. Sie enthält weiterhin einige Tips und Tricks für Aufnahmen auf ein Videoband. Auch einige Anwendungsbereiche werden kurz angesprochen, aber die eigentliche Anwendung ist nur von Ihrer Phantasie begrenzt. Erfreulich ist auch der Preis, der bei

Erfreulich ist auch der Preis, der bei nur ca. 550,- DM liegt.

Falls Sie noch Fragen zu diesem außergewöhnlichen Gerät haben, oder einfach etwas mehr über Genlocks wissen möchten, wenden Sie sich am besten an Steffen Kramer.

Bestellen können Sie das PAL-Genlock-Interface der Münchner Firma Electronic-Design bei der folgenden Adresse:

electronic-design Steffen Kramer Biedersteiner Straße 59 Telefon: 089-3618011 8000 München 40

Future Vision M. Stutz Fr.- Veith-Str. 21 6128 Höchst Tel. 06163-1278

Preis: ca. DM 550,-

ENDE

DER AMIGA MIT 68030

Ein Rechner der vierten Generation?

War noch vor wenigen Monaten der 68020 das Höchste der Gefühle, so ist der Status Quo der Technik heute der 68030. Dieser Prozessor ist durch eine neue Achitektur noch einmal doppelt so schnell wie der 68020, ohne auf die Kompatibilität zu diesem zu verzichten. Was bringt solch ein Wunder der Technik im Amiga?

ie zum Test vorliegende Karte wird von der Firma Ronin Research and Development Inc. gebaut. Gleich der erste Blick auf die Karte zeigt die Ähnlichkeit zu dem im Februar getesteten Modell. So ist in der Tat die 68030-Karte eine aufgebohrte 68020iger. Dies wird durch einen einfachen Adaptersockel erreicht, der das unterschiedliche Pinout der beiden Prozessoren aus dem Hause Motorola aufeinander anpaßt. Da sich beide Prozessoren in der Art der verlangten Signale sehr ähnlich sind, kann die Adapterplatine passiv

h sind, kann die Adapterplatine passiv aufgebaut werden. Das von Ronin verwendete Platinchen zeigt dies deutlich, außer einigen Pullup-Widerständen enthält die Platine keine weiteren Bauteile.

Die restlichen Eigenschaften der 68030-Karte entsprechen somit den Daten des 68020-Boards. Der auf der Karte zur Verfügung gestellte 32-Bit-Bus kann mit den Erweiterungen der 68020-Karte benutzt werden. Eine der sinnvollsten Anwendungen ist hier ein 32 Bit breites Ram, auf das der 68030

mit seiner vollen Geschwindigkeit zugreifen kann. Ein weiterer Geschwindigkeitsvorteil ergibt sich durch den mit 14.32 MHz im Vergleich mit dem Amiga (7.14 MHz) doppelt so hohen Takt der 68030-Karte. Allerdings bietet auch die 68020-Karte schon dieses Feature. Ein viel wichtigerer Aspekt ist die Tatsache, daß Erweiterungen am 32-Bit-Bus mit der höheren Taktfrequenz arbeiten.

Als zweiter Prozessor ist der 68881 auf der Platine enthalten. Obwohl in Sachen mathematischer Koprozessor der 68882 den neusten Stand der Technik repräsentiert, ist man, wie die Benchmarkzeiten zeigen, auch mit dem 68881 bestens ausgerüstet. Der Koprozessor wird mit einen unabhängigen Takt versorgt, so daß der verwendete 16 MHz-Typ gegen eine 20 MHz-FPU getauscht werden kann.

Lieferumfang

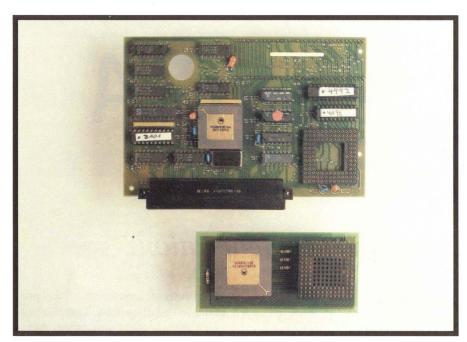
Wir erhielten die Karte direkt auf der Messe. Zu der Karte gehört eine Diskette, auf der neben einigen Utilities auch diverse Demos und Patchprogramme enthalten sind. Eine Einbauanleitung war nicht vorhanden. Allerdings ist das kein Hindernis für den Anwender. Die Karte ist in der vorliegenden Form für den Amiga 1000 ausgelegt und paßt durch ihr besonderes Aussehen nur in einer Richtung auf den Prozessorsockel des Amiga. Spezielle Versionen sind auch für Amiga 500 und 2000 lieferbar, standen jedoch für den Test nicht zur Verfügung.

Einbau

Die 68030-Karte wird in den Sockel des 68000 gesteckt. Hierzu muß das Gerät durch Entfernen der Deckelschrauben geöffnet werden. Der 68000 kann dann durch vorsichtiges Hebeln mit einem Schraubendreher aus seiner Fassung entfernt werden. Die 68030-Karte wird nun in den freien Sockel eingesetzt, und der Rechner ist wieder einsatzfähig.

Testlauf

Nach durchlaufenem Selbsttest verlangt der 68030-Amiga wie gewohnt nach der Kickstart-Diskette, die auch ohne Probleme gebootet wird. Nach Einlegen der Workbench meldet sich der Rechner ganz normal. Auch die ersten getesteten Programme versahen ihren Dienst mit dem 68030, als hätten sie nie etwas anderes getan. Die Probleme, die sich mit einem 68030 ergeben, sind grundsätzlich dieselben, die ein Anwender auch mit dem 68020 hat. So verwenden immer noch viele Programmierer, entgegen den ausdrücklichen Warnungen in der originalen



Der 68030er in voller Pracht

Amiga-Dokumentation, den MOVE SR,<ea> Befehl, der vom 68020/68030 im Gegensatz zum 68000 nicht im User-Modus ausgeführt werden kann. Auch die zweite Unverträglichkeit von vorhandener Software mit dem 68030 liegt im Programmierstil mancher Softwareentwickler begründet. Der 68030 verwendet, wie auch der 68020, einen Cache, in den er schon während der Abarbeitung des momentanen Befehls die nächsten Anweisungen nachlädt. Verwendet man nun selbstmodifi zierenden Programmcode, wird zwar das Codesegment im Speicher verändert, zur Ausführung kommt jedoch das sich schon im Cache befindliche unveränderte Programmstück.

Allerdings ist zur Ehrenrettung der Programmierer zu sagen, daß solche Probleme bei neuerer Software fast nicht mehr auftreten, zumal ein Commodore-eigener 32 Bit-Amiga (68020) wohl nicht mehr lange auf sich warten läßt.

Geschwindigkeit

Der 68030 bringt im Vergleich zum 68020 zirka die doppelte Rechenleistung. Die Entwickler des 68030 trennten hierzu ersteinmal das interne Bussystem des Prozessors nach Daten und Adressen. Zusätzlich wurde ein zweiter Cache eingerichtet, so daß jetzt sowohl Daten als auch Befehle über die Prefetch-Technik schneller dem

Steuer- und Rechenwerk zugeführt werden können. Auch verfügt der 68030 über einige interessante Möglichkeiten, mit der Außenwelt in Verbindung zu treten, die jedoch im Amiga nicht genutzt werden können. Insbesondere sind hier der Burstmode und ein neuer 32 Bit-Zyklus zu nennen. Der Burstmode erlaubt die Datenübertragung von vier Langworten in nur fünf Taktzyklen. Der zweite angesprochene Buszyklus arbeitet mit dem neu eingführten STERM-Signal und ist gegenüber dem 68020 um zirka 30% schneller.

Aufgrund der vom Amiga vorgegebenen Hardware können diese Features des 68030 jedoch nicht ausgenutzt werden. Die gemessenen Zeiten der schon in Heft 2/88 durchgeführten Benchmarks zeigen dies auch nur allzu deutlich. Ein 68030 ist mit einer ihm nicht zugedachten Drumherum-Hardware nicht schneller als ein 68020. Sezielle Besonderheiten wie zum Beispiel die eingebaute MMU und eine Speicherverwaltungslogik, deren Einsatz gerade für den Multitasking-Multiuser-Betrieb eine feine Sache wären, können hier nicht genutzt werden, zumal das Betriebssystem des Amiga von solchen Dingen keinerlei Ahnung

32 Bit-Zauber

Als ein viel leistungsfähigeres Mittel, die Rechenleistung des Amiga zu erhöhen, ist eine Speichererweiterung zu sehen, die vom 68020/68030 mit der vollen Breite des Datenbusses von 32 Bit bedient werden kann. Zu diesem Test stand eine solche Speichererweiterung in der Größe von einem Megabyte zur Verfügung. Die Speicherkarte muß beim Systemstart über den Addmem-Befehl eingebunden werden und steht danach dem System zur Verfügung. Wird eine Applikation in diesem Speicher gestartet, so zeigt die 68030-Karte erst, was in ihr steckt. Die Benchmarkzeiten zeigen eine Steigerung um mehr als 100% zur 16 Bit-Version.

Eine solche Erweiterung ist hiermit auf jeden Fall in Verbindung mit der 68030-Karte zu empfehlen.

Alle Zeiten wurden mit den Benchmarks aus Heft 2/88 gemessen. Die zweite Tabelle enthält die Zeiten mit 32 Bit Speicher.

Benchmark	68000	68030	68030/68881	handoptimie	ert
1	0,10.06	0,04.66	0,04.75	0,04.69	[min]
2	0,18.20	0,09.75	0,09.75	0,09.76	[min]
3	2,07.06	1,19.09	0,12.94	0,11.98	[min]
Benchmark	68000	68030	68030/68881	handoptimie	ert
1		0,02.19	0,02.16	0,02.11	[min]
2	-	0,03.84	0,03.85	0,03.81	[min]
3		0,41.38	0,12.09	0,11.85	[min]



Die 32-Bit-RAM-Erweiterung

Problemfälle

Im Test hat sich außer der oben beschriebenen Software-Unverträglichkeit auch noch ein weiteres Problem ergeben. Viele externe Speichererweiterungen arbeiten nicht mit dem 68030 zusammen, da er über ein bei Speicherzugriffen verändertes Timing im Vergleich zum 68000 verfügt. Gerade die weitverbreitete Golembox wollte nicht korekt mit dem neuen Prozessor zusammenarbeiten, es zeigten sich seltsa-

me Phänomene, wie z.B. ein Read/ Write-Error beim Zugriff auf die Ramdisk. Der Hersteller der Golembox hat dieses Problem allerdings erkannt und bietet ab sofort eine neue Serie der Speichererweiterung an. Leider erreichte uns das Testexemplar nicht mehr rechtzeitig.

Fazit

Die 68030-Karte von Ronin ist zwar auf dem neuesten Stand der Technik, was den Prozessor betrifft, eine merkliche Steigerung vom 68020 ist jedoch nicht zu vermerken. Die Karte kann also hauptsächlich als Entwicklungssystem für andere 68030-Computer gesehen werden, zumal keine jetzt erhältliche Anwendersoftware die Fähigkeiten des 68030 ausnutzt. Auch läßt die Architektur des Amiga den 68030 aussehen wie einen Formel 1-Motor in Opas Traktor. Die mit 2195.- DM angenehm billige Karte kann jedoch ohne weiteres für die ersten Gehversuche in

der Welt der 68030-Rechner dienen, allerdings sollten Sie sich auf jeden Fall eine 32-Bit Speichererweiterung mit ins Reisegepäck stecken.

DER AMIGA MIT 68030

- + leichter Einbau
- + 32-Bit Bus
- + doppelte Taktfrequenz (14 MHz)
- Probleme mit externen
- Speichererweiterungen
- grundsätzlich keine Verbesserung zur 68020 Karte (siehe Text)

Anbieter: IM Frankfurt, Tel.: 069/410071

Preis: 680x0 Karte 1443.- DM 68020 Prozessor hierzu 545.- DM 68030 Prozessor hierzu 2195.- DM 32 Bit Speicher 2 MB 2495.- DM

> 2 MB Aufrüstung (auf 4 MB) 795.- DM

> > ENDE

Prg. für alle AMIGA-Modelle – Exzellent in Struktur, Grafik, Sound – alle Prg. in Deutsch –

ASTROL. KOSMOGRAMM

Nach Eingabe von Namen, Geburtsort (geografischer Lage) und Geburtszeit werden errechnet: Sternzeit, Aszendent, Medium Coeli, Gestirnestände im Tierkreis, Häuser nach Dr. Koch/Schäck (Horoskop-Daten mit Ephemeriden. Außer dem Bildschirmdisplay kann Ausdruck auf 2 DIN A4-Seiten erfolgen: davon 1/2 Seite allgemeines Persönlichkeitsbild mit Partnerschaftskriterien und 1/2 Seite Tierkreisdiagramm (Horoskop). Alle Planeten mit Sonne und Mond. Für alle Berufs- und Hobby-Astrologen eine unentbehrliche Arbeitserleichterung. 78, -

BACKGAMMON

68, -

BIOKURVEN

Zur Trendbestimmung der Biorhythmen und des seelisch-/geistig-/körperlichen Gleichgewichts mit Druck des Kurvendiagramms von oben nach unten in beliebiger Länge.

In der rechten Blatthälfte das Diagramm, links eine Auswertung des Gesamtpotentials für jeden Tag. Werte für bestimmte Tage auch auf dem Bildschirm. Ausführliche Beschreibung der wissenschaftlichen Grundlagen. Ideal für Partnervergleiche. 58, —

GESCHÄFT

- Bestellung, Auftragsbestätigung, Rechnung, Lieferschein, Mahnung, 6 Briefrahmen mit Firmendaten zur ständigen Verfügung (Anschrift, Konten usw.,

Menge/Preis, Rabatt/Aufschlag, MwSt., Skonto, Verpackung, Versandweg usw.) Mit Einbindung von abgespeicherten Adressen und Artikeln. 198, –

GELD

Man wählt mit der Maus unter
 25 Rechenroutinen in den Bereichen: Anlage – Kapital – Vermögensbildung – Rentensparen
 Rendite – Lasten – Zinsen/Zinseszinsen – Kredit – Hypotheken – Laufzeit – Amortisation – Ratenzahlung – Wertverlust – Nominal- und Effektivzinsen – Ausdruck vollständiger Tilgungsraten – Diskontierung – Devisen/Sorten – Konvertierung



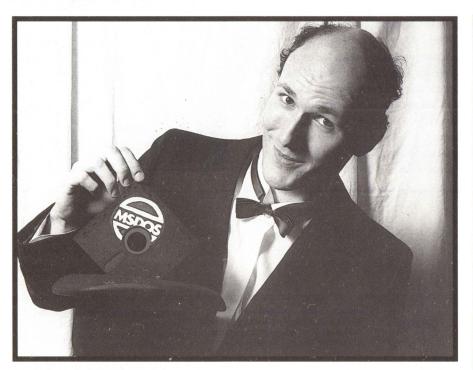
KALORIEN-POLIZEI – Nach Eingabe von Größe, Gewicht, Geschlecht, Arbeitsleistung erfolgt Bedarfsrechnung und Vergleich m. d. tatsächlichen Ernährung (Fett, Eiweiß, Kohlehydrate), Idealgewicht, Vitalstoffe, auf Wunsch Ausdruck. 58. –

Dateiverwaltungen:

Dateiverwaitungen:	
Adressen	69,-
Bibliothek	88,-
Diskothek	78,-
Dokumente	98,-
Exponate	118,-
Galerie	118,-
Inventur, Fibu-gerecht	118,-
Lagerartikel	88,-
Museum	118,-
Videothek	78.

usw. usw. - Fordern Sie mit Freiumschlag unsere Liste an! Im Computer-Center oder bei uns zu obigen unverbindlich empfohlenen Preisen + DM 3. - bei Vorkasse oder DM 4.70 bei Nachnahme





In dieser Folge von PC-INTIM soll einmal ein Überblick über alles gegeben werden, was im Zusammenhang mit dem Speicher und dessen Verwaltung steht. An sich erscheint dieses etwas banal, doch hat man beim PC mit einigen Besonderheiten zu "kämpfen".

iner der wesentlichen Bestandteile eines Rechners ist der "Speicher für wahlfreien Schreib-und Lesezugriff", kurz auch RAM (Random Access Memory) genannt. Ohne diesen könnte man nur einen Computer bauen, der ein bestimmtes, bei der Herstellung vorgesehenes Programm, abarbeiten kann, ohne jegliche Zwischenspeicherung

von Werten. Jedem wird einleuchten, daß dies nur für bestimmte Aufgaben sinnvoll ist, aber für ein universell einsetzbares Gerät völlig unbrauchbar wäre. Aus diesem Grunde ist jeder Computer mit mehr oder weniger RAM ausgestattet. Galt vor einigen Jahren ein Rechner mit 16 KByte als überaus luxuriös ausgestattet, so ist man heute bei manchen Rechnern

schon bei einigen MByte angelangt (siehe AMIGA). Die Gattung der IBM-Kompatiblen unterliegt hier einer Beschränkung auf 640 KByte.

Die Gründe dafür sollen im folgenden aufgezeigt werden, und außerdem soll die Verwaltung und Verwendung dieses Speichers beim Sidecar oder Bridgeboard einmal erläutert werden.

Kilo ist nicht gleich Kilo

Wer sich nicht so genau mit der Hardware eines Rechners auskennt, wird sich vielleicht schon einmal über die etwas seltsam anmutende Zählweise in der Digitaltechnik gewundert haben, denn hier ist ein Kilo nicht gleich tausend (wie z.B. bei Kilogramm, Kilometer, Kilopond usw.), sondern steht für das 1024fache der Grundgröße. Der Grund dafür liegt in der binären Arbeitsweise der Maschinen. Mit einer Stelle kann man im dualen Zahlensystem genau zwei Werte unterscheiden. Mit einer zweistelligen Zahl kann man 2 hoch 2, also 4 Werte darstellen, mit einer dreistelligen 2 hoch 3 (8) und so weiter. Führt man dies nun so fort (Tab.1), so sieht man, daß man mit 10 Stellen im Dualsystem genau bis 1024 zählen kann. Da der Faktor 1000 sich nicht "stellenfüllend" darstellen läßt, hat man diese Abweichung vom normalen Sinn in Kauf genommen, womit man sich einen gewaltigen Vorteil bei der Handhabung verschafft hat.

Genauso verhält es sich mit dem Mega. Mega (10 hoch 6) ergibt sich aus Kilo x Kilo, entspricht demnach 2 hoch 20 gleich 1048576.

Exp. zur Basis 2 Äquivalent	Dezimal-
0	0
1	2
2	2 4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1024
16	65536
20	1048576
24	16777216
Tabelle 1:	CONTROL OF STREET

Splittergruppen

Der größte Unterschied in der Speicherverwaltung von PC und AMIGA besteht wohl in der Segmentierung des Speichers beim PC. Ursache ist die Unfähigkeit des PC's, Zahlen größer 2 hoch 16 in einem Register zu halten. Dies würde nun die Berechnung von Adressen und dergleichen auf 64 KB beschränken. Um diese Schranke zu überwinden ließ man sich nun einen Trick einfallen. Wenn man die Adresse aus zwei Werten bildet, von denen der eine mit einem Faktor gewichtet wird, kann man so diese Grenzen überwinden. Um nun einen Adressraum von 1 MByte zu erreichen, braucht man eine 20stellige Binärzahl.

Wenn man also zwei 16-Bit-Zahlen hat, muß man die eine mit dem Faktor 16 multiplizieren, dann beide addieren, und schon kann man alle gewünschten Zahlen darstellen. Das Multiplizieren und Addieren erledigt der 8088 sogar von alleine. Man muß nur die richtigen Register benutzen. Die Notation der Adressen sieht dann wie folgt aus:

Segment	XXXXh	:	Offset	XXXXh

Um aus diesen beiden Angaben die effektive Adresse zu erhalten, fügt man an das Segment eine 0 an und addiert es mit dem Offset. Das Anhängen einer 0, also das Multiplizieren mit 10h entspricht genau der Gewichtung der Segmentadresse. Ein konkretes Beispiel:

Das	Segment lautet:
	1234h
Der	Offset dazu:
	0005h
Die	effektive Adresse lautet dann

	12340h
+	0005h
	12345h

Diese Art und Weise der Berechnung läßt eine mehrfache Darstellung ein und derselben Adresse zu. So sind z.B. folgende Adressierungen völlig identisch:

SEG.	OFF.
1234h	0005h
1000h	2345h
1111h	1235h
1030h	2045h

Alle Adressen zeigen auf ein und dieselbe Speicherzelle. Dennoch können gewisse Probleme auftauchen. Man stelle sich einmal vor, eine Routine des Betriebssystems hat den Einsprung 0070:0010, beginnt also bei der effektiven Adresse 710h. Äquivalent wäre also auch der Einsprung 0000:0710. Wenn in dieser Routine nun ein intersegmenterer Sprung (ein Sprungbefehl, der nur einen Offset zum Segmentanfang als Parameter hat) vorkommt, landet man nur mit der Notation 0070:0010 für den Einsprung an der richtigen Stelle.

Lautet der Befehl zum Beispiel JMP 326, so führt der Rechner die Routine bei 0070:0326 (also eff. 00A26h) weiter, wobei im zweiten Falle der Programmzeiger auf 0000:0326 (eff. 00326h) zeigt. Speziell beim "wilden" Einprung ins BIOS muß man auf solche Dinge achten.

Manch einem wird dies furchtbar kompliziert vorkommen, doch sollte man nicht die Vorteile der Segmentierung übersehen: Programmzeiger und Sprungzeiger können in WORD-Variablen abgelegt werden. Das läßt die Sache überschaubarer bleiben. Ebenso bequem kann z.B. für Daten ein gesondertes Segment benutzt werden.

An die Grenzen gestoßen

Wie oben schon aufgeführt, kann der PC 1 MByte Speicher ansprechen. Dieser Speicher kann als RAM oder ROM angelegt sein. Bei der Entwicklung des IBM-PC's hat man nun die Festlegung getroffen, daß der Bereich von 0 - 640 KByte für RAM reserviert ist und der

Rest des Adressraums (640 KB - 1 MByte) für bestimmte Betriebssystemzwecke dienen soll. Dazu zählen zum Beispiel der Videospeicher, BIOS und BIOS-Erweiterungen. Eine grobe Übersicht der Belegung ist in Tab.2 zu sehen.

Da man zu den Entstehungszeiten des PC's nicht vermutet hat, daß Arbeitsspeicher in absehbarer Zeit so preiswert wird, hat man den RAM-Bereich leider etwas mager gewählt (zur Erin-



Tab.2 Speicherverteilung beim PC

nerung: Die ersten IBM hatten 16 KByte und einen Kassettenrecorder als Massenspeicher). Zudem setzte sich die anfängliche Idee, im Bereich von CC000h bis EFFFFh zusätzliches ROM zu installieren, nie richtig durch. So hat man heute mit dem kleinen RAM zu kämpfen, wobei man normalerweise noch etlichen Adressraum brachliegen hat, den man aber für RAM-Speicher nicht benutzen kann. MS-DOS kann nämlich im Gegensatz zu AMIGA-DOS nur durchgehendes RAM benutzen. Da im oberen Adressraum die benutzten Areale überall verstreut liegen, können die Zwischenräume nicht mit RAM genutzt werden.

Wo ist was?

Um nun einen genaueren Überblick über die Belegung des Speichers zu geben, wollen wir uns von den niedrigen Adressen nach oben tasten.

00000h - 003FFh

In diesem Bereich sind die Interruptvektoren abgelegt. Für jeden Interrupt sind jeweils 2 Byte als Offset- und 2 Byte als Segment-Adresse reserviert. Die Adresse des Interruptvektors erhält man, wenn man die Interruptnummer mit vier multipliziert. Zu beachten ist die Intelnotation, bei der HiByte und LoByte vertauscht sind. Lautet das Doppelwort z.B. 23 01 70 00, so beginnt die Routine bei 0070:0123. Von den 256 möglichen Interrupts sind bei weitem nicht alle belegt. Die Interrupts sind in folgende Bereiche unterteilt:

INT Oh - INT 7h Prozessor-Interrupts INT 8h - INT Fh Interr. vom Interrupt-Controller INT10h - INT1Ah BIOS-Interrupts INT1Bh - INT1Ch Anwender-Interrupts INT1Dh - INT1Fh Tabellen vom BIOS INT20h - INTFFh MS-DOS Interrupts 00400h - 006FFh Dieser Bereich dient zur Ablage von Systemvariablen und Daten des DOS.

Hier einige interessante Speicherstellen:

```
0040:0000 - 0040:0001
Portadresse 1. serielle
Schnittstelle
0040:0002 - 0040:0003
Portadr.
           2. ser. Schnittst.
0040:0008 - 0040:0009
Portadr. 1. paralle Schnittst.
0040:000A - 0040:000B
Portadr.
            2. par. Schnittst.
0040:0010 - 0040:0011
Equipment-Word (siehe Tab. 3)
0040:0013 - 0040:0014
Größe des Arbeitsspeichers
                              in
KByte
0040:0049
aktueller Videomodus
0040:0066
aktive Farbpalette
00700h - nnnnnh
```

An diese Stelle werden die residenten Teile des Betriebssystems geladen (io.sys, msdos.sys und der residente Teil von Command.com. Da diese Programme von DOS-Version zu DOS-Version unterschiedliche Längen haben, kann keine Endadresse angegeben werden.

```
nnnnh - 9FFFFh
```

Dieser Bereich steht beim Ausbau mit 640 KByte als RAM zur freien Verfügung. Mit 512 KByte endet das RAM bei 7FFFFh, mit 256 KByte bei 3FFFFh.

```
A0000h - AFFFFh
```

Ist der Rechner mit einer EGA-Karte ausgerüstet, so befindet sich hier der Bildschirmspeicher.

```
B0000h - B0FFFh
```

Bildschirmspeicher einer MDA-Videokarte.

```
B8000h - BBFFFh
```

Bildschirmspeicher des CGA.

```
B0000h - BFFFFh
```

Bildschirmspeicher der Hercules-Karte.

```
C8000h - CBFFFh
```

Reserviert für Festplatten-BIOS. Unterteilt in die Bereiche

C8000h - C9FFFh für den ersten Hard-DiskController und

CA000h - CBFFFh für den zweiten Controller.

```
CC000h - EFFFFh
```

Reserviert für ROM-Erweiterungen.

```
FC000h - FFFFFh
```

BIOS des Rechners.

Neben dieser Standardbelegung findet man bei Sidecar oder Bridgeboard noch einige andere Belegungen. Diese sind:

```
A0000h - AFFFFh oder
D0000h - DFFFFh oder
E0000h - EFFFFh
```

64 KByte RAM zum Datenaustausch (für Janus-Disk, Aread, Awrite usw.). Die ersten vier Kilobyte RAM (x0000h - x0FFFh) fungieren jedoch als ROM und werden von DOS als BIOS-Erweiterung angesehen. Die Lage dieses Speichers kann zwischen den drei Bereichen umgeschaltet werden. Allerdings ist nur bei Verwendung des Segments E000h die volle Funktionsfähigkeit des Janustreibers gegeben.

```
F0000h - F3FFFh
```

Parameter-RAM (16 KByte). In diesem Bereich werden Variablen und Parameter zwischen AMIGA und PC ausgetauscht, die zum Datenaustausch notwendig sind. Dazu zählen z.B. Parameter für den Festplattenzugriff des AMIGA oder Zugriffsbeschränkungen für das Datenaustausch-RAM (damit die Werte nicht vor dem Auslesen durch die eine CPU von dem anderen Prozessor überschrieben werden).

Des weiteren sind die Video-RAM-Bereiche auf dem Interface jeweils verdoppelt (MDA B0000h - B1FFFh, CGA B8000h - BFFFFh

Bit B	edeutung
0	Diamora Madua an/aua
1	Diagnose-Modus an/aus
L L	8087 vorhanden
2+3	RAM auf Motherboard (* 64k)
4+5	Videostartmodus: 0-none, 1-40x25 cga,
	2-80x25 cga, 3-mda
6+7	Anzahl Diskettenlaufw1
8	unbenutzt
9+10	Anzahl serieller Schnittstellen
11	unbenutzt
12	Gameport vorhanden
13	unbenutzt
14+15	Anzahl paraller Schnittstellen

Tabelle 3: Das Equipment-Word verrät die Hardwarekonfiguration.

VON BERND MARIENFELDT

VOM TRACKDISK-DEVICE

zur `singenden Floppy`

Dieser Artikel soll einen Einblick in die fantastische Welt der Amiga-Floppy-Disk geben. Es wird sowohl das Arbeiten mit dem Trackdisk-Device als auch das direkte 'Ansprechen' des CIA-Ports mit DMA-Zugriff erklärt. So kommen Sie z.B. in den Genuß, Ihre Floppy 'singen' zu hören. Als Schmankerl wird Ihnen zum Schluß als Belohnung noch gezeigt, wie Sie Programme mit bis zu 1024 Bytes in den Bootblock einbinden können.

lso, jetzt nur eins nicht tun: Weiterblättern! Zuerst sollte klargestellt werden, daß dieser Artikel, obwohl die Listings in Assembler erstellt sind, keineswegs nur für Assembler-Profis gedacht ist. Die Listings sind so gut dokumentiert, daß sie jeder weitgehend verstehen muß und daß das daraus erworbene Wissen in eigene Taten umgesetzt werden kann. So bleibt auch der letzte Basic-Freak nicht verschont. Wenn alles nichts hilft und man ganz verzweifelt, bleibt immer noch der Weg, die Listings einfach abzutippen und in eigene Programme einzubinden. Leute. die schon fundierte Kenntnisse über die Grundlagen besitzen, können diesen Teil überspringen, aber nicht vergessen: Auch der Meister fällt nicht vom Himmel!

Grundlegendes...

Das Wort Floppy sagt wohl jedem etwas, aber über 'das ist doch das Ding, wo man die Scheiben reinlegt und einlädt', kommen die meisten nicht hinaus. Floppy ist eine abgekürzte Redewendung für Floppy-Drive, was soviel heißt wie Disketten-Laufwerk. Aber allzu genau wollen wir auch nicht sein, denn uns kommt es ja auf die Fakten an. Der Amiga besitzt ein Laufwerk des Typs 3 1/2 Zoll, was bedeutet, daß die Größe der Disketten 3 1/2 Zoll beträgt. Sie haben eine Speichergröße von 901120 Bytes. Nun, wie kommt diese Zahl zustande?

Ganz einfach...

Die Disketten werden in 80 Spuren, beim Amiga nennt man diese Spuren Zylinder, unterteilt. Jeder dieser Zylinder ist wiederum in 11 Sektoren aufgeteilt. Schließlich muß man noch wissen, daß der Amiga zwei Schreib-/Leseköpfe besitzt, also von unten und oben lesen/schreiben kann. Nimmt man nun die Erkenntnis zur Hand, daß jeder Sektor 512 Bytes groß ist, sprich so viel Daten fassen kann, kommt man leicht auf die erwähnte Zahl:

80*11*512*2 = 901120 Bytes.

Um später zu wissen, wie man auf einen bestimmten Sektor zugreift, sollten Sie noch wissen, wie der Amiga diese Daten auf die Diskette schreibt und natürlich, wie er sie liest: Sektor 0 liegt auf Zylinder 0/Seite 0(unterer Kopf), Sekt. 1 auf Zylinder 0/Seite 0... Sektor 12 (Block 11, die Sektoren können auch in Blocks von 0 - 1759 eingeteilt werden) liegt auf Zyl. 0/Seite 1. So entsteht ein abwechselnder Schreib/ Lese- Rhythmus (unten/oben). Das sollte als Einführung genügen... halt, da wäre ja noch zu sagen, daß maximal 3 Laufwerke angeschlossen werden können. Mit dem internen (was Sie ja hoffentlich haben, oder was haben Sie für einen Amiga?) sind das dann 4 Laufwerke, die der Amiga verwalten kann. (Ganz schön !!).

Die Laufwerksprogrammierung...

des Amiga läßt sich auf zweierlei Arten bewerkstelligen, fangen wir also

mit der einen Methode an:

Hier bedient man sich des sogenannten Trackdisk-Devices. Wie der Name schon sagt handelt es sich hier um ein Device des Amiga, was natürlich spezielle Handhabung erfordert. In Tabelle 1 finden Sie eine genaue Auflistung der I/O-Struktur mit Distanzwerten. Falls Sie es nicht wissen: um mit einem Device arbeiten zu können, benötigt man eine solche Struktur. Sie dient der Kommunikation zwischen Programm und Amiga. Bevor Sie aber darangehen, das Device zu öffnen, was für die Arbeit notwendig ist, benötigt man einen MessagePort, über den sämtliche Nachrichten übertragen werden. Diesen Messageport kann man auf zwei Arten bekommen:

Die erste sehen Sie in Listing 1. Hier liegt der Msgport (das tut er natürlich immer) im 68-Adreßdistanzwert zum Windowzeiger. Hierfür müssen Sie natürlich zuvor ein Window geöffnet haben.

Die zweite Methode wäre, ihn mit der Findtask-Routine zu erhalten; dies sehen Sie in Listing 2.

