COMMODORE-WELT
Nr. 9/87-September
4. Jahrgang-DM 5,50 / ÖS 46 / SRF 5,50

64'er

20 / 64 / 128
16 / P4 / 116

Das unabhängige Commodore-Magazin

Optimal:
Textomat
Amiga

Kinderleicht:
Eproms
brennen



Kreativ:
Grafiksoftware
für den C 64

Intensiv:
Computer
kennengelernt

HEFT IM HEFT:
Das PC-Magazin CA!

INHALT COMMODORE-WELT Nr. 9/87

GRÜSS GOTT



Das berühmte „Sommerloch" ist diesmal spurlos an uns vorübergegangen. Manche Redaktionsmitglieder schreiben dies dem schlechten Wetter zu, andere glaubten sowieso nicht daran. Jedenfalls hatten wir alle Hände voll zu tun und auch zu schreiben. Wie schon im letzten Heft kurz angesprochen, lieferte Ariolasoft den bisherigen Spieleknüller des Jahres: Barbarian auf dem Amiga. In diesem Heft können Sie nachlesen, wieso dieses Spiel alles Bisherige in den Schatten stellte und bei Amiga-Freaks für nächtelange Kämpfe am Computer sorgen wird.
Um bei den Knüllern zu bleiben: Vor kurzem erhielten wir von einem Leser drei Disketten als Programmeinsendung. Randvoll mit — wir glaubten es zunächst selbst nicht — Comics der Spitzenklasse! In mühevoller Kleinarbeit hatte sich unser Leser Thomas Pohl vor sein Koala-Pad gesetzt und wirklich hervorragende Grafiken entworfen. Mehr hierüber lesen Sie im nächsten Heft, in welchem wir Ihnen einige der Comics vorstellen werden.
Einen Satz nach oben hat unser bisheriger Programmredakteur Andree Herman (Foto) gemacht: Nachdem er mit dubiosen Verbindungen verschiedener Rechner (er betreibt drei Computer an einem 128 D-Laufwerk) für Aufsehen sorgte, war sein Aufstieg perfekt. Nach einigen Monaten, die er mit der Bearbeitung von Leserprogrammen zubrachte, hat er sich nun als Redaktionsassistent bei uns in die Pflicht nehmen lassen. An Arbeit für ihn mangelt es mit Sicherheit nicht . . .
Da geht es ihm ebenso wie der übrigen Redaktion. Uns steht nämlich die Systems bevor. Zum ersten Mal können wir uns freuen, mit unseren Lesern über einen eigenen Stand in Kontakt zu treten. Ein wenig Glück war allerdings schon dabei. Die Messeplätze der Münchener Computermesse sind nämlich schon seit geraumer Zeit vergeben und es bedurfte großer Anstrengungen, um überhaupt noch einen Platz zu ergattern. Trotzdem, wir haben es geschafft. Wo Sie uns finden? In Halle 21. (Siehe unser Titelbild)
Womit wir Ihnen auch diesmal wieder viel Spaß bei der Lektüre dieser Commodore-Welt wünschen.
Ihr Commodore Welt-Team



Andree Herman

IMPRESSUM

COMMODORE-WELT

erscheint monatlich in der CA-Verlags GmbH (i.G.)

CHEFREDAKTEUR (verantwortlich für den Inhalt): ANTON KULT

Ressort 16/116/P4: Alfons Mittelmeyer

Ressort VC20/C64: Anton Kult

Ressort 128: Harald Beiler

Ressort Amiga: Torsten Seibt

REDAKTION UND STÄNDIGE MITARBEITER: Peter Basch, Harald Beiler, Rosemarie Huber, Lothar Miedel, Alfons Mittelmeyer, Michael Reppisch, Rudolf Schmid-Fabian, Torsten Seibt, Hermann Wellesen, Bernd Welte

GESCHÄFTSFÜHRER (und verantwortlich für Anzeigen): Werner E. Seibt

ANSCHRIFT FÜR ALLE VERANTWORTLICHEN:
Postfach 1161
8044 Unterschleißheim
Telex: 5214428 cav-d
Es gilt Preisliste Nr. 7 vom 1.2.1987
Media-Unterlagen bitte anfordern.

ANZEIGENREPRÄSENTANZ:
Verlagsbüro Wolf,
Veilchenweg 2
7909 Dornstadt
Tel. 07348/21394

ANZEIGENSACHBEARBEITUNG:
Angelika Kilches-Wienstein
Tel.-Nr. der Anzeigenabteilung: 089/18 40 22

VERTRIEB: Verlagsunion Wiesbaden

©1987 by CA Verlags GmbH (i.G.)
Printet in Germany

INHALT COMMODORE-WELT Nr. 9/87

TEST & TECHNIK

GAME-FEVER:
TopGun, Ninja, Quartett: Ein buntgemischtes Trio, von uns für Sie getestet
ab Seite 4

SARACEN:
Der Kampf des letzten Kreuzritters gegen die Ungläubigen
ab Seite 7

BARBARIAN:
Unglaublich, welche Grafiken auf dem Amiga möglich sind. Animation wie in einem Zeichentrickfilm ab Seite 8

THE SYDNEY AFFAIR:
Das Manko ist eingebaut: Wer einmal den Mörder kennt . . .
ab Seite 10

BATTLE OF ANTIETAM:
Wer Ballerspiele mit Kriegs-Simulationen liebt – hier ist der amerikanische Bürgerkrieg ab Seite 11

OCP ARTSTUDIO II:
Phantastisch, was dieses Paket aus dem alten C 64 alles herausholt. Ein Software-Paket, das kaum noch zu übertreffen sein wird
ab Seite 12

TEXTOMAT AMIGA:
Da wird die Konkurrenz ganz schön zu beißen haben, was Data Becker da vormacht
ab Seite 16

DIE URKUNDE:
In „Die Erbschaft" mußten Sie eine Million Dollar gewinnen, jetzt können Sie kassieren. Vielleicht, denn da gibt es etliche Probleme
ab Seite 65



Getestet: TopGun und vieles mehr.

THE LAST WARRIOR:
Muß es sowas noch geben? ab Seite 66

DOLMETSCHER:
Von Vizawrite oder Textomat auf den SV-Text? Sie müssen nicht neu eintippen, ein Programm macht das ab Seite 71

PULSAR:
Eprom-Brennen kinderleicht gemacht, mit allem Drum und allem Dran ab Seite 72

SPEICHER-ERWEITERUNG:
512 zusätzliche Kilobyte für den Amiga 500 – und etliche Zugaben mehr ab Seite 77

RKT-PRINTERFACE:
Eines für Alle. Macht endlich Schluß mit dem Druckerwirrwarr
ab Seite 68

SERIE – SERVICE & REPORT

GLOSSE:
Endlich enttarnt – die wahre Bedeutung der Fehlermeldung Ihres Computers ab Seite 14

SEITE 16:
Rettungsanker für Delete, Einblicke ins Disketten-Bam, Tips zu Script Plus und vieles mehr auf unserer Seite 16, aus technischen Gründen in diesem Heft
ab Seite 18

DRUCKERROUTINE:
Druckeranpassung für Datei 128

DIRECTORY-ANZEIGE:
Unser kleines Utility löscht den Speicher nicht!

FASTBACKUP 1541:
Nur noch dreimal wechseln, um eine randvolle Diskette zu kopieren! ab Seite 22

SERIE:
C 128 intern – Was Sie alles mit dem Videochip des 128 anstellen können
ab Seite 48

ERROR:
Fehlermeldungen, was sie bedeuten und wie man sie umgeht
ab Seite 51

NEWS:
Rund um Commodore – Neuheiten und Wissenswertes
ab Seite 54

COMPUTERWISSEN:
Ist er intelligent oder nicht? Wir bringen Grundlagenwissen näher ab Seite 55

LISTINGS

TEXTER 128:
Ein Programm, das mehr kann als so manches Profipaket
ab Seite 28

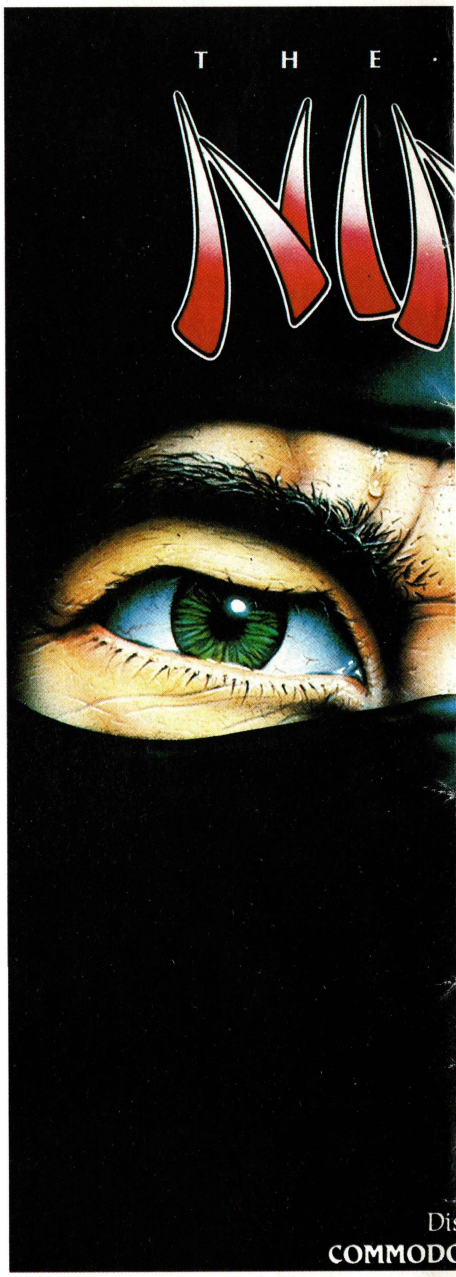
TELEFONBUCH:
Dateiverwaltung für den Amiga, in Amigabasic ab Seite 34

SENSOR:
Wie gut ist Ihr Gedächtnis? Der Computer prüft's
ab Seite 43

SKEET:
Tontaubenschießen hört sich einfach an, aber . . . ab Seite 46

GAME- FEVER I



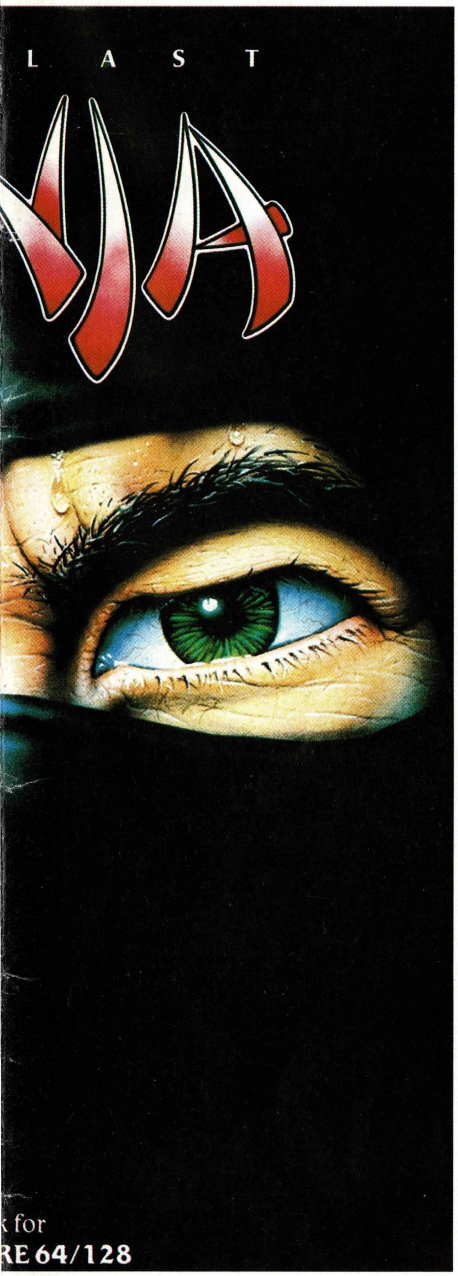


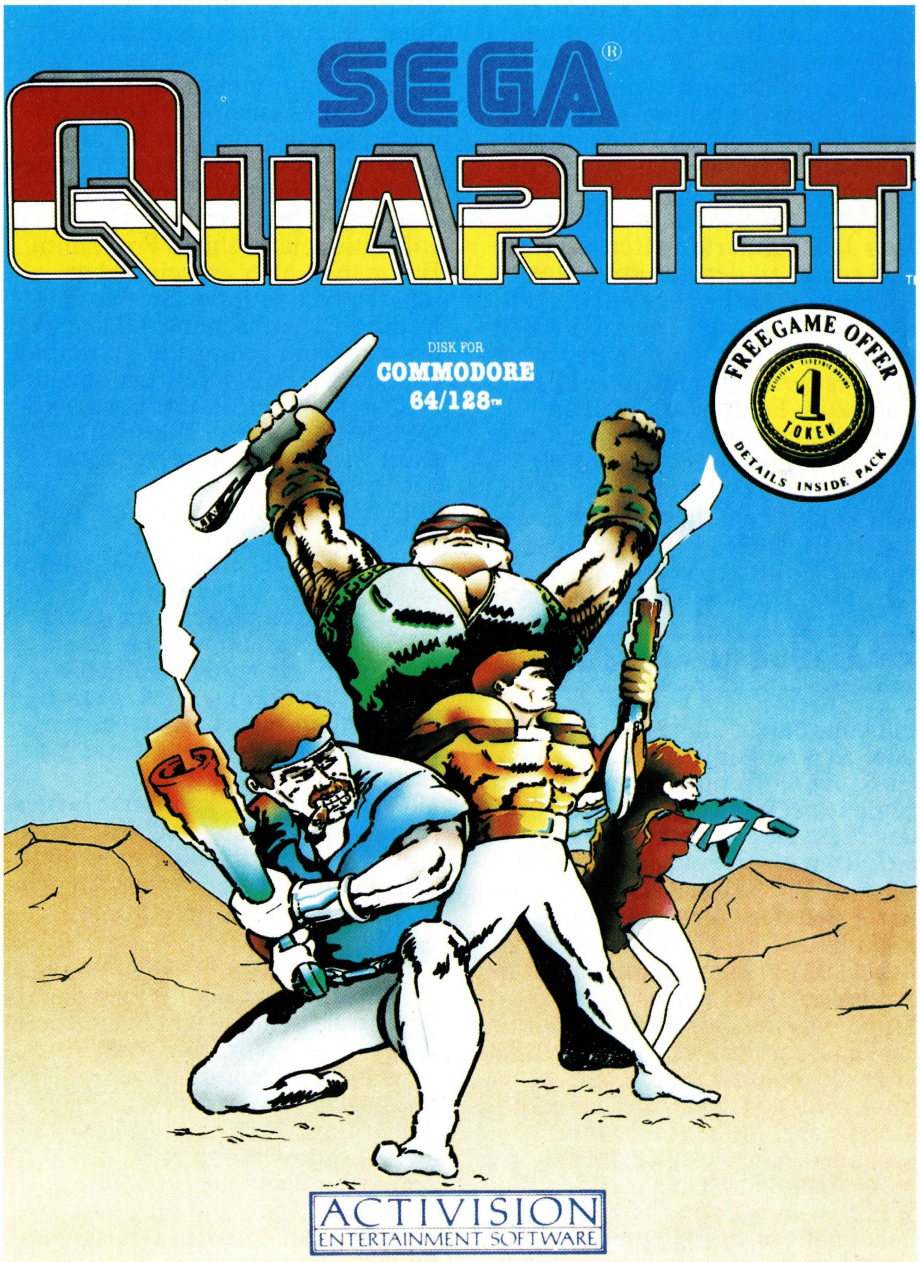
TOPGUN

Wer weiß nicht mehr, als er mit schweißgebadeten Händen im Kinosessel saß und den irren Flugmanövern von Maverick und seinen Navykumpanen am blauen Himmel zusah?
Die Softwarefirma Ocean, bekannt von Miami Vice und Knight Rider, brachte jetzt ein Game mit dem gleichnamigen Filmtitel in den deutschen Softwarehimmel.