Allerdings handelt es sich bei Methode 2 nicht um einen richtigen Msgport, er wird als Replyport bezeichnet. Am besten verwendet man den Port von einem Window, das natürlich geöffnet werden muß (kleiner Tip am Rand: Ein Window kann auch unsichtbar sein, wenn Sie die Flags entsprechend setzen!). Nun geht es aber daran, das Trackdiskdevice zu öffnen. Hier wird die IOstruktur nach a1 (68000 Adress-Register) geladen, sowie der Devicename, hier "trackdisk. device", nach a0 (68000-Adreß-Register) gebracht. Danach ruft man die Opendevice-Routine der Execlibrary auf. Hier bietet sich auch an zu testen, wieviele Laufwerke angeschlossen sind. Man öffnet einfach 4 mal den TRDDevice mit unterschiedlicher Laufwerksnummer (in d0 übergeben und unterschiedlicher IO-Struktur, dO = Daten-Register), und so läßt sich dann in d0 testen ob das Laufwerk angeschlossen ist (siehe Errors in Tabelle2). Nachdem dies geschehen ist, folgt das Einfachere. Um beispielsweise einen Sektor zu lesen, verwendet man den DoIO-Aufruf (auch SendIO möglich). Hier übergibt man Diskbuffer, also wohin die Daten gelesen werden, die Länge (hier 1*512

```
2:
        LISTING 1
3:
    ;
 4:
 5:
               BOOTCOPY
 6:
 7:
        Written by Bernd Marienfeldt
 8:
        (c) KICKSTART '88
 9:
10:
     START:
     movem.1
11:
              d0-d7/a0-a6, -(sp)
                                   ; Register auf SP Retten
12: move.1
                                    ; Execbase nach a6
              4.a6
13:
    move. 1
              #$2c00.d0
                                   ; Speichergroesse nach D0
              #$10003,d1
                                    ; Chip+Clear+Public nach D1
14:
     move.1
15:
     isr
              -198 (a6)
                                    : Speicherres. mit Allocmem
16:
              Fehler
                                    ; Wenn 0 Fehler bei Spei-
     beq
17:
                                    ; cherzuweisung => raus!
18:
     move.1
              d0.diskbuffer
                                    ; Speicherblockadr. retten
19:
20:
    move.1
              4.a6
                                    ; execbase nach a6
              intlib, a1
                                      'intuition.library' -> A1
21:
22:
              -408 (a6)
                                    ; Library oeffnen.
     isr
23:
     move.1
              d0, intbase
                                   ; Zeiger in DO nach Basis
24:
     lea
              Graphlib, al
                                   ; 'graphics.library'
25:
     jsr
              -408 (a6)
                                    ; Library oeffnen
     move.1
              d0, Graphbase
                                    ; und Zeiger nach Basis.
27:
28:
              intbase, a6
     move.1
                                   ; Intbase nach a6
29: lea
              newwindow, a0
                                   ; Windowstruktur laden
30: jsr
              -204 (a6)
                                    ; Window oeffnen
31:
              #0,d0
                                   ; Fehler ?
     cmp.1
32: beq
              fehler
                                   ; ja => ende
33: move.1
              d0, window
                                   ; Windowzeiger retten
34: move.1
              d0.a0
                                   ; Wzeiger nach A0
35:
                                    ; Wzeiger nach Al
     move.1
              d0, a1
36: move.1
              86(a0), msgport
                                   : MsgPort bei 86-Distanz
37: move.1
              50(a1), wrastport
                                    ; WRastport bei 50-Distanz
38:
     move.1
                                    ; Execbase nach a6
              4, a6
39.
    lea
               devicename, a0
                                    ; Device (TRDdev.) laden
40: move.1
               #0,d0
                                    ; Unit 0 (int. Drive) -> d0
                                    ; IO/Struktur nach al
              IOrequest, al
41:
    lea
42:
                                     Kein Flags => loeschen!
     clr.1
              d1
                                    ; Device Oeffnen
43:
     isr
               -444 (A6)
44:
     BeschDat1:
45:
     dc.1
               1,177,30,string1,14; Farbe,x,y,string,laenge
              1,118,40,string2,28; "
46: dc.1
                                            11 11
              1,12,60,string3,55 ; "
47: dc.1
                                   ; 2x durchlaufen (einmal
48: move.1
              #2.bcount
49:
                                   ; dbra eh immer durch)
50: lea
              BeschDat1, a5
                                    ; obige Daten nach A5
51: jsr
              Beschrifter
                                    ; Beschrifter aufrufen
52:
     warten0:
               #6,$bfe001
                                    ; Bit 6 im CIA-Port testen
53: btst
                                    ; Falls nicht gesetzt nach
54:
     bne.s
               warten0
55:
                                    ; warten0=>Schl.
56:
              TD Read
                                    ; Zylinder 0 lesen
     jsr
57:
                                    ; Motor ausschalten
              TD motoroff
     isr
58:
     warten1:
59:
     btst
               #6, $bfe001
                                    ; Bit 6 wieder testen
60 .
     bne s
              warten1
                                    : no Reaktion => Schleife
               TD Write
                                    ; Auf Zylinder 0 schreiben
61: jsr
62:
               TD MotorOff
                                    ; Motor ausschalten
     jsr
63:
     close:
64:
     move.1
               4, a6
                                    ; execbase, a6
                                    ; IOStruktur nach Al
65: lea
               IORequest, a1
66:
     jsr
               -450 (a6)
                                    ; Device schliessen
67: move.1
              intbase.a6
                                    : Intbase.a6
               window, a0
                                    ; Windowzeiger nach A0
68: move.1
69:
               -72 (a6)
                                    ; Window schliessen
     jsr
                                    ; execbase, a6
70:
     move.1
               4,a6
              intbase, al
71:
     move.1
                                    ; Intbase, a6
72:
     jsr
               -414 (a6)
                                    ; Library schliessen
73:
74:
                                    ; Execbase nach a6
     move.1
               4,a6
```

```
move.1
               #$2c00.d0
                                    ; Speichergroesse nach D0
 75.
               diskbuffer, al
                                      Speicherblock nach A1
 76:
      move.1
               -210 (a6)
                                    ; freigeben. mit Freemem
 77:
      jsr
 78:
                                    ; Kein Test, da eh zu spaet
 79:
 80:
      movem.1 (sp)+, d0-d7/a0-a6
                                    ; Register vom SP holen
 81:
      rts
                                    ; Zurueck zum SEKA oder CLI
 82:
 83:
 84: Beschrifter:
                                    : Beschrifter Funktion
                                    ; Durchlaufszahl nach D7
 85: move.1
               Bcount, d7
 86:
      BeschrLoop:
                                    ; Graphik-Library-Base ->A6
 87:
      move.1
               graphbase, a6
 88: move.1
               wrastport, a1
                                    ; Wrastport nach A1
                                    ; Wert in A5 nach d0
 89: move.1
               (a5) +, d0
 90:
                                    ; (PenFarbe), eins weiter.
               -342 (a6)
                                   ; Setpen (PenA-Farbe setzen)
 91:
     isr
 92: move.1
               (a5) +, d0
                                    ; A5 nach d0 (X-Pos) +1
 93:
     move.1
               (a5) + , d1
                                    ; A5 nach d1 (Y-Pos) +1
 94:
      jsr
               -240 (a6)
                                    ; Cursor setzen
 95: move.1
               (a5) +, a0
                                    ; A5 -> a0(string) +1
                                    ; A5 -> d0(Stringlaenge) +1
 96:
               (a5)+,d0
     move.1
 97:
      jsr
               -60 (a6)
                                    ; Textschreiben
 98:
      dbra
                                    ; Wert in d7 erreicht ?
               d7, BeschrLoop
 99.
     rts
                                    ; fertig? zurueckspringen.
100:
     ; WICHTIGE ANMERKUNG ZUM TRDDEVICE:
101 .
102:
     ; Beim Commando Format (11) duerfen im Offset- und
103:
     ; Laengenregister nur Vielfache von 5632 stehen, da
104:
      ; der Device nur in der Lage ist einen halben Cylinder,
105:
     ; also 11 Sektoren, zu formatieren (schreiben)!!!
106:
     ; EBENSO ist beim Commando Write(3) zu beachten, dass
     ; hier im Offset- und Laengenregister nur Vielfache von
108:
109: ; 512 stehen duerfen, da das Device und auch das
     ; Laufwerk (AMIGA) nur in der Lage ist
110:
111:
      ; ganze Sektoren zu lesen!!!!
112:
                                  ; LeseFunktion
113:
      TD read:
114:
      move.1 4,a6
                                    : Execbase nach A6
115:
                                    ; IOstruktur nach A1
     lea
              IORequest, a1
116: move
              #2,28(A1)
                                   ; Command(2) setzen
117:
      move.1 #$2c00,36(a1)
                                    ; Laenge (512*11*2) setzen
118:
      move.1
              diskbuffer, 40 (A1)
                                    ; Datas (Ladeadresse)
119:
      move.1 #$0000,44(a1)
                                    ; Offset (Cylinderadr.) Cyl.0
                                    ; DoIO aufrufen
120:
     jsr
              -456 (A6)
121:
                                    ; Ruecksprung
122:
123:
      TD Write:
                   ; Schreib-Routine - Eigentlich wird
124:
                      : ja formatiert.da Command = 11. erspart
125:
                      ; das vorformatieren - formatiert einfach
126:
                      ; mit Daten im Speicher.
127:
      move.1
               4.a6
                                   ; Execbase, A6
128:
      lea
               IORequest, a1
                                   ; IOstruktur nach A1
129:
               #11,28(a1)
                                    : Command=11 => formatieren
      move
                                   ; Laenge (512*11*2=$2c00)
130:
      move.1
               #$2c00,36(a1)
131:
      move.1
               diskbuffer, 40 (A1)
                                    ; Databuffer wie bei Read
132:
      move.1
               #$0000,44(a1)
                                    ; Und wieder Zylinderoffset
133:
               -456 (A6)
                                    ; DoIO aufrufen
      isr
134:
      rts
                                    ; Ruecksprung
135:
      TD MOTORoff:
136:
                                    ; Motorauschalten
137:
      move.1
                                   ; Execbase, A6
                                   ; IOstruktur nach Al
138:
      lea
               IORequest, a1
      move.1
139:
               #0,36(A1)
                                    ; Laenge 0 => auschalten
140:
     move
               #9,28(a1)
                                    ; Command 9 =>Motor an/aus
                                    ; DoIo aufrufen
141:
      jsr
               -456 (a6)
142:
143:
144:
145:
      ; Namen fuer Librarys und Zeiger zum Zwischenspeichern
146:
      Fehler:
      Devicename: dc.b 'trackdisk.device', 0
```

Bytes), die Leseposition (auch Offset, hier 512 Bytes), sowie das Kommando 'Lesen' (eine Auflistung der Kommandos finden Sie in Tabelle 3). Danach ruft man DoIO auf, und schon liest der Amiga von Disk. Allerdings sollte man nicht vergessen, das Laufwerk wieder auszuschalten! Am besten sehen Sie sich das alles an Listing 1 an, wo ein ganzer Zylinder (Seite 0 und 1) gelesen und geschrieben sowie der Motor einund ausgeschaltet wird.

Wollen Sie mehrere Laufwerke benutzen, müssen Sie natürlich auch mehrere IO-Strukturen 'initialisieren', demnach mehrmals Opendevice mit unterschiedlicher Drivenr. sowie unterschiedlicher IO-Strukturen aufrufen. Zu guter Letzt möchte ich noch darauf hinweisen, daß man nie vergessen sollte, Speicher für den Diskpuffer zu reservieren, da schon beim Anschluß einer Speichererweiterung oder beim 2000'er/500'er die ja mehr als 512 Kbyte RAM haben, es zu Totalabstürzen kommt. Der Grund dafür liegt in der Tatsache, daß das Trackdiskdevice den Blitter benutzt, der ja bekanntlich nur auf die unteren 512 Kbyte, also Chipmem, zugreifen kann. Ein Umkopieren der Daten in das besagte Chipmem ist ohne weiteres möglich, ich habe aber aus Gründen des Programmumfangs darauf verzichtet.

Jetzt sollten Sie eigentlich in der Lage sein, mit dem Trackdisk-Device zu arbeiten. Bald wird man aber feststellen, daß man bei der Benutzung sehr stark eingeschränkt ist (ganz von der Zeit, die ein Trackdiskzugriff länger dauert, abgesehen). So kann man beispielsweise keine sogenannten Fremdformate lesen oder gar über Zylinder 80 hinauslesen/-schreiben. Diese Probleme lassen sich allerdings mit der zweiten versprochenen Methode, das Laufwerk anzusprechen, beseitigen. Die sogenannte 'direkte Programmierung' des Laufwerks unterteilt sich selbst in 2 Teile:

Der erste Teil besteht aus der Adressierung des CIA-A/CIA-B (Typ 8520, ähnlich wie der des C-64), der zweite aus der Datenübergabe an die DMA-Verwaltung. Fangen wir mit dem ersten Teil an. Hier werden also ersteinmal Grundeinstellungen oder Kommandos direkt an das Laufwerk erteilt. Es ist hier z.B. möglich zu testen, ob sich der Schreib-/Lesekopf auf Zylin-

der 0 befindet, oder ob die Diskette schreibgeschützt ist. So läßt sich weiter die Steprichtung des Kopfes angeben, das Laufwerk auswählen, die Seite angeben und gar der Kopf zum Steppen bringen. Eine Auswahl des jeweiligen Bits (eine schöne Aufgliederung aller benötigten CIA-Bits finden Sie in Tabelle 4) findet durch Setzen und Löschen des Bits statt.

Selbstverständlich läßt sich der gleiche Effekt auch durch 'andi'- oder 'or'-Verknüpfungen erreichen. Will man z.B. den Motor (DF0) einschalten, so löscht man zuerst Bit 3 in \$bfd100 und setzt es danach wieder (bclear, bset). Daraufhin löscht und setzt man Bit 7 in \$bfd100 für MotorOn! Alle weiteren Bits werden ebenso gesetzt/gelöscht. Tests der Bits lassen sich recht komfortabel mit 'btst' erreichen.

Das war also der erste Streich, der zweite folgt sogleich...

Jetzt werden alle Daten für den DMA-Zugriff übergeben: Das DSKLEN (\$dff024) wird mit 0 beschrieben. Das DSKEN im DMACON(\$8210 nach \$dff096, 2 hoch bit 15-set, 2 hoch bit 9dmaenable, 2 hoch bit 4-diskenable) wird beschrieben. Der Datapuffer wird ins DSKPTL(\$dff020) mit Langwort geschrieben, oder auch erst Low, dann High, jetzt wird das DSKLEN-Register mit den richtigen Werten beschrieben, das Gleiche nochmal (nötig!), nun warten, bis Disk-DMA beendet ist, danach DSKLEN wieder auf Low setzen (dieser Vorgang wird deutlich, wenn man sich Tabelle 5 ansieht).

Kombiniert man also die DMA-Handhabung mit der direkten CIA-Programmierung, ist man in der Lage, alles Gewünschte mit dem Laufwerk anzustellen. Natürlich von der Hardwareseite aus gesehen. Die Software (z.B. den Kopierschutz oder das Kopierprgogramm) muß man allerdings selbst schreiben.

Hier noch einige Angaben zur Vervollständigung des Bisherigen:

Es gibt noch das DSKBYTR-Register (\$dff01a lesen), das dazu dient, den Status des Diskkontrollers (den man ja die ganze Zeit programmiert) zu erfassen (Tabelle 5). Um auf Synchronmarkierungen (spez. bei Kopierprogrammen) zu warten, oder um die DMA-Übertragung erst ab einem bestimmten Zeitpunkt zu beginnen, dient das DSKSYN-Register (\$dff07e), welches

```
Graphlib: dc.b 'graphics.library',0
     Intlib:
                 dc.b 'intuition.library',0
149:
150:
      String1:
                dc.b 'BOOTCOPY V.1.0',0
                dc.b 'WRITTEN BY BERND MARIENFELDT', 0
151: String2:
               Dc.b '* INSERT SOURCE - LEFTMOUSE - INSERT
152: String3:
                     DEST. - LEFTMOUSE *'
153:
154:
     Diskbuffer:dc.1 0
155 .
     Graphbase: dc.1 0
156:
     Wrastport: dc.1 0
157:
     Intbase:
                dc.1 0
158:
     Window:
                dc.1 0
159:
     Bcount:
                dc.1 0
160:
161: Newwindow:
                           ; Windowstruktur Windowdaten
                120.50
                           ; Startposition
162: dc.w
163:
     dc.w
                470,100
                           ; Breite und Hoehe des Windows
164:
                           ; Farbe fuer Menue und Gadgets
     dc.b
                0.1
165:
     dc.1
                $0300
                          ; IDCMP-Flags
166:
                           ; ($0300=Windowclose+Menuepick)
167:
                $1007
     dc 1
                           ; Aussehen des Windows
168: dc.1
                           ; Pointer zu speziellem Gadget
169:
     dc.1
                           ; Pointer zu user check mark
                0
170:
     dc.1
                0
                           ; Pointer zum Title
171:
     dc.1
                0
                           ; Pointer zum Screen
172:
     dc.1
                0
                           ; Pointer zu Superbitmap
173:
     dc.w
               1,1
                           ; Kleinste Groesse
174:
                470,100
     dc.w
                           ; Groesste Groesse
175:
     dc.w
                $0001
                           ; ScreenTyp
176:
                           ; Genaueres ueber Windowstrukt.
177:
                            ; und Intuition bitte
178:
                            ; Fachliteratur entnehmen.
179:
     IORequest:
180:
     dc.1 0
                   ; Input/Output-Struktur fuer Device-
181 .
     dc.1 0
                   ; Kommunikation. Dient dazu, Werte zu
182: dc.b 5
                  ; holen, bekommen und zwischenzuspeichern.
183:
     dc.b 0
                  ; Wichtig fuer TrdDiskDevice ist auch der
184:
     dc.1 MN node ; Msgport, (erhaeltlich ueber das Window)
185: Msgport:
186: dc.1 0
187:
     dc.w 0
188: dc.1 0
                      ; Device
                     ; UnitNr. (0,1,2,3)
189: dc.1 0
190:
     dc.w 0
                     ; Command-Flag
191:
     dc.b 0
                     ; Flags
192: dc.b 0
                     ; Errors (Hier Errors testen)
193: dc.1 0
                     ; Actual-Field
194:
     dc.1 0
                     ; Laenge (Laenge der Daten)
195: dc.1 0
                     ; Datas (Datenbuffer)
196: dc.1 0
                     ; Offset (Cylinder)
197: dc.1 0
                     ; Reserviert
198:
     dc.1 0
                      ; Reserviert
199: MN node:
                   dc.b 'port',0 ;Portname
200:
     even
```

```
1:
 2:
     : LISTING 2
 3.
     ; 'Find Port' fuer Trackdiskdevice
 5:
     ; BEMERKUNG: Nicht so abtippen, dient nur als Beispiel!
 6:
 7:
 8:
    move.1
                4.a6
                                  ; execbase, a6
 9:
     sub.1
                a1, a1
                                  ; Zeiger auf eigenen Task
10:
     isr
                -294 (a6)
                                  : FindTask
11:
                d0, Replyport+16 ; Port+16
12:
13:
     move.1
                4,a6
                                  ; Mit Addport Port dem System
14:
     lea
                Replyport, al
15:
     jsr
                -354 (a6)
                                  ; 'hinzufuegen' .
```

```
17:
18:
19:
20:
                IOstruktur, al
                                  ;Beispiel an DoIO
     lea
21:
     move.1
                #replyport, 14(a1)
22.
23:
     Commando setzen
24.
     Buffer
              setzen
25:
     Laenge
              setzen
26:
     Offset setzen
27:
28:
     DoIO aufrufen
29:
30:
                replyport, al
     lea
                                ;Port wieder entfernen
31 .
     jsr
                -360 (a6)
     replyport: blk.b 10,0
33:
```

```
1:
     ; LISTING 3
 3:
 4:
                HEAD-MUSIC
 5:
    ;
 6:
            Written by Bernd Marienfeldt
 7:
            (c) KICKSTART '88
    ;
 8:
            SEKA-ASSEMBLER
     ;
 9:
10:
    START:
11:
    movem.1
              d0-d7/a0-a6, -(sp)
                                   ; Register -> StackPointer
12:
               #$4000,$dff09a
                                   ; Interrupts ausschalten
13:
                                   ; => voller Diskzugriff
              #3,$bfd100
14:
                                   ; DFO-Auswahl Bit setzen
15:
    belr
              #3. $bfd100
                                   ; und loeschen => Auswahl
16:
     Frombegin:
17:
              #msize.d5
    move.1
                                    : Anz.durchlaufene Daten
18:
    lea
              Musicdatas, a5
                                    ; Musikdaten -> A5 laden
19:
     Start1:
20.
    move.1
               (a5)+, frequenz1
                                    ; Frequenz1 setzen, +1
21: move.1
               (a5)+, frequenz2
                                    ; Frequenz2 setzen, +1
22:
    Start2:
23:
               #0,$bfd100
     bset
                                     ; SteppenBit setzen und
24:
    bclr
                                     ; loeschen => Steppen
              #0, $bfd100
                                    ; Wait-Routine
25:
              wait
     jsr
26:
     btst
              #6,$bfe001
                                    ; L-Maustaste testen,
27:
    beq
              finito
                                    ; gedrueckt = Ende
     cmp.1
                                    ; Frequenz1 = 0 ?
28:
              #0, frequenz1
29:
    beq
              aus
                                    ; Fequenzen neu setzen
30:
     sub.1
              #1, frequenz1
                                     ; Frequenz1 - 1
31:
    bra
              start2
                                     ; 'Bra start2': mit alten
32:
    Aus:
                                     ; Werten weiterarbeiten.
33:
              d5, start1
                                     ; Bis Musikdaten abgear-
34:
                                     : beitet sind nach start1
35:
                                     ; verzweigen. (Musikdaten
36:
                                     ; minus 1, da Schleife
37:
                                     ; mindestens einmal
38:
                                     : durchlaufen wird.
39:
40:
41 .
    Finito:
               (a7)+,d0-d7/a0-a6; Register von Stack holen
42:
    movem.1
43:
     rts
                                 ; Ausstieg =>SEKA
44:
45:
     Wait:
                                 : Dies Wait-Schleife spart
46:
                                 ; im Gegensatz zur Delay-
47:
                                   Funkt. das Oeffnen der
48:
                                 ; DosLibrary.
49:
    move.1
              frequenz2, d3
                                 ; Frequenz 2 nach D3
50:
     wait1:
51:
                                 ; Fequenz = 0, falls ja nach
     beq
               auswait
52:
                                 ; auswait => Ende Warteschl.
53.
     sub.1
              #1.d3
                                 ; Frequenz 2 -1
54:
     bra
              wait1
                                  ; Goto wait1 => SCHLEIFE
55:
     Auswait:
```

das Datenwort enthält, an dem die DMA-Übertragung stattfinden soll. Bei einem einfachen Kopierprogramm würde es ausreichen, die Daten in den Speicher zu lesen und anschließend wieder auf Disk zu schreiben. Sollen aber Daten im Speicher manipuliert werden, müssen diese erst dekodiert und anschließend wieder kodiert werden, da sie in einem anderen Format auf der Diskette stehen, als sie der Amiga im Speicher verarbeitet. Hierzu dient das ADKCON und ADKCONR-(\$dff09e-schreiben \$dff010-lesen). Zur Erläuterung sei wieder Tabelle 5 erwähnt.

Für 100% Freaks

Abschließend möchte ich noch für die 100% Freaks das interne Datenregister DSKDAT (\$dff026 schreiben, \$dff008 lesen) erwähnen. Hierhin werden die beim DMA-Zugriff zu transportierenden Daten vom Amiga intern 'befördert'. Dies erlaubt den relativ, komfortablen und vor allem schnellen DMA-Zugriff, da die Datenregister nach dem 'Transport' sofort wieder neu beschrieben werden können (Abfrage, ob er schon fertig ist, am besten über ein Interrupt lösen). Dies war nun viel Theorie auf einmal, und viele werden sagen, daß das doch alles zu kompliziert sei.

Die singende Floppy

Trotzdem möchte ich noch einmal darauf hinweisen, daß selbst der BASIC-Anwender am Ende damit was anfangen kann, der nur die Hälfte versteht. Viel Theorie, wenig Praxis: weit gefehlt: Bevor Sie in den Genuß von Listing 3 kommen, sollten Sie noch folgendes lesen: Ja, Sie haben oben schon richtig gelesen, "singende Floppy", nein - nicht, daß jetzt Ihre Floppy zu singen anfängt, aber zumindest eine Melodie ist zu vernehmen. Im Ernst. Das Amiga Disklaufwerk verbirgt im Innersten einen sogennanten Steppmotor, der dazu verwendet wird, den Schreib-/Lesekopf den Spuren entlang zu 'steppen' (fahren). Um Töne mit dem Laufwerk zu erzeugen, macht man sich diesen 'kleinen Helfer' zunutze. Bitte noch nicht in Ohnmacht fallen -Ihr Floppy-Kopf wird es überle-

Der Trick des Ganzen - denn wie man

sieht, handelt es sich um ein relativ kleines und einfaches Programm - besteht darin, dem Kopf so schnell hintereinander die Anweisung 'steppen' zu geben, daß er nicht mehr mitkommt, und in Folge dessen anfängt zu vibrieren - dies erzeugt dann einen Ton. In der Praxis sieht das folgendermaßen aus: Der Amiga teilt dem Steppmotor die Anweisung mit zu steppen - er fängt also an. Jetzt bekommt er sofort wieder eine Anweisung zu steppen, der Motor 'überschlägt' sich, er kommt nicht mit und kann demzufolge keine vollständige Bewegung ausführen. Ist allerdings der Abstand zwischen diesen Anweisungen zu groß, gehen die Töne von hohen Frequenzen in niedrige über, sprich - bei sehr großen Wartezeiten (Abstand zwischen Steppvorgängen) hört man das Laufwerk richtig steppen). Das Programm 'HEADMU-SIC' ist einfachster Art und soll auch nur verdeutlichen, wie das Ganze in der Praxis zu vollziehen ist. Unbegrenzten Phantasien, wie Programmen, die nach Eingabe der Töne (in richtig notierter Schreibweise) die Melodie auf die Diskette schreiben, seien keine Grenzen gesetzt. Hier sind Ihre Phantasie und Ihr Einfallsreichtum gefragt.

Zum Listing 3:

Frequenz 1 ist die Anzahl der Steppvorgänge (dauert je nach Frequenz, also Anzahl, unterschiedlich lang!), Frequenz 2 der Abstand zwischen Steppvorgang (n) und Steppvorgang (n+1). Diese Frequenz ist die eigentliche Tonfrequenz, da ja der unterschiedliche Abstand zu verschiedenen Tönen führt. Im Label Musicdatas werden die Daten in Langwörtern aufgeführt. Möchte man also selbst ein Lied komponieren, braucht man nur hier verschiedene Daten einzusetzen (erst Frequenz 1, Frequenz 2 u.s.w.). Die Anzahl der Daten errechnet das Programm selbst. Zum Schluß seien noch zwei Hinweise angebracht:

- 1. Durch die Angabe von 0 in Fequenz 1 und einem Wartewert in Frequenz 2 ist es möglich, eine kurze Pause zwischen den Tönen einzulegen (ersichtlich bei 'Alle meine Entchen') - der Fachmann nennt das punktierte Noten.
- 2. Dadurch, daß im Programm eine ständige 'btst'- Abfrage auf die linke

```
56: rts
                                 ; Zurueckspringen nach
57:
                                  'jsr wait'.
58:
59:
     ; ** DATAS fuer 'DEUTSCHE HYMNE-THEME' **
60: Musicdatas:
61: MSTART:
62:
    dc.1 1300,100,700,90,1100,80,1250,90
63:
    dc.1 1150,75,1100,81,700,90,500,110
    dc.1 900,102,1400,59,1200,67,1100,76
     dc.1 1000,82,950,92,600,82,500,106
65:
66:
     dc.1 1300,65
67:
68:
69:
     ; ** DATAS fuer 'ALLE MEINE ENDchen-THEME' **
70:
71:
     dc.1 900.100.900.90.900.80.900.75
72:
     dc.1 1300,66,0,$8000,1200,66,0,$2000
73:
74: dc.1 1000,60,0,$4000,800,60,0,$4000
75: dc.1 800,60,0,$4000,800,60,0,$4000
76:
     dc.1 1200,66,0,$13000,1000,60,0,$4000
77:
    dc.1 800,60,0,$4000,800,60,0,$4000
78: dc.1 800,60,0,$4000,1200,66,0,$4000
79:
     dc.1 800,75,0,$4000,800,75,0,$4000
80:
     dc.1 800,75,0,$4000,800,75,0,$4000
81: dc.1 1200,90,0,$4000,1200,90,0,$2000
82:
     dc.1 700,66,0,$2000,700,66,0,$2000
83:
     dc.1 700,66,0,$2000,700,66,0,$2000
     dc.1 1000,100
84:
85:
    MEND:
86:
87:
     Frequenz1:
                    dc.1 0
                                    ; Labels fuer Frequenz 1
88:
                                   ; und Frequenz 2
                                   ; Hier werden die Werte
89:
    Frequenz2:
                    dc.1 0
90:
                                    ; gespeichert.
                              "graphics.library",0
91 .
                 dc.b
     GraphLib:
                                    ; Name fuer Graphics-Libr.
92:
                                    ; (minus 1, wegen dbra)
93:
    Msize = [[Mend-Mstart]/8]-1
```

```
1:
    ; LISTING 4
2:
3.
4:
               BOOTROUTINE
5:
    ;
 6:
7:
             $4000c
                                 ; Spezifischer Seka OpCODE, fuer
    ora
             $4000c
                                   ; absolutes Laden nach Adressen
8:
    load
9:
    move.w
             #$0002,28(a1)
10:
    clr.1
             44 (a1)
             #$40000,40(a1)
                                  ; Abs. Adresse mit 40-Distanz
11: move.1
                                 ; nach Al
12:
13:
    move.1
             #10240,36(a1)
                                  ; Execbase (4) nach A6
14: move.1
             4,a6
                                  ; DOIO Aufruf
15:
    jsr
             -456 (a6)
                                  ; nach $40036 springen
16:
    jmp
             $40036.L
17:
                                  ; hier liegt Programm.
18:
    PRGanfang:
19:
    ;******* Hier beginnt Bootprg.
20:
    START:
21: movem.1 d0-d7/a0-a6,-(sp)
                                  ; Register auf Stack
                                  ; Als Bootprogrammm dient
22:
    nichts:
23:
    btst #$6,$bfe001
                                  ; einfache Mausabrage
                                  ; (auf linke M-Taste)
24:
    bne.s nichts
    ;**********
25:
    ENDE:
26:
27:
    lea
             Graphlib+4 (pc), a1
28: move.1
             #" dos", (a1)
    movem.1 (sp)+,d0-d7/a0-a6
29:
                                  ; Register von Stack
30:
    lea
             Graphlib+5 (pc), a1
                                  ; FindResident
             -96 (a6)
31:
    isr
32:
    move.1
             d0, a0
                                  ; Ergebnis nach A0
             22(a0),a0
    move.1
33:
```

```
34: clr.l d0
35: rts ; Ausstieg aus Programm weiterbooten
36:
37: GraphLib: dc.b "graphics.library",0
38: ;Name fuerGraphics-Libr.
```

Tabellel: Allgemeine Input/Output-Struktur für Devicehandhabung Addrdistanz: Orig. Name: Hinweis: 0 Succ Pointer auf darauffolgenden Eintrag in Str. 4 Pred Zeiger auf vorhergehenden Eintrag in Str. 8 Type Typ des Eintrags 9 Pri Priorität 10 Pointer auf Namen Name Zeigt auf ReplyPort 14 Readreply 18 MnLenght Node-Länge 20 IO Device Zeigt auf Device-Node IO Unit 24 Drivenummer (0.1.2.3) 28 IO Command Gibt Device Kommando an IO Flags Flags 30 31 IO Error Hier sind Fehler auszulesen 32 IO Actual Anzahl der übertragenen Bytes 36 IO Length So viele Bytes mu der Amiga übertragen 40 IO Data Pointer zum Datenpuffer IO Offset Bei TRDdevice Kopfposition 44

Fehler können auch bei jeweiliger Funktion im Datenregister DO getestet werden. ErrorNr: FehlerName: Hinweis: 20 NotSpecified Fehler nicht deutbar 21 Kein Sector Header auf Disk 22 BadSecPreamble Fehler in Sector Vorspann 23 Fehler in Sector ID BadSecID 24 BadHdrSum Falsche Header-Checksumme 25 BadSecSum Falsche Sektor-Checksumme 26 TooFewsSecs Ungültige Zahl von Sectoren auf Zylinder 27 BadSecHdr Kann den Sector-Header nicht lesen 28 WriteProt Diskette ist schreibgeschützt (Fehler!) DiskChanged 29 Diskette wurde gewechselt (Fehler!) SeekError 30 Während Zylindersuche konnte Cyl nicht gef.werden 31 Nicht genug Speicher für Operation NoMem 32 BadUnitNumber Falsche Sektornummer 33 BadDriveType Kein Amiga 3.5 Zoll Laufwerk 34 DriveInUse Laufwerk schon bei Arbeit (nur ein Task möglich) 35 Befindet sich in Reset-Phase PostReset

Tabelle 2: Mögliche Fehler im IO_Error Feld

CNummer:	Name:	Hinweis
2	CMD Read	Lesen von Diskette
3	CMD Write	Schreiben auf Diskette
4	CMD_Update	Daten, die noch im Speicher sind auf Diskette schreiben
5	CMD_Clear	Zylinderpuffer löschen
9	TD Motor	Motor ein/ausschalten
10	TD Seek	Schreib/Lesekopf auf bestimmte Position
11	TD Format	formatieren mit Daten im Speicher
12	TD-Remove	Interrupt installieren, der bei einem Diskettenwechsel ausgegeben wird
13	TD-Changenum	Ermittelt die Anzahl der Diskettenwechsel
14	TD Changestate	Ist Diskette eingelegt ?
15	TD Protstatus	Ist Diskette schreibgeschützt ?
16	TD RawRead	Lesen von unbearbeiteten Disketteninhalt

Maustaste gemacht wird, ändern sich natürlich die Daten für die Noten, wenn Sie diese Abfrage entfernen. Noch ein TIP: Wählen Sie bei diesem Programm Werte für Frequenz 2 (Notenhöhe) zwischen 1-200, zu große Werte hören sich nicht gerade wohltuend an. UND: nicht gleich M-Taste drücken, so schnell geht ein Amiga-Laufwerk auch nicht kaputt (meines ist der beste Beweis!).

Nachdem Sie nun auch noch Einblick in die Umleitung der PAULA-Leitung auf das Disklaufwerk bekommen haben, sollten Sie sich dazu hinreißen lassen, den Artikel voll zu Ende zu lesen, denn ich verrate Ihnen noch kurz, wie Sie Programme mit bis zu 11264 Bytes auf den wohlbekannten Bootblock schreiben können. Mit folgenden Nachteilen:

Sie benötigen einen C-Mon, das Programm muß in Assembler geschrieben sein. Da ich allerdings glaube, daß der C-Mon ein weitverbreitetes Hilfsmittel ist, sollte dem ungetrübten Spaß kein Hindernis mehr im Wege stehen. Um Verwirrung zu vermeiden, gehe ich jetzt nicht auch noch auf das Kapitel des Bootblocks ein, da für den Normal-Anwender eine Anleitung genügen sollte:

- Tippen Sie Listing 4 ab.
- Fügen Sie Ihr Programm (Assembl. prg.) nach 'Prgstart' ein.
- Speichern Sie das Ganze ab.
- Laden Sie Seka (oder anderen Assembl.) mit 'run SEKA'. (run !!).
- Laden Sie das zuvor abgespeicherte Programm ein.
- Laden Sie C-Mon.
- Legen Sie Disk ein, auf die das Bootprogramm soll.
- Geben Sie im C-Mon '<t 40000 0 0' ein.
- liest Cyl. 0 nach \$40000
- Assemblieren Sie Ihr Programm im Assembler.
- Geben Sie im C-Mon 'b 40000' ein erstellt nun neu Bootchecksumme.
- Geben Sie im C-Mon '>t 40000 00' ein
- schreibt Daten ab Adresse \$40000 nach Zyl. 0 auf Disk.
- FERTIG!!

TIP: Notfalls sollten Sie das FA-STRAM abschalten, da absolut geladen wird.

Jetzt sollte sich Ihr Bootprogramm auf dem Bootblock befinden, und nach einem Reset es ablaufen (Disk muß zu diesem Zweck natürlich eingelegt sein!). Denkbar wäre es z.B., Ihr komponiertes Liedchen auf den Bootblock zu schreiben. So hören Sie z.B. sofort nach dem Einlegen die deutsche Hymne. Sie können übrigens Ihre so erstellten Bootprogramme recht einfach mit 'BOOTCOPY V1.0' (Listing 1) kopieren. Ihrer Kreativität und Phantasie steht nichts mehr im Wege.

Ich hoffe, Sie haben alles verstanden. Und nun viel Spaß beim Komponieren von Beethovens Neunte...

		(ohne Dekodierung)
17	TD_RawWrite	Schreiben von unbearbeiteten Daten (ohne Kodierung)
18	TD_GetDriveTyp	Laufwerkstyp (1 = 3 $1/2$ oder 2 = 5 $1/4$)
19	TD GetNumTracks	Gesamtzahl der Tracks
20	TD_AddChangeInt	Interruptroutine bei Diskettenwechsel installieren
21	TD remChangeInt	Interruptroutine entfernen
22	TD_LastComm	Ermittelt letztes ausgeführtes Kommando

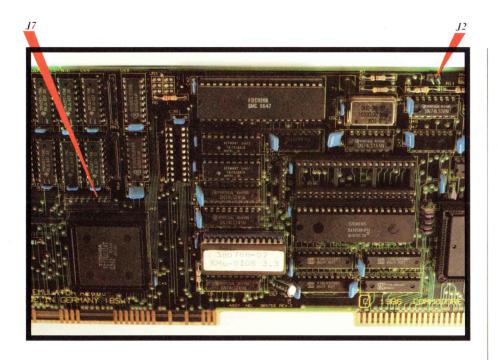
CIA-Port:	Adresse:	Bit:	Name:	Hinweis:
A	\$bfe001	5	DSKRDY	Drive ist bereit
A	\$bfe001	4	DSKTRACKO	Diskkopf über Zylinder 0
A	\$bfe001	3	DSKPROT	Disk ist Schreibgeschützt
A	\$bfe001	2	DSKCHANGE	Disk wurde aus Drive genommen
В	\$bfe001	7	Motor ein,	/aus
В	\$bfd100	6	DSKSEL3	Drive 3 ausgewählt
В	\$bfd100	5	DSKSEL2	Drive 2 ausgewählt
В	\$bfd100	4	DSKSEL1	Drive 1 ausgewählt
В	\$bfd100	3	DSKSEL0	Drive 0 ausgewählt
В	\$bfd100	2	DSIDE	Kopfseite 0 = oberer Kopf
В	\$bfd100	1	DSKDIREC	0 nach innen, 1 nach au en
В	\$bfd100	0	DSKSTEP	Steppsignal
В	\$bfd100	4	DSKINDEX	Wird benutzt um Level

Register:	Name:	Hinweis:
\$dff020 \$dff022		Daten von/zur Diskette obere Bits 16-18 Daten von/zur Diskette untere Bits 0-15
\$dff024	DSKLEN	Bit 14 :WRITE = Daten in DSKPTL/H auf
		Disk schreiben Bit 0-13 :LENGHT= Länge der zu lesenden Daten
\$dff01a	DSKBYTR	Bit 15 :BYTEREADY= Signalisiert, da_ das Datenbyte der unteren 8 Bits gültig ist.
		Bit 14 :DMAON = Zeigt ob DiskDMA eingeschaltet ist
		Bit 13 :DSKWRITE = Zustand von Write in DSKLEN
		Bit 12 :WORDEQUAL= Diskdaten gleich DSKSYNC Bit 11-8 :nicht benutzt
\$dff07e	DCKCANC	Bit 7-0 :DATA = Daten von Diskette Hier ist Datenwort, an dem Datenübertragung
Jairo/e	DSRSTNC	beginnt.
\$dff09e	ADKCON	schreiben
\$dff010	ADKCONR	
		Bit 15 :SET/CLR =1 löschen, 0 setzen
		Bit 14-13:PRECOMP = Enthält den 'PRECOMP' - Wert
		Bit 12 :MFMPREC =1 GCR, 0 MFM Kodierungstyp Bit 11 :UARTBRK =Gehört nicht zum Diskkont.
		Bit 10 :WORDSYN =1 schaltet Disksync ein
		Bit 9 :MSBSYNC =1 scalltet bisksync ein
		Bit 8 :FAST =Taktrate des Diskkontrol.
		1 für 2 ms/bit MFM-Kodier.
		0 für 4 ms/Bit GCR-Kodier.
		Bit 7-0 :AUDIO =Bits für Audio-Hardware
\$dff026	DSKDAT	Internes Datenregister, für Diskdaten (schreiben)
\$dff008	DSKDATR	Internes Datenregister, für Diskdaten

VON CHRISTIAN KELLER

KLEINE BASTELEIEN AM BRIDGEBOARD

Haste Töne - der PC bringt einen Laut hervor



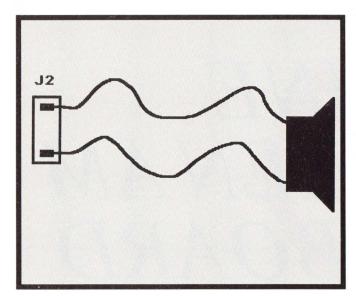
Besitzer eines Bridgeboards haben sich vielleicht schon darüber geärgert, daß sie nicht in den Genuß der extraordinären Soundmöglichkeiten des PC-Teils kamen. Diesbezüglich soll jetzt Abhilfe geschaffen werden, außerdem gibt es nun den langersehnten Resettaster.

Voraussetzung für die im folgenden beschriebenen Aktionen ist ein geöffneter A2000. Man beachte hierbei die Garantiebedingungen. uerst will ich mich dem vergessenen Lautsprecher widmen. Es wurde tatsächlich nur der Quäker weggelassen, der komplette Treiber sitzt schon auf der Platine. Man muß nur einen kleinen Lautsprecher (32 Ohm) an J2 (Stecker am oberen rechten Rand des Bridgeboards) anschließen. Mit einem Potentiometer in der Anschlußleitung kann man sogar die Lautstärke regeln.

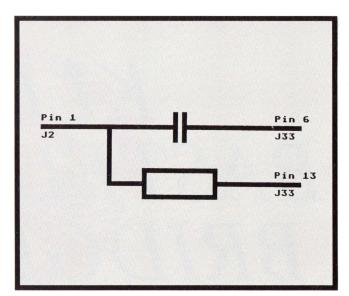
Eine andere, etwas elegantere Lösung ist die Nutzung des Soundkanals des AMIGA. Dazu verbindet man Pin 1 des J2 über einen 0,1 μF/50 V-Kondensator mit Pin 6 des J33 auf dem AMIGA-Motherboard. Ebenfalls verbunden wird Pin 13 von J33 mit Pin 1 von J2, diesmal über einen 470 Ohm-Widerstand (0,25 Watt). J33 sitzt dabei auf der Hauptplatine genau hinter der Buchse des seriellen Ports. Durch diese Modifikation wird der PC-Ton auf den rechten Kanal des AMIGA-Audioteils gemischt und über einen dort angeschlossenen Lautsprecher ausgegeben.

Ob er will oder nicht -Unterbrecher am PC

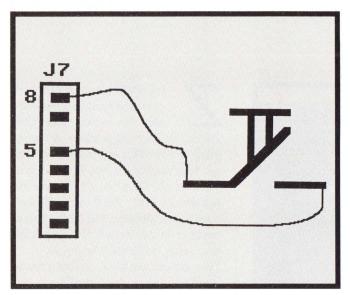
Ein weiterer Punkt, der oft vermißt wird, ist der hardwaremäßige Reset-Taster. Beim Sidecar ganz standardmäßig einzubauen, bedarf es beim Bridge-



Anschluß eines eigenen Lautsprechers.



Tonausgabe über den AMIGA.



Anschluß eines Reset-Tasters..

board doch einer genauen Hardwarekenntnis. Grund dafür ist der fehlende Taktbaustein 8284, der zwecks Miniaturisierung in einen Custom-Chip integriert wurde. Doch auch dieser Custom-Chip bietet dieselben Anschlüsse an, die wir am 8284 benötigt haben. Dieser Pin ist schon auf einen Stecker geführt, so daß der Anschluß eines Resettasters ganz ohne Löten auf der Platine vonstatten geht (wichtig für die Garantie). An Steckverbinder J7 schließt man an Pin 5 und Pin 8 den Taster (Schließer) an und sucht sich ein nettes Plätzchen für den Taster. Weiter ist nichts zu tun. Ab jetzt kann man auch das Bridgeboard resetten, ohne den Amiga neu hochfahren zu müssen.

ENDE

VON ROLAND FOERSTER

TURBO PASCAL

Neues aus Borland

Borlands Turbo Pascal 3.0 ist einer der weitverbreitetesten Pascal-Dialekte im MS-DOS-bzw. PC-DOS-Bereich. Will man jedoch große Programme oder gar ganze Programmpakete in Turbo Pascal 3.0 erstellen, merkt man schnell, daß es auch einige große Nachteile besitzt. Nun erscheint Turbo Pascal 4.0. Behebt es diese Nachteile?

ufgrund der Mängel der 3.0-Version entwickelten Softwarehäuser Erweiterungen, sogenannte Toolboxen, die diese Fehler beheben sollten. Ein Beispiel dafür ist die Extender Toolbox, die es erlaubt, Arrays größer als 64 kB zu dimensionieren, und die auch den Quellcode nicht mehr dieser magischen MS-DOS Grenze von 64 kB unterwirft. Als anderes Beispiel wäre hier noch die Graphic Box anzuführen, in der eine Vielzahl von Grafik-Unterroutinen (Procedures und Functions) bereitgestellt werden. Doch nun ist es soweit, nach langer Vorankündigung ist die neue Version von Turbo Pascal, Turbo Pascal 4.0, auf dem Markt erhältlich. Man muß tatsächlich von einem neuen Produkt reden, da es bezüglich der Entwicklungsumgebung (Benutzerober-

Produkt reden, da es bezüglich der Entwicklungsumgebung (Benutzeroberfläche) wenig von seinem Vorgänger übernommen hat. Der Lieferumfang von Turbo Pascal 4.0 ist weitaus mächtiger als der seines Vorgängers. Man erhält drei Disketten und zwei je 400 Seiten starke Handbücher.

Das erste Buch enthält einige nützliche Informationen und gibt eine ausführliche erste Einführung. Im zweiten Band wird es dann spezieller, d.h., es werden die einzelnen Sprachelemente dieses Pascaldialekts, teilweise anhand kurzer Beispiele, erklärt. Zusätzlich findet man auch eine genaue Beschreibung der 186 schon in Units (siehe weiter unten) enthaltenen Funktionen und Prozeduren. Auf den Disketten befinden sich zahlreiche Beispiele, und die Datei TURBO.TPL, die die mitgelieferten Units sowie zwei Versionen des neuen Compilers enthält, eine commandline-Version und eine integrierte Entwicklungsumgebung (Editor, Compiler und Linker in einem). Für die commandline-Version muß der Rechner mit mindestens 256 kB RAM bestückt sein. Um mit der Entwicklungsumgebung arbeiten zu können, benötigt man als Minimum 384 kB RAM.

UNITS

Eine Unit ist eine Sammlung von Variablen, Konstanten, Typen, Proceduren und Funktionen. Mit der Einführung von Units in Pascal vollzieht sich eine starke Annäherung Pascals an

Modula-2, eine Weiterentwicklung von Dr. N. Wirth, die das modulare Konzept von Pascal noch erweitert. Wer schon mal in Modula-2 programmiert hat, findet sich sofort mit dieser Spracherweiterung zurecht.

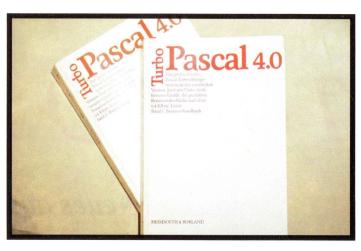
Eine Unit besteht aus einem Interface und einem Implementation-Teil. In Modula-2 wären die Entsprechungen das definition-modul und das implementation-modul. Alle Informationen im Interface-Teil sind unitübergreifend bekannt (public). Die im Implementation-Teil enthaltenen Informationen sind lokal (privat). Eine Unit ist nicht mit einer Includedatei zu verwechseln. Bei Includedateien wird nur der Source unter einem anderen Namen gespeichert. Bei der Compilierung wird dann der gesamte in der Includedatei enthaltene Programmtext übersetzt.

Eine Unit ist mehr eine Art object-Datei, die praktisch in das Hauptprogamm gelinkt wird. Man erkennt Sie im Directory an der Extension '*.TPU'. Mit Hilfe des TPUMOVER's kann man selbsterstellte Units in die im Lieferumfang enthaltene Unitlibrary TURBO.TPL einbinden. Ein Beispiel dafür finden sie in Listing 1. Sie enhält eine Funktion, um die Fakultät zu berechnen. In Listing 2 wird dann die Funktion fak aus dieser Unit benutzt, um ein Pascal'sches Dreieck zu erstellen

Mitgeliefert werden die Units crt, printer, dos, graph, turbo3 und graph3. 'crt' enhält die IBM-spezifischen I/O-Funktionen. Sie kann durch eine Andere ersetzt werden. Um einen Drucker ansteuern zu können, muß man die Unit printer benutzen. Sie stellt die Variable '1st' zur Verfügung und weist sie dem Druckertreiber zu, so daß man die wri-

te und writeln-Prozeduren benutzen kann. In der Unit dos sind Prozeduren wie gettime und getdate enthalten. Es existiert auch eine intr-Funktion, mit der man von Pascal aus Interrupts auslösen kann. Desweiteren gibt es noch die Möglichkeit, MS-DOS-Systemaufrufe über die procedure msdos auszuführen. Die graph-Unit besteht aus über 50 Funktionen und Prozeduren, die die Graphic Adapter CGA, MCGA, EGA, VGA, Hercules, AT&T und 3270 unterstützen. In der graph-Unit sind die Funktionen GraphDriver und Detect enthalten, mit denen man vom Programm aus feststellen kann, welcher Graphic Adapter momentan zur Verfügung steht, so daß der Quellcode universell gehalten werden kann. Desweiteren hat man die Möglichkeit, mit Windows zu arbeiten, verschiedene Schriftsätze auf dem Bildschirm zu benutzen, Kuchengrafiken zu erstellen und noch viele andere grafische Spielereien durchzuführen.

Die turbo3-Unit enthält einige Prozeduren, die Turbo Pascal 4.0 mit der Vorgängerversion kompatibel halten sollen. Aufzuführen wären hier die Funktionen Normvideo, Lowvideo, HighVideo, Longfilesize, Longseek, MemAvail, MaxAvail, IOResult und Kbd. Teilweise wird dadurch der Sprachumfang zu einer Multimenge aufgebläht, d.h. es gibt mehrere Funktionen, die die gleichen Ergebnisse liefern. Will man beispielsweise einen Buchstaben von der Tastatur ohne Echo einlesen, hat man die Möglichkeiten, dies entweder mit Hilfe der turbo3 zu programmieren (Read(Kbd,Buchstabe)), oder durch die Funktion ReadKey (Buchstabe: =ReadKey) aus der Unit crt. Ein großer Fehler ist es jedoch, daß die Funktion Buflen nicht mit übernommen wurde. Sie gestattete es, den Tastatureingabepuffer zu begrenzen, was bei der Maskenerstellung sehr von Vorteil war. Durch die Unit graph3 kann man alle in Turbo Pascal 3.0 zur Verfügung stehenden Grafik-Routinen nutzen. Zum Schluß kommt noch ein großer Vorteil des Unit-Konzepts zum Vorschein: Jede Unit kann bis zu 64 kB groß sein. Da ein Programm beliebig viele Units enthalten kann, ist dadurch die effektive Programmgröße nur durch den Arbeitsspeicher begrenzt. Dies ist eine Das neue Erscheinungsbild von Turbo Pascal



sehr nützliche Erweiterung. Ferner können nun Fremdhersteller Units erstellen und verkaufen, ohne den Sourcecode preiszugeben, was bisher nicht möglich war.

Sie können außerdem getrennt vom Hauptprogramm compiliert werden. Dies ist dann praktisch, wenn man Projektverwaltungsroutinen zur Verfügung gestellt bekommt. Turbo Pascal 4.0 bietet dafür die folgenden Routinen:

PROJEKT-MANAGEMENT

Da ein Projekt unter Umständen aus einer Vielzahl von einzelnen Units besteht, muß man darauf achten, daß auch immer die aktuellen Versionen der Units in das Hauptprogramm gelinkt werden. Eine Möglichkeit, dies sicherzustellen, wäre, alle in dem Programm vorkommenden Units neu zu compilieren. Dazu müßte man allerdings wissen, welche Units im Programm vorkommen, und da Units unter Umständen wieder andere Units benötigen, kann das Durchforsten eines Programmes sehr zeitaufwendig werden. Turbo Pascal stellt eine Hilfe zur Verfügung, die sich grep nennt. Grep ist ein Werkzeug, mit dem man eine oder mehrere Dateien nach einem Textstring durchsuchen kann. Es gibt aber noch ein besseres Mittel, bei dem man nicht alle Units per Hand compilieren muß. Es nennt sich build. Mit build werden alle benötigten Dateien neu compiliert. Wenn man weiß, daß sich einige Units seit dem letzten Ediervorgang nicht verändert haben, sollte man das make-Utility benutzen; C-Programmierer werden sie schon

kennen. Make compiliert die benutzten Files in Abhängigkeit des Datums und der Zeit des letzten Ediervorgangs. Stimmen diese nicht mit dem Object-Code-Datum und der ObjectCode-Zeit überein, so wird der File mitcompiliert, andernfalls wird er nur gelinkt. Schließlich gibt es noch die Möglichkeit des bedingten Übersetzens, d.h. man kann über Compilerdirektiven den zu compilierenden Source verändern.

Was mache ich mit meinen alten Programmen?

Keine Angst, man muß sie nicht wegwerfen. Im Lieferumfang befindet sich ein Programm namens UPGRADE. Mit dieser Utility ist es möglich, Pascal-Programme der Version 3.0 in Programme der Version 4.0 zu Übertragen. Upgrade ist eine Art Konverter: Altes Programm 'rein - neues Programm 'raus. Doch halt, ganz so einfach geht das nicht! Der Quellcode wird zwar in eine Form gebracht, die einem 4.0er Programm ähnlich ist, sofort lauffähig ist das Programm jedoch meistens nicht. Ein bißchen Handarbeit ist noch notwendig. Um diese Handarbeit zu minimieren, gibt es eine Reihe von Parametern, die man richtig einstellen muß. Hier wäre zu erwähnen: / 3, /J, /U. Der Parameter /3 bewirkt, daß die Unit turbo3 benutzt wird, wenn eine der folgenden Anweisungen im Quellcode auftaucht: kbd, cbreak, memavail, maxavail, longfilesize, longfilepos, longseek. Der Schalter /J erzeugt eine Journaldatei, die zusätzliche Hinweise für die Programmumwandlung gibt. Mit /U kann man den alten Quelltext in Units unterteilen,

was sehr empfehlenswert ist, wenn es sich um große Programme handelt. Wenn man diesen Parameter benutzen will, muß man jedoch sein altes Programm etwas modifizieren. Doch nun noch ein Nachteil der neuen Turbo-Version: Turbo 4.0 unterstützt keine Overlays. Wer ein Programm in Overlays aufgeteilt hat, weil der Programmtext größer als 64 kB war, hat keine Probleme mit der Konvertierung, da er nur aus den Overlays Units machen muß. Wer jedoch das komfortable Overlaykonzept aus Gründen der Arbeitsspeicherknappheit des Rechners gewählt hat, hat keine Chance, daß sein Programm unter Turbo Pascal 4.0 ohne gründliche Neuüberlegungen bezüglich der Speichervervaltung lauffähig sein wird.

Die integrierte Entwicklungsumgebung

Nach dem Aufruf von Turbo aus der Kommando-Ebene präsentiert sich eine Maske, die sehr viel Ähnlichkeit mit der von Turbo C hat. Der Editor ist nach wie vor fast identisch mit dem von WordStar. Es existiert eine Pickliste, mit der man sehr schnell die letzten 8 bearbeiteten Dateien in den Editor bekommt. Die ganze Umgebung ist mit Hilfe von pop up/ - pull down-Fenstern gestaltet. Die Unterpunkte in den einzelnen Menüs sind entweder über die Cursor-Tasten zu erreichen, oder durch Eingabe des ersten Buchstabens des gewünschten Punktes aufzurufen. Der Editor hat gegenüber seinem Vorgänger einen schönen Zusatz bekommen: Er merkt sich die Position im Text, bei der man im letzten Ediervorgang stehengeblieben war. Das erspart viel Sucharbeit bei der Erstellung großer Programme und erleichtert das Debuggen.

ARITHMETIK

Turbo Pascal 4.0 unterstützt folgende Datentypen für ganze Zahlen: den longint, einen 32 Bit großen Integer mit Vorzeichen, den Typ shortint, ein Byte mit Vorzeichen, und den normalen Integer, eine 2 Byte lange Zahl mit Vorzeichen. Es gibt auch noch den Typ Byte, das ist eine 1 Byte große Zahl ohne Vorzeichen, und den Typ word (unsigned, 2 Byte). Als Real-Typen stehen solche zur Verfügung, die 4, 6, 8 oder 10 Bytes benötigen. Die 6-Byte-Variablen sind die normalen Reals mit 11-stelliger Genauigkeit und einem Bereich von E-38 bis E+38. Für die anderen Typen (single (4 Bytes), double (8 Bytes), extended und comp (10 Bytes)) benötigt man einen Arithmetikcoprozessor 80x87. Durch die Bereitstellung dieser Typen sind die Versionen Turbo BCD und Turbo-87 überflüssig geworden. Für BCD-Arithmetik steht der Typ comp zur Verfügung. Er liefert Werte im Bereich von E-18 bis E+18, d.h. volle 18 Stellen. Das dürfte für die meisten finanzmathematischen Anwendungen im DM-Bereich ausreichend sein. Bemerkenswert ist auch noch, daß der normale Real-Typ mit dem Coprozessor verträglich ist, d.h. man kann mit 6-Byte-Reals und 80x87 arbeiten.

Hersteller: Borland Vertrieb: Heimsoeth

```
1: LISTING 1
 2:
 3:
                     { unit name, unter diesem Namen sind alle im Inter- }
 4: unit mathe;
                    { face Teil aufgeführten Funktionen zugänglich
 5:
 6:
       { An dieser Stelle könnten auch andere units über uses benutzt werden }
 7:
 8 .
 9: interface
10:
        Function fak (n: Integer): Real;
                                          { nur hier müssen die Variablen }
                                          { deklariert werden
11:
12:
13: implementation
14:
        function fak:
15.
          var i: Integer;
17:
              s:real;
18:
          begin
            if n=0 then fak:=1
19:
20:
             else
21:
              begin
                s:=1:
22:
                for i:=2 to n do s:=s*i;
23:
24:
                fak:=s;
25:
               end;
26:
           end;
27:
28: begin
      { Hier können Befehle stehen, die die Functionen initialisieren. }
29:
30:
       { Dieser Teil wird nur ein einziges Mal ausgeführt.
31:
32:
33.
    { Dieser Source wird automatisch durch das reservierte Wort unit }
34: { als *.TPU Datei compiliert }
```

Listing 1: Einige neue Befehle praktisch angewandt.

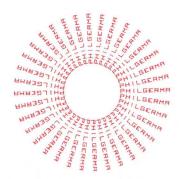
```
Das Programm erzeugt dann folgenden Output. Die Funktion 'fac' taucht nicht
im Sourcecode auf. Die unit 'mathe' wurde getrennt compiliert.
1
1
     1
1
     2
            1
        3
               1
1
     3
1
     4 6
                         1
                          5 1
1
     5 10 10
                         15
1
           15
                  20
                                 6
        21
                                      7
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1
1 9 36 84 126 126 84 36 9
1 10 45 120 210 252 210 120 45
                                                          1
                                                              1
                                                         10
```

Abbildung 1

```
1: LISTING 2
2:
3:
4: Program Binomi;
5:
6: { zu Ehren von B.Pascal erstellt dieses Programm ein "Pascal'sches Dreieck }
8: uses mathe; { Stellt die function fak zur verfügung }
9:
10: var
11: n,k:Integer;
12: bnk:Real;
13.
14: begin
15: for n:=0 to 10 do
16:
       begin
        for k:=0 to n do
17:
18:
          begin
          bnk := fak(n)/fak(n-k)/fak(k);
19:
20:
            Write(bnk:6:0,' ');
21:
           end;
22:
        Writeln;
23:
        end;
24: end.
```

Listing 2: Das Programm erzeugt ein Pascal'sches Dreieck

ENDE



MCC PASCAL (Metacomco) — Pascal Compiler ISO 7185 Standard. Single Pass Compiler, schnell und effizient. Die AmigaDos Routinen können voll im Pascal eingebunden wer-den. MCC Pascal Prog. können mit MCC Assembler oder Lat-tice C gelinkt werden. Handbuch 200 Seiten . . . DM 198,00

MCC ASSEMBLER (Metacomco) — Professioneller Makro Assembler, der den vollen Motorola 68000 Instruction Set unterstützt.

Mit Editor, Linker und AmigaDOS-Routinen DM 168,00

LATTICE C (Lattice) — Bewährter C-Compiler der USA-Firma Lattice, Standardprodukt in der IBM-Welt. Kompatibel auf vielen Rechnern, gut für professionelle Entwicklungen. Kerningham/Ritchie Standard. Fließkommaarithmetik mit 16 Stellen Genauigkeit. Die neue Version 4.0 enthält Assembler Umfang .

TRUE BASIC (True Basic) — Modernes strukturiertes Basic mit Grafik. Hdb. 500 S. DM 398,00

PHILGERMA IHR SPEZIALISR FÜR MIGA COMPUTERSPRACHEN

AZTEC C68k/am-p Professional (Manx) — Neueste Version 3.6 dieses bekannten C-Compilers, Er umfaßt optimierenden 35 dieses bekannten C-Compilers. Er umfatt optimierenden C-Compiler, Assembler, Linker, Bibliotheken und Beispiele. Unterstützung des 68020 und des 68881 Prozessors. Hervorragendes engl. Handbuch 400 S. DM 398,00

AZTEC C68k/am-d Developer (Manx) - Zusätzlich Make.

LIBRARY SOURCE (Manx) Quellcodes für C-Compiler
DM 598,00

AC/FORTRAN77 (absoft) — ANSI X3.9-78 Standard Fortran77 Compiler für 68000 Prozessoren. Zusätzliche Optionen. Fließkommarithmetik 16 Stellen nach IEEE Standard. Overlays, virtuelle Arrays und Debugger. Handbuch 300 S. - ANSI X3.9-78 Standard DM 598,00

AC/FORTRAN77-68020/68881 (absoft) DM 1198,00
 Vizawrite Textverarbeitung
 DM 198,00

 Page Setter Desktopprogramm
 DM 378,00

 Publisher 1000 Desktopprogramm
 DM 498,00

 Synthia Synthesizerprogramm
 DM 228,00

 Dynamic Drums
 DM 168,00

 Sonix 1.4 Musikprogramm
 DM 188,00

 DM 118,00
 DM 118,00
 Aegis Audiomaster DM 118.00 Aegis Audiomaster
Instant Music Kompositionsprogramm
Deluxe Music Construction Set
Deluxe Music + Instant Music DM 69,00 DM 248,00 DM 258,00

NATÜRLICH HABEN WIR AUCH SPIELE

NATURLICH HABEN WIR AUCH SPIELE
Galactic Invasion DM 49,00; The Guild of Thieves DM 69,00;
Test Drive DM 79,00; Racter DM 99,00; Ogre DM 59,00; Grand
Slam Tennis DM 89,00; Deja Vu DM 89,00; Go DM 69,00; World
Games DM 69,00; Winter Games DM 69,00; Super Huey DM
49,00; Xenon DM 69,00; Jinxter DM 69,00; Uninvited DM
79,00; Shadowgate DM 89,00; Defender of the Crown DM
79,00; Sinbad DM 89,00; Pawn DM 69,00; Terrorpods DM
69,00; One-on-One DM 39,00; Archon II DM 69,00; Chessmaster 2000 DM 89,00; Bureaucrazy Infocom Textadventure DM 89,00; Arkanoid DM 69,00; Portal SF DM 89,00; Bard's Tale DM 99,00; Flight II DM 119,00; Scenery Disks Nr. 7 u. 11 je DM 59,00; Balance of Power DM 89,00; Jet DM 119,00; Raster Bike Action DM 29,00 - und viele andere!

PHILGERMA PROLOG ab Anfang Mai lieferbar! Fordern Sie

MODULA II Standard (TDI) V3.0 -- Diese umfangreiche

MODULA II Developer (TDI) V3.0 — Zusätzlich symbolischer File Decoder, Cross Referencer, Modula CLI, Utilities für IFF und ILBN DM 268,00

K-SEKA Assembler (Number)
POWER WINDOWS V2.0 Softwaretool zur Feiligen
mierung f. Assembler, C, M2. DM 198,00
TX ED Texteditor PAL DM 98,00
DISK-2-DISK c64-Disk. lesen DM 198,00
DOS-2-DOS IMB-Disk. lesen DM 128,00
THE C64-EMULATOR wahlw. für Amiga 500/2000 oder für Amiga 1000 incl. Interface DM 128,00

THE C64-EMULATOR wahlw. für Amiga 500/2000 oder für Amiga 1000 incl. Interface DM 128,00

THE C64-EMULATOR wahlw. für Amiga 500/2000 oder für Amiga 1000 incl. Interface DM 128,00

THE C64-EMULATOR wahlw. für Amiga 500/2000 oder für Amiga 1000 incl. Interface DM 128,00

THE C64-EMULATOR wahlw. für Amiga 500/2000 oder für Amiga 1000 incl. Interface DM 128,00

Amiga 1000 incl. Interface

DELUXE PAINT II (PAL) deutsch

AEGIS VIDEO TITLER zum Erstellen von bewe DM 268 00 SCULPT 3D Ray-tracing- und Animationsprogr.
SCULPT 3D Ray-tracing-Programm
ANIMATE 3D Animation zu Sculpt
AGGIS ANIMATOR: MAGCIS DM 298.00 ANIMATE 3D Animation 20 Sculp AEGIS ANIMATOR+IMAGES VIDEOSCAPE 3D Animationsprogramm PAL... X-CAD professionelles CAD-Programm INTRO CAD cAD-Programm DM 248.00 DM 378,00 DM 998,00

DM 148,00 AUSZUG AUS UNSERER HARDWARE-LISTE DM 368,00 DM 438,00 DM 398,00 10 Disketten 3.5" 2DD 3M Scotch DM 29,00

Fordern Sie unsere umfangreiche Preisliste an. Händler bitte

Bestellungen bitte an:

PHILGERMA GmbH & Co. KG, Barerstr. 32 8000 München 2, Telefon (089) 281228

Bei Bestellungen unter DM 200 beträgt der Versandkostenanteil DM 4,80. Nachnahme DM 3,20. Lieferung ins Ausland nur gegen Vorkasse (Überweisung oder Euroscheck).

Besuchen Sie unseren Softwareladen in der Barerstr. 32!