TopGun kann allein oder zu zweit, d.h., gegeneinander, gespielt werden. Dem TopGun-Beginner würde ich allerdings empfehlen, die zweite F14 Tomcat nicht dem Computer anzuvertrauen. Viel besser sollte eine stillstehende Tomcat zum Üben mißbraucht werden.
Nachdem sich jeder seine fliegerischen Fähigkeiten angeeignet hat, kann es losgehen.
Jedes Kampfflugzeug ist mit den gleichen Waffensystemen bestückt. Jede Maschine besitzt eine 20mm-Maschinengewehrkanone, Sidewinder-Raketen und Leuchtraketen, um abgefeuerte Sidewinder zu verwirren. Wird nämlich eine Sidewinder richtig abgefeuert, dann kann der Angegriffene sich nur durch Leuchtraketen aus dieser mißlichen Lage befreien.
Immer wenn eines der Flugzeuge abgeschossen wurde, starten zwei neue F14 von je einem Flugzeugträger.
Dieses Bild wird zwar keinen Schönheitspreis gewinnen, aber es erhöht auf jeden Fall den Spielwitz und die

FOR C 64 & 128





Motivation der Freizeitpiloten. Nachdem die Maschinen gestartet und mittlerweile schon über den Wolken sind, wird in die, vom Spieler beeinflußbare, Kampfsimulation umgeschaltet. Nun sind zwei Flugzeuge in einem geteilten Bildschirm zu sehen; nichts Besonderes, wird man meinen. Sobald der Joystick (in diesem Fall wohl besser Steuerknüppel) bewegt wird, wird die einmalige und fast ruckfreie 3D-Vektorgrafik der F 14 sichtbar. Es ist sicherlich nicht verkehrt, sich erst einmal sämtliche Flugzeuglagen reinzuziehen. Außer den schon genannten Waffensystemen besitzt jede Tomcat eine große Anzahl von Signalisierungs- und Überwachungsinstrumenten.
Zu diesen Geräten gehören z.B. der Raketen-Warnanzeiger, der Triebkraftmesser, der Flugzeughöhenanzeiger und Radar.
Ohne diese Instrumente ist das Finden und das Zerstören von feindlichen Maschinen fast unmöglich.
Zu Zweit macht TopGun natürlich am meisten Spaß. Wer will denn nicht seinen Gegner neben sich sitzen haben und bei seinen wilden Joystickmißhandlungen und den verknoteten Fingern vor Schadenfreude herzlich lachen?
Den Jungs von der Indizierungsstelle wird dieses großartige Simulationsprogramm wahrscheinlich nicht gefallen und somit bald auf der Liste der verbotenen Games landen.

Andree Herman

THE LAST NINJA: DIE RACHE DES NINJA

Nach etwa 200 Jahren Knechtschaft unter dem Geschlecht der Fujiwara, dessen Hof an Prunk jedes andere Shogunat weit übertraf, beschlossen die Bauern und der arme Teil der Bevölkerung, in die Gebirge auszuwandern, wo sie die Ninjutsu, die Gemeinschaft der Kämpfer des Schattens, bildeten. Vier Jahrhunderte später, als sie sich schon zu den gefürchtesten Kämpfer ihrer Epoche entwickelt hatten, war wieder einmal die Zeit gekommen. Alle zehn Jahre, so verlangt es ihr Kult, muß eine Pilgerfahrt zum Schrein des Weißen Ninja unternommen werden, um ihm zu huldigen und sich in die Lehren der Koga Rollen zu vertiefen. Leider kam diesmal etwas dazwischen. Dieses Etwas hat den Namen Kunitoki, stammt aus dem Geschlecht der Ashikaga und ist

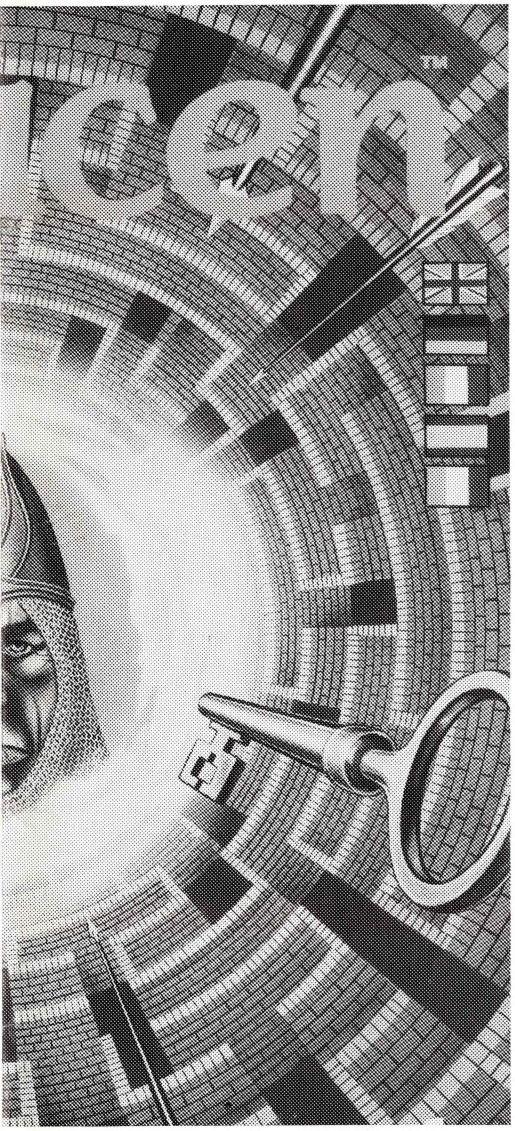
ANLEITUNG IN GUTEM DEUTSCH

Shogun, der zu allem Überfluß noch böse ist. Die Ninja sind ihm schon lange ein Dorn im Auge. Diese Gelegenheit packt er beim Schopf und lauert ihnen bei ihrer Pilgerfahrt auf. Kein einziger überlebt dieses schändliche Blutbad. Doch in einem hatte sich der Ultrabösewicht verkalkuliert: Bevor die Ninjas auf ihre Pilgerfahrt gingen, beauftragten sie Armakuni, jetzt der letzte ihrer Sippe, mit der Bewachung des Bunkinkan-Schreins, was bedeutet, daß er zurückbleiben mußte. Als Armakuni vom Tod seiner Brüder erfuhr, schwor er, blutige Rache am Shogun und allen seinen Anhängern zu nehmen. Alsbald machte er sich auf den Weg zur Insel Li Fen, die eine natürliche, nahezu unbezwingbare Festung abgibt. Doch er kannte von früher her einen geheimen Durchgang, der ihn zum Palast von Li Fen führen würde. Als er jedoch dort ankam, war nichts mehr so wie früher. Nun war er ganz allein auf sich gestellt und hatte mit vielen unbekannten Gefahren zu rechnen.
Dies war die eindruckvolle Geschichte zum Spiel „The Last Ninja". Sie vermittelt dem gespannten Spieler schon einen ungefähren Eindruck in das Ziel des Spiels.
Hat er sich durch die sehr ausführliche Anleitung gekämpft, die in gutem Deutsch und außerordentlich detailliert geschrieben ist, so weiß der Spieler zwar einiges über das Spiel, aber noch lange nicht alles. Das Laden des Spiels klappte bei uns allerdings nicht ganz ohne Probleme, da es einen sehr guten Kopierschutz hat, der den Eindruck erweckt, als überliste er sich manchmal selbst. Erst nach mehreren Versuchen präsentierte sich uns ein vollständig lauffähiges Programm. Hat man es aber erst einmal geschafft, so erlebt der Spieler eine freudige Überraschung. Gleich zwei hervorragend gelungene Titelbilder, eins für die Softwarefirma und eins für das Spiel selbst, erscheinen abwechselnd auf dem Bildschirm. Erst dann wird das Hauptprogramm nachgeladen.
Und hier offenbart sich dem Spieler Grafik vom Feinsten. Auf einem relativ klein gehaltenen Bildschirm erscheint ein Ausschnitt der Szenerie, in der sich unser Ninja gerade befindet. Außerhalb werden sämtliche Anzeigen, wie zum Beispiel die Energie des Ninjas, seine Waffen, die Wunden seines momentanen Feindes und noch einiges mehr dargestellt.
Mit dem Joystick kann man seinen Ninja durch sechs verschiedene Gegenden führen, die allerdings nicht einzeln aufgerufen werden können, sondern hart erkämpft werden müssen.
Die Szenerie ist grafisch einmalig dargestellt. Man sieht seinen Kämpfer und dessen Umgebung aus der Perspektive von rechts oben. Die umgebende Szenerie ist sehr vielfältig ausgefallen. Es gibt nahezu alles, was man sich denken kann. Dazu gehören schön animierte Springbrunnen, einfache Felsen, Bäume, Statuen und und und.
Das alles wird in einer Grafik dargestellt, die sich keiner auf dem C 64 erahnt hätte. Jedes Einzelteil, jeder Schatten wurde berücksichtigt. Dazu kommt, daß mit der farblichen Gestaltung nicht gespart wurde. Deshalb ist es nicht ratsam, das Spiel auf einem Grünmonitor zu betreiben, da man dann vieles versäumt.
Es gibt Unmengen von verschiedenen Sprites, die ebenfalls sehr farbenfroh ausgefallen sind. Alle Sprites sind hochauflösend gezeichnet. Das Beste an ihnen ist aber ihre Animation (Bewegung). Sie erreicht fast Amiga-Qualität und dürfte nach Impossible Mission neue Maßstäbe in dieser Sparte setzen. Alle Bewegungen sind außerordentlich sanft. Man erkennt auch nach dem genauestem Hinsehen keinen Hopser und nicht das geringste Ruckeln. Auch der Sound läßt eigentlich keine Wünsche mehr offen. Schon bei den Titelbildern wird dem Spieler ein gut komponiertes Musikstück präsentiert, das allerdings im Vergleich zu dem, das beim Spiel abläuft, noch nichts ist.



Es gibt sehr viele verschiedene Schlagtechniken, die sich auf die Waffe beziehen, die man gerade zur Hand hat, und Waffen gibt es eine ganze Menge. Die Palette reicht von der eigenen Handkante über das Schwert bis zum Wurfstern.
Neben der großen Anzahl an Schlag- bzw. Wurftechniken sind auch noch einige komplexe Bewegungsabläufe wie Salto und Deckung angesagt. Wegen der reichen Zahl an Bewegungsmöglichkeiten ist die Joysticksteuerung sehr komplex und deshalb

nicht einfach zu bedienen. Alle acht Richtungen des Joysticks in Kombination mit dem Feuerknopf werden voll ausgenutzt. Anfangs macht die Steuerung noch eine Menge Probleme, aber mit der Zeit (und das kann schon dauern) lernt man sie. Ansonsten reagiert die Steuerung sehr präzise und auch schnell genug, um mit allen Gegnern fertig zu werden, auch wenn man es anfangs nicht glauben will. Ein Manko von The Last Ninja



soll nicht unerwähnt bleiben. Gerade am Anfang ist das Spiel sehr sehr schwer. Es kann sehr lange dauern, bis man an einigen Feinden vorbeikommt. Da kann manchmal gewaltig Frust aufkommen.
Ansonsten ist die Motivation von The Last Ninja sehr hoch einzuschätzen. Auch nach längerem Spielen wird das Spiel nicht langweilig, da es immer wieder Neues und Faszinierendes zu bieten hat.
Kommt man lange nicht weiter, so neigt man leicht dazu, den Joystick oder irgend etwas Greifbares zu packen und durch die Luft zu schleudern.

FAZIT

The Last Ninja ist ein Spiel, das in Sachen Grafik und Animation auf dem C 64 neue Maßstäbe setzen dürfte. Es ist sicher keine leichte Aufgabe, ein derartig gut gelungenes Spiel zu übertreffen. Daß das Spiel komplex ist, sieht man schon an seinem Umfang. Es kommt selten vor, daß ein Aktion- und Geschicklichkeitsspiel zwei Diskettenseiten belegt. Wer ein Fan von Karate- und Ninjaspielen ist, der sollte sich dieses Programm keinesfalls entgehen lassen.
Wer es sich kaufen will, der kann es bei Rushware und Ariolasoft beziehen. *(mn)*

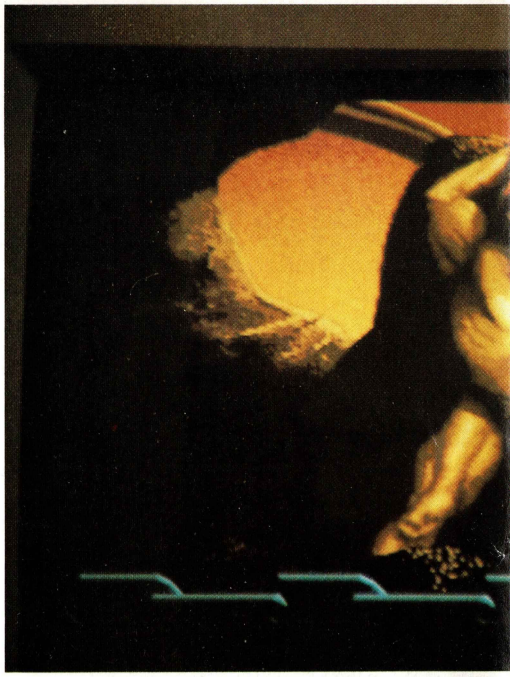
SARACEN: DER KAMPF DES KREUZRITTERS

Vor langer Zeit, etwa vor 800 Jahren, wurden überall auf der Welt Glaubenskriege geführt. So mußten damals natürlich auch die edlen Kreuzritter mit von der Partie sein und auch ihren Beitrag leisten, indem sie die bösen ungläubigen Sarazenen zum Christentum bekehrten. In Saracen verkörpern Sie natürlich einen dieser christlichen Ritter, der, egal wie er ist, ins Morgenland eilt, um seinen Mitstreitern im Kampf gegen die Ungläubigen zu helfen. Dort angekommen, eilen Sie natürlich gleich zum Hauptpalast, um den bösen Anführer unschädlich zu machen. Der Weg dorthin ist aber nicht gerade einfach. Unser edler Ritter muß sich erst durch hundert verschiedene Irrgärten kämpfen, in denen ein übler Haufen von fanatischen Wächtern wartet. Ihre einzige Waffe ist ein Bogen mit überdimensionalen Kräften. Mit dessen Pfeilen können Sie sogar Wände zerstören und explosive Kettenreaktionen auslösen. Die Pfeile dazu fallen Ihnen aber nicht von alleine in die Hand, sondern müssen erst gefunden werden.
Aber nicht nur böse Dinge warten auf unseren edlen Ritter, sondern auch einige sehr hilfreiche.
Nun zum Spiel selbst:
Nachdem man Saracen geladen hat, was allerdings eine ganze Weile dauert, da kein Turbolader eingebaut wurde, hat man die Möglichkeit, zwischen hundert verschiedenen Leveln zu wählen. Danach kann entschieden werden, ob lieber mit dem Joystick oder der Tastatur gespielt wird. Hat der Spieler das alles hinter sich gebracht, so kann er ins Spiel einsteigen.
Und hier erlebt man eine seltsame Überraschung. Die ganze Spielszene kommt doch irgendwie sehr bekannt vor und erinnert recht stark an das Spiel Boulderdash. Dazu kommt, daß Saracen von der selben Firma programmiert wurde.
Nun ja, jedenfalls muß der Spieler eine Figur durch einen Irrgarten führen, der mal nicht mit Erde gefüllt ist.
Die Grafik von Saracen ist sehr schlicht ausgefallen. Neben massiven Mauern existiert noch ein mehrfarbiges Ritterspritе, das aber enorm schlecht animiert ist. Bewegt man den Joystick, so hüpft unser Ritter von Kästchen zu Kästchen durch den Irrgarten. Auch Bewegungen wie Tippen mit dem Fuß oder das Augenzwinkern fehlen hier.
Die übrigens sehr seltsam anmutenden Soldaten bewegen sich genau in der gleichen Weise im Quadrat, wie die Wesen in Boulderdash. Auch im Aussehen haben sie enorme Ähnlichkeiten mit diesen. Hätte der Spieler nicht seine Anleitung, so würde er wohl kaum auf die Idee kommen, diese seltsamen Kreaturen als Wächter der Sarazenen zu bezeichnen. Neben den Soldaten und dem eigenen Ritter gibt es, wie oben erwähnt, noch andere Gegenstände. Diese treten aber nur vereinzelt auf und machen wegen ihrer Größe einen etwas plumpen Eindruck.
Die Hintergrundgrafik beschränkt sich während des ganzen Spiels auf eine verschnorkelte Mauer vor einem leeren Hintergrund.
Erwähnenswert ist auch das rekordverdächtig schlechte Scrolling. Erreicht die Spielfigur einen bestimmten Punkt, so bewegt sie sich nicht mehr, sondern der Bildschirm außenrum. Hierbei muß gesagt werden, daß das Scrolling des Bildschirms etwa der Animation des Ritters gleichkommt.
Der Sound besteht lediglich aus ei-