Wichtiger Hinweis

Folgende Video- und Computerspiele sind von der Bundesprüfstelle, Bonn, indiziert:

Battlezone Beach Head Beach Head II Blue Max Castle Wolfenstein Commando Commando Libya Part I Desert Fox **Eroticon** Falcon Patrol Falcon Patrol II Flyerfox F 15 Strike Eagle Friday the 13th G.I. Joe I + II Girls they want to have Fun

Green Baret

Paratrooper

Nice Demo

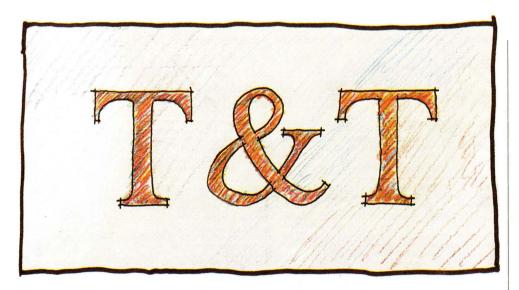
Hitler Diktator

Protector II Raid over Moscow Rambo II River Raid S.D.I Seafox/Seawolf Sex Games Silent Service Skyfox Soldier One Speed Racer Stalag I Swedish Erotica Stroker Tank Attack Teachbusters Theatre Europe 1942 Trainer

Porno Dia Show

Der Verlag behält sich vor, bei Softwareangeboten indizierte Spiele ersatzlos zu streichen

91



TIPS & TRICKS

NICHT KOPFLOS

Um eine Datei auszudrucken, wird man normalerweise den Befehl TYPE benutzen. Wenn man jedoch mehrere Dateien ausdrucken und die Listings aufheben will, dann weiß man oft nach einiger Zeit nicht mehr, wie die Dateien heißen. Wer die Ausdrucke nicht von Hand beschriften will, der kann diese Aufgabe mit folgendem Batch-File lösen:

```
.key datei/s
.bra (
.ket )
if exists (datei)
echo >prt: "Listing der Datei (datei): *n"
type (datei) to prt:
echo >prt: "*n"
else
echo "*nDatei nicht gefunden!*n=======*n"
endif
```

Kurze Erklärung: '.key' bestimmt die Übergabeparameter für die Befehlssequenz. '.ket' ändert das Zeichen '>' in eine Klammer, weil das Größerzeichen für die Umlenkung der Ausgabe auf den Drucker benötigt wird. '.bra' ändert entsprechend das '<' Zeichen. '>prt:' lenkt die Ausgabe des TYPE-Befehls auf den Drucker um. Überall, wo '(datei)' steht, wird der Name des Files eingesetzt. '*n' ist ein Steuerzeichen und bewirkt einen Zeilenvorschub.

Zweckmäßigerweise gibt man diesem Programm den Namen 'drucke' und kopiert es in das C-Verzeichnis der Workbench-Diskette. Nun kann mit 'execute drucke ...' eine beliebige Datei ausgedruckt werden.

Wenn man außer dem Namen der Datei auch das Entstehungsdatum ausdrucken will, dann muß die erste Echo-Anweisung wie folgt geändert werden:

```
list (datei) dates quick to prt:
```

Wer hiermit etwas experimentiert, der wird sicherlich noch einige interessante Varianten finden.

Thomas Stümpfig, Möglingen

ALERTS IM GRIFF

Jeder kennt Sie, die im oberen linken Eck auftauchenden 'System Requester', die eine Warnmeldung des Betriebssystems beinhalten.

Eine Möglichkeit der Beantwortung dieser Anfrage ist das Anklicken eines der Felder mit der Maus. Doch wenn man gerade im CLI tippt oder gar keine Maus zur Hand hat, dann kommt diese Meldung recht ungelegen. Mit der Tastenkombination <linke AMIGA-Taste>+<v> kann jedoch das RETRY-Feld angewählt werden. Entsprechend bewirkt <linke AMIGA-Taste>+ CANCEL.

Wilhelm Bergmann, Hamburg

GE-WOHN-HEITEN AM AR-BEITS-PLATZ

Wer viel mit dem AMIGA arbeitet, hat sich sicherlich schon eine spezielle Workbench erstellt. Viele der Einstellung werden dabei mit den PREFERENCES vorgenommen, so z.B. die Workbenchfarben, die Mausattribute, die Tastatureinstellungen, die Werte der seriellen Schnittstelle, der Name des verwendeten Druckertreibers und

anderes mehr. Je mehr man sich an diese Einstellungen gewöhnt hat, umso ärgerlicher ist es, wenn man eine andere Diskette bootet und plötzlich alle Einstellungen verändert sind. Jetzt kann man entweder auf allen diesen Disketten die Einstellungen ändern, oder man kopiert sich einfach die Datei SYSTEM-CONFIGURATION in den DEVS-Ordner der betreffenden Diskette. Dann braucht man noch nicht einmal mehr die mit knapp 60 KByte recht langen PREFERENCES auf der Diskette zu belassen.

MEHR SPEI-CHER

Besitzer eines Sidecar oder eines Bridgeboards können rund 115 KByte Speicher zusätzlich erhalten, wenn folgende Änderungen in der STARTUPSEQUENCE vorgenommen werden:

Befehl BINDDRIVERS löschen ADDMEM 200000-21DFFF einfügen

Durch diese Maßnahme wird das Dual-Ported-Ram der Sidecar bzw. des Bridgeboards freigegeben, und der Speicher kann vom AMIGA benutzt werden.

Christian Keller, Asbach

ERSTE HILFE BEI INTERLACE

Auf einem normalen Monitor wie z.B. dem Modell 1081/1084 von Commodore ist das Flackern des Interlace-Modus nicht über längere Zeit erträglich. Wenn man sich nun nicht einen nachleuchtenden Monitor leisten kann oder will, dann kann durch Umstellen der Farben in Graustufen ein Verminderung des Flimmerns erreicht werden.

DIMMER FÜR POWERLED

Das folgende kleine Programm ist eher als Witz gedacht. Es dimmt die PowerLED des AMIGA 'stufenlos' von hell nach dunkel. Allerdings benötigt dasProgramm in der hier gezeigten Version zuviel Rechenzeit, um noch ein sinnvolles Arbeiten am 'gedimmten' Rechner zu erlauben.

PortA ist der Zeiger auf den für die LED zuständigen Port des AMIGA. Über die Befehlsfolgen in der Funktion 'dim' wird das Bit 2 dieses Ports abwechselnd ein- und ausgeschaltet. Bei entsprechendem Timing entsteht der Eindruck einer stufenlosen Helligkeitsregulierung.

```
main()
#include "exec/types.h"
#include <functions.h>
                                                 int i, j;
                                                 PortA
                                                                      (UBYTE
UBYTE *PortA;
                                           *) 0xbfe001;
void dim(promil)
int promil;
                                                 for (;;)
                                                      for (i=0; i<1000; i+=10) dim(i);
                                                          for(i=1000;i>0;i-
       *PortA=0x02 | *PortA;
                                          =10) dim(i);
       for (i=0; iiii; i++);
       *PortA=0xfd & *PortA;
                                                  return(0);
       for (i=promil; i<1000; i++);
```

RESET-FESTE RAM-DISK

Eine resetfeste RAM-Disk ist eine sehr große Hilfe im täglichen Gebrauch mit dem AMIGA und im Kampf gegen den allseits gefürchteten GURU. Glücklicherweise existiert ein solches

Programm als Shareware u. a. auf der KICKSTART PD 35. Die Installation ist schnell bewerkstelligt, indem man folgende Arbeiten vornimmt:

Zuerst müssen die Programme 'mountlist' und 'asdg.vdisk.device' in das DEVS-Verzeichnis der Boot-Diskette kopiert werden. Danach wird in der Startup-Sequence die Ram-Disk mit 'mount vd0:' als 'VD0:' angemeldet. Sie kann nun wie jedes andere logische Gerät angesprochen werden. Ihre Eigenschaften sind die der normalen Ram-Disk, jedoch mit dem Vorteil, daß die Daten nach einem Reset bzw. Rechnerabsturz nicht verlorengehen (zumindest in den meisten Fällen!). Eine sinnvolle Startup-Sequence hätte dann z.B. folgendes Aussehen:

```
mount vd0:
if not exists vd0:c
makedir vd0:c
copy df0:c vd0:c
endif
path add vd0:c
```

Hiermit wird der C-Ordner der Boot-Diskette in den neuerstellten Ordner auf der Ram-Disk kopiert. Besonders wichtig ist hierbei, daß kontrolliert wird, ob dieser Ordner schon auf der Ram-Disk existiert, denn dann ist der Kopiervorgang nicht mehr notwendig. Thomas Becker, Höxter

LESERFORUM

Briefe an KICKSTART

Liebe Leserinnen und Leser,

die uns erreichende Flut von Telefonaten ist beim besten Willen nicht mehr zu schaffen. Wir sind daher gezwungen, feste Zeiten anzugeben, zu denen unsere Redakteure für Sie zur Verfügung stehen. Sie können an folgenden Tagen Fragen an Redaktionsmitglieder stellen: Dienstags von 14-17 Uhr

und Donnerstag von 14-17 Uhr

Bitte haben Sie dafür Verständnis, daß wir an anderen Tagen keine Fragen zu Softoder Hardware-Problemen entgegennehmen können.

Ein aufmerksamer Leser bemerkte einen Fehler in der April-Ausgabe der Kickstart. Uns erreichte folgender Brief, den wir Ihnen nicht vorenthalten möchten.

Bezug:

KICKSTART 4/88, Seite 11, "Disketten Errors selbst repariert"

Sehr geehrte Damen und Herren, bezugnehmend auf obengenannten Artikel darf ich Ihnen für diese, wie übrigens auch für alle anderen Artikel in Ihrer Zeitschrift, sehr praxisnahe und detailgetreue Schilderung meinen herzlichen Dank übermitteln. Schon oft haben mir Ihre klaren und leicht verständlich geschriebenen Artikel über manche Hürde im Umgang mit meinem Amiga hinweggeholfen.

Allerdings ist Ihnen in obigen Artikel ein fataler Fehler unterlaufen. Sie haben schlichtweg vergessen, auf zwei eminent wichtige Dinge hinzuweisen:

- 1. Es muß unbedingt sogenanntes Naß-schleifpapier verwendet werden.
- 2. Nach dem Schleifen ist ein Auftrag von Klarlack in mindestens zwei Schichten (kreuzweise) notwendig. Werden diese Punkte nicht beachtet, so ergeben sich folgende Beeinträchtigungen: Da die bei Verwendung von

normalem Schleifpapier entstehenden Staubpartikel nicht restlos entfernbarsind, sind diese später auf dem Bildschirm als kleine Punkte sichtbar.

Ohne Lackauftrag entsteht eine seidenmatte Schicht (Nebel) auf dem Bildschirm. Dies wirkte sich bei mir besonders schwer aus, als ich bei der Bearbeitung einer Kundendatei mit der so reparierten Diskette nach einem bestimmten Begriff suchte. Stets gab das Programm folgenden Requester aus:

Nebeldiskette in Drive 1 weitere Bearbeitung wegen Bodennebel nicht möglich. Weiter mit Suchscheinwerfer? Programm beenden?

Nach Lackierung der Diskette mit 2Komponenten-DD-(diskdrive) Lack war der Fehler behoben. Meine Disketten erzeugten wieder einwandfrei glänzende Bilder am Bildschirm.

Die Verwendung von farbig-lasierendem Lack sollte den Besitzern von monochromen Monitoren empfohlen werden. Auf diese Weise können einfarbige Grafiken leicht in vielfarbige verwandelt werden. Es ist jedoch genau zu ermitteln auf welcher Spur das Programm die entsprechenden Pixel

abgelegt hat, da diese dann im Einzelpixelsprühverfahren bearbeitet werden müssen.

Robert Höfler Nürnberg

Anmerkung der Redaktion:

Es ist unbedingt der oben angesprochene Lack zu verwenden, da andere Marken die Magnetschicht der Diskette anlösen und so die Diskette endgültig zerstören.

Betrifft Netzteil des Amiga 2000

Sehr geehrte Damen und Herren, da ein Amiga 2000 nicht gerade billig ist, stellt sich die Frage, ob es sinnvoll ist, den Computer gegen Überspannungen aus dem Stromnetz zu schützen. Haben Sie hierzu entsprechende Informationen? Außerdem wüßte ich gern, ob es wirklich gefährlich für den Amiga ist, wenn man ihn für weniger als fünf Sekunden ausschaltet. Falls dies der Fall ist, könnte ja zum Beispiel ein kurzer Stromausfall den Computer beschädigen.

Thomas Stümpfig Möglingen

Grundsätzlich sollte der Rechner nicht weniger als fünf bis zehn Sekunden abgeschaltet werden, um dem Netzteil und den Bauelementen auf der Platine die Zeit zu geben, um sich zu entladen, so daß beim wiederholten Einschalten ein definierter Ausgangszustand eingestellt ist (stromlos). Die sehr selten vorkommenden 'kurzen' Netzausfälle sind zwar eine zusätzliche Belastung der Bauelemente, die ihre Lebenszeit verkürzt, dies macht sich jedoch im praktischen Betrieb nicht bemerkbar. Die von Ihnen erwähnte Maßnahme, den Rechner gegen Überspannung und Netzausfall zu sichern, wird bei größeren Anlagen in der Tat praktiziert. Der Preis für ein solches Gerät übersteigt jedoch die Anschaffungskosten für einen Amiga 2000 bei weitem, so daß diese Lösung hier nicht sehr sinnvoll erscheint.

Die Red.

Roboter am Amiga?

Sehr geehrte Damen und Herren, da ich mich für Robotersteuerung am Computer (Amiga 2000) interessiere, aber nach mehreren Anfragen keine Information über dieses Thema erhalten habe, hoffe ich, daß Sie mir weiterhelfen können.

Sollte Ihnen ein Verlag mit Roboterliteratur oder eine Firma bekannt sein, die Roboter, Interfaces oder Bauanleitungen vertreibt, so teilen Sie es mir bitte mit. Kann man eventuell ein Atari-ST-Interface am Amiga 2000 benutzen?

Werner Klingler Straß Die Firma Fischer-Technik bietet eine große Palette an Interfaces und Bausätzen für rechnergesteuerte Greifarme und Produktionssimulationen an. Software ist nach dem heutigen Stand unter anderem für den C-64, den IBM-PC und den Atari-ST verfügbar. Software für den Amiga ist in Arbeit. Da die Centronics-Schnittstelle des Amiga 2000 dem auch beim IBM verwendeten Standard entspricht, sollte es bei ausreichenden Programmierkentnissen kein Problem sein, entsprechende Software selbst zu schreiben.

Wir planen jedoch in nächster Zukunft einen ausführlichen Artikel über dieses Thema.

Die Red.

Monitore

Normalerweise muß man, wenn man mit einem Amiga 2000, PC-Karte und Grafikkarte arbeiten will, einen zweiten Monitor kaufen. Kann man mit einem Multisync-Monitor auskommen, wenn man über ein Y-Kabel und Umschalter den Amiga und eine Karte mit analogen Ausgang (z.B. VGA) anschließt?

Stefan Helbig Rosenheim Sie können unter bestimmten Umständen mit dem angesprochenen Y-Kabel beide Bildausgänge auf dem Monitor darstellen. Dies ist immer dann gegeben, wenn beide Signale die Bildinformation auf dieselbe Art liefern (z.B. beide Signale analog 0.7 V). Da der Eingangswahlschalter des Monitors dann in derselben Stellung gehalten werden kann, ist ein einfaches Y-Kabel mit Umschalter ausreichend. Wollen Sie jedoch verschiedene Bildsignale auf demselben Monitor darstellen (z.B. Amiga-Analog-Signal und CGA-TTL-Signal), ist ein Monitor erforderlich, der getrennt schaltbare Eingänge für das jeweilige Signal besitzt. Ein solcher Monitor ist zum Beispiel der in einer der letzten Ausgaben getestete Mitsubishi EUM-1471A.

Die Red.

A & O

Endlich Software für Profis!?

Man hört ja schon von vielen Seiten, daß dem Amiga noch die "wirklich professionelle Software" fehlt, so z.B. neben guten Textverarbeitungen und DTP-Software auch Programme, die für professionellen Einsatz im Beruf konzipiert sind. Zugegeben, auf diesem Gebiet ist der Amiga nun wirklich noch dünn bestückt. Doch allen, die mit dem Gebiet der Nachrichtentechnik irgendwie zu tun haben, kommt das Ingenieurbüro Dr. Christ in Bremen 'zu Hilfe'. Das Programm 'A&O' (Analyse und Optimierung) für analoge Netzwerke erleichtert die Arbeit an analogen Netzwerken. Damit ist es das, meines Wissens, einzige Programm, das sich im Moment überhaupt mit solchen Simulationen beschäftigt (einmal abgesehen von Logic Works, das nur digitale Schaltungen verarbeitet). Nach dem ersten Einladen und Testen des Programms fiel die erstaunlich kurze Rechenzeit auch für kompliziertere Frequenzanalysen auf (wenn ich da an meine letzte Nachrichtechnik-Klausur denke...), und das, obwohl das Programm 'nur' in C geschrieben ist.

nsgesamt kann man mit 'A&O' bis zu 128 Knoten auf einmal verwalten (hört sich zwar nach recht wenig an, ist aber in der Praxis meist voll ausreichend), wobei die Schaltungen jeweils in einzelne Untergruppen aufgeteilt werden müssen.

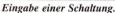
Die Steuerung des Programms wurde ziemlich gut gelöst, nach einer kurzen Einarbeitungszeit läuft alles 'wie geschmiert'; die Art, wie die Eingaben über die Tastatur verwirklicht wurden, könnten für eine neue tastaturgesteuerte Intuition-Oberfläche Modell stehen. Dazu kommen noch diverse Menüs, die zur Steuerung von Grafik, Protokoll etc. verwendet werden. Allerdings passierte es während der Eingabe der Frequenzen einige Male, daß Fehleingaben abgefangen wurden, was jedoch den Gesamteindruck nicht schmälert und laut Autor in Kürze korrigiert wird.

Alle Netzwerk-Daten sowie die Graphen werden in verschie- denen Fenstern zur Verfügung gehalten, so daß man sich jederzeit über die Schaltung informieren kann.

In den Bildern 2 bis 6 sind einige verschiedene Ausdrucke vom Programmablauf abgebildet. Die Eingabe der Schaltung (Bild 1, nicht von A&O!) wird in Bild 2 dargestellt, die Daten der Schaltung in dB mit Phasenwinkel Ol sieht man in Bild 3, Bild 4 und 5 zeigen das Verhalten der Schaltung bei Schwingungen im Bereich von 10 bis 1e6 Hertz sowie von 1e9 bis 1e13 Hertz. Und schließlich zeigt Bild 6 die Bildschirmmaske 'Optimierung'.

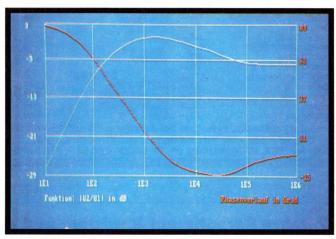
Alles in allem ist A&O ein hervorragendes Programm, das sich durch hohe Geschwindigkeit und gute Benutzer-





führung auszeichnet. Des weiteren fiel positiv auf, daß recht große Netzwerke verwendet und berechnet werden können, ohne sie in einzelne Rechenteile zerlegen zu müssen. Und auch die Optimierung wird in ihrer Qualität wohl vielen Hardwareentwicklern Arbeit ersparen.

Einzig nachteilig fiel eigentlich das recht dünne Handbuch auf, das zwar alle Funktionen anspricht, allerdings in den Erklärungen etwas kürzer gefaßt



Verhalten der Schaltung bei Schwingungen im Bereich 10 bis 1 e6 Hertz.

ist. Dafür enthält es allerdings einen Beispiel-Teil, der von jedem Benutzer sorgfältig durchgearbeitet werden sollte, da er dank seiner Ausführlichkeit das restliche Handbuch fast überflüssig macht. Alle Semi- und Vollprofis auf dem Gebiet der analogen Netzwerke werden dieses Programm zum Preis von DM 265.- wohl bald nicht mehr missen wollen. Und für alle, die vor dem Kauf gern mal 'reinschnuppern' wollen, wird auch eine Demo-Version

zum Preis von DM 20.- angeboten, die alle Funktionen bis auf Speichern und Drucken beinhaltet und mit der man sich einen guten Eindruck von dem Programm machen kann.

Bezugsquelle: Dr. Uwe Christ Klawitterstr. 27 2800 Bremen 61 Tel.: 0421/825487

ENDE

Public Domain Software * **AMIGA MS-DOS**

Über 600 Disketten lieferbar: Fish 1 - 134, Panorama 1 Faug 1 - 53, Amicus 1 - 22, Talfun 1 - 50, ES-Soft 1 - 55 Rei ner Wolf RW-Disks 1 – 30, Chiron Conceptions 1 – 40, ACS 1 – 23, Tornado-Spez. 1 – 30, Kickstart 1 – 57, TBAG 1 – 7 - 4, Casa Mi Amiga, sowie Winners Cycle System, Amuse, Amigazin, Juice Magazin, AAA u.v.a. Disks mehr

Einzeldisk bis 10 Stück DM 4.85 30 Stück 60 Stück DM 4,80 DM 4,70 bis 90 Stück 120 Stück DM 4,60 DM 4,50 150 Stück DM 4.40 auf 3,5"-Disketten 2DD

Achtung neu! Ray-Tracing-Construction-Set V2.0, siehe Amiga 1.88, S. 117. Komplettpaket 3 Programmdisks & 2 Katalogdisks & ausgedruckte deutsche Anleitung für DM-29,95 inkl. Porto.

Achtung neu! Bei Abnahme ab 30 Disketten kostenios für den Anfänger oder Profi ... CLI-Hilfe auf Diskette, lesen, kopieren, editleren, sortieren, drucken, renamen und vieles mehr, ähnlich wie CLIMATE oder ZING ... DirUtil

Nur 145, – für jedes Paket mit 30 PD-Disketten, inkl. Porto. Verpackung und CLI-Hilfe DirUtil, bei Vorkasse (V-Scheck oder Bar). Zum Beispiel: Paket Nr. 1a = Fred Fish Nr. 1 – 30

Paket Nr. 1b = Fred Fish Paket Nr. 1b = Fred Fish
Paket Nr. 1c = Fred Fish
Paket Nr. 1d = Fred Fish
Paket Nr. 3 = Panorama
Paket Nr. 4 = Faug Hot I
Paket Nr. 7 = Kickstart
Paket Nr. 8 = Taifun
Paket Nr. 8 = Fee Seft Faug Hot Mix Kickstart Paket Nr. 9a = ES-Soft Paket Nr. 9b = Paket Nr. 10 = Paket Nr. 11 = ES-Soft Chiron Conc. Nr. 31 Nr. 1 Tornado-Spez. Nr. Oder Sie stellen sich Ihr ganz persönliches Paket aus unserem Amiga PD-Katalog zusammen.

UWE SCHMIELEWSKI

- Ihr Public Domain Archiv für Amiga
Haroldstr. 71 · 4100 Duisburg 1 · Tel. 0203/37 64 48

BTX *0203376448# · Fax 02 03 / 35 96 90

2 Katalog-Disketten mit Information über Inhalt der Programme für Amiga 500/1000/2000 gegen DM 5, – in Briefmarken/bar/V-Scheck anfordern!

Spezial-Katalog über Original PC-StG-Public Domain- & Shareware-Programme für den Amiga mit PC-Karte oder mit MS-DOS-Transformer gegen DM 5, – in Briefmarken/bar/ in Briefmarken/bar/ V-Scheck anfordern

Am gleichen Tag des Bestelleingangs erfolgt der Versand unserer Kataloge

Versandkosten PD-Disketten:

Porto für Inland/Ausland Nachnahme für Inland Nachnahme für Ausland DM 14, -

Kombinierte Anwendung! Inhalt des Paketes: Datenbank, Textverarbeitung, Rechnerkopplung, Datenfernübertragung, Text-Editor, Disk&Harddisk-Monitor, HD-Backup, CAD, Ray-Tracing, Antivirus-Prog und vieles mehr.

Angebot des Monats: DM 45,— (V-Scheck), DM 49,— bei NN incl. Porto und Verpackung.

LAUFWERKE FÜR AMIGA 3 1/2" Extern, Metallgehäuse,

helle Front, durchgef. Port m. 299,-Schraubverr., Abschaltbar 3 1/2" Intern, komplett m.

Einbausatz u. Anleitung 239,-5 1/4" Extern, Metallgehäuse, helle Front, 40/80 Spur, durchge 369.-Port m. Verr., Abschaltbar

SPEICHERERWEITERUNG FÜR AMIGA 500

512 KB Ram, akkugep. Uhr und Abschaltung

1 MB Extern, angep. Metallgeh.,
Abschaltb. 698,-2 MB Extern, (Profex), durchg. Port, Abschaltbar 848,- COMPUTERLEITUNGEN

23,-Druckerkabel für alle Amiga Monitorleitung Amiga/Scart 25,-

Emulatorkabel 19,90 19,-Bootselector DF 0/DF 12

NEU IN UNSEREM ANGEBOT

Sound Digitizer Stereo f. a. Amiga Midi-Interface a. A. Mouse-Pad, Antistat., 16,90 rutschfest 23,90

Disketten NO NAME 2DD Weitere Angebote auf Anfrage. Preisänderungen vorbehalten.

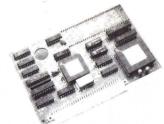
Versand per Nachnahme: Rainbow Data, Am Kalkofen 1, 5603 Wülfrath, Tel. 0 20 58/13 66

SPRACHEN/ENTWICKLI Metacomco Assembler Metacomco Pascal Lattice C-Compiler Vers. 4.0	UNG 159, 239,	King of Chicago Kings Quest III Mission Elevator Ports of Call Shadowgate Terrorpods Test Drive	64, 76, 59, 99, 66, 79,
SPIELE		The Guild of Thieves	69,
Bad Cat	54,	The Feary Tale	119,
Barbarian (Psygnosis)	66,	The Pawn	72,
Bard's Tale	89,	Ultima III	72,
Bureaucracy	89,	Ultima IV	72,
Chessmaster 2000	82,	Uninvited	79,
Dark Castle	69,	Western Games	59,
Defender of the Crown	79,		
Flight Simulator II	119,	DRUCKER	
Garrison II	64,	NEC P 2200	1049,
Goldrunner	69,	Epson LQ-500	998,
Impact	44,	Epson LX-800	699,
Jinxter	72,	Star LC 10	649,
Sofort kostenios Pre	isliste	bei Abteilung KS anfor	dern! 📆

Computer & Zubehör Versand Gerhard und Bernd Waller GbR

Kieler Str. 623, 2000 Hamburg 54, 2 040/570 60 07 + 570 52 75

Hurricane! DM 1998,-



- Für AMIGA 500, 1000, 2000
- Amiga läuft mit hohem Systemtakt (16 od.20 MHz)
- Superschnell: bis 1000% schneller.
- Schnellstes Turbo-Board auf dem Weltmarkt!!!
- 32 Bit-Ram (100 ns)

nur DM 2498,-



Borsigallee 18 6000 Frankfurt/M. & 069-410071/72 Schweiz: MICROTRON Bahnhofstraße 2 CH-2542 Pieterlen Tel 032 872429

2 Jahre Garantie auf Hardware und geprüfte Software mit Zertifikat, beides in bester Qualität unterscheiden uns zu Billigangeboten

AMIGA aktuell — Dieter Hieske Ladenlokal Schillerstraße 36 6700 Ludwigshafen-Oggersheim Telefon (06 21) 67 31 05

Öffnungszeiten: Montag—Freitag 9.30—12.00 Uhr, Montag—Freitag 12.00—18.00 Uhr, Samstag 9.00—13.00 Uhr. Danach Anrufbeantworter.

Public Domain Software Kopierservice lieferbar sind: Fish 1-128, Panorama 1-51, Faug 1-53, Amicus 1-20, Amuse 1-3, Chiron 1-39, TBAG 1-7, Auge 4000 1-14, Kickl 1.2V 1-30, ACS 1-39, Safe 1-4, Tornado 1-30, Taifun 1-50 usw.

Kopierpreise, wenn wir die Disketten 2DD mitliefern — Preise per Stück: 1-9 Disk 7,00 / 10-19 6,50 / 20-29 6,00, 30-49 5,50 / 50-99 5,00 / 100-200 4,50 ab 200 4,00, Fishserie 1-128 DM 539,00 (4,21)

Kopierpreise, wenn Sie uns Ihre Disketten per Einschreiben liefern — Preise per Stück: 1-9 4,00 / 10-19 3,50 / 20-29 3,00 / 30-49 2,50 / 50-99 2,00 / 100-199 1,50 ab 200 1,00.

Software Anwender

Sculpt 3D 182,90 / Animate 3D 298,00 / Forms in Flight 158,95 / Pagesetter D 298,00 / Videotitler 178,00 / Butcher D 99,00 / Photon Paint 239,00. Software Spiele

Software Spiele
StripPoker II + DM 42,95 / Destroyer
DM 68,90 / Giana Sisters DM 51,90 /
Ferrari Formula 1 DM 76,95 / BMX
Simulator DM 54,90 / Pink Panther DM
54,90 / Vampires Empire DM 54,90 /
Doubble Bubble DM 54,90 / Strike
Force Harrier DM 67,50 / Winter Olympic 88 DM 54,00. Neuheiten täglich,
bitte anrufen. Es lohnt sich.

Hardware

A500: 512 Kb Speicher, Uhr absch.
DM 249,00, A1000: 2 MB ext. Box DM
999,00, A500: 2 MB ext. Box DM
999,00, Laufwerk 3,5" DM 339,00, A500:
2 MB Profex 777,00, Bootselektor DF1
und DF2 je 19,95

Versand per Nachnahme + DM 8,00 Porto, Versand per Vorauskasse ohne Versandkosten

Preisliste und Public Domain Inhaltsverzeichnis (70 Seiten) kostenlos.



COMPUTERSOFT

AMIGA ARCADE GAMES		AMIGA SPORT GAMES	
Backlash	64,90	Grid Start	34,90
Battle Ship	64,90	Indoor Sports	79,90
Big Deal	79,90	Soccer King	29,,95
Blasterball	34,90	Thai Boxing	34,90
Brainstorm	34,90	Winter-Olympiade '88	64,90
Clever und Smart	64,90	AMIGA ADVENTURE	
Cogans Run	49,90	Dark Carstell	79,90
Defender of the Crown	86,00	Gnome Ranger	49,90
Destroyer	94,90	Hellowoon	74,90
ECO	79,90	Jinxter	79,90
Garrison II	69,90	Kings Quest I+II+III	79,90
Gee Bee Air Rally	99.00	Leisuresuit Larry	64,90
Insanity Flight	79,90	AMIGA STRATÉGIE	
Into the Eagles Nest	64.90	Kampfgruppe	89,00
Jinks	59,90	Ogre	79,90
Joe Blade	34,90	Roadwar 2000	64,90
Jump Jet	49,90	Roadwar Europa	79,90
Larrie and the Ardies	49,90	Tetris	64,90

HITS * HITS * HITS

Ports of Call	99,00	Great Giana Sisters	59,90
Strike Force Harrier	79.90	Balance of Power	89.90
King of Chikago	99,90	ECO	79,90
Leviathan	64.90	AMIGA ANWENDER	
Mercenary Compendium	79,90	Aegis Audiomaster	129,90
Mike — the Magic Dragon	29,95	Aegis Draw Plus	498,00
Moebius	79,90	Aegis Sonix Version 2.0	159,00
Phalanx II	29,95	Aegis Video Title	359,00
Phantasie III	59,00	Deluxe Paint II (deutsch)	259,00
Pinball Wizard	49,95	Digi Paint (deutsch)	169,00
Roadwars	64,90	Druckeranpassung CP-80X	59,90
Sky Blaster	64,90	Instand Music	149,00
Slaygon	64,90	Logistix (deutsch)	399,90
Star Wars	64,90	Mause Matte	19,90
Strange New World	49,90	Sculpt 3D Pal Version	229,00
Terramax	64,90	Sculpt Animate 3D	349,90
The Great Giana Sisters	59,90	Silver	299,90
Thunderboy	59,90	Sound Sampler A500/2000	149,90
Tolteka	64,90	The 64 Emulator 500/2000	169,90

** Wir suchen noch Programmautoren ** Preisänderungen vorbehalten ** CSJ COMPUTERSOFT GMBH



An der Tiefennede 27, 3000 Hannover 1, Tel. Bestellservice (05 11) 88 63 83 Händleranfragen erwünscht, sofort CSJ NEWS anfordern (Computertyp angeben und Briefmarken 1,50 DM beilegen). Versand Inland Vorkasse 3,00 DM (Euroscheck in DM) per Nachnahme 7,00 DM.



- Software zur Konfiguration von verschiedenen SCSI-Hard-Disk Laufwerken.
- Hyperschnell durch DMA. (Direkter Speicherzugriff)
- Mit 3.5" HD-Laufwerken kann eine File-card aufgebaut werden!!
- Ab Lager lieferbar, versandkostenfrei!! natürlich von:



Borsigallee 18 6000 Frankfurt/M. & 069-410071/72 Schweiz: MICROTRON Bahnhofstraße 2 CH-2542 Pieterlen Tel. 032872429





ABO

Absender (Bitte deutlich schreiben)				
Vorname/Name				
Straße/Nr.				
PLZ/Ort				

Postkarte

mit 60 Pf. frankieren

Heim-Verlag

Heidelberger Landstr. 194

6100 Darmstadt-Eberstadt Telefon 06151/56057



Einzelheft- u. Monatsdisketten Bestellung



Einzelheft- u. Disketten Service

Absender (Bitte deutlich schreiben)	
Vorname/Name	
Straße/Nr.	
PLZ/Ort	

mit 60 Pf. frankieren

Postkarte

Heim-Verlag

Heidelberger Landstr. 194

6100 Darmstadt-Eberstadt Telefon 06151/56057



Kontaktkarte



Kontaktkarte

Bitte Adresse der Firma, bei der Sie Informationen, oder etwas bestellen möchten, auf der

Absender (Bitte deutlich schreiben)

Vorname/Name

Beruf

Straße/Nr.