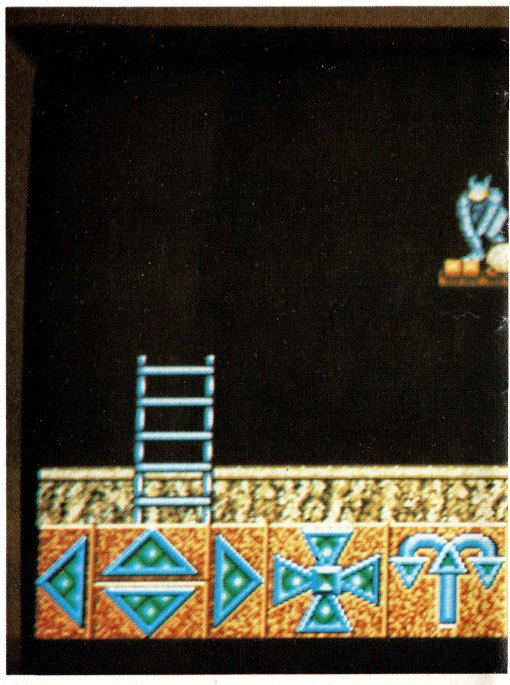
Bitte lesen Sie weiter auf Seite 79

GRAFISCH PERFEKT: BARBARIAN

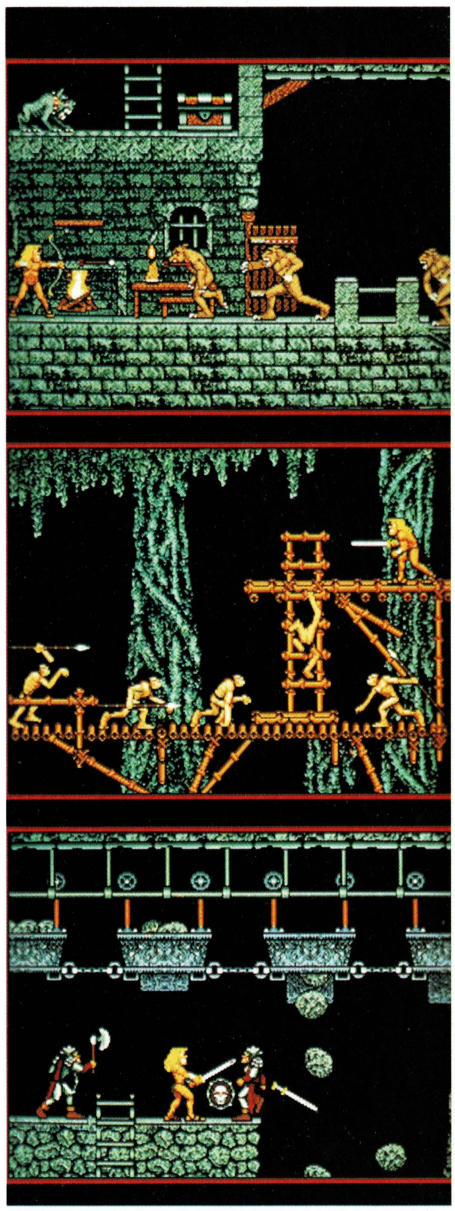
Rollenspiele sind „in". Gerade auf dem Grafik-Computer Amiga erfreuen sich diese Games wachsender Beliebtheit. Einen neuen Vorstoß wagt nun Psygnosis, in Deutschland vertrieben von Ariolasoft, mit dem Spiel „Barbarian" und kontert hiermit gleichzeitig in die Richtung der Cinemaware-Produkte von Mindscape. Eines ist sicher: Barbarian setzt einen weiteren Maßstab in Hinsicht auf die Qualität von Spielen auf dem Amiga.





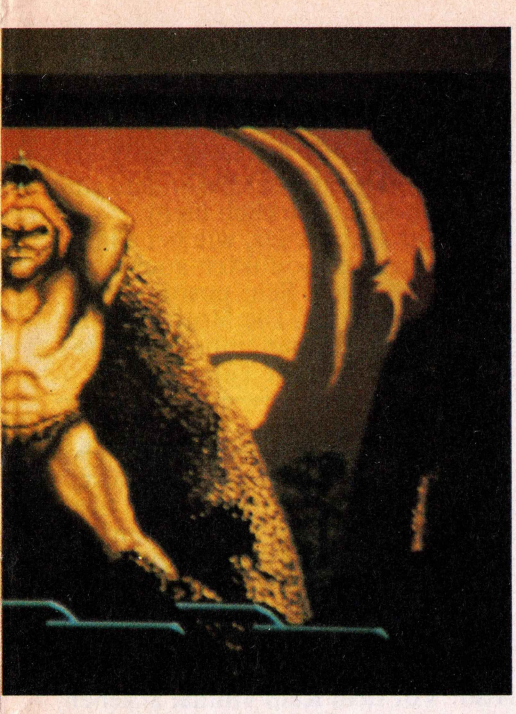


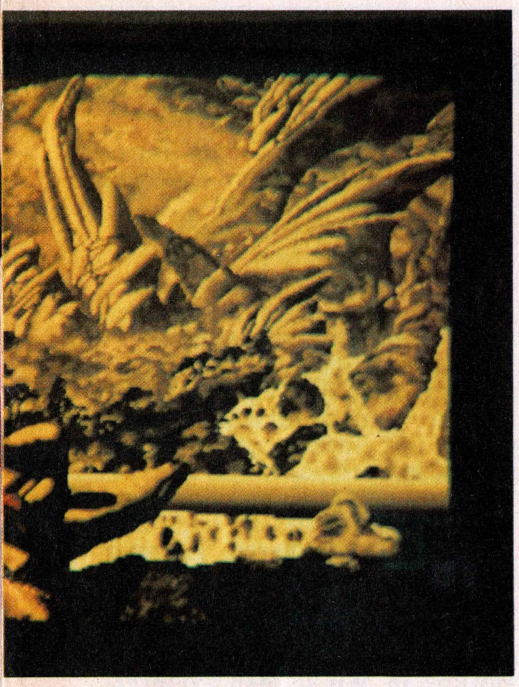
Zuerst sei bemerkt, daß Barbarian auf dem Amiga nichts mit dem gleichnamigen Game auf dem C 64 gemein hat. In disem Amiga-Spiel muß sich Hegor, der Held des Geschehens, durchkämpfen.
Gleich vorneweg die Story, die dem Käufer leider nur in englischer Sprache anhand des mitgelieferten „Geschichtsbuches" vermittelt wird:
Hegor, anerkannter Frauen- und auch sonstiger Held, bricht auf in das Reich seines schurkischen und unglaublich bösen Onkels, welcher seinerzeit Hegors Vater unter Zuhilfenahme eines fiesen Drachens gekillt hatte. Dessen unterirdisches Reich zieht sich über unzählige Ebenen und Räume hin, bewacht von den widerlichsten Kreaturen (die natürlich aus jedem Eindring-

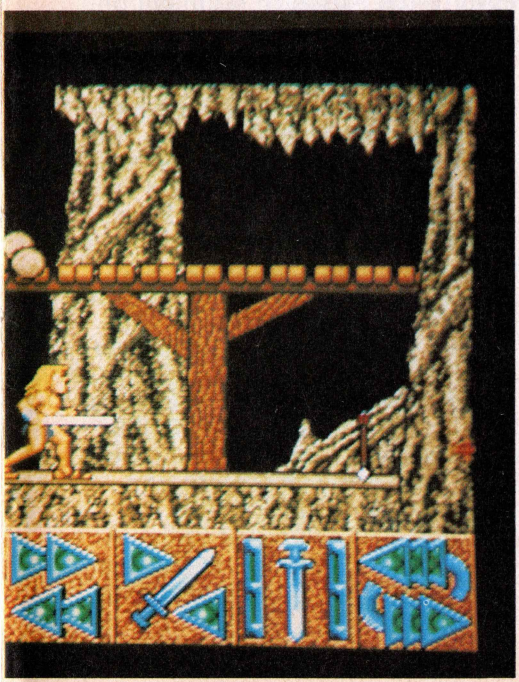


KREATUREN ALLERORTEN

ling Kleinholz machen wollen). Auch der eingangs erwähnte Drache haust irgendwo tief unten in diesem Labyrinth, ist allerdings vor lauter verspeisten Eindringlingen so fett geworden, daß er nicht mehr durch die Gänge paßt und sich zwangsläufig immer in der selben Ebene herumtreibt.
Hegors Aufgabe ist klar: Den bösartigen Onkel aufzutreiben und zu vernichten. Klar, daß bis zum Erreichen desselben das gesamte Labyrinth samt seinen Bösewichtern durchwandert werden muß.
Und genau am Eingang dieses Labyrinthes beginnt, nach einem lautstarken und grafisch sehr gut aufgemachtem Eingangsbild, die Mission unserer Spielfigur.
Gleich zu Anfang fällt die hervorragende Grafik und Animation des Spieles auf: Hegor, etwas größer als ein Daumennagel, stiefelt mit superknappem Tanga-Höschen und muskelbepackt durch die Landschaft. Ein grimmiges Grinsen aufgesetzt, verfügt er über einige sehr differenzierte Fortbewegungsmöglichkeiten, die der Spieler mittels Maus an der unteren Befehlsleiste anklicken kann:
– Normaler Gang
– Laufen
– Springen und
– Fliehen







Es ist wunderbar anzusehen, wenn Hegor aus vollem Lauf zu einem Salto ansetzt und über die Szenerie fetzt. Ebenfalls lustig ist die Option, mit der Hegor zum Fliehen gebracht wird: Mit angstverzerrtem Gesicht und unter Ausstoß eines panikartigen Schreis verläßt unser Held mit ausgestreckten Armen das Geschehen in die entgegengesetzte Richtung. Das einzige Problem beim Fliehen besteht darin, daß Hegor alles bisher Aufgesammelte stehen und liegen läßt. Auf der Menüleiste stehen auch noch zwei Kampfoptionen (angreifen und verteidigen) sowie ein Punkt zum Stoppen und zur Richtungsänderung bereit. In einer zweiten Menüleiste, die durch Drücken der rechten Maustaste erreicht wird, kann Hegor noch dazu veranlaßt werden, Dinge aufzuheben oder wieder zu verlieren sowie bestimmte Objekte aus seinem Sammelsurium zu benutzen.
Die verschiedenen Räume, die jeweils einen ganzen Bildschirm füllen, fallen durch ihren Abwechslungsreichtum auf. Es handelt sich um Höhlen, Sumpflandschaften, Holzgerüste und vieles mehr, die ebenfalls grafisch hervorragend dargestellt werden. Auch die jeweiligen Figuren, die es auf dem Weg in Richtung des bösen Onkels zu beseitigen gilt, bestechen durch abwechslungsreiche Darstellung.
Hegor hat es mit Kreaturen unterschiedlichster Ausführung, von kuttentragenden Monstern über Nashörner, Bulldoggen und waffenstarrenden Rittern bis hin zu Phantasiemonstern, in steigend scheußlichem Aussehen, zu tun.
Die Gefahren, die auf Hegor lauern, sind nicht nur die angesprochenen Monster, welche dem Barbaren ständig nach dem Leben trachten. Der miese Onkel hat auch für zahlreiche sonstige Fallen gesorgt: Steinschlag und herabfallende Eisengitter sowie einstürzende Holzpfade machen der Spielfigur das Leben zusätzlich schwer, diese müssen unter allerlei artistischen Einlagen übersprungen werden. Des weiteren gibt es auch bei verschiedenen Figuren nur eine bestimmte Möglichkeit, diese zu beseitigen: Der Drache in den unteren Räumen läßt sich beispielsweise nur durch mehrere gezielte Pfeilschüsse besiegen. Aus diesem Grund ist es für die Spielfigur auch unbedingt notwendig, die herumliegenden Pfeile aufzusammeln, ohne die der ebenfalls zu findende Bogen selbstverständlich wirkungslos ist.

GEFAHREN LAUERN ÜBERALL

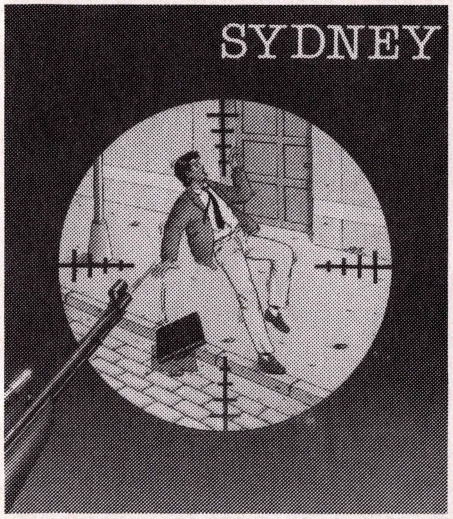
Dem Spieler stehen bei Barbarian genau drei Figuren zur Verfügung. Um bei diesem Punkt zu bleiben: Es muß nämlich jedesmal, wenn Hegor zum dritten Mal sein Leben aushauchte, aus dem Anfangsbild begonnen werden.
Bei dieser Unzahl von Räumen mit den darin enthaltenen Gefahren bringt dies natürlich bei zunehmender Übung einigen Frust mit sich, da mühsam die einzelnen Levels nochmals durchkämpft werden müssen. Mit einer Abspeicherung im Spiel wäre dieses Problem gelöst, man könnte in dem Level weitermachen, in welches man sich zuvor angestrengt vorgearbeitet hatte. Dies ist allerdings der einzige Mangel, der uns bei Barbarian auffiel.
Was das Spiel so herausragend auch aus den neueren „Cinema-Ware"-Produkten abhebt, sind die erstklassige Grafik von Figuren, Hintergrund sowie die programmierten Bewegungsabläufe. Die Animation von Barbarian läßt sogar in Angesicht professioneller Spielautomaten wirklich nichts mehr zu wünschen übrig. Die Maussteuerung über die Menüleiste ist überdies „narrensicher" und äußerst effektiv.
Innerhalb des Spiels wurde auf eine blutrünstige Darstellung verzichtet, welche ja beim namensgleichen C 64-Game auf einige Proteste gestoßen war. Dies holten die Autoren aber dummerweise in der mitgelieferten Anleitung wieder auf, die in einigen Passagen wirklich nicht gerade zimperlich mit rollenden Köpfen und dergleichen umgeht.
Dies sollte allerdings den Spaß an diesem rundweg gelungenen Produkt nicht vermiesen können.

FAZIT

Barbarian beweist einmal mehr die Leistungsfähigkeit des Amiga, was Grafik und deren Animation betrifft. Der Spieler fühlt sich in einen hochwertigen Zeichentrickfilm versetzt, in welchem er selbst Hauptrolle und Regie übernimmt. Gerade in Hinsicht auf die „Konkurrenz" seitens Mindscape dürfte Barbarian wohl eine neue Runde im Genre der Rollenspiele eingeläutet haben. Überdies ist Barbarian in der Bedienung lange nicht so umständlich und langatmig wie beispielsweise das vergleichbare „Sindbad". Durch die exzellente Steuerung und den unglaublichen Abwechslungsreichtum lohnt sich die Anschaffung dieses Games allemal, Langeweile kommt hier so schnell nicht auf. *Torsten Seibt*

THE SYDNEY AFFAIR

Und schon wieder ein Fortsetzungsspiel eines erfolgreichen Programmes aus dem Hause Infogrames. Als Unteroffizier der Gendarmerie in St. Etienne haben Sie Ihr Können schon im Fall „Vera Cruz" bewiesen. Da Sie den Mord erfolgreich aufgeklärt haben, übertragen Ihnen Ihre Vorgesetzten auch „The Sydney Affair". Dieser Fall soll noch schwieriger aufzuklären sein als Vera Cruz: Ein Mann wird mit einer Kugel im Kopf auf St. Etiennes Straßen aufgefunden. Als scharfsinniger Kriminalist wird Ihnen sofort klar: Der Mann ist erschossen worden. Der Mörder, der vom gegenüberliegenden Gebäude aus geschossen hat, kannte wohl die Gewohnheiten des armen Opfers, das jeden Tag zu dieser Zeit am Gebäude auf dem Weg zur Arbeit vorbeigehen mußte. Zur Beweisaufnahme begeben Sie sich an Ort und Stelle. Ausgerüstet mit einer Lupe können Sie die Leiche und die Gegenstände, die auf der Straße herumliegen, genauer untersuchen. Die Lupe wird mit den Cursortasten bzw. mit dem Joystick gesteuert. Drückt man auf F7, wird der Teil des Bildschirms, auf dem die Lupe liegt, vergrößert, und man kann nun Aufschriften oder sonstige Details genauer erkennen. Zuerst wird die Straße untersucht, dann können Sie auch die Wohnung, aus der geschossen wurde, durchstöbern. Seien Sie bei diesem Teil des Spiels sehr sorgfältig, denn es müssen alle Indizien gefunden werden, um den Mörder überführen zu können. Wenn Sie sicher sein können, alle Indizien gefunden zu haben, dann drücken Sie die F1-Taste, und schon wird der zweite Teil des Spiels geladen. Sie befinden sich nun auf der Gendarmerie von St. Etienne, die mit den modernsten Ermittlungsapparaten von ganz Europa ausgerüstet ist. So steht Ihnen zu den herkömmlichen Methoden zur Täterfindung, wie Vergleich der Indizien, Kreuzverhöre oder Expertisen, Ballstikuntersuchungen oder Autopsien von Leichen, auch noch das Computersystem Diamant zur Verfügung. Auf dem Bildschirm ist eine Fernschreiberseite zu sehen, auf der alle Mitteilungen an Sie ausgegeben werden. Am unteren Bildschirmrand befindet sich eine Kommandozeile, auf der alle Befehle aufgelistet sind. Mit dem Diamantcomputersystem können Sie sich mit anderen Dienststellen in Frankreich in Verbindung setzen. Diese Nachrichten müssen so präzise wie möglich verfaßt werden. Wird die Mitteilung von der betreffenden Dienststelle verstanden, wird eine Antwort auf Ihrem Fernschreiber ausgegeben. Wenn Sie über einen Drucker verfügen, können Sie sich eine Menge Schreibarbeit ersparen. Durch Tastendruck werden alle Informationen auch auf dem Drucker ausgegeben. Mit dem Diamantcomputersystem haben Sie die Möglichkeit, diverse Dienststellen anzurufen, so zum Beispiel die Dienststelle für Gerichtliche Nachforschungen, das Bundeskriminalamt oder die Gendarmerie von St. Galmier.