PLZ/Ort

Telefon Vorwahl/Rufnummer

Postkarte

Bitte freimachen

Firma

Straße/Postfach



Abonnement

für mindestens 1 Jahr (11 Hefte) zum ermäßigst (Ausland: Nur gegen Scheck-Voreinsendung D!) Der Bezugszeitraum verlängert sich nur dann un digt wird.	ren Preis von jährlich DM 70,— frei Haus. M 90,— Normalpost.) n ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekür
	Gewünschte Zahlungsweise bitte ankreuzen
	☐ Bequem und bargeldlos durch Bankeinzug
Name	Konto-Nr. BLZ
Vorname	
	Institut Ort
itraße/Nr.	☐ Ein Verrechnungsscheck über DM liegt bei.
PLZ Ort	
	Garantie: Diese Bestellung kann ich schriftlich innerhalb einer Woche (rechtzeitige Absendung genügt) widerrufen. Dies bestätige ich durch meine 2. Unterschrift.
Datum Unterschrift	Datum Unterschrift



ABO



Einzelheftu. Monatsdisketten Bestellung

KICKSTART können Sie direkt beim HEIM-VERLAG zum Einzelheft-Preis von DM 7, – (zuzüglich Gebühr für Porto und Verpackung) nachbestellen. Bearbeitung nur gegen beigefügten Scheck über den entsprechenden Betrag (keine Überweisung).

Jan.	Febr.	März	April	Mai	Juni	Juli/Aug.	Sept.	Okt.	Nov.	Dez.		
											1987 = DM	
											1988 = DM	
+ Ge	ebühr f	für Po	rto u.	Verp							= DM	,
\square s	check	in Hö	ihe								zus. DM	liegt bei

Disketten Service

Alle Programme, die in KICKSTART veröffentlicht wurden, sind auf Disketten erhältlich. Die Disketten enthalten die Programme von jeweils 2 KICKSTART-Ausgaben. Bestellen Sie durch ankreuzen die gewünschten Disketten

Preis je	Juli/Aug.	Sept./Okt.	Nov./Dez.	Jan./Febr.	März/Apr.
Diskette 19,– DM	87	87	87	88	88

Lieferung: gegen beigefügten Scheck zuzügl. 5,- DM Versandkosten



Einzelheft- u. Monatsdisketten Bestellung



_	- 1	1									*							
	lch	ı b)1	tt	e	u	m	wei	te	re	In	O	rn	la	t1	01	ne	en

☐ Ich gebe folgende Bestellung auf

in Bezug auf Ihre Anzeige in Kickstart Heft _____ Seite __

Menge	Produkt/Bestellnummer	DM	gesamt DM
	,		
			*

Abgesandt am:



Firma:		
		,
Bemerku	ngen:	

Bemerkungen:							

Datum, Unterschrift (für Jugendliche unter 18 Jahren der Erziehungsberechtigte)



Kontaktkarte

H A		14
		K
		1
	4 D	

Kontaktkarte

Absender (Bitte deutlich schreiben)

Vorname/Name

Beruf

Straße/Nr.

PLZ/Ort

Telefon Vorwahl/Rufnummer

Postkarte

Bitte freimachen

Eirman

Straße/Postfach

- ---

Or



Kurzmitteilung



Kurzmitteilung

Postkarte

Bitte freimachen

Absender (Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Telefon

Merlin Computer GmbH KICKSTART Redaktion Industriestraße 26

6236 Eschborn



PD Bestellung



PD Bestellung

Postkarte

Bitte freimachen

Absender (Bitte deutlich schreiben)

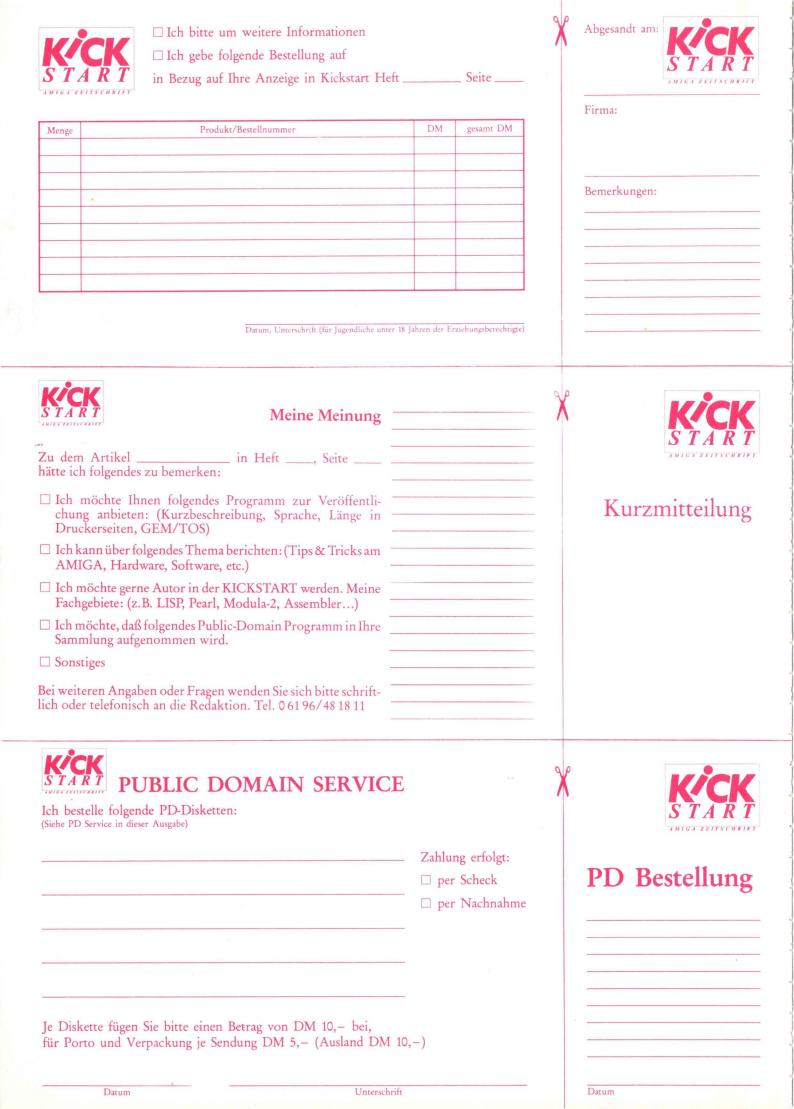
Vorname/Name

Straße/Nr.

PLZ/Ort

Merlin Computer GmbH KICKSTART Redaktion Industriestraße 26

6236 Eschborn





ir begrüßen Sie zu der neuen Ausgabe der KICKS FÜR INSIDER.

Die darin enthaltenen Programme dienen zum einen als Beispiel zum Selbstprogrammieren, zum anderen sind sie natürlich auch sehr nützlich in der Anwendung. Der nette nebenstehende Herr zeigt deutlich, daß die KICKS recht heftig auf ihn wirkten. Wir wünschen auch Ihnen damit einen ebenso eindrucksvollen Erfolg.

INHALT

GUT GEPACKT

IFF-Format, na und ?!

Seite108

BIST DU ABER GROß GE-WORDEN

DIR-Befehl mit Längenangabe

Seite 100

KOMFOR-TABLE CURSOR-POSITIO-NIERUNG

für C-Programmierer

Seite104

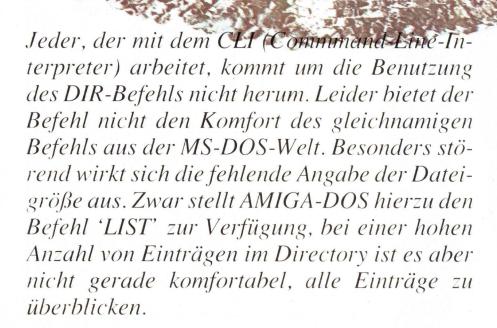
AUF-GERISSEN

CLI auf voller Größe

Seite 113

BIST DU ABER GROß GEWORDEN

DIR-Befehl mit Längenangabe



as abgedruckte Programm schafft Abhilfe. Es ist eine Mischung aus beiden Befehlen. Das Programm gibt in zwei Spalten aufgegliedert die Directoryeinträge aus und gleichzeitig die Größe der ausgegebenen Datei an. Durch diese Ausgabe ist es dem Anwender möglich, viele Dateien und deren Größe auf dem Bildschirm zu überblicken . AMIGA-Anwender werden sicherlich dieses Manko des CLI-DIR-Befehls schon gerügt haben.

Das Programm ist nach dem Assemblieren lediglich 948 Bytes groß und wird dadurch schnell eingeladen.

Das 'modifizierte DIR' wurde mit dem Profimat-Assembler erstellt. Zu Beginn des Listings werden die Offsets für die Bibliotheksroutinen und zwei MACROS definiert, wobei dem MACRO 'CallSys' besondere Beachtung geschenkt werden sollte. Dieses ist nicht identisch mit dem gleichnamigen 'CallSys' aus dem Include-File. Nach den Definitionen folgt das Hauptprogramm, von wo die Unterprogramme

aufgerufen werden; der Programmablauf wird dadurch übersichtlicher und ist leicht zu ändern. Die Unterprogrammnamen lassen leicht auf die Aufgabe des jeweiligen Unterprogramms schließen.

Ansonsten bietet das dokumentierte Listing genug Informationen über den Programmablauf. Vergessen Sie nicht, das assemblierte Programm auf Ihre Workbench-Diskette in den Ordner 'c' zu kopieren. In diesem Ordner ist das komfortable 'DIR' immer abrufbereit. Der Name des Programms bleibt Ihnen überlassen, wir in der Redaktion haben es 'SDIR' getauft.

```
FileSize
              von Matthias Kühn
                 für KICKSTART
LVOOpenLibrary EQU -$0228
LVOCloseLibrary EQU -$019e
LVOLock
               EOU -$0054
LVOUnLock
               EQU -$005a
LVOExamin
               EQU -$0066
LVOExNext
               EOU -$006c
LVOOutput
               EQU -$003c
               EQU -$0030
LVOWrite
ExecBase
               EOU
ACCES READ
                        -2
               EOU
CallSys
             MACRO $\Sub,$\Base
     MOVE.L \Base, A6
      JSR
             LVO\Sub(A6)
      ENDM
Spalte MACRO $\Zif1,$\Zif2
     MOVE.B #$0d, (A0)+
     MOVE.B #$9b, (A0)+
     MOVE.B #\Zif1, (A0)+
     MOVE.B #\Zif2, (A0)+
     MOVE.B #$43, (A0)+
      ENDM
MOVEM.L DO/AO/SP, Inital DO
BSR OpenDos_
BSR
      Output
BSR
     Lock
     Directory_
BSR
BSR
     Files
Error:
BSR
     Unlock
BSR
     CloseDos
MOVE.L Inital SP, SP
RTS
OpenDos
              DO
      CLR L
                                     ; Version ist
egal
      LEA
             DosName, Al
                                     : Name der
Library
     CallSys OpenLibrary, ExecBase
                                     ;Library öffnen
      MOVE L DO, DosBase
                                     ; Adresse retten
      BEO
                                     ; bei Fehler Ende
      RTS
      TST.L
             DosBase
                                     ;ist Library
geöffnet
      BEO
              \nein
                                      ; wenn ja
     MOVE.L DosBase, A1
     CallSys CloseLibrary, ExecBase
                                     ; dann schließen
nein: RTS
                                     ; sonst nicht
Output
     CallSys Output, DosBase
                                     ; momentanen
```

```
OutputFile
      MOVE.L DO, Output
                                       ; holen und
retten
      BEO
               Error
                                       ; wenn keiner
da, dann Ende
      RTS
     MOVEM.L Inital D0, D0/A0
; CommandLenght und Buffer
      ADD.L
              D0, A0
                                      ; Pointer auf
Ende
      SUB. L
               #1,A0
                                       ; und eins
weiter, dann
      MOVE . B #0, (A0)
                                      ; LockInput
mit 0 abschließen
      MOVE.L ACCES_READ, D2
                                       ; wir wollen
nur lesen
     MOVE.L Inital_A0,D1
                                      ;Pointer auf
CommandBuffer
     CallSys Lock, DosBase
                                      ; Dir Zugriff
holen
      MOVE.L DO, Lock
                                      ; und retten
      BEO
              Error
                                       ; gibt es
keinen, dann Ende
      RTS
Unlock
      MOVE L Lock D1
                                      :DirZugriff
holen
      BEQ
               \Ende
                                       ; gibt es
keinen, dann Ende
      CallSys Unlock, DosBase
                                       ; sonst
schliesen
\Ende RTS
Directory :
      MOVE . L Lock , D1
                                       ; DirZugriff
holen
     MOVE.L #FileInfoBlock,D2
                                      ; Pointer auf
Buffer
      CallSys Examin, DosBase
                                       :DirInfo
      TST.L
              DirEntryType
                                      ; testen ob es
ein Dir ist
      BMI
              Error
                                       ; wenn nicht
dann Exit
      LEA
              DirText, Al
      LEA
              WriteBuffer, A0
      MOVE.W #4,D0
                                       ;5 Schleifen
\Dir MOVE.W (A1)+, (A0)+
                                      ; 'Directory:
in Buffer schreiben
      DBRA
            DO.\Dir
      LEA
              FileName, A1
                                       ;Dir in
Buffer schreiben
\next MOVE.B (A1)+, (A0)+
      BNE
              \next
      SUB. L
              #1,A0
      MOVE.B #'/', (A0)+
; Steuerzeichen in Buffer schreiben
      MOVE.B #$a, (A0)+
      MOVE.B #$d, (A0)+
     MOVE.L A0, TextPointer
              Write
                                      ; und ausgeben
```

RTS Files :		
_	GetFile	;Fileinfo
TST.L	D0	;war noch eins
THE REPORT OF THE PARTY OF THE	\1Ende	;wenn nein
dann Ende LEA	WriteBuffer, A0	;Pointer auf
WriteBuffer BSR	WriteFile	;vorderes File
schreiben Spalte	12','4'	;Steuerzeichen
schreiben	Size, D0	;Size holen
	\1Size	;ist Size = 0,
BSR	WriteDir	;'Directory'
schreiben BRA	\1	; sonst
\1Size BSR	WriteSize	;Bytelänge
schreiben \1 Spalte	`3','8'	;Steuerzeichen
MOVE . L	A0, TextPointer	; TextPointer
	GetFile	;FileInfoBlock
	TextPointer, A0	;TextPointer
	D0,-(SP)	;war kein File
da BEQ	\2	;dann Ende und
D0 auf Stack BSR	WriteFile	; sonst
hinteres File	schreiben	
	`6','2'	;wie oben
MOVE.L	Size, D0	
BNE BSR	\2Size WriteDir	
BRA	\2	
\2Size BSR		
Ende	#\$a,(A0)+	; CR und LF ans
	#\$d, (A0) +	
MOVE.L BSR	A0, TextPointer Write	
TST.L	(SP)+	;weiteres File
? BNE	Files_	;wenn ja, dann
an Anfang		glade parameters and the
\lEnde RTS GetFile:		;sonst Ende
MOVE.L holen	Lock, D1	;DirZugriff
MOVE.L Buffer	#FileInfoBlock,D2	;Pointer auf
CallSys	ExNext, DosBase	;FileInfoBlock
holen		
WriteFile: LEA	FileName, A1	;FileName
eintragen \next MOVE.E	3 (A1)+, (A0)+	;solange
Zeichen holen BNE	bis \next	;ein Null
kommt SUB.L	#1,A0	dann Pointer;
ein zurück RTS		
WriteDir:	DirText, Al	;'Directory'
schreiben		1000000
Ordnung	#' ', (A0) +	;ein Space zur
MOVE.W	#8,D0	;9 Schleifen

THE RESIDENCE AND RESIDENCE	THE RESERVE TO SHARE THE PARTY OF THE PARTY	Name and Address of the Owner, where	AND DESCRIPTION OF THE PERSON NAMED IN	
\next	MOVE.B			; Zeichen holen
		DO, \nex	t	A. 表示的复数形式 1886 · 第二章的
	RTS			
WriteS				
	MOVE . B			;Leerzeichen in D5
		Divisor	, A1	
	MOVE . W	#8,D4		;9 mal Schleife
durchl			e della	and the second of the second of the second of
\10er	MOVE . L			erster Divisor
	CMP.L			;ist Dividend kleiner
		\Nicht		;wenn nein
	MOVE.W	#31,D3		;32 Schleifen
durchl	aufen	20		
		D2		
/roob	ASL.L			;D0 um 1 nach links und
	ROXL.L			;in D2 schieben
100	CMP.L			;paßt Divisor in D2
		\ADR		;wenn ja
	SUB.L	DI, DZ		;dann Divisor von D2
subtra	hieren	#1 50		
1200	ADD.L DBRA ADD.B	#I,D0		;und 1 zu D0 addieren
\ADR	DBKA	#\$30 PG	P	in ACCIT was dela
				; in ASCII umwandeln ; ASCII schreiben
	MOVE I		T	; ASCII schreiben ; Rest -> Dividend
	MOVE.L MOVE.B			
	BRA			;D5 umschalten
Mich	BRA t MOVE.B		1+	;' 'oder '0' schreiben
	DBRA			, oder o schreiben
(Hexc	ADD.B			;in ASCII umwandeln
	MOVE.B			; in ASCII umwandein ; und schreiben
	RTS	DO, (AO)	T	, und schreiben
Write				TO STATE OF STATE OF
MIICE_		Output	D1	; OutputHandle
holen	MOVE. L	Output,	DI	, output handre
noten	MOVE I	#WriteE	Suffer, D	2 ;Pointer auf
TextAr		MMIICEL	direr, D	z ,Foincer aui
TEACH		TextPoi	nter, D3	;TextEnde in D3
			Suffer, D	
ermitt		WILLCEL	Juliel, D	, and Lange
Camao		****** *)D	
	CallSvs	write.	Joshase	: Text, ausgeben
	CallSys RTS	write,	osbase	;Text ausgeben
; ****	RTS			
;****	RTS			;Text ausgeben paar Daten **********
	RTS ******** DATA	*****	*** ein	paar Daten **********
	RTS ******** DATA	*****	*** ein	
	RTS ****** DATA me: ALIGN	*****	**** ein 'dos.li	paar Daten ************************************
DosNa	RTS ****** DATA me: ALIGN	****** DC.B	*** ein	paar Daten ************************************
DosNa	RTS ****** DATA me: ALIGN xt: ALIGN	****** DC.B	**** ein 'dos.li	paar Daten ************************************
DosNan DirTe	RTS ****** DATA me: ALIGN xt: ALIGN	DC.B DC.B	**** ein 'dos.li 'Direct	paar Daten ************** brary',0
DosNan DirTe	RTS ******* DATA me: ALIGN xt: ALIGN or: 00000,100	DC.B DC.L DC.L	**** ein 'dos.li 'Direct	paar Daten ************** brary',0
DosNan DirTe: Divis 100000	RTS ******** DATA me: ALIGN xt: ALIGN or: 000000,100	DC.B DC.B DC.L 0000000,:	**** ein 'dos.li 'Direct	paar Daten ************************************
DosNan DirTe: Divis 100000	RTS ******** DATA me: ALIGN xt: ALIGN or: 000000,100	DC.B DC.B DC.L 0000000,:	**** ein 'dos.li 'Direct	paar Daten ************************************
DosNan DirTe: Divis 100000	RTS ******** DATA me: ALIGN xt: ALIGN or: 00000,100 *********	DC.B DC.B DC.L 0000000,:	**** ein 'dos.li 'Direct	paar Daten ************************************
DosNan DirTe: Divise 100000	RTS ******** DATA me: ALIGN or: 00000,100 ******** BSS 1_D0	DC.B DC.B DC.L 0000000,: DC.L *******	**** ein 'dos.li 'Direct 10000000 10000000 **** unc	paar Daten *************** brary',0 cory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos **
DosNam DirTe: Divise 100000 ;****	RTS ******** DATA me: ALIGN xt: ALIGN or: 000000,100 ******** BSS 1_D0 1_A0	DC.B DC.B DC.L 0000000,: DC.L ********	**** ein 'dos.li 'Direct 10000000 1000000,1 **** unc	paar Daten *************** brary',0 .ory:' .1000000 .0000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen
DirTe: Divis: 100000 ;**** Inita Inita	RTS ******** DATA me: ALIGN or: D00000,100 ******* BSS 1 D0 1 A0 1 SP	DC.B DC.L DO000000,: DC.L ********* DC.L DC.L	**** ein 'dos.li 'Direct 10000000,1 100000,1 **** unc	paar Daten ************************************
DosNan DirTe: Divis: 100000 ;**** Inita Inita Inita	RTS ******** DATA me: ALIGN or: 000000,100 ******* BSS 1_D0 1_A0 1_SP se	DC.B DC.L 0000000,: DC.L ******** DC.L DC.L DC.L DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** unc 0 0 0	paar Daten ************************************
DosNan DirTe: Divis: 100000 ;**** Inita Inita Inita DosBa	RTS ******** DATA me: ALIGN or: 000000,100 ******* BSS 1_D0 1_A0 1_SP se	DC.B DC.L DO000000,: DC.L DC.L DC.L DC.L DC.L DC.L DC.L DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0	paar Daten ************************************
DosNan DirTe: Divis: 100000 ;***** Inita Inita Inita DosBa Outpu Lock	RTS ******** DATA me: ALIGN or: 00000,100 ******* BSS 1_D0 1_A0 1_SP se t	DC.B DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0	paar Daten ************************************
DosNan DirTe: Divis: 100000 ;***** Inita Inita Inita OosBa Outpu Lock TextP	RTS ******** DATA me: ALIGN or: 00000,100 ****** BSS 1_D0 1_A0 1_SP se t	DC.B DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0 0 0	paar Daten ************************************
DosNan DirTe: Divis: 100000 ;**** Inita Inita Inita Outpu Lock TextP	RTS ******** DATA me: ALIGN or: 00000,100 ****** BSS 1_D0 1_A0 1_SP se t	DC.B DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0 0 0	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis: 100000 ;**** Inita Inita Inita Outpu Lock TextP	RTS ******** DATA me: ALIGN or: 00000,100 ****** BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben	DC.B DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0 0 0	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis 100000 ;**** Inita Inita Inita OosBa Outpu Lock TextP FileI: gesch: DiskK DirEn	RTS ******** DATA me: ALIGN or: 00000,100 ******* BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType	DC.B DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0 0 0 werden d 1,0 1,0	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis 100000 ;**** Inita Inita Inita OosBa Outpu Lock TextP FileI: gesch: DiskK DirEn FileN	RTS ******** DATA me: ALIGN or: 000000,100 ******* BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000,1 100000,1 **** unc 0 0 0 0 0 werden d 1,0 1,0 108,0	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis 100000 ;**** Inita Inita Inita Outpu Lock TextP FileI: gesch: DiskK DirEn FileN Prote	RTS ******** DATA me: ALIGN or: 000000,100 ******* BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame ction	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000,1 100000,1 **** und 0 0 0 0 0 werden d 1,0 1,0 108,0 1,0	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis: 100000 ;****: Inita Inita Inita Output Lock TextP FileI: gesch: DiskK DirEn FileN Prote Entry	RTS ******** DATA me: ALIGN or: 000000,100 ******* BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame ction	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0 werden d 1,0 1,0 108,0 1,0 1,0 1,0	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis: 100000 ;****: Inita Inita Inita JosBa Outpu Lock TextP FileI: gesch: DiskK DirEn FileN Prote Entry Size	RTS ******** DATA me: ALIGN or: 000000,100 ******* BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame ction Type	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0 0 0 werden d 1,0 1,0 1,0 1,0 1,0 1,0 1,0	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis: 100000 ;**** Inita Inita Inita JosBa Outpu Lock TextP FileI gesch: DiskK DirEn FileN Prote Entry Size NumBl	RTS ******** DATA me: ALIGN or: 000000,100 ******* BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame ction Type	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0 0 0 werden d 1,0 1,0 1,0 1,0 1,0 1,0 1,0 1,0	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis: 100000 ;****: Inita Inita Inita Output Lock TextP FileI: gesch: DiskK DirEn DiskK DirEn Prote Entry Size NumBl Days	RTS ******** DATA me: ALIGN or: 000000,1000 ******* BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame ction Type ocks	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0 0 0 0 0 0 0 1,0 1,0 1,0 1,0 1,0	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis: 100000 ;***** Inita Inita Inita JosBa Outpu Lock TextP FileI: gesch: DiskK DirEn FileN Prote Entry Size NumBl Days Minut	RTS ******** DATA me: ALIGN or: 000000,1000 ******* BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame ction Type ocks	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000,1 1000000,1 **** und 0 0 0 0 0 0 0 0 0 1,0 1,0 1,0 1,0 1,0 1	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis: 100000 ;***** Inita Inita Inita JosBa Outpu Lock TextP FileI: gesch: DiskK DirEn FileN Prote Entry Size NumBl Days Minut Tick	RTS ******** DATA me: ALIGN or: 00000,100 ****** BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame ction Type ocks e	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000,1 1000000,1 **** und 0 0 0 0 0 0 0 0 0 1,0 1,0 1,0 1,0 1,0 1	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis: 100000 ;***** Inita Inita Inita JosBa Outpu Lock TextP FileI: geschi DiskK DirEn FileN Prote Entry Size NumBl Days Minut Tick Comme	RTS ********* DATA me: ALIGN or: 00000,100 ****** BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame ction Type ocks e nt	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000 100000,1 **** und 0 0 0 0 0 0 0 1,0 1,0 1,0 1	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse
DosNan DirTe: Divis: 100000 ;***** Inita Inita Inita JosBa Outpu Lock TextP FileI: geschi DiskK DirEn FileN Prote Entry Size NumBl Days Minut Tick Comme	RTS ******** DATA me: ALIGN or: 00000,100 ****** BSS 1_D0 1_A0 1_SP se t ointer nfoBlock rieben ey tryType ame ction Type ocks e	DC.B DC.L DC.L	**** ein 'dos.li 'Direct 10000000,1 1000000,1 **** und 0 0 0 0 0 0 0 0 0 1,0 1,0 1,0 1,0 1,0 1	paar Daten ************************** brary',0 .ory:' ,1000000 10000,1000,100,10 d etwas Platz für Infos ** ;dosCmdLen ;dosCmdBuf ;Rücksprungadresse

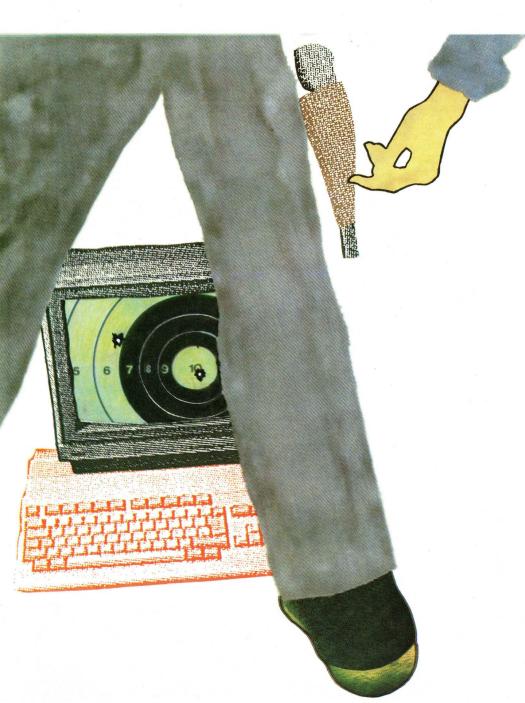
KORREKTUR

Ärztlicher
Kunstfehler.
Leider wurde
im Artikel
'Binäre
Antibiotika'
(KICKSTART
4/88, Seite 84)
Listing 1
doppelt
abgedruckt.
Hier ist
es nun in voller
Pracht.

```
2: ; * BootBlock AntiVirus
3: ; * gegen SCA und
4: ;* ByteBandit
5: ;* written 1988 by
6: ; * Garry Glendown
   ; * (c) KICKSTART
11: move.l 4,a6
                       ; Standardvorbereitung
12: lea.1 dosname(pc),a1; vor dem Booten
17: move.1 22(a0),a0
18: movem.1 d0-d7/a0-a6,-(a7) ; Register retten
   move.w #0,$dff180
                          : Bildschirm schwarz
                          ; ExecBase
                         ; CoolCapture ?
                          ; wird benutzt !
24: move.w 148(a6),d0
                          ; IntVek[5] = VBI Int.
25: cmp.w #$fc,d0
                          ; Kickstartbereich ?
27: move.1 550(a6),d0
                          ; KickTagPtr ?
                          ; nicht benutzt !
30: move.1 #0,550(a6)
                         ; KickTagPtr löschen
31: move.1 #0,554(a6)
                          ; KickCheck löschen
                          ; Bildschirm flimmern
34: move.w #$f00,$dff180 ; Bildschirm rot
35: move.1 $7ecc4,d0
36: cmp.1 #$444f5300,d0 ; noch aktiv?
37: bne.s ende
38: move.1 #$01020304,$7ecc4; deaktivieren
                         ; Bildschirm flimmern
40: move.w #$f0,$dff180
41: btst #6,$bfe001
                          ; bis Maustaste
42: beg.s ende
43: move.w #$f,$dff180
48: movem.1 (a7)+,d0-d7/a0-a6; Register zurück
52: moveq #-1,d0
55: dc.b 'dos.library',0
Listing 2: Behandlung einer infizierten Diskette mit Medikamenten der 'Binären Antibiotika
```

CURSOR-POSITIONIERUNG

Komfortables für C-Programmierer



Vielen BASIC-Programmierern ist der Befehl 'LOCATE' sicher ein Begriff. Mit seiner Hilfe kann der Cursor an eine beliebige Stelle im Fenster gesetzt werden. Der Befehl verlangt lediglich die Eingabe einer X- und einer Y-Koordinate. Nicht anders verhält es sich mit der hier vorgestellten Funktion. Sie ist jedoch für C-Programmierer gedacht, da keine ähnliche Funktion unter C bereitgestellt wird. Der Cursor wird in einem Dos-Fenster auf die Position X/Y gesetzt.

ie nötige Geschwindigkeit erreicht das Programm dadurch, daß es in Assembler gehalten ist. Die Funktion kann direkt in den C-Code eingebunden oder auch dazu gelinkt werden. Beim Einbinden ist jedoch der Aztec C-Compiler vonnöten.

Der Funktion müssen drei Parameter übergeben werden:

Input

fh = Zeiger auf den FileHandle
des Dos-Fensters

X = X-Koordinate des Cursors

Y = Y-Koordinate des Cursors

Beispiel:

SetCrs(OutPut(), 20, 20);

Das Programm muß unter Aztec C folgendermaßen assembliert werden:

as SetCrs.asm

Einbinden

Das Einbinden des Assemblerprogramms in ein C-Sourceprogramm muß wie folgt vonstatten gehen:

#asm
...SetCrs.asm...
#endasm

Die Labels '#asm' und '#endasm' müssen gesetzt werden, damit der Compiler erkennt, daß ein Assemblerteil folgt.

Ebenso kann das kleine Assemblerprogramm zu einem bestehenden Programm dazugelinkt werden. Der Syntax hierzu lautet:

ln Test.o SetCrs.o -lc

Das abgedruckte Listing ist kommentiert, so daß Sie schnell erfahren können, was in welcher Zeile vonstatten geht.

```
1 ; (c) KICKSTART 1988
 2
3 ; SetCrs()
 4 ;
5 ; Setzt den Cursor an die Position x/y im DOS-Fenster
7 ; INPUTS: fh - FileHandle des Fensters
            x - x-Position (Spalte)
8 ;
 9;
                  - y-Position (Reihe)
             У
10 ;
11 ;
12 ; MODIFICATION HISTORY:
13 ;
               Name:
14 ; Datum:
                                   Beschreibung:
15 ; -
16 ; 23.01.1988 Andreas Trappmann
                                    AMIGA-Version 1.0
17 ;
18
   _LVOWrite equ
19
                      -48
20
21
             public _DOSBase
22
23
              public
                       SetCrs
24 _SetCrs:
              move.1
                      4(sp),d1
                                    ; FileHandle
25
26
             move.w
                     8(sp),d0
                                    ; x-Position
27
              lea
                      xpos, a0
                                    ; Adresse laden
                      _convert
28
                                    ; umwandeln
              bsr
29
                                    ; y-Position
30
             move.w
                     10(sp), d0
31
                      ypos, a0
                                    ; Adresse laden
              lea
                      _convert
32
              bsr
                                    ; umwandeln
33
                                   ; Pufferadresse laden
34
             move.1
                     #buffer, d2
35
              moveq
                      #7,d3
                                    ; Laenge laden
                     _DOSBase, a6
                                    ; DosBase laden
36
             move.1
37
              jsr
                      LVOWrite(a6)
38
              rts
39
   _convert: and.1
                      #$ff, d0
                                    ; auf Byte beschraenken
40
              divu
                      #10,d0
41
                      conv1
                                    ; in ASCII wandeln
42
              jsr
                                    ; Rest holen
                      d0
43
              swap
                     #'0',d0
              add.b
                                    ; in ASCII wandeln
44 _conv1:
             move.b d0,(a0)+
                                    ; in Puffer schreiben
45
46
              rts
47
                       0
48 buffer:
              ds
49
              dc.b
                       $95
50 ypos:
              dc.b
                       0,0
51
              dc.b
                       1; "
52 xpos:
              dc.b
                       0,0
53
              dc.b
                       'H'
```

INSERENTEN-VERZEICHNIS

	Seite		Seite		Seite
ALCOMP		HIESKE	98	SOFTWARE 2000	28
ARIOLASOFT	19	IDEE SOFT	73	SCHMIELEWSKI	97
CVS	38	IM	28, 44, 98	VIDEO LOFT	50
CSJ	98	KUPKE	132	WALLER	98
DATA-BECKER	2, 39	PDC	66	WSW	38
GTI	131	PHILGERMA	91	YELLOW	38
HEIM	47	RAINBOW DATA	97		

DERGROSSEKICKSTART

HARDWARE WETTBEWERB

Der Amiga ist ein offenes System, es bietet reichlich Schnittstellen zur Außenwelt. Nach den Standardschnittstellen, wie RS 232 und **Druckerport**, sind RGB Port und der Expansionport die interessantesten Verbindungen für Peripheriegeräte. In der Theorie mag das den Anwender begeistern, auf der praktischen Seite fehlen jedoch einfach die entsprechenden Geräte und Erweiterungen. Wir wollen genau da ansetzen und mit diesem Wettbewerb Abhilfe schaffen.