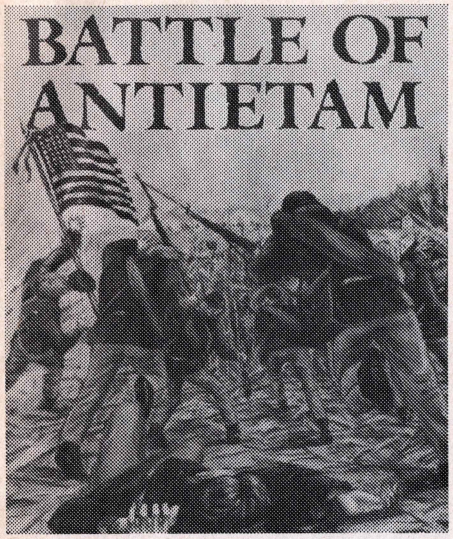
Diese Möglichkeit werden Sie sehr bald zu schätzen wissen.
Mit dem Diamant-System können Sie Nachforschungen über Personen anstellen, aber auch schon aufgeklärte Fälle miteinander vergleichen. Die klassischen Fahndungsmethoden werden Ihnen in vielen Fällen genauso weiterhelfen.
Wenn Sie unter der Option Aussage den Namen und die Adresse einer Person eingeben, muß diese eine Aussage machen. Es ist ebenso möglich, unter der Option Expertisen Autopsien oder Ballistikuntersuchungen durchzuführen. Wenn Sie genügend Beweismaterial gesammelt haben, bedienen Sie sich der Option Verhaftung. Die sollten Sie allerdings nur benutzen, wenn Sie ganz sicher sind, den Täter überführt zu haben. Reichen die Beweise nicht aus, gibt Ihnen das der Computer ganz unmißverständlich zu verstehen. Die Grafik ist für ein solches Kombinations- bzw. Puzzlespiel sicherlich ausreichend, aber besonders gut gelungen ist sie wirklich nicht. Sound ist mal wieder, wie bei anderen Infogramesprodukten auch, nicht vorhanden. Sehr positiv: Alle Texte in Deutsch. Die Motivation ist relativ hoch, da man unbedingt wissen möchte, wer der Täter ist. Hat

FAZIT

man das Spiel jedoch einmal gelöst, macht der Fall Sydney keinen Spaß mehr, weil man den Täter ja dann schon kennt. Der Fall Sydney (englischer Originaltitel: The Sydney Affair) ist ein recht originelles Puzzlespiel. Jedoch ist dieses Spiel bestimmt nicht jedermanns Sache. Also: Vor dem Kauf einmal ansehen.

BATTLE OF

Die Firma Stratetic Simulations bringt schon seit Jahren hochwertige Strategiespiele heraus.
Battle of Antietam ist solch eine Kriegsspielsimulation.
Der Krieg ist ein so ernstes und trauriges Thema, das nicht Gegenstand eines Spiels sein sollte.
Wenn man unbedingt für seinen Computer ein Strategiespiel haben möchte, dann sollte man es doch einmal mit einer Sportspielstrategiesimulation versuchen. Es muß doch nicht unbedingt Krieg sein, oder?
Wir würden uns freuen, auch mal Ihre Meinung zu diesem Thema zu hören, schreiben Sie uns doch mal, oder rufen Sie Mittwochs zwischen 15 – 19.00 Uhr unter der Telefonnummer (089) 1298013 in unserer Hotline an. Wir werden die interessantesten Aussagen sammeln und gegebenenfalls veröffentlichen.
Trotz meiner Abneigungen gegen Kriegssimulationsspiele habe ich versucht, einen einigermaßen objektiven Softwaretest zu schreiben: Bei Battle of Antietam handelt es sich um eine Kriegsspielsimulation aus dem amerikanischen Bürgerkrieg. Es gibt 2 verschiedene Parteien, die sich bekämpfen müssen. Da wäre zum einen die graue Partei, die aus Robert E. Lee und seinen 40000 müden, halb verhungerten Soldaten besteht. Die andere Partei, die blaue Partei, besteht aus General George Mc Clellan, dessen Armee 87000 Mann stark ist. Sie können nun eine der beiden Armeen steuern und ihr zum Sieg verhelfen. Die Schlacht hält von 6 Uhr morgens bis 5 Uhr nachmittags an. Wird es Ihnen gelingen, durch kluge takti-



ANTIETAM

sche Anweisungen Ihre gewählte Armee zum Sieg zu führen? Sie können offensives oder defensives Feuer anordnen, ebenso können Sie einige Patrouillen an strategisch wichtigen Punkten absetzen. Es ist wichtig, seine Soldaten an Brücken zu postieren. Nachschubbatterien des Feindes sind sofort unter Feuer zu setzen. Kontrollieren Sie Ihre Artillerie, die Kanonen sind Ihre wohl wirksamsten Waffen.

DREI SCHWIERIGKEITSGRADE

Battle of Antietam verfügt über 3 Schwierigkeitsgrade: Basic Game, Intermadiate und Advanced Level. Im Basiclevel sind die Spielregeln vereinfacht, die Züge können sehr schnell ausgeführt werden, da der Computer nicht lange zur Berechnung des Gegenzuges braucht.
Alle Vorgänge sind für beide Gegner gleichermaßen sichtbar. Bei Battle of Antietam stimmen laut Aussage der englischsprachigen Anleitung alle Daten, wie Anzahl der eingesetzten Soldaten und Waffen, örtliche Begebenheiten etc., mit dem historischen Äquivalent überein.
Battle of Antietam wird in einer Riesenverpackung angeboten. Darin befinden sich die knapp 30seitige englische Anleitung, ein Prospekt mit einer Auflistung aller Stratetic Simulationsprodukte und eine Papptafel, auf der man das ganze Schlachtfeld sehen kann (auf dem Computerbildschirm sieht man nur Ausschnitte). Auf dieser Tafel befindet sich auch auf der Rückseite eine tabellarische Zusammenfassung mit den wichtigsten Parametern und Daten über Battle of Antietam.
Nach dem Laden des „Spiels" gelangt der „Spieler" in ein Menü, in dem er Größen wie Schwierigkeitsgrad, die Art des Spiels (also Advanced Computer Reaction oder Basic Reaction des Computers), Zeitlimit, aber auch Symbol bzw. Icondarstellung, wählen kann.
Ich habe während meines Tests den Computer gegen sich selbst spielen lassen. Zu Beginn des Spiels erfolgt eine Bilanz, wieviele Tote und wieviele Verwundete die Schlacht bisher schon gekostet hat (das soll wohl eine Art Highscoreliste sein. Igitt! Wie geschmacklos!). Hat man sich von diesem kleinen Schock erholt, fängt auch schon eine Armee mit der Truppenbewegung an.
Artillerie, Fußvolk, Reiter und Materialträger können mit einem quadratischen Cursor an den strategisch wichtigen Punkten plaziert werden.
Glaubt man, aus dieser Position den Krieg gewinnen zu können, zieht der Gegner, und schon kann das Gemetzel beginnen.
Die Grafik ist, wie bei allen Simulationsspielen dieser Art, sehr schlecht, was aber auch nicht verwunderlich ist.
Der Speicher des Computers wird für die umfangreichen Berechnungen benötigt. Für eindrucksvolle Grafiken ist dort kein Platz. Ebenso wurde auf den Sound verzichtet.
Die Motivation ist nur dann hoch, wenn man Spaß an derartigen „Spielchen" hat. Ich kann mir aber vorstellen, daß sogar der hartgesottenste Kriegsspielsimulationsfreak, hat er einmal den Computer in einer Schlacht bezwungen, keinen Spaß mehr an Battle of Antietam findet.

FAZIT

Das hier getestete Strategiespiel ist weder schlechter noch besser als andere Spiele dieser Art. Battle of Antietam entspricht dem üblichen Standard der Stratetic-Simulationsspiele, der, relativ gesehen, sehr hoch ist. Man bekommt für sein Geld, möchte man es tatsächlich für ein solches Kriegsspiel ausgeben, etwas geboten.
Wer unbedingt noch ein gutes Strategiespiel sucht, der sollte aber, das sage ich zum wiederholten Male, auf ein friedliches zurückgreifen.
Als Beispiel waren da Phantasie II, Computer Baseball oder Computer Quarterback von der selben Firma zu nennen.

C 64/128 + X = Druck in Vollendung

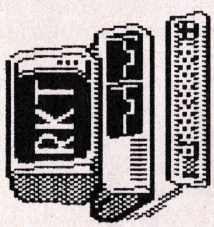


PRINTERFACE

Printerface GH 8708 248,- DM
Printerface GH 8732 298,- DM
Printerface GH 8724 128,- DM

X =

Infomappe gegen
selbstadressierten, mit
DM 2,50 vorfrankierten
DIN A 4 Kuvert



Rolle-Kommunikations-Technik
Postfach 710844 Teletex: 897 310
8000 München 71 Telex: 178 97 310
Tel: 089/79 51 10 Btx: *089795110#
Telefax: 089/7 91 86 21

DER C64 ALS GRAFIK-KÜNSTLER

Obwohl schon geraume Zeit auf dem Markt, ist es in neuer, verbesserter Version erneut erschienen: das englische „Grafikkunst-Studio" von Rainbird.



Es wird immer schwieriger, den verwöhnten C-64-User in Bezug auf Grafikprogramme zufrieden zu stellen: Zu viele gibt es bereits. Seit dem Erscheinen von „Paint Magic" zieht ein Trend wie ein roter Faden durch den Software-Markt des C 64: Grafik-, bzw. Mal- und Zeichenprogramme: Eines besser als das vorhergehende, mit noch mehr Funktionen, mehr Komfort, mehr Farbmöglichkeiten. Wir wollen hier nicht alle aufzählen, eines hat sich bei unserem Test dieses Software-Paketes herausgestellt: Es wird für künftige Programmierer oder Software-Häuser sehr schwierig sein, das „Advanced OCP Art Studio" in punkto Farbenvielfalt und Eingabekomfort noch zu übertreffen.

ZWEI PROGRAMME – ZWEI DISKETTEN

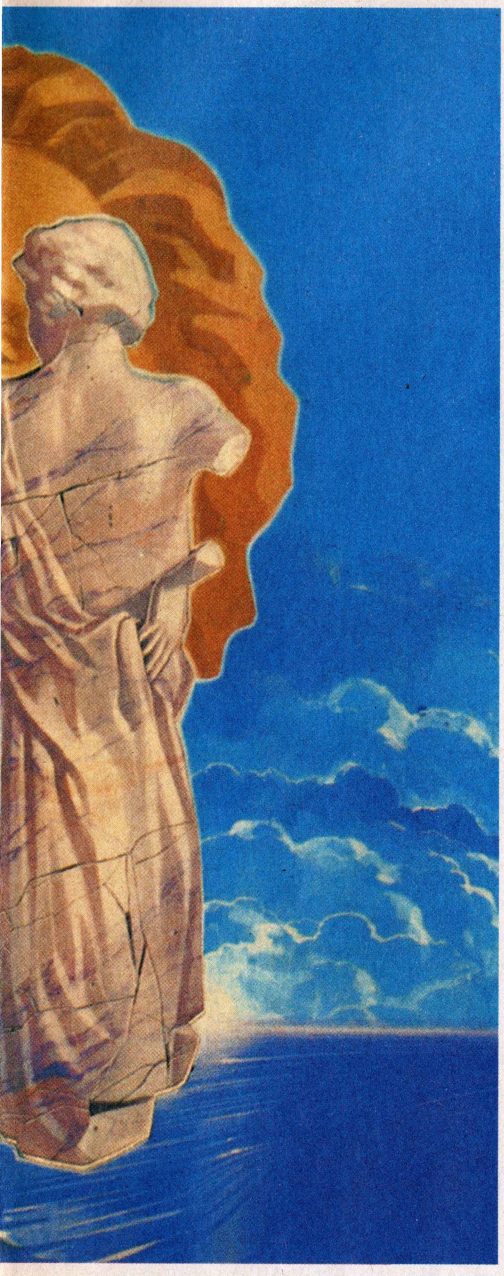
Geliefert wird das Software-Paket (DM 99,–) mit zwei Disketten und zwei Handbüchern. Es handelt sich nämlich im Prinzip um zwei Programme, das „Advanced Art Studio" und „OPC Art Studio", die allerdings einander sehr ähnlich sind. (Das letztere war die ursprüngliche Version.) Die beiden Mal- und Zeichenprogramme lassen sich wahlweise per Tastatur, Koala Pad, Mouse oder Joystick betreiben. Für unseren Test haben wir uns für die letztere Möglichkeit entschieden, weil das sicher dem Verbraucherverhalten am nächsten kommt.
„Advanced" bedeutet so viel wie: fortgeschritten. Der Fortschritt besteht darin, daß nun der Multicolormodus (Sie wissen: 3 Zeichenfarben + Hintergrundfarbe lassen sich darstellen!) aktiv ist, was dem Grafik-Freak sofort erfreulich ins Auge sticht. Bevor Sie jedoch zu arbeiten anfangen, sollten Sie als Besitzer eines Druckers unbedingt innerhalb des Installationsprogramms „Studioconfig" die Druckerparameter festgelegt haben (notfalls unter Zuhilfenahme des entsprechenden Handbuchs), um keine Probleme beim Ausdrucken Ihrer Grafik zu haben. (Ein „Absturz" des gesamten Programmes kann da leicht passieren, peinlich, wenn Sie dann Ihre Grafik noch nicht abgespeichert haben!)
Wer von den C-64-Besitzern den „Startexter.64" kennt, dem wird die Installation des Druckertreibers recht bekannt vorkommen, da prak-

VORHER INSTALLIEREN: DEN DRUCKERTREIBER

tisch nach dem selben Prinzip verfahren wird. Nach Angabe der für Ihren Drucker gültigen Parameter wird das Druckerfile auf die Diskette zurückgeschrieben (vorher Schreibschutzaufkleber entfernen oder mit Sicherheitskopie arbeiten!). Danach läßt sich aus dem Installationsprogramm heraus das Hauptprogramm laden.
Bevor's aber losgehen kann, haben sich die findigen Software-Entwickler noch einen pfiffigen Kopierschutz ausgedacht: Sie müssen nämlich jedesmal, bevor Sie ins Hauptprogramm mit allen seinen Mal- und Zeichenfunktionen kommen, bestimmte Worte als „Paßwort" eingeben, die sich an den unterschiedlichsten Stellen des mitgelieferten Handbuches befinden. Nach drei Fehlversuchen verabschiedet sich das „Advanced Art Studio" mit einem System-Reset! Wir finden so einen Kopierschutz auf alle Fälle gelungener als den mancher Software-Häuser, der die Floppy oder deren Schreib-/Lesekopf doch unter meist häßlichen Geräuschen in Mitleidenschaft zieht.
Das „Zeichenblatt" erscheint danach mit einer Menüleiste am obe-

ZEICHENBLATT MIT MENÜ

ren Bildschirmrand. Die einzelnen Funktionen können jetzt mit dem Joystick und dem Pfeil-Sprite auf dem Bildschirm ausgewählt und „angeklickt" werden.
Augenblicklich erscheint dann das „Pull-Down"-Menü mit den Unterfunktionen, die wiederum mit dem Joystick und einem reversen Balken ausgewählt werden.
Es klappt genauso wie beim Amiga, sogar der grafische Aufbau der „Windows" präsentiert sich verblüffend ähnlich.
Die Menüleiste bietet nun eine Anzahl von Arbeitsmöglichkeiten, wie „Colour" (Wahl der Zeichen-, Hintergrund- und Rahmenfarben), „Paint" (hier wird die Strichstärke Ihrer Zeichnung definiert), „Fill" (nun werden die verschiedenen Füllmuster = Patterns der Grafik ausgewählt), bis hin zur „Text"-Funktion (Grafik mit unterschiedlicher Schriftart beschreiben) und der „UNDO"-Möglichkeit (den letzten Befehl rückgängig machen), um nur einige der insgesamt 11 Wahlarten zu nennen.
Durch die Funktion „Magnify" können Sie Teile des Grafikbildes „zoomen", d.h., ausschnittsweise vergrößern (es sind drei Stufen vorgesehen). Hier wählen Sie mit dem Joystick die Farbe aus, die Sie zum Setzen der Pixels benutzen wollen und plazieren diesen Bildpunkt in-



nerhalb des vorgegebenen Rasters per Druck auf den Feuerknopf. Somit ist grafische Filigranarbeit ohne weiteres möglich.
Der eigentliche Zeichenmodus wird mit „Shapes" aufgerufen, in diesem Menü läßt sich festlegen, ob Sie Linien, Rechtecke oder beispielsweise Kreise zeichnen möchten. Dies geschieht dann in der vorher ausgewählten Zeichenfarbe und der vorgesehenen Strichstärke.
Nun kann es passieren, daß man mit den vom Programm vorgegebenen Füllmustern oder dem Aussehen

KOMFORTABLE ROLLÄDEN

der Schriftzeichen nicht zufrieden ist. Es lassen sich aus den entsprechenden „Rolladen"-Menüs heraus jederzeit entsprechende Editoren (Pattern Editor und Font Editor) aufrufen, mit denen Sie sich eigene Füllmuster oder Zeichensätze für den Text in Ihrer Grafik entwerfen können. Der Font-Editor z.B. hat neun Buchstabengrößen, diese können fett oder kursiv dargestellt werden.