Wer kann mitmachen?

Es ist jede Person an unserem Hardwarewettbewerb beteiligt, die bis zum Einsendeschluß einen fertigen Prototyp und die Schaltungsbeschreibung einsendet. Die Schaltung muß natürlich frei von Rechten Dritter und darf bis jetzt noch nicht in einer anderen Art veröffentlicht worden sein.

Was darf es sein? Gefragt sind bei dieser Ausschreibung alle Arten von Hardware. Als Anregung hier eine kleine Liste mit Vorschlägen:

- Speichererweiterungen
- MIDI-Schaltungen
- Stereo-Verstärker
- Schnittstellentreiber
- Programmiergeräte (z.B. PAL Brenner)
- Video oder HF Modulatoren
- Soundsampler
- Videodigitizer
- Meßwerterfassung
- Ein/Ausgabe Port
- Soundsampler
- Digitizer
- Meßwerterfassung
- Ein/Ausgabe-Port

oder was Ihnen hier sonst noch einfält

Der Phantasie der Entwickler sollen hier keine Grenzen gesetzt werden. Auch kleinere Projekte werden hier berücksichtigt, zumal auch der Bauteileaufwand sowie die Kosten der Schaltung in die Bewertung mit eingehen. Das Speeder-Projekt in dieser Ausgabe gibt Ihnen ein weiteres Beispiel für -wie könnte es aussehen-.

Das müssen Sie tun. Schicken Sie Ihre Schaltung mit Beschreibung und Prototyp an folgende Adresse:

MERLIN Computer Gmbh

Kennwort Kickstart-Hardwarewettbewerb

Industriestr. 26, 6236 ESCHBORN Der Einsendeschluß ist der 16.6.88, es gilt das Datum des Poststempels.

Die Gewinne

Wir haben Preise im Wert von 6600,-DM ausgesetzt.

1. Preis 3000,- DM

2. Preis 2000,- DM

3. - 10. Preis je 20 PD Disketten nach freier Auswahl aus unserem Angebot Teilnahmebedingungen:

Die Vergabe der Preise erfolgt durch eine aus Redaktionsmitgliedern gebildete Jury. Der Rechtsweg ist ausgeschlossen.

Alle eingesandten Schaltungen, Schaltpläne und Prototypen werden nach ihrer Beurteilung durch die Jury an die Einsender zurückgeschickt.

Der Teilnehmer gibt durch seine Einsendung die Erklärung ab, daß die Schaltung frei von Rechten Dritter ist. Das Copyright der Preise 1. und 2. geht an die Merlin Computer Gmbh. Bei allen übrigen Einsendungen bleibt das Copyright beim jeweiligen Autor. Sollten unter diesen Projekten für eine Veröffentlichung intessante Vorschläge sein, dann werden wir uns mit dem Autor in Verbindung setzen, um ihm ein Angebot zu machen.

Mitarbeitern der Merlin Computer Gmbh und deren Angehörigen ist die Teilnahme untersagt.

KICKSTART DISKETTEN-SERVICE

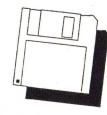
Für AMIGA-Fans, denen das Abtippen der abgedruckten Programme zu viel Mühe bedeutet, gibt es jetzt eine Alternative, denn alle in der KICKSTART abgedruckten Programm-Listings (bei Compilersprachen natürlich auch die ausführbaren Program-

me) befinden sich auf den KICKSTART-MONATSDIS-KETTEN. Auf jeder Diskette sind die Programme von zwei Monaten enthalten. Der Preis liegt bei nur DM 19.- pro Diskette zuzüglich DM 5.- Versandkosten.



Juli/Aug. 87

Fractals Business-Grafik Funktions-Plotter



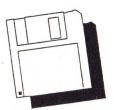
März/April 88

Röhrenschoner (Bildschirmabschalter)
Jump (IF-Befehl für's CLI)
Find (Datei-Sucher)
Fractale Landschaften (Basic)
Systemfonts in Basic
Colorhunter
DigiSound (Basic)
Viruschecker
(SCA,Byte Bandit)



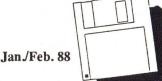
Nov./Dez. 87

MovieMaker C-Shell Tastenbelegung Vokabeltrainer Viruschecker

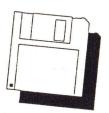


Sept./Okt. 87

3D-Rotation Farbeditor BAM-Copy Puzzle (Spiel)



TRON (Spiel)
KICKS (Requester,
Scroll-Demo in Basic,
Out, Fullscreen)
Konverter (AmigaASCII - PC-ASCII)
SetPCDate (Uhrzeit
für PC)

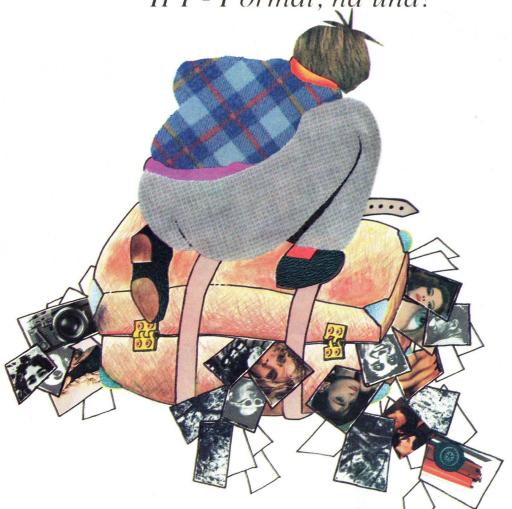


Für die Bestellung können Sie die Postkarte in der Heftmitte benutzen!



GUT GEPACKT

IFF- Format, na und?



Gelobet sei das IFF-Format, so heißt es hier, so heißt es da. Und wirklich, Kompatibilität zwischen grundverschiedenen Zeichenprogrammen ist eine feine Sache. Aber besonders für BASIC-Programmierer resultiert daraus ein großer Nachteil: Die Dateien sind komprimiert, und deshalb vom BASIC schlecht ladbar. Was macht man aber, wenn man unbedingt sein DeluxePaint-Machwerk als Titelbild für sein neues Programm braucht?

un, eine Lösung wird von Commodore auf der EX-TRAS-Diskette mitgeliefert und heißt LoadILBM-SaveACBM. Heureka, werden da manche rufen, beim Ausprobieren ziehen sich die Gesichter aber in die Länge:

Extrem lange Ladezeit und ein ellenlanger Quelltext trüben die Freude am eigenen Titelbild! Doch es gibt ja von diesem Programm aus die Möglichkeit, das ILBM (=Interleaved Bitmap)-Bild als ein sogenanntes ACBM-Bild zu speichern, was wiederum Amiga Continous Bitmap heißt. Was ist daran anders? Das ILBM-Format klaubt für jeden Punkt das entsprechende Bit aus allen Bitplanes zusammen, im ACBM-Format wird Bitplane nach Bitplane komplett gespeichert (und geladen), was für den Computer einfacher ist und somit schneller geht.

Es befindet sich auch ein Programm names LoadACBM auf der Diskette, und tatsächlich, das Bildladen geht schon recht flott. Aber auch hier ist die Laderoutine sehr offen gehalten und deshalb sehr umfangreich. Außerdem: wer will in SEINEM Programm schon einen FREMDEN Quelltext, von dem er nur weiß, daß er und nicht, wie er funktioniert. Und: Meistens geht es nur darum, ein stehendes Bild zu laden und auf den Bildschirm zu bringen, dazu werden dann weder Paletten-Rotationen noch sonstige Features gebraucht, sondern nur die Farbpalette und der Bildschirmspeicher.

Lange Rede, kurzer Sinn: Wie bekommt man die Farbpalette und den Bildschirmspeicher Plane für Plane auf die Diskette?

Beide Probleme lassen sich leicht lösen, wenn das Bild erst einmal auf dem Bildschirm ist. Dann müssen nur noch Startadressen der Planes und der Farbtabelle ermittelt und die entsprechenden Speicherbereiche auf Disk gebannt werden.

Der Dreh: Man läßt einmal LoadILBM SaveACBM für sich arbeiten und nimmt dann die lange Ladezeit in Kauf. Natürlich hat man das Programm vorher etwas präpariert, und brav werden die gewünschten Daten auf Diskette abgelegt.

Der AMIGA verwaltet seinen gesamten Bildschirmaufbau in Datenstrukturen, also Feldern aus ein paar Bytes, die je nach Funktion Namen besitzen. Da gibt es zum Beispiel eine Struktur, die für unser aktuelles Basic-Ausgabe-Window zuständig ist. Die Startadresse dieses Datenblocks, des sogenannten RASTPORTs, erhält man mit der Funktion WINDOW(8). Im Rastport stehen Dinge wie Linienmuster, Text-darstellungsarten etc. und eben auch ein Zeiger auf die BITMAP. Den Zeiger errechnet man mit PEEKL (WINDOW(8)+4).

Addiert man zu diesem Zeiger noch 8, dann kann man dort die Startadresse der ersten Bitplane herauslesen (siehe Programm 1,2).

Hurra! Das Ziel unserer 'Schnipseljagd' im Speicher des AMIGA ist erreicht. Die Zeiger auf die anderen 1 bis 4 Planes folgen in 4 Byte-Abständen. Das schnelle Laden und Speichern übernimmt eine DOS-Routine, die wir

über den LIBRARY-Befehl anzapfen. Um so erfolgreich auf der Diskette walten zu können, muß allerdings das Datenfile auch mit einer DOS-Routine geöffnet und geschlossen werden. Kein Problem, hier die Syntax:

```
Öffnen:
handle&=xOpen& (SADD (Filename$), Modus)
handle& wird bei den folgengen Operationen gebraucht.
Filename$ ist der Filename+chr$(0), z.B. "HUGO.PIC"+CHR$(0)
Lesen:
Erg&=xRead& (handle&, Start&, Länge&)
handle&
Start&
Länge&
         gibt an, wieviel Bytes eingelesen werden.
Schreiben:
Erg&=xWrite(handle&, Start&, Länge&)
Schließen:
xClose (handle&)
handle& ist der Wert der xOpen&-
```

```
1: REM **** Fast-Save ****

2: REM by Simon Wiest ILYM&I

3: REM - KICKSTART -

4:

5: REM Dieses Programm in LoadILBM-SaveACBM

6: REM ab dem Label 'SaveACBM:' einfuegen

7:

8: DIM planes&(5)

9: Laenge&=(scrWidth%*scrHeight%)/8

10: Farben&=2^scrDepth%

11:

12: Rastport&=WINDOW(8) 'Adressen fuer
```

```
13: Bitmap&=PEEKL(Rastport&+4)
                                       ' Bilddaten
                                         errechnen
14: FOR T=1 TO scrDepth%
15:
     planes&(T)=PEEKL(Bitmap&+4+4*T)
16: NEXT T
17:
18: Windowx&=WINDOW(7)
                                       ' Adressen fuer
19: Screenx&=PEEKL (Windowx&+46)
                                       ' Farbdaten errechnen
20: Viewport&=PEEKL(Screenx&+48)
21: Colormap&=PEEKL(Viewport&+4)
22: Farben$=ACBMname$+".Farben"+CHR$(0)
23: ACBMname$=ACBMname$+CHR$(0)
24:
25: WHILE INKEY$="":WEND
                                       ' Tastendruck
26:
27: REM Teil 1: Bilddaten
28:
29: handle&=XOpen& (SADD (ACBMname$), 1006)
30:
31: FOR T=1 TO scrDepth%
     Erg&=xWrite&(handle&,planes&(T),Laenge&)
32:
33: NEXT T
34:
35: CALL xClose(handle&)
36:
37: REM Teil 2: Die Farben
38:
39: handle&=XOpen& (SADD (Farben$), 1006)
40:
41: Erg&=xWrite&(handle&,Colormap&,Farben&*2)
42 .
43: CALL xClose& (handle&)
44 .
45: RETURN
46: REM **
```

Wie die Funktionen im Programmtext angegeben werden, ist aus den Programmbeispielen ersichtlich. Wichtig ist, daß die angegebenen Datentypen (z.B. '&' für Long-Integer) immer angegeben werden. Wenn man also die Startadressen der Bitplanes errechnet hat, kann es losgehen:

Eine Bitplane ist (Breite*Höhe)/8 Bytes groß, also bei einer Auflösung von 320*200 z.B. 8000 Bytes.

Als erstes wird das Bilddatenfile geöffnet, dann Plane für Plane abgespeichert und danach wieder geschlossen. Diese Arbeit macht der erste Teil des Programms 1.

Mit den Farben geht es ähnlich, es existiert ein Datenfeld namens COLOR-MAP, in dem die Farbpalette gespeichert ist. Um an sie zu gelangen, muß man über die Struktur WINDOW (mit WINDOW(7) errechenbar), dann über **SCREEN** (PEEKL (WINDOW (7)+46)) zum VIEWPORT (PEEKL (WINDOW(7)+46)+44) und von dort dann zur COLORMAP gehen, die dann bei PEEKL(PEEKL(WIN-DOW(7)+46)+44+4) beginnt. Diese Adresse ist die des ersten Farbtopfs. Die weiteren Farb-Register folgen in 2 Byte-Abständen.

So, jetzt öffnet man wieder eine Datei, schreibt die Farbwerte (Start& ist COLORMAP,Länge& ist Farben*2) und schließt sie wieder.

Jetzt besitzt man eine Bild- und eine Farbdatei. Um das Bild vom eigenen BASIC-Programm auf den Bildschirm zu bringen, muß man nur noch einen Screen einrichten, Adressen ermitteln, Bildschirmspeicher laden, Palette laden, fertig! Programm 2 zeigt eine Möglichkeit.

Und nun viel Spaß beim Titelbildmalen und Ausprobieren...

```
1: REM **** Bildlader *****
    REM von Simon Wiest ILYM&I
 2:
 3.
    REM
            - KICKSTART -
 4:
 5:
    CHDIR "df0:"
 6:
7: LIBRARY "graphics.library"
 8:
    LIBRARY "dos.library"
 9:
                                                'DOS-Routinen
10: DECLARE FUNCTION xOpen& LIBRARY
11: DECLARE FUNCTION xRead& LIBRARY
12:
13: DIM planes& (5)
14:
15: Tiefe=5
                                                '32 Farben
16:
     Breite=320
                                                '320 Spalten
17: Hoehe=200
                                                '200 Zeilen
18: Modus=1
                                                'LOW-Res, No Interlace
19.
20: Laenge&=(Breite*Hoehe)/8
                                                'Groeße einer Bitplane
                                                'Anzahl der Farbtöpfe
21:
     farben&=2^Tiefe
22: daten$="df0:IFF/KingTut.BPLS"+CHR$(0)
                                                'Bilddaten-Datei
23: farben$="df0:IFF/KingTut.BPLS.Farben"+CHR$(0) `Farb-Datei
24:
25: SCREEN 2, Breite, Hoehe, Tiefe, Modus
                                                'Screen vorbereiten
```

```
26: WINDOW 2,,,6,2
27:
28: Rastport=WINDOW(8)
                                                'Verschiedenes fuer
29:
    Bitmap=PEEKL (Rastport+4)
                                                'Bilddaten errechnen
30: FOR T=1 TO Tiefe
31:
     planes&(T)=PEEKL(Bitmap+4+4*T)
32:
    NEXT T
33:
34:
    Windowx&=WINDOW(7)
                                                'Verschiedenes fuer
35:
    Screenx&=PEEKL(Windowx&+46)
                                                'Farbdaten errechnen
36: Viewport&=Screenx&+44
37:
    FOR T=0 TO farben&-1
38:
                                                 'Verdunkeln
39:
      PALETTE T, 0, 0, 0
40: NEXT T
41:
    Handle&=xOpen&(SADD(daten$),1005)
42:
                                                 'Bild oeffnen
    FOR T=1 TO Tiefe
                                                 'Tiefe-mal..
43:
     Erg&=xRead&(Handle&,planes&(T),Laenge&)
44:
                                                 `...eine Bitplane laden
45:
    NEXT T
46:
    CALL xClose (Handle&)
                                                'Bilddatei schließen
47:
48: F$=STRING$ (66, 33)
49:
   Handle&=xOpen&(SADD(farben$),1005)
50:
                                                 'Farbdatei oeffnen
51:
     Erg&=xRead&(Handle&, SADD(F$), farben&*2)
                                                 'Farben lesen
    CALL xClose (Handle&)
                                                'Farbdatei schließen
52:
53:
    CALL LoadRGB4& (Viewport&, SADD (F$), farben&) 'Neue Farben
54:
55:
    WHILE INKEY$="":WEND
                                                 'Auf Taste warten
56:
57:
     WINDOW CLOSE 2
58:
                                                 'Finito !
59: SCREEN CLOSE 2
60: END
61:
62:
```

WE WANT YOU!

Wie schon auf der Titelseite erwähnt, wird diese Rubrik zu einem ständigen und wichtigen Bestandteil dieser Zeitschrift. Gedacht ist sie für all diejenigen, die ihren Rechner selbst programmieren, und dazu Tips und Anregung gebrauchen können. Diese Rubrik kann aber nur dann bestehen, solange viele Leser (Sie eingeschlossen) sich daran beteiligen. Wir fordern Sie deshalb auf, Ihre Ideen in einen Umschlag zu stecken und auf dem schnellsten Weg zu uns zu schicken. Vorraussetzungen gibt es praktisch keine: es muß lediglich interessant sein und so dokumentiert, daß auch andere User daraus einen Nutzen ziehen können. Auch bei den Sprachen gibt es keinerlei Einschränkungen. Ob C, BASIC, PASCAL, Assembler, Modula-2, Fortran, Forth, LISP oder andere Sprachen bleibt Ihnen überlassen. Veröffentlichte Programme werden natür-lich angemessen honoriert.

Bitte beachten Sie bei der Einsendung: Schicken Sie den Quelltext und das evtl. kompilierte Programm sowie die Dokumentation auf Papier und Diskette. Die Diskette wird zurückerstattet. Einsendungen direkt an die Redaktion:

MERLIN-Computer GmbH, Redaktion KICKSTART 'KICKS' Industriestr. 26, D-6236 Eschborn, Tel: 06196 / 481811

AZTEC-C FÜR AMIGA **VERSION 3.6**

Wußten Sie, daß eines der verbreitesten und komplexesten Betriebs-

systeme - UNIX - in C geschrieben ist? Wußten Sie, daß auch das Betriebssystem des AMIGA größtenteils in C geschrieben wurde? SOFTWARE DEVELOPMENT SERV Wußten Sie, daß C eine der wichtigsten und modernsten Programmiersprachen ist? Wußten Sie, daß Aztec-C einer der schnellsten und leistungsfähigsten Compiler für den AMIGA ist? Wußten Sie, daß jetzt Aztec-C in der Version 3.6 verfügbar ist? Möchten Sie mehr darüber wissen? Dann schicken Sie uns einen ausreichend frankierten Rückumschlag und Sie erhalten ausführliche Information. Up-Date-Service für alle MANX-Kunden auch bei uns. Fragen Sie nach!

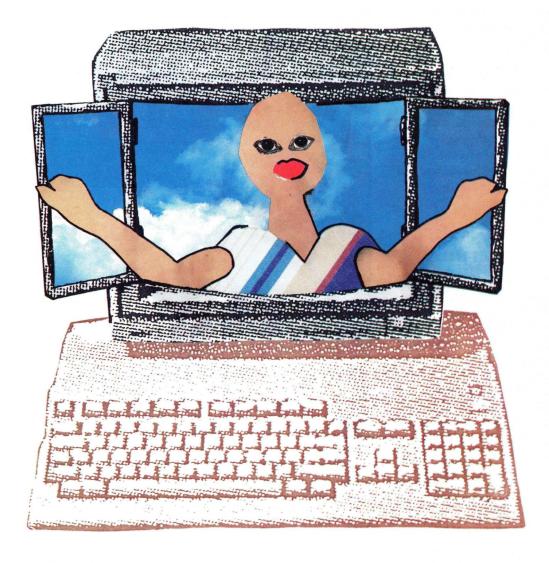
HIERMIT BESTELLE ICH:		
☐ AZTEC-C68K/AM-P PROFESSIONAL SYSTEM ☐ AZTEC-C68K/AM-D	FÜR DM 398	NAME:
DEVELOPER SYSTEM AZTEC-SDB SOURCE LEVEL DEBUGGER	FÜR DM 598 FÜR DM 149	VORNAME:
Versandkosten: Inland DM 7,50 Ausland DM 10,- Auslandbestellungen nur gegen Vorauskasse Nachnahmegebühr DM 3,70		ORT:
☐ Vorauskasse ☐ Nachnahme		UNTERSCHRIFT:



MERLIN COMPUTER GMBH INDUSTRIESTRAßE 26 6236 ESCHBORN TEL. 06196/481811

AZTEC-C IST EIN EINGETRAGENES WARENZEICHEN VON MANX SOFTWARE SYSTEMS

AUFGERISSEN!



An dieser Stelle wird ein interessantes Utility-Programm vorgestellt. Es kann zum einen als CLI-Befehl genutzt und zum anderen in eigenen Programmen leicht eingebunden werden. Das Programm bringt das aktuelle Fenster auf die maximale Größe, also über den ganzen Bildschirm.

ird das Programm als CLI-Befehl genutzt, muß nicht zur Maus gegriffen werden, sondern durch einfachen Aufruf des Befehls kann das aktuelle CLI-Fenster auf die maximale Größe gesetzt werfen. Natürlich kann der Befehl auch in der Startup-Sequence (Stapeldatei, auf die das System beim Booten zugreift) installiert werden. Das aktuelle CLI-Fenster wird daraufhin sofort vergrö-

ßert. Mit einigen Tricks können allerdings auch andere Fenster vergrößert werden. Dazu ist lediglich eine kleine Stapel-Datei vonnöten, die folgendermaßen arbeitet: Zunächst muß etwas Zeit mit dem WAIT-Befehl "geschunden" werden, danach folgt der Programmaufruf von 'BigCli' (Name des Programms, von der Redaktion gewählt, kann natürlich frei bestimmt werden). Nach dem Starten dieser Stapeldatei mit 'EXECUTE' kann mit der Maus ein anderes Fenster angewählt werden. Das neue aktuelle Fenster wird nach Ablauf der mit WAIT eingestellten Zeit automatisch größer.

Die zweite Möglichkeit, das Programm zu nutzen, ist für Programmierer gedacht. Dazu muß lediglich der Source von BigCli in das selbstgeschriebene Programm eingebunden werden. Dadurch ist es möglich, zu jeder Zeit das aktuelle Fenster auf die maximale Größe zu setzen. denn nicht immer sind die vordefinierten Werte des Fensters die momentan gewünschten.

Programmierung

Das Programm ist mit dem Aztec-C-Compiler erstellt. Die Anpassung an den Lattice-Compiler macht aber keine Schwierigkeiten. Ebenso ist das Programm extrem kurz, nach einigen Deklarationen wie dem Einbinden von verschiedenen 'include'-Dateien und Zuweisungen von zwei Strukturen mit Zeigern beginnt schon das Hauptprogramm. Die Intuition-Library wird geöffnet, da sie als einzige benötigt wird (SizeWindow()), des weiteren werden die Zeiger des aktuellen Bildschirms bzw. des Fensters ausgelesen, die Koordinaten gesetzt, mit der Funktion SizeWindow() die neue Größe definiert, und fertig ist das Programm. Die Compileroptionen beschränken sich auf ein Minimum:

```
cc Programmname.c
In Programmname.o -lc
```

```
(c) KICKSTART 1988
       Diese Funktion bringt das aktuelle Fenster auf
      die maximale Groesse
2
      MODIFICATION HISTORY:
                                         Beschreibung:
      16.01.1988 Andreas Trappmann
       AMIGA-Version 1.0
14
    #include <exec/types.h>
    #include <intuition/intuitionbase.h>
18
    #include <intuition/intuition.h>
                                   *OpenLibrary();
   struct IntuitionBase
                             *IntuitionBase
                                                = NULL;
   main()
       struct Screen
                          *screen;
       struct Window
                          *window;
       long dx, dy;
       IntuitionBase=(struct IntuitionBase*)
       OpenLibrary ("intuition.library", OL);
       if (IntuitionBase)
          /* Zeiger holen */
          screen = IntuitionBase->ActiveScreen;
          window = IntuitionBase->ActiveWindow;
          /* linke Window-Ecke auf Koordinaten (0/0)
          legen */-
          dx = window->LeftEdge;
          dy = window->TopEdge;
          MoveWindow (window, -dx, -dy);
43
          /* Window auf max. Size */
          dx = screen->Width - window->Width;
          dy = screen->Height - window->Height;
46
          SizeWindow (window, dx, dy);
47
48
          CloseLibrary (IntuitionBase);
49
```

SIDEWINDER

Ein Luxusballerspiel

Das gesamte Universum ist 'mal wieder bedroht. Ein Sternenkreuzer mit dem bezeichnenden Namen Starkiller rast auf die Sonne zu, um sie zu zerstören. Sie müssen als Pilot eines ultramodernen Raumschiffs in den riesigen Sternenkreuzer fliegen und ordentlich aufräumen. Wenn Sie in den fünften Abschnitt des Sternenkreuzers geflogen sind, können sie die Zentraleinheit des gigantischen Raumschiffs einäschern. Nur so kann das Unheil vereitelt werden. Eine solche Hintergrundstory nach dem Muster "Held muß Bedrohung von der Erde oder dem Universum abwehren" ist um nahezu jedes Ballerspiel gestrickt.



Den Fighter muß der Spieler schon recht geschickt steuern, um heil den nächsten Level zu erreichen.

wirken durch die perfekte Grafik realistisch. Der Sound ist ebenso großartig: Titelmelodie und Soundeffekte während des Spiels klingen beeindruckend. Spaß hat man durch die technische Brillianz und die fünf verschiedenen Schwierigkeitsgrade für lange Zeit. Sidewinder ist ein Spiel ohne Mängel, und so etwas gibt es selten: Ein Superhit für Ballerfans!!!

(Carsten Borgmeier)

ei SIDEWINDER von MA-STERTRONIC ist die Hineigentlich tergrundstory überflüssig, denn der Spieler ist auch ohne sie bestens motiviert. SIDEWIN-DER ist nämlich um ein Ballerspiel mit Effekten und Gameplay vom Feinsten. Man fliegt mit seinem Raumschiff bei vertikalem Scrolling durch den Sternenkreuzer. Eine Reihe von Geschützstationen feuern rote Kugeln unentwegt auf das eigene Raumschiff. Mit dem Daumen auf dem Feuerknopf erwehrt man sich seiner Heldenhaut. Man muß die Geschützstationen mehrmals treffen, dann kommt es zu einer grafisch sehr eindrucksvoll in Szene gesetzten Explosion. Die Geschütztürme sind

aber nicht die einzigen Gefahren. Es gibt unzählige Jagdschiffe, die sich auf das Raumschiff stürzen. Man hat alle Hände voll zu tun, um allen Feinden auzuweichen und um sich den Weg freizuschießen.

Sidewinder besteht aus fünf verschiedenen Leveln, die alle andere Schikanen bereithalten und grafisch komplett verschieden gestaltet sind. Am Ende jeden Levels fliegt man in einen kleinen Gang, der Amiga lädt den nächsten Level, und weiter geht es mit purer Action.

Technisch ist Sidewinder brilliant. Vertikales und horizontales Scrolling sind butterweich. Die Gegner sind herrlich animiert, und die Explosionen

Über den Wolken

Der JET Kampfflugsimulators

ehmen Sie Platz im Cockpit eines F-16 oder F-18 Kampfflugzeugs. schnallen und das Rauchen bitte einstellen, denn hier geht es heiß her. Doch zunächst einmal Grundlegendes zum Programm selbst: JET ist keine Neuentwicklung, das Programm existierte für andere Rechner schon länger, wurde aber erst jetzt für den AMI-GA umgeschrieben. Dabei wurde die Software nicht, wie sonst oft üblich, einfach für den jeweiligen Rechner adaptiert, sondern hier wurde einiges verändert und verbessert, um den vielfältigen Möglichkeiten des AMIGA gerecht zu werden. Dabei wurde trotz der Implementierung vieler nützlicher Optionen auf einen kurzen kompakten Programmcode geachtet, so daß der JET mit 512 Kbyte Speicher auskommt und somit auf jedem AMIGA lauffähig sein dürfte. Auch ergaben sich keinerlei Schwierigkeiten mit eventuell angeschlossenen Speichererweiterungen.

Doch nun zur Programmbeschreibung und dem, was das Programm für den AMIGA so interessant macht. Unter der Ausnutzung von Intuition können mehrere Windows gleichzeitig auf den Screen gebracht werden, um z.B. verschiedene Beobachtungsstandpunkte einzunehmen oder sich einen Überblick über die gesamten Vorgänge während eines Flugs zu verschaffen. Dazu gehören ein zweites 3D-Window, das verschiedene Standpunkte, wie Cockpit, Kontrollturm oder die Sicht aus einem hinter- bzw. nebenherfliegenden Flugzeug heraus, annehmen kann, außerdem Fenster für Radar, Landkarte und Raketenauge. Eine Zoom-Funktion, kann dabei auf



allen Fenster angewendet werden. Selbst erstellte und häufig gebrauchte Bildschirmeinstellungen können abgelegt und per Tastendruck aktiviert werden. Dabei handelt es sich - je nach Flugzeugtyp F-16 oder F-18 - um Luftbzw. Boden-Kampf sowie freien Flug. Für jeden Modus werden unterschiedliche Landschaften und Szenarien bereitgestellt. Wem das zu wenig Abwechselung bedeutet, kann mit den Scenery Disks des Flight-Simulators II von Sublogic noch andere Landschaften und Flughäfen einladen. Voraussetzung ist allerdings die Kenntnis der genauen Nord- und Ost-Koordinaten der einzelnen Flughäfen, will man nicht irgendwo im "Grünen" herumirren. Sind die Koordinaten bekannt, so ist es kein Problem mehr, mit dem JET auch in New York, Miami, Dallas oder San Francisco herumzufliegen.

Der Modus "Combined Attack" dürfte nicht nur die interessanteste und abwechslungsreichste Landschaft aufweisen, sondern auch vom Spielvergnügen her den höchsten Ansprüchen gerecht werden. Ist ein Modus ausgewählt, gelangt man in das zweite Menü, in dem der Schwierigkeitsgrad festgelegt wird. 0 steht für Practice und erlaubt das Fliegen im Übungsmodus,

d.h. eventuelle Abstürze werden ignoriert. Gegner sind zwar vorhanden, schießen aber in diesem Modus nicht. Der Practice-Modus dient nicht nur dem Üben und Kennenlernen der Landschaft, sondern auch dem Erlernen der umfangreichen Tastaturbefehle und ihrer Auswirkungen, die gerade am Anfang etwas verwirrend sein können. Mit Level eins (leicht) steigen Sie dann schon richtig ins Kampfgeschehen ein. Hier sind die Gegner zwar noch recht schwach, und man selbst muß schon einge Treffer erhalten, um abzustürzen. Dies ändert sich allerdings von Level zu Level recht schnell, und im neunten Level (schwierig) gilt es, höllisch aufzupassen, will man sich länger am Himmel halten. Das Instrumenten-Display wurde platzsparend um den eigentlichen Screen herumgelegt, ist aber trotzdem sehr Übersichtlich und gut gestaltet. Da das Programm keine Funkfeuer simuliert, konnten die dafür nötige Instrumente eingespart werden. Der künstliche Horizont wurde auf das Cockpit-Fenster gelegt, und die anderen Anzeigen für Treibstoff, Höhe, Steigwinkel, Geschwingigkeit, Schub, Kompass, Uhrzeit, Level, Score und die verschiedenen Warnlampen für Fahrwerk, Crash Down, Raketen- und Feind-Alarm wurden so gestaltet, daß fast der ganze Bildschirm für Beobachtungszwecke nutzbar ist. Eine recht nützliche Angelegenheit sind die ADF- und DME-Instrumente; sie geben ständig die Richtung und Entfernung zur Landebahn an. Das Visier besteht aus einem weißen Kreis, der gleichzeitig als Entfernungsmesser fungiert. Die Entfernung zu dem jeweils gerade anvisierten Punkt wird gemessen, und je nach Reichweite der verschiedenen Waffen färbt sich der Kreis dann teilweise rot ein und signalisiert damit, daß das Ziel innerhalb der Reichweite dieser Waffe liegt. Bleibt der Indikator weiß, ist die Entfernung zum angestrebten Ziel zu

Ähnlich verhält es sich mit der Anzeige für die Raketen. Sie erscheint in Form von eckigen Klammern und ist ständiger Begleiter gegnerischer Flugzeuge. Fliegt man nun genau auf einen Gegner zu, so wechselt die Farbe von weiß auf schwarz, und erst jetzt sollte man seine Rakete abfeuern. Ist dies geglückt, wird der Indikator rot, und



das gegnerische Flugzeug wird nun, unabhängig von eventuellen Ausweichmanövern, von einer Rakete verfolgt. Anfliegende gegnerische Raketen verhalten sich ebenso. Ihre Spur sollte mit dem Radar gut verfolgt werden, um durch geschicktes Ausmanövrieren die Rakete von der Verfolgung abzubringen.

Die Steuerung des Flugzeugs erfolgt wahlweise über Maus, Tastatur oder einen an Port 2 angeschlossenen Joystick. Da kein bestimmter Modus festgelegt werden muß, sind alle Eingabegeräte gleichzeitig nutzbar. Bei Steuerung mittels Joystick muß die Geschwindigkeit allerdings über Tastatur oder Maus eingestellt werden.

Angenehme Steuerung

Dank eines sehr guten Flugverhaltens, es gleicht dem einer steuerbaren Rakete, lassen sich die Maschinen, bei entsprechendem Schub, sehr einfach steuern. So können extreme Kurven geflogen werden, ohne ein Absinken der Maschine zu verursachen. Dies gibt dem Benutzer vermehrt die Möglichkeit, die vielen Fenster und verschiedenen Beobachtungsstandpunkte zu nutzen, zumal auch beim Betrieb von mehreren Fenstern gleichzeitig, eine Verringerung in der Geschwindigkeit der Simulation kaum warzunehmen ist. In wieweit dies jedoch aus Gründen der Übersichtlichkeit sinnvoll ist, bleibt jedem selbst überlassen. Auch besteht die Möglichkeit, bestimmte Bodenlinien aus der Darstellung herauszunehmen, um die Simulation zu beschleunigen.