TEILE AUS DER GRAFIK HERAUSTRENNEN

Ebenso wie bei anderen C 64-Grafikprogrammen ist auch beim „Art Studio" eine Möglichkeit vorgesehen, Ausschnitte aus einer Grafik als „Window" zu definieren (die Größe bestimmen Sie selbst), und diesen Teil dann an anderer Stelle wieder einzufügen. Man könnte beispielsweise Teilbildelemente auf Disk speichern, um diese dann später problemlos zu einem neuen Bild zusammenzuhängen oder zumindest darin verwenden.
Jede aufgerufene Funktion (also Shape, Fill, Paint usw.), bleibt so lange aktiv, bis eine andere „angeklickt" wird. Mit der "RUN/STOP"-Taste kommen Sie auch aus jeder Funktion heraus, bei der Eingabe von Text (unter diesem Menüpunkt) wird durch diese Taste der Cursor bei Zeilenende auf die nächste Zeile gesetzt. Der Druck auf den Feuerknopf nützt hier nämlich nichts.

GRAFIKBILDER AUSDRUCKBAR

Haben Sie nun den bereits eingangs erwähnten Druckertreiber für Ihren Drucker eingestellt und abgespeichert, steht einem Ausdruck Ihrer Grafikbilder nach Wahl der Menüfunktion „Print" nichts mehr im Wege. Das „Advanced Art Studio" hat die Werte des Commodore-Druckers MPS 801 vorgegeben. Ein Anschluß am Centronics-Anschluß und dem User-Port Ihres C 64 ist genauso möglich, wie über ein serielles Kabel. Die Lage des Bildes auf dem Druckerpapier (in der Mitte, links – rechts außen) kann ebenfalls von Ihnen bestimmt werden. Bei unserem Redaktionsdrucker klappten die Testausdrucke (nach vorheriger Definition des Installationsprogramms „Studioconfig") einwandfrei.

UNWESENTLICHE ABWEICHUNGEN – OCP ART STUDIO

Das andere Malprogramm auf der zweiten Diskette, „Ocp Art Studio" bietet im Vergleich zum „Advanced" nichts Anderes.
Es arbeitet im HIRES-Modus (hochauflösende Grafik) und fällt dadurch auf, daß der Menüpfeil, der auch hier wahlweise per Tastatur, Joystick, Maus oder Koala Pad betrieben werden kann, ein wenig langsamer als sein Pendant beim erweiterten Programm ist. Das mag sich für nervöse Zeitgenossen aber sogar als Vorteil entpuppen, da sich die einzelnen Menüpunkte auf der Leiste (ebenfalls oben) leichter treffen und anwählen lassen.
Auch hier ist für einen zufriedenstellenden Druckerbetrieb die Änderung des Installationsprogrammes wichtig.
Die Menüleiste am oberen Bildschirmrand ist ein wenig anders aufgeteilt, das ebenfalls sehr ausführliche Handbuch (allerdings beide in Englisch!) ist so aufgebaut wie das andere, so daß sich alle Funktionen leicht und verständlich nachvollziehen lassen.

BILDER LADEN OHNE HAUPTPROGRAMM

Auf beiden Disketten ist das Hilfsprogramm „Screenloader" enthalten, mit dem sich die Grafikbild-Files auch ohne Aufruf und Laden der beiden Mal-Hauptprogramme auf den Bildschirm bringen lassen. So können Sie z.B. eigene Programme entwickeln (Adventures, Lehrdemos und ähnliche), sie mit Hilfe dieses „Screenloaders" laden und auf den Bildschirm bringen.

FAZIT

Das ergänzte „Advanced Ocp Art Studio" ist eines der besten Grafikprogramme für den C 64, das wir bisher getestet haben.
Einen Nachteil hat es allerdings (falls man es als solchen bezeichnen will): es arbeitet mit nur einem Grafikbildschirm. Andere (Hi-Eddi, Starpainter usw.) bieten in dieser Hinsicht mehr. Das hat schon seine Vorteile, vor allem, weil man eine Grafik in ihren Entstehungsstadien in einem anderen Bildschirm zwischenspeichern kann. (Manchmal sehr hilfreich, falls Sie versehentlich Ihre Grafik kurz vor Vollendung gelöscht haben!). Beim „Advanced OCP Art Studio' empfiehlt sich daher, die Bildfiles auf Diskette zwischenzuspeichern.
Wer sich daran nicht stört, dem können wir „Advanced Ocp Art Studio" nur empfehlen. Zu beziehen über den Fachhandel oder direkt bei Ariolasoft. *(hb)*

ENDLICH ENTTARNT- DIE WAHRE BEDEUTUNG DER FEHLERMELDUNG IHRES COMPUTERS

Unser freier Mitarbeiter B. H. aus G. hat in zeitraubender Kleinarbeit die „echte" Bedeutung verschiedener Basic-Fehlermeldungen (immer mit "ERROR" gekennzeichnet) Ihres Computers getestet und erarbeitet. Das Ergebnis dieser schweißtreibenden Arbeit, die sich über 36 Monate hinzog, lesen Sie bitte im folgenden Artikel. Unschwer wird der eine oder andere feststellen, daß sie von den im Handbuch beschriebenen Bedeutungen erheblich abweichen. Da können Sie mal wieder sehen, daß auch auf gar nichts mehr Verlaß ist.



BAD DISK
diese Fehlermeldung tritt ab und zu auf, wenn Sie im Hintergrund – während des Computerns – Ihr Radio oder die Stereoanlage laufen haben. Wann diese Fehlermeldung auftritt, liegt am persönlichen Musikempfinden Ihres Rechners. Unser Tip: Schaffen Sie sich einen Walkman mit Kopfhörern an!

BAD SUBSCRIPT
das bedeutet „unleserliche Unterschrift" auf einem mit dem Textverarbeitungsprogramm „Witze-Schreiber" erstellten Brief oder Schriftstück. Abhilfe bietet hier sich zu bemühen, so zu schreiben, wie man's in der Schule gelernt hat. (Übrigens: Unsere Tests ergaben, daß diese Meldung auch häufig bei einem mit „Wort Fink" erstellten Dokument auftritt!)

BREAK
läßt sich am besten erklären mit „Halt die Klappe", und bedeutet, daß der Computer momentan keinen Bock auf Eingaben hat.

CAN'T CONTINUE
der Computer kann nicht fortfahren, in der Regel ist das Benzin alle. Bitte suchen Sie unverzüglich mit ihm die nächste Tankstelle auf.

CAN'T RESUME
der Computer ist überfordert, aus Ihren hochgeistigen Texten oder Eingaben ein abschließendes Resumee zur Beurteilung zu ziehen. Bitte verfassen Sie den Eingabetext neu und vermeiden Sie geschraubte Ausdrücke und Fremdwörter!

DEVICE NOT PRESENT
Obwohl vom Computer so gewünscht, haben Sie keine weiße Zeichenfarbe bei der Eingabe verwendet. Bitte einstellen!

DIVISION BY ZERO
eine spezielle Meldung, die nur bei Großrechneranlagen von militärischen Dienststellen (Verteidigungsministerium, Kreiswehrersatzämter ect.) im Betriebssystem verankert ist. Sie wird in der Regel dann ausgegeben, wenn ein vorgesetzter Offizier sich Auskunft über den Ausbildungsstand und die Einsatzbereitschaft bestimmter militärischer Einheiten, Division, verschaffen will.

FILE DATA
bei einem Hardware-Fehler, bei dem Sie gezwungen sind, das Computergehäuse zu öffnen, zeigt Ihnen der Rechner genau, bei welcher Schraube Sie die Nagelfeile Ihrer Freundin ansetzen sollen (einen passenden Schraubenzieher haben Sie ja sowieso nicht zur Hand, stimmt's?),

FILE NOT FOUND
trotz intensivster Suche in Handtasche, Badezimmer oder Küchenschrank, wurde die entsprechende Nagelfeile nicht gefunden.

FILE NOT OPEN
die verwendete besagte Feile taugt nichts (zu stumpf oder zu klein) und sollte gegen eine andere ausgetauscht werden. (Evtl. eine passende beim Versandfachhandel bestellen!)

FILE OPEN
es wird versucht, ein bereits offenes Computergehäuse mit einer Nagelfeile zu öffnen. Der Vorgang ist ohne Verzögerung abzubrechen.

FORMULA TOO COMPLEX
Sie verwenden ein viel zu großes Druckerpapier. Abhilfe schafft hier zweimaliges Falten nach rechts, dann das Programm nochmals starten!

ILLEGAL DEVICE NUMBER
das Programm, mit dem Sie gerade im Computer sind, ist eine Raubkopie! Es wird dringend empfohlen, sich beim Fachhandel das Original zu besorgen.

ILLEGAL DIRECT
das soeben erworbene Originalprogramm ist eine Raubkopie eines anderen Originalprogramms. Hier hilft in der Regel nur Ausschalten, das Innere des Computers (Chips, Platinen usw.) auszubauen und das Leer-Gehäuse fortan als Blumenkasten (z.B. für den Balkon) zu verwenden.

ILLEGAL QUANTITY
diese Fehlermeldung erscheint, wenn Sie mehr als 10(!) Stunden vor dem Computer sitzen,

um ein Listing abzutippen. Dem Computer reicht es jetzt.

LOAD ERROR
der Computer entschuldigt sich für die letzte Fehlermeldung

MISSING FILE NAME
zum schnellen Auffinden der bereits vorhin beschriebenen Nagelfeile (zum Öffnen des Computers) sollte dieser immer ein eigens definierter Name gegeben werden, z.B. „Blaues Feilchen" oder so. Fehlt dieser, tritt diese Fehlermeldung auf.

NEXT WITHOUT FOR
tritt nur dann auf, wenn Sie mit dem Computer das beliebte Kartenspiel „17 + 4" spielen und noch eine Karte nehmen wollen, obwohl Sie schon vier Stück haben.

NOT INPUT FILE
der Computer schämt sich, und bittet Sie, sein Gehäuse nicht zu öffnen.

NOT OUTPUT FILE
das Gegenteil zur vorherbeschriebenen Fehlermeldung. Das Öffnen seines Gehäuses mit einer Nagelfeile tut dem Computer sichtlich wohl, in der Regel tritt diese Fehlermeldung dann auf, wenn Sie damit aufhören.

OUT OF DATA
falls Ihnen mitten in einem Programm plötzlich einfällt, Ihren Computer zu fragen, wo er denn eigentlich hergestellt wurde, erscheint diese Fehlermeldung. Er war von Ihrer Frage so überrascht, daß er sie zwar beantwortet (eben aus „Data"), aber mit dieser Fehlermeldung aussteigt. Anmerkung in eigener Sache: Da wir den Ort „Data" auf keiner Landkarte gefunden haben, sind wir für Hinweise aus unserem Leserkreis sehr dankbar!

OUT OF MEMORY
im Prinzip wie vorher, wird aber in aller Regel von größeren Rechnern wie PCs und auch vom 64 ausgegeben. Auch diesen Ort konnten wir nicht ausfindig machen, vermuten aber, daß es irgendwo in Alaska sein könnte.

OVERFLOW
wird bei zu häufigem und übermäßigem Biergenuß während der Arbeit am Computer gemeldet. („overflow" = zu deutsch etwa „es fließt dir ja schon bei den Ohren heraus")

REDIM D'ARRAY
ein französischer Adliger (1723–1789), der als Pionier aus der Gründerzeit der Computerei als Erster entdeckt hat, daß zum Betrieb eines Rechners nur *ein* Schalter genügt. (Die bisherigen Modelle wurden immer mit zwei Schaltvorrichtungen geliefert, eine zum Ein- und eine zweite zum Ausschalten.) Das wirkte sich natürlich auf Benutzerfreundlichkeit und Handhabung des Gerätes denkbar negativ aus. Tritt bei Ihnen diese Fehlermeldung auf, so hat Ihnen Ihr Computerhändler noch so ein Uralt-Gerät aus dieser Zeit angedreht.

RETURN WITHOUT GOSUB
diese Meldung befindet sich erst seit jüngster Zeit in verschiedenen Computer-Systemen, genau: seit den Olympischen Spielen 1984 in Los Angeles. Dort wurden in nahezu perfekt ausgereifter Form alle sportlichen Ereignisse computermäßig erfaßt und gemessen. Die obenstehende Fehlermeldung tritt dann auf, wenn ein Bodenturner (oder -turnerin) zwar einen Salto rückwärts ganz ordentlich ausgeführt hat, aber nicht auf dem Kopf zu stehen gekommen ist. Den Preisrichtern wird so angezeigt, daß hier ein Punktabzug vorzunehmen ist.
Falls Sie eine Turnhalle oder Ihren Fitnessraum als Arbeitsplatz für Ihren Computer gewählt haben, kann diese Fehlermeldung schon ab und zu einmal auftreten.

STRING TOO LONG
Hiermit wird ein Hardware-Fehler festgestellt. Diese Meldung bedeutet, daß das Kabel vom Netzteil Ihres Computers zur Steckdose der Stromversorgung viel zu lang ist. Bitte ein kürzeres Kabel anbringen. Wir haben es zwar noch nicht getestet, sind aber der Meinung, daß sich dies auch sehr positiv auf die Geschwindigkeit Ihres Rechners auswirkt, da die Daten jetzt durch ein kurzes Kabel auch viel schneller übertragen werden.

SYNTAX
ebenfalls ein Computer-Pionier aus der Frühzeit, aus Gallien stammend. So lange er lebt, verstand er es meisterlich, zu verhindern, daß irgendein Vokabel-Trainings-Programm für Latein (auch „Latein-Kurs" genannt) auf den Markt, geschweige denn in die Technik kam. Da sich das nach seinem Tode jedoch grundlegend geändert hat, tritt diese Fehlermeldung häufig noch bei Latein-Lernprogrammen auf. Abhilfe schafft hier nur, leise, aber stetig, die französische Nationalhymne zu summen.

TOO MANY FILES
Sie haben versucht, das Gehäuse Ihres Computers mit mehr als einer Nagelfeile zu öffnen (vielleicht, weil Sie auch mehr als eine Freundin haben!). Eine genügt vollkommen.

TYPE MISMATCH
heißt wörtlich übersetzt „Gib doch nicht so mieses Zeug ein!". Eine Erklärung erübrigt sich.

UNDEF'D STATEMENT
Ihr Computer hat soeben bemerkt, daß Sie, obwohl Sie mit ihm arbeiten, nicht aus Bayern stammen. Hier hilft einzig und allein nur, bei der zuständigen Behörde die bayerische Staatsangehörigkeit zu beantragen. (Als ständiger Zuschauer des Fernsehmagazins „Scheibenwischer" sind Ihre Chancen jedoch denkbar ungünstig!).
Bis zum Abschluß aller Formalitäten (in der Regel 4 – 5 Jahre) reagiert Ihr Computer auf keinerlei Eingaben mehr und läßt sich auch nicht ausschalten, außer Sie legen das für die Stromversorgung zuständige Elektritzitätswerk lahm.

UNDEFINED FUNCTION
vom Rechner wurde festgestellt, daß Sie als Operator für ihn denkbar ungeeignet sind. In den meisten Fällen schafft hier Abhilfe, wenn Sie mal Ihren Sprößling oder Ihre Frau ranlassen!

VERIFY
diese Meldung besitzt eine Zwitterstellung. Sie wird nämlich angezeigt, wenn Ihr Computer von der Qualität Ihrer Eingaben so begeistert ist (Abkürzung für: very fine = sehr schön), daß er einen freudigen Stromstoß durch alle Chips und Platinen jagt, was unweigerlich dazu führt, daß ein „Interrupt" (=Unterbrechung) ausgelöst wird. Obwohl sehr zufrieden mit Ihnen, bringt der Rechner doch eine Fehlermeldung. Wir wollen damit ganz klar wieder eine der vielfältigen Ungereimtheiten gewisser Computerhersteller – und Software-Entwickler des entsprechenden Betriebssystems anprangern. Leider konnten wir diese Fehlermeldung mit all ihren Auswirkungen und Nebenerscheinungen bisher noch nicht austesten, da sie eigenartigerweise während unserer Redaktionstätigkeit noch niemals aufgetreten ist . . . . .