Der 'Combined Attack-Modus' dürfte wohl die interessanteste und abwechslungsreichste Landschaft bieten. Zwischen einigen dreidimensionalen Pyramiden können das Flugverhalten des JET und die exzellente Steuerung ausprobiert und genossen werden. Dazu fliegen Sie am Anfang am besten ungestört im Übungsmodus, denn diese Landschaft hat wirklich einiges zu bieten. Neben den schon angesprochenen Pyramiden befinden sich noch eine ganze Reihe von Bodenzielen, wie Raketensilos, Hafenanlagen, Brücken, Straßen, Stromleitungen, Umspannwerke sowie ein Kernkraftwerk in der Szenerie. Das Meer, ein Fluß, ein weiterer Flughafen und ein kleiner Wald runden dann das eindrucksvolle und recht realistische Landschaftsbild ab. Die Grafiken entsprechen denen des Flightsimulators II von Sublogic und geben somit keinen Anlaß zur Kritik. Wird im Combined-Attack-Modus ein Level (1-9) gewählt, so müssen, um im Schwierigkeitsgrad zu steigen, alle primären Bodenziele zerstört werden. Das Unternehmen wird immer wieder von Wellen angreifender gegnerischer Kampfflugzeuge gestört, die es möglichst schnell zu vernichten gilt, um sich dann wieder den anderen Zielen zuwenden zu können. Hat man im Laufe der Mission seine Waffen verschossen oder wird der Treibstoff knapp, muß man auf der Heimat-Basis landen, um sich mit Sprit und neuen Waffen zu versorgen. Doch Vorsicht, gegnerische Jets nehmen die Verfolgung auf und beschießen Sie noch in unmittelbarer Nähe der eigenen Landebahn. Mit steigendem Schwierigkeitsgrad werden die feindlichen Flugzeuge immer aggressiver, und das eigene Flugzeug kann mit immer weniger Treffern abgeschossen werden. Am Anfang des Spiels stehen dem Piloten vier Flugzeuge zur Verfügung, und nach jeweils 10.000 Punkten erhält er einen weiteren Jet. Wird der eigene Jet einmal abgeschossen, dann blinkt der ganze Bildschirm, begleitet von einem Warnton, der darauf hinweist, jetzt sofort mit dem Schleudersitz auszusteigen. Gelingt dies innerhalb einer gewissen Zeit nicht, so ist die Mission, egal wieviel Flugzeuge noch übrig sind, beendet, und es gilt, wieder von vorne zu beginnen. Doch nicht nur die Flugzeuge stellen eine massive Bedrohung dar, Gefahr droht auch von den Raketenbasen am Boden. Aber warum immer gegen den Computer kämpfen? Spielen Sie zur Abwechslung

einmal gegen einen Freund oder Bekannten. Dazu sind allerdings zwei Rechner nötig. Die Daten können entweder über ein Modem oder über ein direktes Kabel übertragen werden. Letztere Möglichkeit setzt aber ein Nebeneinanderstellen der Rechner voraus. Das nötige Kabel besteht nur aus drei Leitungen sowie den Steckern für den seriellen Port und kann selbst leicht und ohne große Kosten hergestellt werden. So ausgestattet, kann nun mit zwei Maschinen gegeneinander geflogen werden, womit die Flugsimulation noch einen besonderen Reiz erhält. Schade eigentlich, in diesem Modus nur ein bestimmtes Spielfeld zur Verfügung steht und nicht auch die anderen Landschaften in diesen Spielmodus einbezogen werden können. Eine genaue Anleitung zum Verbinden der beiden Rechner befindet sich im Handbuch, das in englischer Sprache verfaßt ist. Fazit: Eine gut gelungene Simulation. Der JET ist somit ein Muß für jeden, den die Fliegerei und ihre Umsetzung auf den Computer begeistert. An der Tatsache, daß z.B. z.B. Fluglinien einen Großteil ihrer Piloten auf Simula-

rechtigung.
Diese Systeme Übertreffen mit ihren Fähigkeiten die eines AMIGA zwar bei weitem, trotzdem verdient auch der JET das Prädikat Flugsimulator.

toren schulen, erkennt man ihre Be-

Leider wurde bei JET der kriegerische Aspekt wieder einmal viel zu groß geschrieben. Das Abschießen von Schiffen, Bodenzielen, etc. und vor allem der Angriff auf ein Atomkraftwerk ist äußerst geschmackslos und kann auch nicht gerade als neu bezeichnet werden. In Sachen Fliegerei gibt es sicherlich noch viele andere interessante Situationen, bei denen man nichts abschießen, nichts zerstören und mal zur Abwechslung niemanden töten muß. Die JET-Programmierer hatten jedenfalls keine neuen Ideen, vielleicht das nächste Mal. (Jürgen Schmidt)

Simulation & Grafik:

*

Idee & Handlung:

* Kriegshandlung führt zur Abwertung

KNISTERNDE EROTIK BEIM POKERN

STRIP POKER II Plus

uch von ANCO gibt es jetzt ein Pokerspiel dessen Reiz nicht nur in dem Spiel an sich liegt. Interessanter erscheint vielmehr die Tatsache, daß der Pokerfan zu Beginn des Spiels zwischen zwei nicht weniger leidenschaftlichen Pokerpartnerinnen wählen darf. Donna und Sam heißen die beiden Hübschen. Nach der Qual der Wahl geht es mit dem Pokern auch schon los. Sam liegt mit einem weißen Pullover vor den Augen des Spielers. Der Grundeinsatz ist 100 DM. Die Karten werden gemischt, und das eigene Blatt erscheint auf dem Bildschirm. Man wettet nun auf sein Blatt, macht Einsätze bis zu 25 Mark, läßt sich neue Karten hinzugeben oder gibt in einer ausweglosen Situation auf. Im Endeffekt muß man beim "Zeigen" ein besseres Blatt aufweisen als die Gegnerin.

Wenn man seine 100 DM verspielt hat, wird es aber erst richtig lustig. Um weiterspielen zu können, muß man ein Kleidungsstück verkaufen und erhält dafür 100 DM. Wenn Sam ihren Pullover verspielt hat zeigt sie sich in Spitzenunterwäsche, die sie dann nach und nach ablegt. Wie es zur überaus hohen Motivation kommt, braucht wohl nicht weiter erklärt zu werden. Man möchte Sam schließlich in natura sehen. Dabei sollte man aber nicht allzu risikofreudig spielen, denn sonst muß man selber



Die Damen sind durchweg sehr gut digitalisiert.

seine Kleidung verkaufen. Das Programm gibt einem genaue Anweisungen, wenn es Zeit ist, seine Schuhe oder ähnliches zu opfern. Ob man den Anweisungen auch Folge leistet, kann der Computer glücklicherweise nicht kontrollieren.

STRIP POKER II PLUS ist ein unterhaltsames Spiel, das durch seine exzellenten Grafiken glänzt. Alle Bilder sind digitalisiert und von einem Grafiker nachgearbeitet worden. Die Resultate sind beeindruckend. Spaß hat man beim Spielen eine ganze Menge, besonders wenn man im Kreise mehrerer Pokerspieler versucht gegen Sam und Donna anzutreten.

Die Motivation ist solange hoch, bis man es geschafft hat Sam und Donna "auszuziehen". Dann wird das Spiel eigentlich uninteressant. Alles in allem ist Strip Poker II Plus ein gut gelungenes Computerspiel mit einer pfiffigen Idee, die zwar nicht neu ist (siehe Hollywood Poker), trotzdem aber für eine Menge Spielspaß sorgt.

Eltern und Moralapostel können beruhigt sein. Die Grafiken von Strip Poker II Plus sind zwar erotisch, aber nicht pornographisch. Ein Spiel mit einem Hauch Erotik ist doch immer noch besser als ein brutales Kriegsspiel, oder?

(Carsten Borgmeier)

BEYOND ZORK

ndlich! Die Zork-Trilogie ist fortgesetzt worden. Beyond Zork ist der lange erwartete Nachfolger von Zork I, Zork II und Zork III, den wohl bekanntesten Adventures der Welt. Diese Fortsetzung wurde von Brian Moriarty geschrieben der durch Wishbringer, vor allem aber durch Trinity bekannt wurde.

Allerdings hat Beyond Zork nicht allzuviel mit seinen Vorgängern zu tun. Es gibt keine unterirdischen Höhlen, auch muß man keine Schätze um ihrer selbst willen sammeln oder irgendwelche Zauberer überwinden. Ziel des Spiels ist es vielmehr, eine Kokosnuß zu finden. Aber nicht irgendeine Kokosnuß, sondern die Kokosnuß von Quendor (Quendor heißt das Land, in dem die ganze Geschichte spielt). In diese soll das gesammelte Wissen der großen Zauberer eingeschlossen werden, um es der Nachwelt zu erhalten, denn das große Zeitalter der Legenden, Zauberer und Abenteuer neigt sich seinem Ende zu. Das einst blühende Land Quendor ist jetzt größtenteils verlassen, die guten Zauberer verlieren langsam ihre Macht, und böse Wesen machen sich breit. Die Aufgabe des Spielers, also Ihre, ist es, diese Kokosnuß zu finden. Es gibt da aber dummerweise ein paar Typen, die gerade das verhindern wollen.

Bei Beyond Zork hat sich im Gegensatz zu den anderen Infocom-Adventures einiges geändert. Zum ersten gibt es Grafik. Insgesamt ist es eine, nämlich ein Titelbild.

Auch gibt es einen Runen-Zeichensatz (keine Angst, nur die Namen der Zaubersprüche, die man nicht verwenden kann, stehen in Runen!). Des weiteren wird endlich von der Window-Technik Gebrauch gemacht. Im Fenster kann man sich entweder die Beschreibung des aktuellen Raumes, die Gegenstände, die man bei sich trägt oder

den Zustand des Spielers anzeigen lassen. Womit schon die nächste Neuerung erwähnt wurde. Am Anfang des Spiels kann man sich seinen eigenen Charakter zusammenstellen. Dieser besteht aus Ausdauer, Stärke, Beweglichkeit, Intelligenz, Mitgefühl und Glück. Auf diese Eigenschaften kann man ein bestimmtes Anfangspotential verteilen und sich so einen möglichst günstigen Charakter zusammenzustellen. Die Eigenschaften verändern sich im Laufe des Spiels. Gewinnt man zum Beispiel einen Kampf mit einem anderen Wesen, so steigt die Ausdauer. Außerdem kann man bestimmte Zaubersprüche, die man auf den aus der Enchanter Trilogie bekannten Schriftrollen findet, erst ab einer bestimten Intelligenzstufe verwenden, oder man benötigt, um irgendwelche wackeligen Türme zu besteigen, eine bestimmte Beweglichkeit. Aber keine Angst, im Verlauf des Spiels gibt es genügend Möglichkeiten, seine Eigenschaften durch Zaubertränke oder bestimmte Aktionen zu verbessern.

Eine weitere Neuerung betrifft all diejenigen, die es schon immer geliebt haben, große, umfangreiche, komlizierte und auf mehrere Ebenen verteilte Karten zu zeichnen. Dies wird bei Beyond Zork von selbst erledigt, und zwar in der rechten oberen Ecke. Dort wird automatisch mitkartographiert, wenn der Spieler sich von einem Raum zu einem neuen bewegt. Als letzte Neuerung lassen sich auch noch die Funktionstasten mit beliebigen Befehlen definieren.

Im Gegensatz zu anderen Infocom-Adventures muß man hier mit anderen Charakteren kämpfen und diese sogar töten. Die Chancen, solche Kämpfe zu überleben, hängen unter anderem von der Bewaffnung, den Fähigkeiten und der Kleidung (z.B. Rüstung oder Mantel) des Spielers ab.

Beyond Zork ist von der Schwierigkeit her sicher sehr nahe an den Expert Level zu legen. Wenn nicht, vielleicht sogar Beyond Expert Level? Beyond Zork ist auf jeden Fall eine Herausforderung für alle Adventure-Freaks und jedem zu empfehlen.

(Andreas Paul)

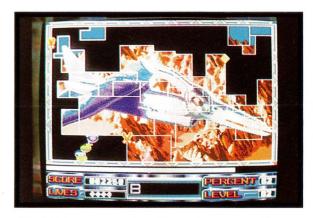
Zum Schluß ein paar, vor allem für Anfänger nützliche Tips:

- Zu allererst sollte man sich eine Waffe besorgen. Ein paar der Wesen lassen sich mit den Waffen, die man anfangs hat, übrigens nicht ohne weiteres töten. Hierfür braucht man zum Beispiel einen wirksamen Zauberstab.
- Hat man die Dienste des Pterodactyls erworben, sollte man sparsam mit ihnen umgehen.
- Man sollte sich auch nicht scheuen, zu versuchen, wirklich alles zu nehmen, was man sieht, selbst wenn es nur ein vager Umriß ist.
- Die Hinweise, die die Verkäuferinnen über die gefundenen Gegenstände geben können, sind oft sehr wertvoll.
- Intelligenz und Stärke sind relativ wichtige Eigenschaften. Entsprechende Zaubertränke sollte man nicht verkaufen.

- Beim Verkaufen von Gegenständen sollte man vorsichtig sein. Sie kosten beim Zurückkauf das Doppelte, was sie vorher eingebracht haben.
- Statische Elektrizität läßt sich mit Stäben und Fellen produzieren.
- Die Lösungen der Rätsel sind sehr wichtig. Das auf dem Felsen ist leicht, die Lösung des Rätsels auf der Klippe stammt aus dem naturwissenschaftlichen Bereich und hat etwas mit dem Wetter zu tun.
- Die Aufmerksamkeit sollte durch überdimensionale Stiefel nicht abgelenkt werden.
- Das Schließen von Türen ist eigentlich immer sinnvoll.
- Wenn Schlüssel angeboten werden, sollte man mehrere Möglichkeiten ausprobieren und ihre unterschiedliche Wirkung auf Krähen testen.

POWER-STYX

Die QUIX-Variante



Farbenfroh gibt sich Powerstyx auf dem AMIGA-Bildschirm.

eline hat mit POWER-STYX ein süchtig machendes Computerspiel geschaffen, das sich an den Klassiker Quix anlehnt. Als Spieler steuert man einen Punkt über ein rechteckiges Feld. Wenn man den Feuerknopf drückt, kann man Linien malen. Jedes Feld, das durch die gemalten Linien eingrenzt wird, zeigt einen Auschnitt ei-

nes Bildes. Ziel des Spiels ist es, auf diese Weise mindestens 75 % des Bildes offenzulegen. Das ganze verläuft natürlich nicht ohne Komplikationen. Einige Rechtecke, die über den Bildschirm huschen, versuchen, die Mühen des Spielers zu behindern. Außerdem gibt es im ersten Level zwei Totenköpfe, die Jagd auf das linienmalende Etwas machen. Hat man 75 % eines

Bildes freigelegt, so, darf man sich an einem neuen Bild versuchen. Da man natürlich alle Bilder ansehen möchte ist die Motivation dementsprechend hoch.

Soundliebhaber kommen allerdings nicht auf ihre Kosten: Eine kurze, digitalisierte Melodie wird während des ganzen Spiels gedudelt, so daß nach kurzer Zeit auch der Hartgesottenste den Lautstärkeregler am Monitor ganz leise stellt. Die Grafiken hingegen sind sehr gut gelungen. Farbenfroh und detailreich präsentieren sie sich dem Spieler.

Powerstyx ist ein unterhaltsames Spiel, das man ruhig einmal ausprobieren sollte.

Info: Ariolasoft

(Carsten Borgmeier)

SPINWORLD

Action ohne Satisfaction

n SPINWORLD von RELINE geht es heiß her. Man fliegt mit einem Raumschiff von links nach rechts, feindliche Geschwader bedrohen den Spieler. Was kann man dagegen tun? Richtig, man ballert und ballert, solange bis es keine Feinde mehr gibt.

Natürlich hat Spinworld auch eine Hin-

tergrundstory, aber mit der möchten ich Sie verschonen. Sie fliegen also von links nach rechts und drücken sich den Daumen wund, denn feindliche Aliens fliegen in Scharen durch den Weltraum. Bei ihren gezielten Vernichtungsmanövern müssen Sie darauf achten, daß die Biester nicht in Sie hineinfliegen und daß Sie nicht gegen Sa-

telliten, die friedlich im Weltraum ihre Runden drehen, scheppern, denn sonst macht es einmal "bumm", und Sie haben ein Raumschiff weniger.

Beim Ballern ist darauf zu achten, daß der Laser nicht überhitzt, denn sonst ist man dem Feind für einige Sekunden schutzlos ausgeliefert. Ganz lustig in Szene gesetzt ist der Pilot ihres Raum-



Auch der Titlescreen ist nicht gerade überwältigend.

schiffes. Ihn sieht man am unteren Bildschirmrand, wie er verbittert an seinem "Freudenknüppel" drückt. Grafisch ist Spinworld zufriedenstellend, aber nichts Außergewöhnliches. Spielefans, die momentan ein gutes Ballerspiel suchen, sollten sich Sidewinder zulegen, denn Spinworld bietet langfristig keine Motivation.

Info: Ariolasoft

Pink Panther

macht AMIGA unsicher!



Paulchen bei der nächtlichen Arbeit.

aulchen Panther dürfte nicht nur jüngeren Lesern ein Begriff sein. Das rosarote Tierchen trieb vor einigen Jahren in einer Zeichentrickserie sein Unwesen. Als Comicheld mußte Paulchen sich ebenfalls schon behaupten. Wo der Panther auch auftaucht, das Herz des Puplikums erobert er im Sturm. MA-GIC BYTES, das erfolgreiche Softwarehaus aus Gütersloh, hat die Lizenz von Paulchen für ein Computerspiel ersteigert. PINK PANTHER ist ein absoluter Superhit. Eine lustige Handlung, excellente Comicgrafik und ein pfiffiger Sound vermögen den Spieler zu unterhalten.

Paulchen ist mal wieder völlig blank. Ein Job als Hausdiener käme ihm da sehr gelegen. Tagsüber könnte er Staub wischen und seinem Chef die Pantoffeln herbeitragen, und nachts, wenn der reiche Knopf schläft, könnte er das Haus ausräumen. Dies bräuchte er nur fünfmal zu wiederholen, dann könnte er sich ein angenehmnes Leben im Süden leisten. Doch bevor seine Träume wahr werden, muß er erst zum Arbeitsamt und sich empfehlen lassen. "Für den Job brauchen Sie nur einen Zylinder, der Hausherr legt Wert auf Außerlichkeiten". Diese Szenen sind natürlich köstlich animiert. Wenn Paule stolzer Besitzer eines Zylinders ist, kann die Arbeit beginnen. Tagsüber wischt er Staub und nachts ... Oh nein! Der Hausherr ist Schlafwandler. Paulchen muß bei seiner "nächtlichen Räumungsaktion" darauf achten, daß der Gute nicht aufwacht.

Während Paul alle Wertgegenstände einsackt, muß er gleichzeitig darauf achten, daß der Hausherr nicht gegen Wände läuft oder am Teppichrand stolpert. Zu diesem Zwecke hat Paulchen ein kleines Glöckchen. Immer, wenn Paule bimmelt, läuft der Hausherr in

seine Richtung. Außerdem kann er Gegenstände wie kleine Brücken und Gummipanther aus seinen Taschen kramen und aufblasen, die sein Opfer auf den richtigen Weg bringen.

Als Spieler sitzt man schweißgebadet am Joystick, sammelt in dem geräumigen Haus, das sich über mehrere Etagen erstreckt, Wertgegenstände ein und sorgt panisch dafür, daß der Hausherr nicht aus seinem Schlaf erwacht. Ich halte PINK PANTHER für eins der besten Actionspiele, die jemals geschrieben wurden. Grafik, Sound, Idee, man hätte es nicht besser machen können. Da das Spiel auch noch ziemlich kniffelig ist, verspricht Pink Panther lange Zeit beste Unterhaltung.

Hersteller: Magic Bytes Info bei: Ariolasoft

(Carsten Borgmeier)

NEUESTE

Vom Spielemarkt

THE GER-MAN FOOT-BALL SIMU-LATION

Von EAS Software, dem Produzenten der schwachen Au-"Ralley torennsimulation Master", kommt ein Abklatsch des legendären Fußballmanagers. Als Manager deutschen Erstligamannschaft soll man mit seinem Team die deutsche Meisterschaft gewinnen. In den Spielerlisten jeder Mannschaft sind alle Namen aufgeführt, wie wir sie aus der Realität kennen. Burdenski spielt bei Werder Bremen, Pfaff bei Bayern München etc... Nicht nur in diesem Punkt ist "The German Football Simulation" realistisch. Der Manager kann die Eintrittspreise fürs nächste Spiel festlegen, gute Spieler einkaufen, schwache Spieler verkaufen. Natürlich hat der Manager auch die Möglichkeit, den Trainer zu wechseln oder im Toto zu wetten. Durch Gewinne im Toto und durch die Eintrittsgelder muß man die Gehälter der Spieler bezahlen und alle anfallenden Nebenkosten bestreiten. Man kann sich vor jedem Spiel Bilanzen und Statistiken ansehen, und wenn man unter Kapitalknappheit leidet, auch Kredite von der Bank aufnehmen. Alle Aktionen werden von einem Menü, das aus zwölf verschiedenen Feldern besteht, eingeleitet. Jedes Feld bedeutet eine andere Aktion. Fußballfans werden schon mit dem Gedanken spielen, sich das Spiel zu kaufen. Meiner Meinung nach lohnt sich eine Anschaffung jedoch nicht. Zu meiner Ansicht komme ich nach einem mehrstündigen Programmtest, bei dem sich folgende Mängel herauskristallisierten: 1. Nach jeder getätigten Aktion muß das Menü neu geladen werden. Man soll doch die Zeit mit Spielen verbringen und nicht mit warten.

2. Nach jeder Wahl wird das Menü in vier Teile zerrissen, die sich aus dem Bild bewegen. Beim ersten Mal ist dieser Effekt sehr eindrucksvoll, aber wenn man sich das nach jeder gewählten Aktion anschauen muß, reagiert man leicht genervt, weil auch hier ständige Wartezeiten auftreten.

3. Actionsequenzen fehlen völlig. Wenn man schon unbedingt beim Fußball-Manager abkupfern muß, sollte man es auch konsequent tun. Eine kurze Actionsequenz, in der das anstehende Match

ausgetragen wird, hätte mehr Spaß ins Spiel gebracht.

4. Das Programm verwaltet die Spielerlisten der einzelnen Teams zwar korrekt, wenn zwei Teams aber gegeneinander spielen (bei meinem Test Bayern München und Werder Bremen) und Bayern beispielsweise ein Tor erzielt, wird als Torschütze der Werderaner Ordenewitz ausgegeben. Das wäre nicht unlogisch, wenn es sich bei dem Tor um ein Eigentor handeln würde. Diese Verwechslungen kommen aber teilweise 2-3 mal pro Spiel vor. Hat man alle Spieler umgepolt?

Alles in allem ist "The German Football Simulation" ein Spiel, auf das man getrost verzichten kann.

Info: Leisuresoft

BATTLES-HIPS

Was, Sie spielen "Schiffe Versenken" noch mit Papier und Bleistift? Das brauchen Sie jetzt nicht mehr zu tun, denn jetzt gibt es BATTLES-HIPS von ELITE. Der Bleistift wurde durch die Maus ersetzt und das Papier durch den Amiga. Ein oder zwei Spieler müssen versuchen, sich gegenseitig ihre Schiffe

zu versenken. Spielt man zu zweit, muß der zweite Spieler wegschauen, wenn der Mitspieler seine Schiffe im Koordinatensytem plaziert. Nachdem auch Spieler 2 seine Schiffe plaziert hat, wählt jeder Spieler mit der Maus Punkte im Koordinatensystem aus, an denen er die Schiffe des Gegners vermutet. Dann kommt der Höhepunkt des Spiel. In einer kurzen animierten Sequenz wird von einem Kriegsschiff auf im Hintergrund befindliche Schiffe des Feindes geschossen. Hat man die Koordinaten eines Schiffes richtig erraten, wird ein Schiff getroffen, was zur Folge hat, daß es langsam sinkt.

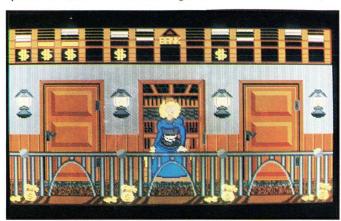
Grafik und der Sound mit digitalisierter, klar verständlicher Sprachausgabe sind vom Feinsten, leider wird das Spiel auf die Dauer langweilig, da sich allé Vorgänge ständig wiederholen.

Info: Profisoft, Leisuresoft

BRANDHEISS!

Kurz vor Redaktionschluß erreichte uns ein Bildschirmfoto eines der neuesten Spiele von RELINE. Den Titel konnten wir leider nicht mehr in Erfahrung bringen. Nur so

viel steht fest: Das Spiel orientiert sich an den Spielautomaten "West Bank" und spielt, wie man unschwer auf dem Foto erkennen kann, im Wilden Westen. Mehr darüber in einer der nächsten Ausgaben.



Freunde des Wilden-Westens kommen auf ihre Kosten.

BLASTER-BALL

BLASTERBALL heißt ein futuristisches Eishockey aus dem Hause MASTERTRO-NIC. Zwei Spieler können sich brandheiße Eishockevduelle liefern. Die Austragungsstätte des sportlichen Wettkampfs liegt aber nicht in einer Eissporthalle, das Szenario wurde in den Weltraum auf einen fernen Planeten verlegt. Die Eishockeyspieler wurde durch Raumschiffe ausgetauscht. Jeder Spieler versucht, mit gezielten Schüssen von seinem Raumschiff aus den Puck ins gegnerische Tor zu lenken.

Info: Leisuresoft

TRACERS

Tracers ist eine Tron-Variante, wobei der Spieler eine sich immer mehr verlängernde Schlange steuern muß. Ziel ist es dabei, die feindliche Schlange in der Bewegungsfreiheit einzuengen. Der Spieler sollte jedoch aufpassen, daß er sich keine eigene Falle baut. Die Grafik ist schlicht, jedoch für das Spiel ausreichent. Die Motivation von

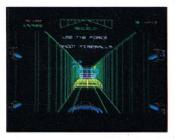
Tracers ist ausgesprochen hoch. Der Sound ist ausgezeichnet und gibt dem Spiel einen weiteren Pluspunkt.

STAR WARS

Der Angriff auf den Todesstern beginnt jetzt auch auf dem AMIGA. Science Fiktion-Fans ist dies sicherlich ein Begriff. Für Uneingeweihte: Das Spiel Star Wars ist an den gleichnamigen Film angelehnt. Der Spieler steuert einen Raumgleiter zunächst durch die Weiten des Weltalls und muß sich den Angriffen Untertanen Lord Vaders erwehren. Ist diese Hürde genommen, fliegt der Gleiter in einen Schacht des Todessterns, und man muß mit seiner Bordkanone in einen bestimmten Trichter treffen, um eine Kettenreaktion auszulösen. Ist auch diese Hürde genommen, so taucht ein neuer Todesstern auf, und das Spiel beginnt von neuem.

Star Wars stammt ursprünglich vom, Atari ST und kann nicht als gelungene Umsetzung bezeichnet werden. Die schlechte Vektorgrafik ist dem AMIGA nicht würdig. Der Sound beschränkt sich auf Baller- und Explosionsge-

räusche. Star Wars gehört mit Sicherheit nicht zu den besseren AMIGA-Spielen.



Vektorgrafik bei Star Wars.

STARWAYS

Von Diamond Games erreichte die Redaktion kurz vor Redaktionsschluß noch ein weiteres Spiel. Die Aufgabe des Spielers besteht darin, eine Spielfigur über einen in 3D-Scrolling gehaltenen Regenbogen mit verschiedenen Hindernisse unbeschadet zu manövrieren. Die Grafik ist schlicht, aber gut, der Sound kann als gut bezeichnet werden.



Gegen die Zeit und Hindernisse muß unser kleiner Held ankämpfen.

Seconds Out...

...ist eine Boxsimulation mit sehr großen Bobs. Der Spieler steuert einen Boxer, der in Rasterdarstellung auf dem Bildschirm zu sehen ist. Die Gegner werden mit der Zeit natürlich stärker, gemeiner und muskulöser. Die Grafik erscheint zum Teil etwas grob, kann aber trotzdem als recht ansprechend gewertet werden. Der Sound ist hingegen mager geraten. Tynesoft hat mit Seconds Out die erste reine Boxsimulation für den

AMIGA veröffentlicht und wird damit sicherlich recht erfolgreich sein.



Der erste Gegner ist noch leicht zu schlagen.

INFO-ADRESSEN:

Die unter den Tests und News aufgeführten Großhändler geben Ihnen Auskunft, wo Sie die getesteten Spiele beziehen können.

Ariolasoft GMBH Postfach 1350 4830 Gütersloh 1 Tel: 05241/803871

Leisuresoft Industriestr. 23 4709 Bergkamen 5 Tel:02389/6071

Profisoft Sutthauser Straße 50/52 4500 Osnabrück Tekl: 0541/53905

Rushware Microhandelsgesellschaft Bruchweg 128-132 4044 Kaarst 2 Tel:02101/6070

Jeden Monat erstellen wir aus den Zuschriften unserer Leser unsere TOP 12, die AMI-GA-Spiele-Hitparade.

Sie gibt den aktuellen Trend der Spiele-Szene wieder und zeigt, wie sich ältere Spiele gegen die neuen behaupten.

Auch diesmal ist Bewegung in die Hitparade geraten, neuer Spitzenreiter ist Ports Of Call, die komplexe Handelssimulation. Der Flightsimulator II konnte seine Führung nicht behaupten und ist auf den zweiten Platz zurückgefallen. Die nächsten Plazierungen sind alles alte Bekannte, bemerkenswert ist der Klassiker Defender Of The Crown (Platz 10), der sich hartnäckig gegen seine Kontrahenten behauptet. Mit Barbarian (12) geht es hingegen immer mehr bergab. Zunächst noch auf Platz 1 der Hitparade, dann auf den Platz verwiesen, ist das Spiel jetzt am Ende zu finden. Auffällig ist, daß sich keine reinen Arcade-Spiele in den TOP 12 befinden.

Um die Teilnahme schmackhaft zu machen, verlosen wir unter allen Einsendungen 12 mal das 'Spiel des Monats'.

Und so können Sie teilnehmen:

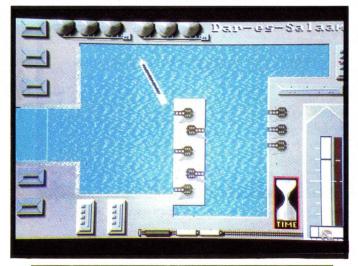
Sie füllen die 'TOP 12 Postkarte' (bitte keine andere) in der Heftmitte aus, kleben eine Briefmarke darauf (diese können wir leider nicht mitdrucken) und werfen sie in den nächstbesten Briefkasten.

Unter allen Teilnehmern (egal, was gewählt wurde) verlosen wir 12mal das 'Spiel des Monats'.Der Rechtsweg ist dabei ausgeschlossen. Einsendeschluß ist der 30. April 1988. Karten, die nach diesem Termin eintreffen, werden im nächsten Monat berücksichtigt. Wir gratulieren den Gewinnern dieses Monats

Je ein 'Ports Of Call' erhalten:

1.(11) Ports Of Call

Das Softwarehaus AEGIS hat mit Ports Of Call eine Handelssimulation der Extraklasse auf den Markt gebracht. Die ausgezeichnete Grafik und der komplexe Spielablauf fesseln den Spieler viele Stunden an den Rechner. Bei Ports Of Call gilt es, seine Handelsflotte gezielt einzusetzen, geschickt zu wirtschaften, bei günstigen Angeboten zuzugreifen und vieles mehr, bis man sich getrost im Sessel zurücklehnen kann.