(hb)

TEXTOMAT AMIGA

Fällt der Name Textomat aus dem Hause Data-Becker, so denkt der Computerfreak unwillkürlich an das gleichnamige Produkt für den C 64. Nun gibt es dieses fast schon legendäre Programm auch für den jüngsten Sproß aus der Commodore-Familie, dem Amiga. Diese Textverarbeitung besticht durch Komfort und Schnelligkeit wie bisher keine andere auf dieser Rechner-Familie.

Ein recht niedriger Preis von 99 DM für ein voll ausgereiftes Produkt sowie kein Kopierschutz für diese Textverarbeitung ist wohl ein Novum im Bereich der Amiga-Software. Das in C geschriebene Programm wurde unter Zuhilfenahme des gleichnamigen Produktes vom Atari ST erstellt, welchem es allerdings einige Features und Möglichkeiten voraushat. Ähnlichkeiten sind also rein zufällig, lediglich, als beim neu formatieren die Atari-Fliege als Wartezeiger auf dem Bildschirm erschien, waren einige Redaktionsmitglieder unangenehm überrascht.
Kommen wir zum Programm selbst. Textomat auf dem Amiga überzeugt durch überragende Geschwindigkeit und funktionelle Aufmachung. Obwohl Textomat sich neben der Tastaturbedienung selbstverständlich auch mit der Maus dirigieren läßt, wurde auf überflüssigen Schnickschnack und bunte Farbgebung (wie sie besonders bei amerikanischer Software zu beobachten ist) bewußt verzichtet. Textomat gibt sich betont sachlich, eben als Arbeitstier.
In vier Pulldownmenüs erhält der Benutzer die Gelegenheit, gewünschte Einstellungen und Arbeitsvorgänge anzuwählen. Dabei wurden glücklicherweise die wichtigsten Punkte wie Diskettenoperationen, Formatiervorschriften oder Schriftartwechsel auch über die Kombination einer Taste mit der Amiga-Taste anwählbar gemacht. Bei einigen Kombinationen wurde allerdings auf die allgemein übliche Syntax verzichtet, so wählt man bei Textomat verschiedene Schriftarten nicht mehr mit Angabe des Anfangsbuchstabens (z.B. U für unterstreichen) sondern unter Angabe einer Zahl. Für Anwender, die bereits mit anderen Textprogrammen auf dem Amiga gearbeitet haben, bedeutet dies sicher eine kleine Umstellung.
Dafür macht einem Textomat das „Durchhangeln" durch verschiedene Untermenüs besonders einfach. So genügt beispielsweise bei den meisten Abfragen, etwa zum Laden oder Speichern von Texten, ein Druck auf die Return-Taste, um einen voreingestellten Wert zu akzeptieren. Soll zum Beispiel der Text, der gerade bearbeitet wird, sicherheitshalber zwischengespeichert werden, so genügt die Angabe des Save-Befehls (Amiga-Taste und F) und das zweimalige Drücken der Return-Taste. Im Gegensatz zu anderen Produkten, die einen vergleichsweise umständlichen Mausklick auf das "OK"-Feld erwarten, spart der Anwender hier enorme Zeit.



Ein Bildschirmfoto zeigt die professionelle Aufmachung des Programmes — Hier in der größtmöglichen Schriftart

SCHNELLIGKEIT IST TRUMPF

Um bei der Zeit zu bleiben: Textomat besticht durch eine Verarbeitungsgeschwindigkeit, die dem 68000er Prozessor des Amiga endlich gerecht zu werden erscheint. Ob es sich um das Scrollen von Texten mittels der Cursortasten, die Umformatierung oder Verschiebung von Textblöcken oder das Suchen und Ersetzen handelt, Textomat ermöglicht endlich eine vernünftige Arbeitsgeschwindigkeit. Dies sollte gerade deswegen so betont werden, weil erheblich teurere Produkte Textomat in der Geschwindigkeit zum Teil um 100 Prozent unterlegen sind.
Ein weiteres Feature offenbart sich, wenn in das „Datei"-Menü geblickt wird. Hier findet der Anwender unter dem Punkt „Modus" die Auswahlmöglichkeit „Brief" oder „C-Source". Programmierer horchen jetzt auf: Unter C-Source ermöglicht Textomat durch eine besondere Behandlung der Klammern, die ja für die C-Programmierung von besonderer Bedeutung sind, das Erstellen von C-Programmen. Hier wird klar, welchen Vorteil dies für Programmierer bedeutet: Einen komfortableren Editor für die Programmierung als ein Textprogramm kann sich wohl keiner wünschen.
„Desktop-Publishing" ist ein Zauberwort, welches in letzter Zeit durch die Verwendung immer leistungsfähigerer Prozessoren zum Tagesgespräch wurde. Textomat geht einen kleinen Schritt hin zu dieser Möglichkeit (obwohl es natürlich nicht sämtliche Features eines Desktop-Programmes bietet): Es ermöglicht

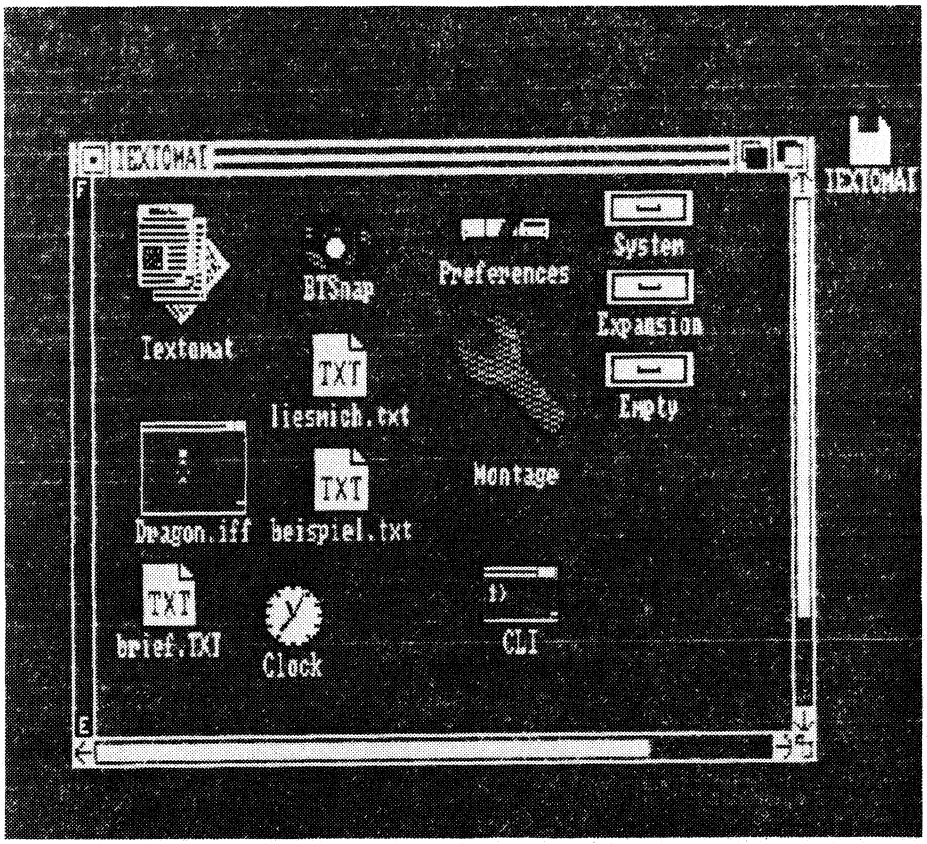
VERWENDUNG VON GRAFIKEN IM TEXT

die Einbindung von herkömmlichen IFF-Grafiken in den aktuellen Text. Hierfür ist eine eigene Ladefunktion im Programm enthalten. Weiterhin wird auf der Programmdiskette ein Utility mitgeliefert, welches aus Programmen heraus Bilder oder Bildausschnitte auskopiert, so daß diese für Textomat einlesbar sind. Dies funktioniert zwar nicht bei allen Amiga-Programmen, stellt aber eine sehr nützliche Erweiterung dar (solche Grafiken können auch mit anderen Malprogrammen weiterverarbeitet werden). Dieses Programm kann übrigens auch auf andere Disketten kopiert werden, um dort seine Arbeit aufzunehmen.
Womit wir bei einem weiteren Punkt wären, dem Kopierschutz. Data Becker hat sich ja in der Szene auch durch die verschiedenen Kopierschütze (die im Endeffekt immer wieder „geknackt" wurden) einen Namen gemacht. Seit neuem verfolgt das Düsseldorfer Haus nun eine neue Linie, in der auf einen Kopierschutz gänzlich verzichtet werden soll. Im Falle des Textomat sieht dies so aus, daß bei Erstbenützung des Programmes eine Art Fragebogen ausgefüllt werden muß, der Name und Anschrift des Kunden sowie die Seriennummer des Produktes enthält. Dieser „Fragebogen" wird dann direkt in das Programm eingetragen. Nach diesem Erstein-

KEIN KOPIERSCHUTZ MEHR

trag kann das Programm beliebig oft kopiert werden (auch aus der Workbench heraus), wobei Data Becker natürlich ausdrücklich auf Sicherheitskopien für den Eigenbedarf besteht. Ob diese neue Art des Schutzes allerdings ihren Zweck erfüllt, bleibt fraglich, denn eintragen kann man bei der Erstbenutzung, was man will. Was allerdings etwas seltsam erscheint, ist die eigentliche „Registrierungskarte" zum Programm.

Diese muß ausgefüllt werden, um, so wörtlich, in den „Genuß des Data Becker Kundenservice zu kommen". Hier gehen die Düsseldorfer wohl doch etwas zu weit, denn was ein Softwarekauf mit den verwendeten Laufwerken, dem installierten Arbeitsspeicher sowie dem benutzten Betriebssystem zu tun hat, entmen, archivieren oder weiter bearbeiten will, einfach über die serielle Schnittstelle an Textomat weiterleiten.
Auch die automatische Silbentrennung des Programmes ist vom Feinsten: Erstens ist uns bislang kein Programm bekannt, welches eine automatische Trennung während der



Eine Hardcopy des Workbench-Bildschirmes mit allen bei Textomat mitgelieferten Programmen.

zieht sich wirklich unserer Kenntnis. Doch diese und noch einige andere Angaben werden auf der besagten Karte vom Kunden verlangt. Dies hat zwar mit dem eigentlichen Programm nichts zu tun, jedoch sollten sich die maßgeblichen Herren bei Data Becker hier noch eine etwas feinfühligere Formulierung überlegen.
Ein nützlicher Punkt erwartet den Anwender unter dem Menü „Datenweg". Hier kann gewählt werden, ob die Lade- und Speichervorgänge wie üblich auf Diskette erfolgen oder über die RS 232-Schnittstelle geschehen. Dies bedeutet, daß Texte direkt von einem Rechner zum anderen übertragen werden können. Gerade bei verschiedenen Computersystemen ist dies ein unschätzbarer Vorteil: So kann ein Anwen-

AUCH RS 232-ÜBERTRAGUNG MÖGLICH

der, der seine alten Texte vom C 64 gerne mit Textomat übernehTexteingabe auf dem Amiga realisiert hat, zweitens funktioniert diese Trennung wirklich nach den Regeln der deutschen Rechtschreibung. Auch die Ausrichtungsmöglichkeiten entsprechen allen Ansprüchen, links- und rechtsbündiger Flattersatz, zentrierter sowie Blocksatz lassen sich realisieren. Dabei geschieht die jeweils angewählte Ausrichtung direkt während der Texteingabe (beileibe keine Selbstverständlichkeit, betrachtet man Konkurrenzprodukte). Auch nachträgliche Umformatierungen, beispielsweise nach der Einfügung eines Textes, werden mit Tastendruck problemlos durchgeführt. Vielschreiber haben einen weiteren Vorteil bei Textomat. Erstens lassen sich die Funktionstasten jeweils dreifach mit jeweils 44 Zeichen belegen (das sind also insgesamt über 1300 Zeichen), zweitens ermöglicht es das Programm per Tastendruck, das aktuelle Datum und die Uhrzeit in den Text zu schreiben. Gerade für Besitzer einer Echtzeituhr bedeutet dies natürlich eine große Hilfe, wenn viele Briefe verfaßt werden. Die Funktionstasten-

DREIFACHE FUNKTIONS-TASTENBELEGUNG MÖGLICH

belegung läßt sich natürlich abspeichern und wird bei jedem Neustart des Programmes automatisch eingeladen. Sie eignet sich also vorzüglich zur Verwendung für sogenannte „Floskeltexte", also immer wiederkehrende Formulierungen (wie beispielsweise „Sehr geehrte Damen und Herren") oder auch als kleine Adressdatei.
Erfreulich ist auch die „Zwischenspeicherung" aktueller Parameter. So merkt sich das Programm automatisch den letztbenutzten Textnamen, es muß nicht jedesmal das komplette Inhaltsverzeichnis der Diskette nachgeladen werden. Auch der Drukkertreiber wird lediglich einmal eingelesen (ganz im Gegenteil zu anderen Amiga-Programmen, welche diesen für jeden Druck neu nachladen). Diese Handhabung spart im Endeffekt sehr viel Zeit und vor allem Nerven: Wer jedesmal auf die Aktion seines Rechners warten muß, nur weil irgendeine Datei von Diskette gelesen werden muß, verliert schnell die Lust und vor allem die Arbeitsfreude.
Kommen wir zum letzten Punkt, dem Handbuch. Dieses erklärt auf über 200 Seiten wirklich alles, was mit dem Programm anzustellen ist. Etwas Kummer kommt lediglich beim nicht ganz vollständigen Stichwortverzeichnis auf. Hierüber läßt sich aber bei der Ausführlichkeit der Erklärungen im Handbuch sicherlich hinwegsehen, der Anwender kommt mit dieser Begleitlektüre sicherlich hinter fast alle Tricks und Kniffe des Programmes.

FAZIT

Textomat Amiga leistet absolut Erstaunliches. Für einen bis dato absolut unüblichen Preis von unter 100,– DM vermag dieses Programm endlich, den Anforderungen eines Vielschreibers auf dem Amiga gerecht zu werden. Die bisher bekannten Programme verwirklichten diese Möglichkeiten (wenn überhaupt) nur in englischer Form oder zu horrenden Preisen. Die Konkurrenz dürfte wohl einiges aufzuholen haben.

Torsten Seibt

PROGRAMME RETTEN NACH DEL

Wie leicht passiert es, daß man sich beim Eingeben eines DEL-Befehls vertippt? Schon fehlt das halbe Programm. Wenn man schon lange daran gearbeitet hat, ist dies sehr ärgerlich. Aber, das Programm ist ja nicht vollständig verschwunden. Bis auf 256 Byte ist alles wieder zu retten. Es gibt zwei Wege dazu. Zunächst geht man mit 'mO' in den Monitor und sucht die 3 Nullen, die den Programmschluß anzeigen, mit dem HUNT-Befehl:
H 1000 2000 00 00 00
Bei einem Programm, das länger als 4 Kilobyte ist, ist die zweite Zahl (2000) durch eine entsprechend höhere zu ersetzen (jeweils um 1000 erhöhen/pro 4 Kilobyte).
Angenommen, die erste Zahl, die angezeigt wird, ist 13E5 (aufschreiben!), dann muß man den alten Programmschluß ab 14E5 suchen, da die folgenden 256 Byte überschrieben wurden:
H 14E5 3000 00 00 00
Wird keine Zahl angezeigt, muß 3000 durch eine höhere Zahl ersetzt werden. Jetzt muß die Lücke von 256 Byte überbrückt werden. Dies geschieht mit dem TRANSFER-Befehl. Befindet sich das alte Programmende bei 1AF7, so lautet die Verschiebung:
T 14E5 1AF7 13E5
Nun gibt es nur noch eine kleine Schwierigkeit. Meist erfolgt das Zusammenfügen der beiden Teile mitten in einer Zeile und so stimmen die Zeilennummern nicht, sondern werden dadurch bestimmt, welche beiden Zahlen an 3. und 4. Stellle der „Klebestelle" stehen. Ist diese Zeilennummer kleiner als die vorhergehenden Zeilennummern, so kann man sie nicht löschen. Deshalb wird eine künstliche Zeilennummer erzeugt. Mit dem „>"-Befehl kann man 8 Byte anzeigen und ändern:
>13E5
(Dies ist die Zahl, die beim ersten Suchbefehl herauskam, sie muß natürlich angepaßt werden.) Nach Return sollte einmal 00 mit sieben weiteren Hexzahlen hinter dem Größerzeichen stehen. Die 3. und 4. Zahl hinter der Null ändert man jeweils in FF um und drückt RETURN.
Um die Linkpointer, die die einzelnen Programmzeilen verknüpfen, richtigzustellen, wird nun ein Hardwarereset gemacht und mit folgender Zeile (RENEW) das Programm restauriert:
POKE4097,1:SYS34840
:SYS34891:CLR
Vergessen Sie nicht, die Zeile an der Schnittstelle zu löschen. Sie hat die Zeilennummer 65535.

DIE SEITE 16 FÜR ALLE 16er!

AUS TECHNISCHEN GRÜNDEN HEUTE AB SEITE 18!

Neue Tips & Tricks für den „starken Kleinen von COMMODORE". Einige wichtige Systemadressen für Assemblerfreaks (und Anfänger). Ein Rettungsboot für DELETE-Vertipper. Einblicke in die Disketten-BAM und Hilfe für SCRIPT/PLUS.



Wer aus irgendeinem Grund eine Abneigung gegen den Monitor hat, kann auch einen anderen Weg einschlagen. Hierzu wird unmittelbar nach dem falschen DEL-Befehl der BASIC-Anfang auf den zweiten Teil angehoben:
POKE43,PEEK(45)
:POKE46,PEEK(46)+1
und mit RENEW die Linkpointer richtiggestellt:
SYS34840:SYS34891:
CLR
Jetzt läßt sich der zweite Teil listen (die erste Zeile muß meistens gelöscht werden) und kann abgespeichert werden. Ein RESET und RENEW stellt den ersten Teil wieder her. Dieser kann dann mit dem zweiten durch Mergen verbunden werden:
POKE43,PEEK(45)–2:
POKE44,PEEK(46):NEW
Laden des 2. Teils und Zurücksetzen des BASIC-Anfangs:
POKE43,1:POKE44,16
Jetzt kann das ganze Programm (dem in der Regel 5 – 7 Zeilen fehlen) abgespeichert werden.

PROGRAMMIERTES GOTO

Während der RESTORE-Befehl auch Variable anstelle von Zahlen zur Angabe der Zeilennummer akzeptiert, muß man bei GOTO immer eine Zahl einsetzen. Dadurch kann man die Sprungadresse im Programm leider nicht auf einfache Weise berechnen, kann aber einen GOTO ZN-Befehl simulieren, wobei das ZN im Programm errechnet oder abgefragt wird.
In unserem Beispielprogramm ist dies in Methode 1 dargestellt. Der Trick besteht darin, den USR-Vektor auf die CONT-Routine umzulenken. Mit Hilfe des RESTORE-Befehls werden dann einige Speicherstellen richtig belegt, was zur Folge hat, daß der nächste Aufruf der USR-Funktion wie ein GOTO ZN wirkt. Am häufigsten wird eine im Programm berechenbare Sprungadresse bei mathematischen Programmen oder bei Spielen, deren Verlauf zufällig sein soll, gebraucht. Man muß aber bei zufällig bestimmten Zeilennummern sorgfältig darauf achten, ob das Sprungziel existiert.
Aber es gibt auch noch eine weniger ungewöhnliche Methode. Diese ist mit Methode 2 vorgestellt. Hier wird der ON...GOTO...-Befehl angewendet. Dabei ist man nicht ganz frei in der Wahl der Sprungadressen, sondern kann nur zwischen bestimmten vorgegebenen Zeilennummern wählen. Dies dürfte jedoch in den meisten Fällen genügen. Besonders nützlich ist die Methode bei Menüauswahl, wobei man Z durch die Lage eines Zeichens in einem

BAM READ

```
10 rem ******* bam read ******
*****                              <hj>
100 dim a%(150),ts%(35,20):rv$
=chr$(18):ro$=chr$(146)            <bd>
110 gosub290:trap270:fori=0to1
43:get#2,a$:a%(i)=asc(a$+chr$(
0)):next:gosub320:a=4              <pg>
120 printchr$(147):char,0,21,r
v$+"000000000111111111122222222
222333333"                         <ha>
130 char,0,22,"123456789012345
678901234567890123456789012345"    <oe>
140 fori=0to20:char,35,i,right
$(" "+str$(i),2):next:printro$     <kf>
150 fort=1to35:z=1:x=0             <ef>
160 m=20:ift>17thenm=18:ift>24
thenm=17:ift>30thenm=16            <lk>
170 fors=0tom                      <fb>
180 w=-((a%(a+z)and2^x)=0):w$=
mid$(str$(w),2):ts%(t,s)=w         <eo>
190 char,t-1,s,w$                  <ca>
200 x=x+1:ifx<8then220             <ei>
210 x=0:z=z+1                      <ih>
220 nexts:a=a+4:nextt:getkeyq$     <fg>
230 scnclr:input"BAM auf Druck
er ausgeben";q$:if q$<>"j"then
270                                <pd>
240 open4,4:fors=0to20:fort=1t
o35:print#4,mid$(b%(t,s),2);:n
extt                               <ji>
250 print#4,right$("    "+str$(
s),4):nexts                        <nb>
260 print#4:print#4"1234567890
123456789012345678901234567890":clo
se4                                <ep>
270 end                            <ip>
280 rem ****** unterprogramme
*****                              <ko>
290 open15,8,15,"i":open2,8,2,
"#"                                <dh>
300 print#15,"u1:"2;0;18;0:rem
 block read (bam)                  <hi>
310 print#15,"b-p:"2;0:return:
rem bufferpointer                  <kh>
320 close2:close15:return          <bl>
```

Vergleichsstring (V$) mit der INSTR-Funktion festlegt.

GOTO ZN MIT TRAP

Wie in den meisten Fällen geht es auch hier noch einfacher und zwar mit dem TRAP-Befehl. Sie wissen ja, daß der TRAP-Befehl bei einem auftretenden Fehler an die angegebene Adresse springt. Und zu unserem Glück ist der TRAP-Befehl mit Variablenangabe möglich. Man braucht also nur "TRAP ZN:*" ins Programm einzufügen. Durch den nicht vorhandenen Befehl "*" wird ein Fehler erzeugt, das Programm springt zur Zeile ZN. Im Beispiel 3 ist gleichzeitig gezeigt, wie zufällige Zeilensprünge mit dem RND-Befehl erzeugt werden können.

UNTERBRECHEN SIE DOCH MAL

Auf vielfachen Leserwunsch wollen wir auf der Seite 16 auch wichtige Systemadressen behandeln. Eine davon ist die Interrupt-Routine bei $CE0E. Hierhin springt der Computer alle 60stel Sekunde, um wichtige Arbeiten zu erledigen (STOP-Taste, Uhr, Cursorblinken...). Will man ihn noch zusätzlich beschäftigen, indem er zuvor immer eine eigene Maschinenroutine durchläuft, so kann man den RESET-Vektor auf die eigene Routine umlenken. Wichtig sind dabei zwei Dinge: Die eigene Routine muß mit JMP $CE0E abgeschlossen werden und es darf während der Umstellung des Vektors kein Interrupt erfolgen. Dies geschieht mit einer kleinen Routine:
Interrupt-Vektor auf $065e verbiegen:

```
. 0675 sei
. 0676 lda #$5e
. 0678 sta $0314
. 067b lda #$06
. 067d sta $0315
. 0680 cli
. 0681 rts
```

Natürlich muß vor dem SYS-Aufruf zum Vektorumstellen (SYS1651) die eigene Routine dort stehen. Als Beispiele dafür haben wir uns zwei nette kleine Programme ausgedacht. Das erste zeigt durch die Rahmenfarbe (steht in $FF19) an, ob der Einfügemodus eingeschaltet ist. Die dabei wichtige Speicherstelle ist 2026 ($07EA, $00 = aus, $80 = ein). Wer viel mit Einfügen arbeitet, wird für diese kleine Hilfe sicher dankbar sein.

```
. 065e lda $07ea
. 0661 bne $066b
. 0663 lda $01
. 0665 sta $ff19
. 0668 jmp$ce0e
. 066b lda $00
. 066d sta $ff19
. 0670 jmp$ce0e
```

Auf ähnliche Weise kann man sich anzeigen lassen, ob ESC aktiv ist. Dies ist vor allem dann hilfreich, wenn die Tastatur schon stark beansprucht ist und die ESC-Taste nicht mehr immer anspricht. In 2027 ($07EB) steht der ASCII-Code des letzten Zeichens. Bei ESC steht dort 27 ($1B). Man kann auf diese Art natürlich auch andere Zeichen abfragen (Menüauswahl bei Maschinenprogrammen).

```
. 065e  lda  $07eb
. 0661  cmp #$1b
. 0663  bne  $066d
. 0665  lda  $00
. 0667  sta  $ff19
. 066a  jmp  $ce0e
. 066d  lda  $01
. 066f  sta  $ff19
. 0672  jmp  $ce0e
```

ZEICHEN AUSGEBEN MIT BSOUT

Eine weitere wichtige Systemadresse ist SFFD2. Diese gibt das Zeichen aus, dessen ASCII-Code im A-Register steht. In BASIC kann man dies ausprobieren, indem man eingibt:
POKE2034,65:SYS65490
Man erhält ein A an der Cursorposition. Um die Anwendung zu zeigen, wollen wir einmal damit den BASIC-Eingabepuffer (512–600) auslesen:

```
. 065e  sei
. 065f  lda  #$6b
. 0661  sta  $0314
. 0664  lda  #$06
. 0666  sta  $0315
. 0669  cli
. 066a  rts
. 066b  sei
. 066c  lda  $0543
. 066f  cmp  #$06
. 0671  bne  $0694
. 0673  ldx  #$00
```

```
. 0675 stx  $d0
. 0677 lda  $0200,x
. 067a beq  $0687
. 067c jsr  $ffd2
. 067f inc  $d0
. 0681 ldx  $d0
. 0683 cpx  #$58
. 0685 bne  $0677
. 0687 inx
. 0688 cpx  #$ff
. 068a bne  $0687
. 068c stx  $0543
. 068f lda  #$0d
. 0691 jsr  $ffd2
. 0694 jmp $ce0e
. 0697 brk
```

Mit dem Monitor eingeben. Die Routine wird mit SYS1630 gestartet und fragt interruptgesteuert die Speicherstelle $0543 (1347) ab, in der die SHIFT/CTRL-Flagge enthalten ist. Diese Speicherstelle enthält normalerweise 0, wenn keine Taste gedrückt ist. Bei der SHIFT-Taste enthält sie 1, bei der C=-Taste 2, bei der CTRL-Taste 4 und bei C= und CTRL-Taste gleichzeitig 6. Wird also die Commodore-Taste zusammen mit der CTRL-Taste gedrückt, so wird der BASIC-Puffer (letzte Eingabezeile) und am Ende ein RETURN ausgegeben ($0D=13). Die kleine Schleife von $0687–$068A ist nur deshalb eingebaut, um Ihnen die Möglichkeit zu geben, den Finger von den Tasten zu nehmen, bevor eine zweite Ausgabe erfolgt. Beachten Sie bei der Anwendung von BSOUT, daß dabei X- und Y-Register verändert und daher gesichert werden müssen. Hat man das Programm aktiviert und testet es mit der Eingabe einer Textzeile (was in der Regel mit SYNTAX ERROR quittiert wird), so erhält man beim Drükken der Commodore-Taste zusammen mit der CTRL-Taste den gleichen Text. Gibt man aber eine BASIC-Zeile im Direktmodus ein, so macht man eine erstaunliche Feststellung: Der ausgegebene Text sieht seltsam verändert aus. Dies liegt nicht an unserer Routine, sondern an der Tatsache, daß

```
GOTO Zeilennummer

-----------------------------------
0 rem ** beispiele fuer goto zn ***
5 rem *** goto zn methode 1 *******
10 input"zahl 1-3";z:zn=z*100:goto4
00
100 print"zeile 100":goto10
200 print"zeile 200":goto10
300 print"zeile 300":goto10
400 poke1281,105:poke1282,141:rem u
sr auf cont
410 restorezn:zn=usr(zn)
500 rem *** goto zn methode 2 *****
510 v$="abc12":print"eingabe "v$:ge
tkeya$:z=instr(v$,a$):ifz=0then510
520 onzgoto600,700,800,600,700
600 print"zeile 600":goto510
700 print"zeile 700":goto510
800 print"zeile 800":goto510
890 rem *** goto zn methode 3 *****
900 a=900+int(rnd(1)*4+.5)*10:trapa
:*
910 print"1":goto900
920 print"2":goto900
930 print"3":goto900
940 print"4":goto900

-----------------------------------
```

Für jeden Computertyp leicht abwandelbar

die BASIC-Befehlsworte in TOKEN (Abkürzungen für die Befehle) umgewandelt werden. Schließlich muß der Interpreter die Zeile ja ausführen können und dazu muß sie ihm zuvor übersetzt werden. Übrigens gelten alle heute behandelten Systemadressen für alle CBM-Computer.

WAS IST EIGENTLICH DIE BAM?

Viele neue Floppy-Besitzer sind unter unseren Lesern dazugekommen. Auch ihnen wollen wir helfen, ihre Diskettenstation besser zu verstehen. Die Informationen stehen auf der Diskette in Paketen (Blocks) zu je 256 Bytes abgespeichert. Diese Blöcke sind in 35 Spuren (Tracks) mit maximal 21 Sektoren (Sectors). In der BAM (Block Availability Map) ist für jeden Block ein Byte vorgesehen. Ist dieses 0, so ist der Block belegt, bei 1 ist er frei. Die BAM steht auf TRACK 18 Sector 0.
Mit dem Programm BAM-READ kann man sich diese Blockbelegung ansehen und ausdrucken lassen. Machen Sie einmal folgenden Vergleich: Zuerst nehmen Sie eine neue Diskette (hier müssen alle Blocks frei sein), dann eine fast leere Diskette (man erkennt, daß die Blöcke nahe TRACK 18 zuerst belegt werden) und zuletzt vergleichen Sie einmal eine vielbenutzte Diskette, auf der oft gelöscht und beschrieben wurde, vor und nach einem COLLECT-Befehl (bei COLLECT werden die von den gelöschten Files noch belegten Blöcke wieder freigegeben).

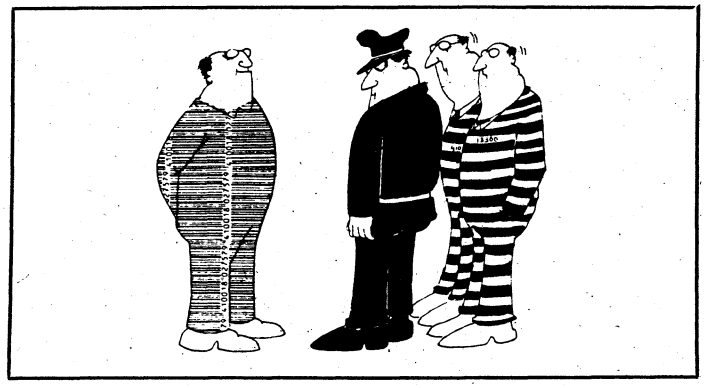
TIPS & TRICKS ZU SCRIPT/ PLUS

Immer wieder kommen Fragen zu SCRIPT/PLUS (das wohl zur Zeit stärkste Textprogramm für den C16/Plus4). Manche Leser klagen über einige Mängel im Programm und über Abweichungen zum Handbuch. Dazu muß zunächst gesagt werden, daß an diesen Mängeln nicht die Entwickler, sondern Commodores Vorliebe für verschiedene ROM-Versionen des Kernal schuld ist (wir werden darüber noch berichten). Es gibt beim C16/ 116 Plus4, soweit uns bekannt ist, mindestens vier verschiedene Versionen (sogar 1 Version mit deutschem Zeichensatz soll es geben). So, wie es im Handbuch beschrieben ist, arbeitet Script/Plus nur mit der damals (als das Programm entwickelt wurde) gängigen ROM-Version zusammen. Kleine Änderungen im ROm führten dann auch zu kleinen Änderungen in der Funktion von Script/Plus. Wenn es Sie interessiert, wann Ihr Script/Plus-Programm fertiggestellt wurde, so drücken Sie einmal im Edit-Modus 'ESC CBM&=' (zuerst die ESC-Taste und dann die Commodore-Taste links unten zusammen mit dem Gleichheitszeichen). Dann erscheint oben links die Script/Plus-Version mit Datum (bei uns steht da: 'V1.13 1mar84').
Wir wollen nun die drei am häufigsten beklagten Probleme aufgreifen und entsprechende Lösungsvorschläge angeben. Diese wurden einer deutschen Befehlsübersicht entnommen, die vom Ingenieurbüro Stechmann (IFI), wo auch Script/Plus vertrieben wird, gegen einen Freiumschlag angefordert werden kann (Sonntagstr. 20, 2152 Horneburg).
PANNING: Bei langen Texten kann man mit der

Schnellscroll-Funktion rascher zu einer weiter unten (oder oben, links, rechts) stehenden Textstelle gelangen. Normalerweise schaltet man das Panning mit ESC gefolgt von P ein und gibt mit der entsprechenden Cursor-Taste die Richtung an. Will man das Panning abschalten, so sollte dies eigentlich mit ESC oder STOP gehen und das Anhalten und Fortfahren mit SPACE. Funktioniert dies bei Ihnen nicht, so drücken Sie zum Abschalten dreimal STOP oder, was oft schneller geht, 3mal kurz eine Cursor-Taste. Anhalten können Sie auch mit ESCP und STOP, um dann mit einer Cursor-Taste das Panning fortzusetzen.