- 2.(1) Flightsimulator II
- 3.(5) The Feary Tale Adventure
- 4.(10) Arkanoid
- 5.(3) Shanghai
- 6.(6) Garrison
- 7.(5) Emerald Mine
- 8.(12) The Bard's Tale
- 9.(8) Test Drive
- 10.(-) Defender Of The Crown
- 11.(2) Barbarian
- 12.(-) Indoor Sports

Uwe Hermann, Diepholz / Christian Nordström, Stein/Traun / Marco Trauner, Reinheim / Michael Bromund, Montabaur / Thomas Hoess, Stuttgart / Andrea Hardy, Frankfurt / Robert Hermann, Marxheim/ Klaus Libig, Aschaffenburg / Stephan Kurz, Freiburg / Marc Ermer, Langenselbold/ Holger Himmelmann, Leimen / Wolfgang Ilzhöfer, Schwetzingen

Einkaufsführer

Hier finden Sie Ihren Commodore/Amiga Fachhändler

Anzeigenschluß für Heft 7-8/88 ist der 13.05.88

1000 Berlin





2000 Hamburg



Gerhard u. Bernd Waller GbR Computer & Zubehör-Shop

Kieler Straße 623 2000 Hamburg 54

2 040/570 60 07 + 570 52 75

2000 Hamburg



Computer Hardware · Software · Zubehör Lilienstraße 32 (beim Mönckebergbrunnen) 2000 Hamburg 1 Tel. (0 40) 33 67 08



NEU: Software Shop RADIX Bürotechnik

Heinrich Barth Str. 13 2000 Hamburg 13 Telefon: 040-441695

2160 Stade



2390 Flensburg



2900 Oldenburg

GOLDT Computerhaus

Donnerschweer Str. 127-129 (gegenüber Weser Ems Halle) 2900 Oldenburg Telefon 04 41 / 88 47 06

2940 Wilhelmshaven

Radio Tiemann

Commodore-Systemfachhändler

Markstr. 52 2940 Wilhelmshaven Telefon 04421-26145

3000 Hannover

COM DATA

Ani Schiffgraben 19 · 3000 Hannover 1 Telefon 05 11 · 32 67 36

3000 Hannover



An der Tiefenriede 27 . 3000 Hannover 1 Tel.:(0511) 88 63 83 . 24 Stunden Service

3500 Kassel

Hermann Fischer GmbH

Commodore-Systemfachhändler

Rudolf-Schwander-Str. 5-13 3500 Kassel Tel. (05 61) 70 00 00



4650 Gelsenkirchen-Horst



Hard- und Software, Literatur Bauteile, Service, Versand

Groß- und Einzelhandel Poststr. 15 · 4650 Gelsenkirchen-Horst Tel. 0209/52572

5000 Köln

BÜRÖ MASCHINEN DIQUI

AM RUDOLFPLATZ GmbH 5000 KÖLN 1

RICHARD-WAGNER-STR. 39 RUF: 0221/219171



5200 Siegburg

Computer Center

Luisenstraße 26 5200 Siegburg Telefon 0 22 41/6 68 54

6000 Frankfurt



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

CE Commodore OKI ATARI TOSHIBA

Bei uns werben bringt

GEWINN



Sprechen Sie mit uns. Heim Verlag 0 61 51 / 56057

BUF

6200 Wiesbaden

Poststraße 25 6200 Wiesbaden-Bierstadt (061 21) 5600 84 fax (061 21) 5636 43



AUTORISIERTER COMMODORE SYSTEM-HÄNDLER



6380 Bad-Homburg

PDC GmbH Produkte u. Details Computerverbund

Louisenstr. 115
Ladenpassage Alter Bahnhof
6380 Bad-Homburg
Tel. 0 61 72 - 2 47 48

Autorisierter Commodore-Systemhändler

6457 Maintal

Landolt-Computer

Beratung · Service · Verkauf · Leasing

Autorisierter Commodore-Händler

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

6551 Fürfeld



6680 Neunkirchen

Shop 64

Computer GmbH

Saarbrücken * Saarlouis Homburg * St. Ingbert

Neunkirchen 06821/23713 Commodore Systemhändler

6700 Ludwigshafen



Dieter Hieske Schillenstrasse 36 6700 Ludwigshafen Tel. 06 21 / 67 31 05

AMIGA USER ONLY

6800 Mannheim



Computersysteme + Textsysteme

6800 Mannheim 24
Casterfeldstraße 74-76

☎ (0621) 85 00 40 · Teletex 6 211 912

7000 Stuttgart

»If AMIGA, go to Schreiber« Stuttgart's starker Computer-Laden.

SCHREIBER COMPUTER

Rotebühlplatz 10 7000 Stuttgart-1 Tel.0711/227099 Neu · Neu · Neu · Neu Im SUBWAY BREUNINGER City

7140 Ludwigsburg

B D T

BÜRO-DATEN-TECHNIK-VERTRIEBS GMBH

Kurfürstenstraße 18 · 7140 Ludwigsburg Telefon 0 71 41 · 2 50 74

7890 Waldshut-Tiengen

hetter=data vervice ambh

Lenzburger Straße 4 7890 Waldshut-Tiengen Telefon 077 51/3094

7900 Ulm

EDV-Systeme Softwareerstellung Schulung Systemhaus: Frauenstr 28 7900 Ulm/Donau Tel 07 31/2 80 76 Telex 712 973 csulm-0



8000 München



COMPUTER + BÜROTECHNIK

COMPUTER - SOFTWARE - PERIPHERIE BERATUNG - TECHN. KUNDENDIENST INGOLSTÄDTER STR. 62 L



MACHINE Gabriele L Beratung · Schulung · Verkauf

Computergraphik Gabriele Lechner • Verkauf

Planeggerstr. 1 · 8000 München 60 Tel. 0 89-834 05 91

8390 Passau

Zimmermann @@ktroland

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

Erfolgreich werben im Amiga-Einkaufsführer

Sprechen Sie mit uns. Heim Verlag 0 61 51 / 56057

BUF

8400 Regensburg

Zimmermann @ktroland

8400 Regensburg Dr.-Gessler-Str. 8 2 0941/95085

8700 Würzburg

SCHC5LL BÜROTEAM

Hardware · Software Service · Schulung

computer center

am Dominikanerplatz Ruf (0931) 50488



Die Firma, die Euch
MarauderII® und GRABBIT®
gebracht hat, hat jetzt in
Deutschland ein Büro.
Schaut mal nach unseren
neusten Titeln.

Friedrich-Spee-Str. 11 8700 Würzburg Tel: 0931-884822

Schweiz

CH-8021 Zürich



Langstrasse 31 Postfach CH-8021 Zürich

Tel. 01/2417373 Telex 814193 seco

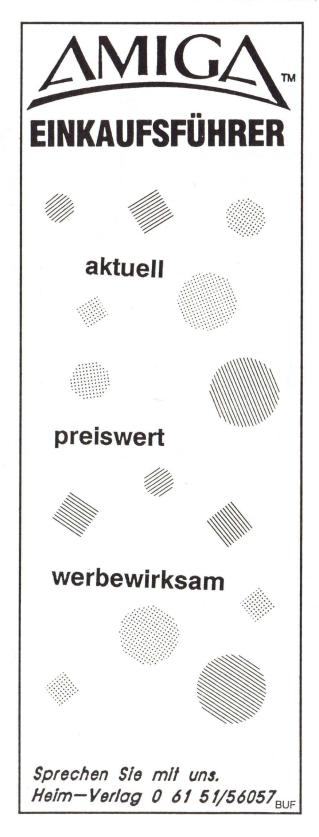
CH-4054 Basel

Wir sind Amiga-Freaks.

Unsere Öffnungszeiten: Von Dienstag bis Freitag 9.30 – 12.30 und 14.00 – 18.30 und am Samstag 9.30 – 16.00



Basel: Tel. 061/39 25 25 · Holeestrasse 87 · 4054 Basel Aarau Tel. 064/22 63 33 · Kasernenstrasse 26 · 5000 Aarau



Liebe Leser,

für den AMIGA gibt es eine unüberschaubare Menge von Public Domain-Sammlungen, so daß es dem Einsteiger völlig unmöglich ist, den Überblick zu behalten. Aus diesem Dilemma soll Ihnen der Public Domain Service der KICKSTART helfen. Die einzelnen Disketten werden nach festen Kriterien zusammengestellt, d.h. daß jede Diskette einen Schwerpunkt hat (z.B. Lehrgänge (Tutor), Bilder-Show, C-Programme, Utilities, Spiele, u. ä.). Außerdem werden Angaben über Programmiersprache, Dokumente usw. gemacht.

DIE NEU-EN

PD 61: GRAFIK-DEMOS

Auf dieser Diskette sind die Beiträge von Allen Hastings, Hobie Oris und Phil Burk zum BADGE KILLER DEMO CONTEST enthalten.

RIPPLES: eine eindrucksvolle Sequenz, die mit VideoScape 3D erstellt wurde.

WAVEBENCH bringt Ihren Bildschirm gewaltig ins Wanken; VIACOM ist auch nicht viel besser!

DROPSHADOW 2.0: von nun an hat jedes Fenster eine Schattenseite, mit Einstellregler für Schattenstärke und -länge.

PD 62: GRAFIK-DEMOS

Eine weitere Diskette mit Werken vom BADGE KILLER DEMO CONTEST. Dieser Wettbewerb hat so viele gute Demos hervorgebracht, die man einfach nicht verstecken kann.

HAGENDEMOS: Die Filme, die Joel Hagen zeigt, sind unbestritten Kunstwerke, denn nicht umsonst wurde "RGB" der Gewinner des BADGE KILLER DEMO CONTEST.

Die beiden Filme wurden mit "The Director" von "The Right Answers Group" erstellt. (benötigt 1 MB RAM!)

BULLY: Diese Demo ist der absolute Wahnsinn. Auf der Workbench läuft ein Boxen-Demo, auf einem anderen Bildschirm blinkt ein Bild von Zipp mit Colorcycling und auf einem weiteren hüpft der AMIGA-Ball. Doch nicht genug, die drei Bildschirme sind auch noch in Bewegung. Sie gleiten auf und nieder und treten dabei abwechselnd in den Vor-

dergrund. Da zeigt der AMIGA was Multitasking bedeutet.

LINEDRAWER zeichnet Bilder, deren Daten einer Datei entnommen werden.

POPLIFE ist eine Variante von LIFE, die sich direkt auf der Workbench abspielt.

PD 63: (Action-) SPIELE

AMOEBA: hinter diesem Namen verbirgt sich eine spielhallenmäßige SPACE INVADERS-Adaption. Obwohl dies ein Spiel der ersten Computergeneration ist, hat es nichts von seiner Spannung verloren. Ein sehr gutes Ballerspiel.

ASTEROIDS ist ein Weltraumspiel, das ebenfalls von Spielhallencomputern umgesetzt wurde. Es ist nicht ganz einfach, die Kontrolle über das Raumschiff zu behalten und es sicher im Asteroidenhaufen zu manövrieren.

EGYPTIAN RUN nennt sich ein Actionspiel, bei dem ein Geländewagen durch die Wüste gesteuert werden muß. Dabei muß man den Sandhügeln und besonders dem Sandstrahl ausweichen, der den Wagen zu unkontrollierten Bewegungen veranlaßt.

PD 64: (Strategie-) SPIELE

BACKGAMMON: bei diesem Strategiespiel können Sie zeigen, ob Sie besser sind als der Computer.

PUSH OVER: hier wird so lange geschoben bis plötzlich und unerwartet einer der Spieler fünf Steine in einer Reihe hat. Ein spannendes Strategiespiel.

PUZZLE PRO, IFF2PICS: zwei Programme, die ein beliebiges IFF-Bild in mehr oder weniger viele Einzelteile zerlegen. Dann liegt es an Ihnen, wie lange Sie brauchen, um es wieder zusammenzupuzzeln. Der Schwierigkeitsgrad ist einstellbar.

PD 65: ICON-TOOLS

Mit diesen Programmen ist es jedem möglich, eigene Icons zu erstellen. Diese können sogar gegen alte ausgetauscht werden. DPICON ermöglicht es, eine mit Deluxe Paint erstellte Brush beliebiger Größe in ein Icon umzuwandeln. Damit kann man Bilder mit dem Malprogramm entwerfen und dann einfach in ein großes Icon umwandeln.

SNAP ist ebenfalls ein sehr interessantes Hilfsprogramm, denn damit kann ein beliebiger Bildschirm in ein Icon umgewandelt werden. Die Größe des Icons ist dabei einstellbar. Bei mehrfarbigen Screens wird außerdem eine Farbreduzierung durchgeführt.

ALTICON gibt dem eigenen Icon den letzten Pfiff, denn hiermit werden zwei Icon-Bilder zu einem 'animierten' Icon zusammengefügt.

Und außerdem noch eine Vielzahl schöner Icons, damit nicht alles gleich aussieht.

DIE KLAS-SIKER

Im folgenden finden Sie eine, nach Sachgebieten sortierte Auswahl des Angebots. Sollten Sie an einer kompletten Liste interessiert sein, dann senden Sie einen frankierten und an Sie adressierten Rückumschlag an die untenstehende Adresse.

UTILITIES:

QuickCopy (gutes Kopierprogramm), DirUtil (File-Copy), FileZap (File-Monitor), DiskZap (Disk-Monitor), DiskSalv (Diskettenretter), System-Monitor, CSH (UNIX-ähnliche Shell)
[PD 7]

CLOCK, PORTAR, MACView, Kickbench, Disassembler, Trakker, Checkmodem, POPCLI und vieles mehr. [PD 25]

Auf dieser Diskette befinden sich einige sehr schnelle und leistungsstarke Editoren (UEDIT, MED, BLITZ) mit dazugehörigen Zeichensatz-Utilities.
[PD 28]

PrtDrvGen (erstellt Drucker-Treiber), DropShadow (jedes Fenster bekommt einen Schatten), MemClear (löscht den Speicher), ScreenSave (speichert den Bildschirm auf Diskette), Compress (komprimiert Programme) [PD 29] ASDG (resetfeste RAM-Disk), FixDisk, ErrorCk (zur Fehlersuche auf der Diskette), DiskCat (erstellt eine Übersicht über die Programme Ihrer Disketten) [PD 35]

AddMem (zum Konfigurieren von Speichererweiterungen), Mem View (zeigt den Speicherinhalt als Grafik an), GetRom, MegaPatch (paßt die Kickstart des AMIGA 1000 für das autom. Erkennen von internen Speichererweiterungen an) [PD 37]

Alles, was Sie zu dem von ELECTRONIC ARTS entwickelten Grafik-Standard (IFF-Format) wissen müssen: Laden, Speichern, Komprimieren, Dekomprimieren. Mit Dokumentationen und Source-Codes in C. [PD 41]



SECTORAMA (ein sehr nützlicher Disketten- und FestplattenMonitor, mit dem verlorene oder zerstörte Daten wiederhergestellt werden können), SILICON (ein sehr komfortabler CLI-Ersatz mit separatem Ausgabe- und Eingabefenster), DBUG (maschinenunabhängiger Debugger von Fred Fish (Source in C)), TIMER (eine Stoppuhr für die Workbench) [PD 47]

CONMAN V1.0 (sehr nützlicher CLI-Ersatz), MOUSEREADER (Texte lesen mit Mausbedienung), TIMERAM, MEMWATCH, DISKMAN V1.0 (Bedienung fast aller Diskettenoperationen mit der Maus, sehr umfangreich!), DIRMASTER V1a (sehr schönes Disketten-Verwaltungsprogramm, viele Funktiont

Einige schöne Grafikdemos und Utilities zu diesem Thema [PD 55]

GOMF (bekämpft den GURU), VCHECK (Viruschecker von Commodore), Journal (Mausaktionen werden aufgenommen), PRINTERSTEALER (leitet Druckerausgabe auf Disk um), Editordemo [PD 57]

Sprachen:

XLISP 1.7 (neueste Version) mit ausführlicher Anleitung (über 50k) [PD 16]

MODULA-2: Pre-Release eines Modula-Compilers mit verschiedenen kleineren Beispielprogrammen, die als Source-Code vorliegen [PD 17] MVP-FORTH und C-FORTH (C-Forth ist ein recht leistungsfähiger FORTH-Interpreter, der auch als Quelltext vorliegt)
[PD 22]

Eine Diskette voll Programmen (Spiele, Grafikprogramme uva.) in AMIGABASIC zum Reinschauen, Verändern, Lernen. [PD 50]

Ein einfacher C-COMPILER, in den man etwas Arbeit stecken muß, damit er läuft. Eignet sich für Interessenten am Compilerbau, da der Source-Code vorliegt. [PD 51]

Eine Sammlung von Programmen, die besonders dem Anfänger zeigen, wie man Intuition programmiert. Die Programme liegen sowohl aus C-Quellcode als auch als fertige Programme vor, die sofort gestartet werden können. [PD 1]

ASM68K (Macro Assembler mit guter Dokumentation), ASM (68010 Macro Assembler wie im AmigaDOS Manual beschrieben), BLINK (bekannter Linker), AS6502 (portabler 6502-Assembler mit Source in C)[PD 56]

ANWENDUNGEN:

MCAD ist ein wirklich gut gemachtes CAD-Programm, daß jedoch nur im Interlace-Modus läuft. Es bietet die einfachen Zeichenfunktionen und Features wie Zoom, Group, Ungroup, Grid, Move, Rotate). Auf der Diskette sind mehrere Dokumente, die das Programm erklären.

[PD 36]

MICROSPELL (überprüft die Rechtschreibung), ACCESS (Terminal), QBASE (Dateiverwaltung) und einige andere Anwenderprogramme [PD 54]

GRAFIK & ANIMATION:

RAY TRACERS: wunderschöne räumliche Bilder, die auf einer VAX berechnet und auf den AMIGA übertragen wurden [PD 11]

Auf zwei randvollen Disketten erleben Sie eine einmalige Dia-Show mit hervorragend digitalisierten futuristischen Bildern in voller PAL-Auflösung. Dazu gibt es stimmungsvolle, sphärische Musik. (IFF-Format)
[PD 26 & PD 27]

Boing!, Rotate, Sparks, Moire, Dazzle, 3DCube, Scales, Sizzlers. Sehenswert ist der Film 'Atari meets AMIGA', der die erste und einzige Begegnung der beiden Computer dokumentiert. Sehr schön ist das Programm LANDSCAPE, das wunderschö-

ne fraktale Berg- und Tallandschaften erzeugt.[PD 40]

Eine einmalige Show, bei der eine digitalisierte Katze in gleitenden Bewegungen über den Bildschirm trabt. Erstellt wurde diese faszinierende Animation mit einem Digitizer, DPaint und VideoScape 3D.[PD 43]



Eine weitere Diskette (siehe auch PD 42) mit phantastischen RAY-TRACER-Bildern, unterlegt mit digitalisierter, fetziger Musik. (IFF-Format)[PD 46]

Die neuesten Demos vom BAD-GE DEMO KILLER CONTEST: hervorragende Grafik und Animation.[PD 58]

Mit drei herrlichen Grafik-Animationen stellt Eric Graham seine Programme Sculpt 3-D und Animate 3-D vor. Die Bilder der Animationen sind nach dem Ray-Tracer-Verfahren berechnet.

[PD 59]

RAY-TRACER DBW-Render ist ein leitungsfähiges Programm zum Erstellen von Bildern nach dem Ray-Tracing-Verfahren. Die Berechnung der Bilder dauert mehrere Stunden, aber die Ergebnisse sind hervorragend. [PD 60]

Diverses:

AMIGA-Tutor, eine Einführung in die Bedienung des AMIGA 500. Ein farbenfroher Lehrgang, der ganz am Anfang beginnt und mit vielen Bildern und Grafiken die Grundbegriffe des AMIGA erklärt. (für Anfänger, kompleu in deutsch)[PD 21]

SPIELE:

YachtC (Würfelspiel für 4 Personen), Puzzle, Missile (verteidigen Sie Ihre Stadt, starker Sound), TriClops (sehr schönes 3D-Spiel), Breakout (3D-Effekt mit Brille), Trek73 (bekannte Star Trek-Variante) [PD 2]

TUNNEL VISION: Werden Sie den Weg durch das Labyrinth finden? REVERSI: eine spielstarke Version des bekannten Brettspiels KLONDIKE: ein Patience-Kartenspiel [PD 34]

Ein echter Leckerbissen für Spiclefans ist das Grafikadventure LARN. In unüberschaubaren unterirdischen Gängen müssen Gold und Schätze gesucht werden, die von Gnomen und Vampiren bewacht werden. Wirklich sehr empfehlenswert! [PD 44] GRAVITYWARS (interessantes Weltraumspiel, bei dem sich zwei Raumschiffe im Kampf gegenüberstehen, viele Optionen!), OTHELLO (sehr schöne Reversi-Variante mit eigenem Fenster!), STREITPATIENCE (Patience-Variante von Hellmut Voelcker), CHESS (spielstarkes Schachprogramm)[PD 45]

Die Programme laufen auf allen AMIGA-Computern mit Kickstart/Workbench 1.2, allerdings sollten mindestens 512k Speicher vorhanden sein (einige Grafik-Demos brauchen 1MB!). Sollten dennoch Einschränkungen gelten, wird dies bei den betreffenden Programmen angegeben.

Versandbedingungen:

Um einen schnellen und problemlosen Versand zu gewährleisten, beachten Sie bitte folgende Punkte:

- Für jede Diskette ergibt sich ein Unkostenbeitrag von DM 10.
- Pro Sendung kommt ein Versandkostenbetrag (für Porto und Verpackung) von DM 5.- (Ausland DM 10.-) hinzu.
- Bei Nachnahme zuzüglich DM 3.70
- Nutzen Sie die Bestellkarte in der Heftmitte,

Anschrift:

MERLIN Computer GmbH KICKSTART Redaktion Postfach 5569 6236 Eschborn

Die Diskettenbestellung kann auch telefonisch erfolgen. Der Versand erfolgt dann per Nachnahme.

Tel.: 06196/48 18 11 (MO.-FR. von 9-17 Uhr)

WANTED! PD-SOFTWARE GESUCHT!

Belohnung: 5 PD-Disketten

Wer aufmerksam das PD-Angebot studiert. wird schnell feststellen, daß fast alle Programme und auch die meisten Grafik- und Sounddemos von amerikanischen und englischen Programmierern erstellt werden. Dies hat unter anderem zur Folge, daß fast alle Kommentare in den Programmen und vor allem die Anleitungen in englisch verfaßt sind. Dies macht es manchem deutschen AMIGA-Einsteiger nicht leicht, den AMIGA und dessen Programmie-

rung kennenzulernen. Wir wollen deshalb noch einmal alle Programmierer im deutschsprachigen Raum aufrufen, ihre selbstgeschriebenen Programme, die sonst in einer dunklen Schublade verschwinden, in die PD-Sammlung zu geben. Als kleinen Anreiz erhält jeder Programmierer, dessen Programm in den KICK-START PD SERVICE aufgenommen wird, fünf PD-Disketten seiner Wahl.

-VORSCHAU-

Juniorprommer

In dieser Ausgabe wurde die reine Hardware des Prommers erklärt, doch damit können Sie natürlich noch nicht viel anfangen. In der nächsten KICKSTART können Sie erfahren, wie es um die Software steht. Grundlageninformationen über die generelle Programmierung von Eproms und natürlich abgedruckte Routinen zum Programmieren von Eproms finden Sie in der Mai-Ausgabe.

Die eigene Stimme aus dem Rechner?

Ein Sound-Sampler stand uns zum Test zur Verfügung. Die technischen Voraussetzungen des Gerätes lassen einiges erwarten. Ob das Gerät in der Praxis seinen Dienst ordnungsgemäß versah, erfahren Sie in der nächsten KICKSTART.

Und sonst...

...gibt es natürlich die neuesten Spiele, interessante Listings, neue KICKS und - ganz klar - die neuesten Berichte über und mit dem AMIGA.

Ab 13. Mai an Ihrem Kiosk! Änderungen vorbehalten!

DISCovery

Auf Entdeckungsreise geht es in der nächsten KICKSTART. Ein komfortabler Disketten-Monitor mußte zeigen, was er kann. Tatsache ist, daß dem Programm eine lange Entwicklungsphase vorangegangen ist, und daß jede Menge Features implementiert wurden. Der Redaktion lag die endgültige Version für einen ausführlichen Test vor. Mehr in der nächsten KICKSTART.

Grundlagen

Wie immer ist die KICKSTART natürlich angefüllt mit Grundlagenartikeln. Die Kurse 'Mit BASIC ans System' und 'Mit dem Rechner auf DU und DU' gehen natürlich weiter, genauso wie der in dieser Ausgabe begonnene Kurs der Daten-Fern-Übertragung (DFÜ).

KICKSTART im ROM für AMIGA 1000

Die Firma Kupke bietet diese interessante Hardwareerweiterung für den AMIGA 1000 an, mit der das Laden des Betriebssystems entfällt. Aber auch andere Kickstart-Versionen sollen sich noch laden lassen. Lohnt sich die Anschaffung dieser Hardwarebox, ist sie dem Alltagsbetrieb gewachsen, oder treten eventuell Probleme auf? Fragen, deren Antworten Sie in der Mai-Ausgabe erfahren.

Impressum KICKSTART

Chefredakteur: Uwe Bärtels (Chefredakteur)(UB) Markus Nerding (Stellvertreter)(MN)

Redaktion: Andreas Krämer (AK) Gerald W. Carda (GC) Harald Schneider (HS) Marcelo Merino (MM) Harald Egel (HE)

Herausgeber: 'MERLIN'-Computer GmbH Industriestraße 26 Postfach 5569 6236 Eschborn Tel.: 06196/481811 FAX: 06196/41137

Redaktionelle Mitarbeiter: Christian Keller (CHK) Andreas Diebold (AD)

Wolf Dietrich (WD) Andreas Suchy (AS Jobst Hermeier (JH) Autoren dieser Ausgabe: Florian du Bois Carsten Borkmeier Roland Foerster Garry Glendown Ralf Görlach Matthias Kühn Dr Volker Kurz Herbert Kunz Bernd Marieenfeldt Andreas Paul Jürgen Schmidt Lars Schmitt Thomas Schlereth Oliver Siebenhaar Michael Sistig Sven Stillich Andreas Trappmann Simon Wiest Public Relations

Auslandskorrespondenz: D. dela Fuente (GB) L. Hennely (USA)

Claus Peter Lippert

Verlag: Heim Verlag Heidelberger Landstraße 194 6100 Darmstadt 13 Tel.:06151/56057 FAX:06151/55689 FAX:06151/56059

Verlagsleitung: Hans-Jörg Heim Anzeigenverkauf: Kyriakulla Margaritis Uwe Heim (Ltg.)

Anzeigenpreise: nach Preisliste Nr.3, gültig ab 1.1.88

Layout, Titelbild, Fabian & Maier

Produktion: Karl-Heinz Hoffmann Susanne Failer Bernd Failer

Fotographie, Titelfoto Rainer Spirandelli, Archiv Druck: Ferling Druck, Darmstadt

Bezugsmöglichkeit: Zeitschriftenhandel, Kauf- und Warenhäuser, Commodore-Fachhändler oder direkt beim Verlag.

KICKSTART erscheint 11 mal im Jahr Einzelpreis: DM 7,- ,ÖS 56,- SFr 7,-Jahresabonnement Inland: DM 70,-Europ, Ausland DM 90,-Luftpost DM 120,-

Alle in der KICKSTART erscheinenden Beiträge sind urheberrechtlich geschützt. Reproduktionen gleich welcher Art, ob Übersetzung, Nachdruck, Vervielfältigung oder Erfassung in Datenverarbeitungsanlagen, sind nur mit schriftlicher Genehmigung des Herausgebers und des Heim Verlages erlaubt.

Programmlistings,
Bauanleitungen und Manuskripte
werden von der Redaktion gerne
entgegengenommen. Sie müssen
frei von Rechten Dritter sein. Mit
ihrer Einsendung gibt der
Verfasser die Zustimmung zum
Abdruck und der
Vervielfältigung. Honorare nach
Vereinbarung. Für unverlangt
eingesandte Manuskripte wird
keine Haftung übernommen.

Sämtliche Veröffentlichungen in KICKSTART erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes, auch werden Warennamen ohne Gewährleistung einer freien Verwendung benutzt.

Für Fehler in Text, in Schaltbildem, Aufbauskizzen, Stücklisten, usw., die zum Nichtfunktionieren oder evtl. zum Schadhaftwerden von Bauelementen führen, wird keine Haftung übernommen.

(c) Copyright Heim Verlag



Piogramm	DM
SPIELE UND SIMULTATIONEN	
☐ Alien Fires	69,00
□ Alien Strike	49.95
☐ Arazok's Tomb	65,00
☐ Arkanoid	69,00
□ Backlash	54,90
□ Bad Cat	59,00
□ Barbarian	74,95
□ Bards Tale	89,00
☐ Battleships	54,95
□ Breach	74,95
☐ Bureaucracy	75,00
☐ Championship Golf	75,00
□ Cogans Run	49,00
□ Crazy Cars	64,90
□ Dark Castle	75,00
☐ Defender of the Crown	85,00
□ Deja Vu	85,00
□ Destroyer	74,95
□ Emerald Mine	29,00
☐ Emetic Skimmer	49,00
☐ Faery Tale Adventure	85,00
□ Feud	29,00
☐ Firepower	49,95
☐ Flight Simulator II	99,00
☐ F.Sim Scenery Disk#7	49,90
☐ F.Sim Scenery Disk#11	49,90
□ Footman	49,95
□ Fußball Manager	74,95
☐ Galactic Invasion	44,95
☐ Garrison	69,00
☐ Girls of Riviera	44,95
☐ Goldrunner	75,00
☐ Gnome Ranger	44,95
☐ Guild of Thieves	64,90
☐ Great Giana Sisters	59,00
☐ Hollywood Poker	39,00
☐ Indoor Sports	74,95
☐ Insanity Fight	75,00
☐ Into the Eagles Nest	75,00
□ Impact	49,95
☐ Jagd auf Roter Oktober	74,95
□ Jet	89,95
□ Jinxter	79,00
□ Jump Jet	44,95
☐ Kikstart 2	29,00
☐ Knight Orc	59,00
☐ Land of Lounge Lizards	90,00
☐ Mercenary	64,90
□ Moebius	79,95
□ Mouse Trap	44,95

Programm	DM
] Phalanx II	29,00
□ Phantasie III	59,95
□ Pinnball Wizard	49,00
☐ Plutos	49,95
Ports of Call	84,95
□ Q-Ball	59,00
Roadwars	54,90
Roadwar Europa	59,95
☐ Seconds Out	54,95
☐ Shadowgate	89,00
Silicon Dreams	59,00
Sinbad	85,00
□ Slaygon	54,95
☐ Space Port	54,95
Space Ranger	29,00
□ Starfleet I	99,00
□ Star Glider	69,00
☐ Super Huey	60,00
Swooper	59,95
☐ Terrorpods	74,95
Test Drive	74,95
The Wall	49,00
☐ Thunderboy	59,00
Time Bandit	54,90
Tolteka	
Uninvited	59,00
☐ Vyper	85,00
Western Games	49,95
	59,00
Winter Games	75,00
Winter Olympiad 88	54,95
World Games	75,00
Xenon	54,90
☐ XR-35	29,00
CHACHECKE	24.00
Art of Chess	64,90
Chessmaster 2000	80,00
Großmeister	59,00
□ Sargon III	95,00
NIMATIONS- UND GRAFIKSOFTWAR EXTVERARBEITUNG UND DESKTOP F	
Aegis Animator/Images	235,00
Aegis Draw Plus	445,00
Aegis Video Titler	249,00
Analytic Art	110,00
Animate 3D	275,00
Animate 3D + deutsche Anl.	310,00
Animator Flipper	70,00
Animator Junior	135,00
Butcher (Deutsch PAL)	110,00
Calligrafonts	99,00
Deluxe Paint II (Deutsch PAL)	249,00
	,

Programm	DM
☐ Desktop Artist	55,00
Digipaint (Deutsch PAL)	138,00
□ Digipic	899,00
☐ Digiview (Deutsch PAL)	440,00
☐ Digiview PAL Software	19,95
□ Director (125,00
☐ Director + deutsche Anleitung	155,00
☐ Express Paint	135,00
☐ Forms in Flight	145,00
☐ Gender Changer	55,00
☐ Graphic Studio	105,00
□ Interchange	85,00
□ IntroCAD	140,00
□ Page Flipper	85,00
□ Photon Paint	195,00
□ PIXmate	120,00
□ Prism +	120,00
□ Professional Page	660,00
□ Sculpt 3D	190,00
□ Sculpt 3D	
☐ TV Show	280,00
	185,00
□ TV Text	179,00
□ Word Perfect Installer	69,00
☐ X-CAD Designer	1145,00
□ Zuma Fonts Volume I	60,00
□ Zuma Fonts Volume II	60,00
□ Zuma Fonts Volume III	60,00
PROGRAMMIERSPRACHEN UND UTIL	
□ AC Basic	360,00
□ AC Fortran	545,00
☐ Aztec C 3.6 (DEV)	595,00
☐ FACC II – Floppy Accelerator	59,95
□ Go – 64	139,90
□ Grabbit	54,00
□ Inovatools#1	140,00
☐ Intswitch	27,50
☐ Lattice C 4.0	385,00
□ LV Backup	120,00
□ M2 Amiga (Deutsch)	350,00
☐ Marauder II (mit Brainfile 10)	69,00
☐ Metacomco Assembler	185,00
□ Metacomco Pascal	185,00
□ Metacomco Shell	135,00
☐ Mimic	69,00
□ Modula 2 (Regular)	185,00
☐ Modula 2 (Developers)	275,00
☐ Modula 2 (Commercial)	545,00
□ Quarterback	135,00
☐ The 64 Emulator	148,00
☐ True Basic	195,00
☐ Turbo Print	89,00

Programm	DM
MUSIKPROGRAMME	
□ Aegis Audiomaster	99,00
☐ Aegis Audiomaster + dtsche. Anl.	134,00
□ Aegis Sonix	135,00
□ Casio CZ Editor/Librarian	225,00
□ Deluxe Music Construction Set	159,00
☐ Deluxe Music Constr. Set (Deutsche PAL)	199,00
□ Dynamic Drums	135,00
□ Dynamic Studio	375,00
DX7 Master Editor/Librarian	275,00
□ D50 Master Editor/Librarian	275,00
□ ECE MIDI Interface	130,00
☐ ECT Sample Ware	145,00
Generic Editor/Librarian	225,00
□ Instant Music	85,00
☐ Keayboard Controlled Sequencer vs1.6	445,00
☐ MIDI Gold (für Amiga 1000)	170,00
☐ Music Mouse	125,00
☐ Pro MIDI Studio V 1.4	345,00
□ Soundsampler	225,00
DATENFERNÜBERTRAGUNG	
□ Aegis Diga	135,00
BUSINESSPROGRAMME	
☐ Acquisition 1.3F	545,00
□ Aegis Impact	150,00
□ Haicalc	105,00
□ Logistix (Deutsch)	340,00
□ Maxiplan 500	249,00
□ Maxiplan Plus	345,00
☐ Microfiche Filer	175,00
BÜCHER	Comme 1
☐ Flying Flight Simulator	19,95
□ Music Through MIDI	39,95
□ Robo City News	4,95
DISKETTEN	1,00
☐ Rhone-Pouleno 3,5" DS/DD Preise auf	Anfrage
VERSCHIEDENES	
☐ Flicker Master	35,00
☐ Lipstick Plus	44,95
□ Megacover (Haube für A500 + Maus)	
☐ Mouse House	
	19,90
□ Mouse Mat	16,50

GTI. Spezialist für AMIGA-Software