REPLACE: Da die Preise für die Floppy 1551 (die übrigens dreimal so schnell ist wie die 1541) inzwischen schon unter die 300,– DM-Grenze gefallen sind, nimmt die Zahl der C16-Besitzer mit Floppy beständig zu (was besonders bei häufiger Textverarbeitung sehr zu empfehlen ist). Hat man einen Text, der zuvor schon einmal unter gleichem Namen abgespeichert war, eingeladen, bearbeitet,und will ihn nun erneut abspeichern, so wird man mit 'REPLACE FILE?' gefragt, ob das bereits vorhandene File auf Diskette überschrieben werden soll. Wenn bei Ihnen die Floppy bei Eingabe von 'Y' arbeitet, so haben Sie Glück und die passende Kernal-Version in Ihrem Rechner. Sonst springt das Programm wieder in den Edit-Modus. Aber auch diesen Fehler kann man leicht umgehen. Wenn Sie die Floppy 1551 haben, können Sie die Sicherheitsabfrage vermeiden, indem Sie anstelle des Filenamens allein diesen zusammen mit dem Klammeraffen (Replace der Floppy) eingeben. Also statt 'name' geben Sie ein '@:name' (Der Doppelpunkt ist wichtig!). Aber Vorsicht bei der Floppy 1541! Hier kann der Klammeraffe Ihre Files auf Diskette zerstören. Die 1541 hat nämlich einen eingebauten Fehler, durch den der @:-Befehl zwar das vorhandene File ordnungsgemäß überschreibt, dabei aber nicht darauf achtet, ob das neue File vielleicht länger ist. Dann nämlich wird nicht nur das File mit dem gleichen Namen, sondern auch das darauf folgende überschrieben. Es passiert dabei nichts, wenn das File als letztes auf einer fast leeren Diskette steht, aber sonst sollte ein anderes Verfahren gewählt werden. Entweder Sie geben jedem File einen anderen Namen (empfehlenswert ist das Datum als Anhang), oder Sie gehen mit CTRL&0 in den Diskettenmodus und löschen das alte File mit 's:name' und speichern dann normal ab. Sie sollten sich ohnehin angewöhnen, ein File nur dann zu löschen oder zu überschreiben, wenn Sie davon eine Sicherheitskopie besitzen. Abgesehen von den seltenen Fällen, bei denen die Floppystation hängen bleibt, oder der Strom ausfällt, ist auch menschliches Versagen nie ganz auszuschließen.

DISKETTENMODUS: Auch hier gibt es ein kleines Problem. Eigentlich sollte man mit STOP zurück in den Edit-Modus gelangen und mit RETURN einen neuen Diskettenbefehl eingeben können. Geht dies nicht, so gibt man statt dessen 3mal STOP bzw. 3mal CTRL&0 ein. Man hat sich schnell daran gewöhnt und empfindet es nicht mehr als Beeinträchtigung.
Trotz dieser kleinen Mängel können wir Script/Plus empfehlen. Sollten Sie uns noch andere Probleme mitteilen, so werden wir versuchen, Ihnen auch dafür Lösungen anzubieten.
Noch ein letzter Tip am Rande: Normalerweise arbeitet man beim Editieren mit der WORD-WRAP-Funktion, d.h., die Wörter werden als Ganzes auf die nächste Zeile übernommen, wenn sie zu lang werden. Dies erhöht die Lesbarkeit des Textes auf dem Bildschirm. Beim Laden eines langen Textes muß der gesamte Text aber 'reformatiert' werden, was etwas Zeit in Anspruch nimmt. Will man nun nur kurz etwas ändern oder anfügen, so kann man sich diese Zeit sparen, indem man vor dem Laden des Textes die Wordwrap-Funktion mit ESC, gefolgt von W ausschaltet und erst danach wieder einschaltet.
Noch eine Korrektur zu unserer letzten Seite 16. Dort berichteten wir Ihnen von einem stärkeren Netzteil für den C16 (das übrigens auch beim Ingenieurbüro Stechmann erhältlich ist). Hierbei handelt es sich nicht um ein Original ZX80-Netzteil. Unser Test-Netzteil war lediglich teilweise aus ZX80-Teilen aufgebaut (wodurch wir diesem Irrtum erlegen sind), aber durch ein anderes Kabel und einige Änderungen im Inneren an den C16 angepaßt. Wie wir erfuhren, soll bei entsprechender Nachfrage ein speziell auf den C16/Plus4 zugeschnittenes Netzteil hergestellt werden.





DRUCKER-ROUTINE

Das im Sonderheft „128 Spezial 5/87" abgedruckte Listing „Datei.128" (für den 80-Zeichenmodus!) ist bei unseren Lesern aufgrund seiner Funktionen und der Bildschirmdarstellung gut angekommen. Eines aber wirkte ein bißchen störend: eingegebene oder gespeicherte Daten (z.B. Adressen) konnten nicht ausgedruckt werden, weil der Autor dies in seinem Programm nicht vorgesehen hatte. Viele Leser baten uns telefonisch, ihnen hier weiter zu helfen.
Günther Kl. Übbung, ebenfalls Leser der Commodore Welt, hat uns just im richtigen Augenblick eine von ihm entworfene Druckerroutine für dieses Programm zugesandt. Sie fügt sich nahtlos in das Originalprogramm ein und wird angewählt unter dem Punkt „Notice" im Hauptmenü. Jetzt erscheint unter dem Hinweis „Instructions" noch ein zusätzliches Menü:

< , > Datensatz − 1 = zurückblättern
< . > Datensatz + 1 = vorwärtsblättern
< < > Datensatz −10 = 10 Datensätze zurück
< > > Datensatz +10 = 10 Datensätze vor
< D > Datensatz drucken

Außer einem komfortablen Vorwärts- bzw. Zurückblättern innerhalb der Datei erhalten Sie durch Betätigen der Taste "D" eben den heißersehnten Ausdruck des Datensatzes (Adresse usw.) auf einem Commodore-Drucker oder einem mit kompatiblen Interface. Die Sekundäradresse ist mit "7" festgelegt, d.h., Klein-/Großschrift. Sollte Ihr Drucker oder Ihr Interface eine andere Sekundäradresse brauchen, sehen Sie bitte im Handbuch nach. Außerdem muß der Drucker über den seriellen Port mit dem Computer verbunden sein (nicht Userport!).
Am besten tippen Sie die im folgenden Listing abgedruckten Zeilen zu "Datai.128" dazu und speichern das gesamte Programm erneut ab.

Günther Kl. Übbing/hb)

```
Druckerroutine DATEI.128     - 1 -


741 ifa$-"d"theny-y:gosub61000 <nk>
780 getkeya$:ifa$-chr$(13)then
color5,2:return:elseifa$<>","a
nda$<>"."anda$<>"<"anda$<>">"a
nda$<>"d"then780                <ee>
811 ifa$-"d"theny-y:gosub61000 <cb>
1130 print" [,] Datensatz=Dat
ensatz-1":print" [.] Datensatz
=Datensatz+1":print" [<] Date
nsatz=Datensatz-10":print" [>
] Datensatz=Datensatz+10"       <hm>
1131 print" [D] Datensatz aus
drucken ";:return               <oo>
1150 print"    [SPACE] = Weite
r suchen":print"    [RETURN] =
 Suche beenden":print" "aa$"a
tte":printc2$left$(qr$,6)" suc
ht z.B nach P1"aa$"atte"        <gl>
1151 printtab(8)rn$"Taste zum
Menue"rf$                       <nl>
1160 print"    Datensatzausdru
ck [D]";:return                 <hi>
61000 close4:open4,4,7:open3,3
:printchr$(19);                 <nf>
61100 forz=1to10:b$=" ":k=39:f
ors=1tok:get#3,a$:b$=b$+a$:nex
ts:printchr$(157)               <lb>
61200 ifmid$(b$,k,1)=" "thenk=
k-1:b$=left$(b$,k):ifk>0then61
200                             <hj>
61300 print#4,b$:nextz:close3:
close4:return                   <lk>
```

DIRECTORY ANZEIGEN

Der C 64 kennt einen komfortablen Befehl zum Laden und Anzeigen des Disketteninhaltsverzeichnisses (Directory), den Sie sicher alle kennen: LOAD"S",8 mit anschließendem "LIST".

```
10 rem directory c 64           <pl>
20 rem by christian kroell      <km>
30 rem    commodore welt        <io>
40 for i=0to101:read x:poke 49
152+i,x:next:new                <ai>
50 data169,1,162,205,160,235,3
2,189,255,169,8,133,186,169,96
,133,185,32,213                 <op>
60 data243,165,186,32,180,255,
165,185,32,150,255,169,0,133,1
44,160,3,132,251                <ao>
70 data32,165,255,133,252,164,
144,208,52,32,165,255,164,144,
208,45,164,251                  <lo>
80 data136,208,233,166,252,32,
205,189,169,32,32,210,255,32,1
65,255,166,144                  <ml>
90 data208,23,170,240,6,32,210
,255,24,144,240,169,13,32,210,
255,32,225,255                  <ld>
100 data240,4,160,2,208,193,76
,66,246                         <ig>
```

Einen Nachteil hat die ganze Sache allerdings: Das „Directory" wird an den Anfang des Basic-Speichers geladen, ein Programm, das sich im Speicher befindet, wird überschrieben und damit gelöscht.
Es gibt nun in Basic eine recht elegante Lösung, die den C 64-Insidern bestimmt auch nicht mehr neu ist: Den Basic-Anfang vor dem Laden des „Directory" hochsetzen, und zwar hinter das Ende des im Speicher stehenden Programms, so daß dieses erhalten bleibt. Ihre Eingabe im Direktmodus muß dann so aussehen:
POKE 44, PEEK(46)+1
Wenn Sie dann das „Directory" auf die vorher besprochene Weise la-

den, können Sie es mit "List" ansehen bzw. auf den Bildschirm bringen. Um wieder in Ihr vorheriges Programm zu kommen, geben Sie bitte ein: POKE 44,8. Das „Directory" ist zwar weg, aber das Hauptprogramm wieder da.
Ein wenig umständlich, aber es funktioniert.
Wem dies trotzdem zu diffizil erscheint, der tippe sich das kleine Listing „Directory C 64" ab.
Die Ausgabe des Diskteninhaltsverzeichnisses bewirkt nun dieses Maschinenprogramm, das Sie jederzeit – direkt oder innerhalb eines z.B. Dateiverwaltungsprogrammes – mit dem Befehl "SYS 49152" aufrufen können. *(hb)*

FASTBACKUP 1541

Dieses schnelle Kopierprogramm ist in der Lage, eine komplette Diskette zu duplizieren. Dazu müssen Sie nur dreimal die Disketten wechseln.
Wie schon aus dem Programmnamen ersichtlich, arbeitet es aber nur mit dem Betriebssystem der Floppy 1541 zusammen, mit der 1570 oder 1571 funktioniert es leider nicht. Es spielt allerdings keine Rolle, ob Sie den C 64 oder den C 128 im 64er-Modus dafür benutzen.
Die Floppy 1541 wird darüber hinaus beim Kopiervorgang um das 10 – 12fache beschleunigt.
„Fastbackup.1541" ist in Maschinensprache geschrieben und belegt den Speicherbereich Ihres C 64 von 2048 ($0800) bis 4369 ($1111). Gestartet wird es mit "SYS 2061". Beim Abtippen des Programms beachten Sie bitte die folgenden Eingabehinweise (den „Checksummer" bitte vorher laden und starten!).

1) Das Listing „Fastbackup-Lader" abtippen und als erstes Programm auf Disk speichern.
2) Jetzt „Fastbackup1541.Source" eingeben, ebenfalls auf Disk speichern.
3) Tippen Sie nun den „MC-Creator" ab und speichern diesen ebenfalls.
4) Starten Sie jetzt das Programm „MC-Creator" (wenn es sich nicht mehr in Ihrem C 64 befindet, muß es selbstverständlich vorher geladen werden).
5) Der Basic-Anfang wird jetzt nach 8192 ($2000) verlegt, dann das Quellprogramm „Fastbackup. Source" geladen und sofort gestartet. Zunächst werden die Daten eingelesen und in die entsprechenden Speicherstellen gePOKEt. Danach entsteht das Maschinenprogramm „Fastbackup.1541" auf Ihrer Diskette. Bitte werden Sie nicht ungeduldig, wenn dieser gesamte Vorgang, ein wenig dauert. Kommt die Floppy wieder zum Stillstand und es erscheint "READY" mit dem Cursor, dann ist alles in Ordnung.
6) Laden Sie jetzt das Hilfsprogramm „Fastbackup. Lader" und starten Sie es mit "RUN". Nun lädt der C 64 das soeben erzeugte Maschinencode-File nach. Es wird automatisch mit "SYS 2061" gestartet.

Nun meldet sich das Kopierprogramm mit dem Hinweis "Quelldiskette einlegen" (damit ist die Disk gemeint, die Sie kopieren möchten). Nach Druck auf jede beliebige Taste beginnt der Kopiervorgang.
Wenn die LED-Anzeige Ihrer Floppy dabei dauernd blinkt, sollte Sie das nicht kümmern. Falls Sie die Bildschirmanzeige nun auffordert, die „Zieldiskette" einzulegen. (damit ist die gemeint, auf die kopiert werden soll), wechseln Sie bitte die Disketten aus und fahren durch Tastendruck mit dem Kopieren fort. Die ganze Prozedur müssen Sie dreimal wiederholen, dann ist die Kopie fertig. Es lassen sich nun beliebig viele weitere Disketten nach dem selben Schema duplizieren, Sie brauchen dazu nur wieder irgendeine Taste zu drücken. Um „Fastbackup.1541" abzubrechen oder ganz zu beenden, schalten Sie bitte kurz Ihren C 64 und die Floppy 1541 aus.
Wichtiger Hinweis: Dieses Kopierprogramm ist nur in der Lage, eigene Daten- oder Programmdisketten zu vervielfältigen, bzw. sich davon Sicherheitskopien zu erstellen. Dazu ist es nämlich gedacht. Falls Sie versuchen sollten, sich damit unberechtigte Kopien von urheberrechtlich und meistens auch kopiergeschützten Originaldisketten anzufertigen, werden Sie leider scheitern, denn das kann das Programm nicht. Schlimmstenfalls kann es Ihnen sogar passieren, daß diese Originaldiskette dadurch beschädigt und unbrauchbar wird. Also besser Finger weg von solchen Manipulationen. *(Ralph Seyfarth/hb)*

```
Fastbackup1541 MC-Creator    - 1 -
10 rem fbackup mc.creator ==64 <eh>
20 rem 09/87 commodore welt=== <ec>
30 rem ======================= <ci>
40 rem 09/87 commodore welt=== <fg>
50 rem                 h.b.=== <fo>
60 rem version 2.0 40z/ascii== <fb>
70 rem c 64 + floppy 1541  === <aj>
80 rem ======================= <fk>
100 printchr$(147)chr$(17)chr$
(17)"poke43,1:poke44,32:poke81
92,0:new"                      <ah>
110 printchr$(17)chr$(17)"load
"chr$(34)"fb.source"chr$(34)",
8"                             <nm>
111 fori=1to5:print:next:print
"run":printchr$(19);           <ca>
120 fori=0to5:poke631+i,13:nex
t                              <kh>
140 poke198,5:end              <oa>
```

```
Fastbackup1541 Lader         - 1 -
10 rem fastbackup1541 lader=64 <fh>
20 rem 09/87 commodore welt=== <ec>
30 rem ======================= <ci>
40 rem 09/87 commodore welt=== <fg>
50 rem                 h.b.=== <fo>
60 rem version 2.0 40z/ascii== <fb>
70 rem c 64 + floppy 1541  === <aj>
80 rem ======================= <fk>
100 printchr$(147)chr$(17)chr$
(17)"poke43,1:poke44,32:poke81
92,0:new"                      <ah>
110 printchr$(17)chr$(17)"load
"chr$(34)"fastbackup.1541"chr$
(34)",8,1"                     <ip>
111 fori=1to6:print:next:print
"sys2061":printchr$(19);       <pe>
120 fori=0to5:poke631+i,13:nex
t                              <kh>
140 poke198,5:end              <oa>
```

```
10 rem fastbackup source====64 <km>
20 rem 08/87 commodore welt=== <eb>
30 rem ======================= <ci>
40 rem 08/87 r. seyfarth    === <fk>
50 rem                      === <hi>
60 rem version 2.0 40z/ascii== <fb>
70 rem c 64 + floppy 1541   === <aj>
80 rem ======================= <fk>


100 fori=2061to4369:reada:poke
i,a:next:goto1480               <pi>
110 data162,255,154,189         <og>
120 data 27,  8,157,247,  0,20
2,208,247, 76,  0,  1,169,  0,
 32,113,168, 76                 <df>
130 data174,167,120,230,  1,16
0,  0,169, 16,133,174,169, 17,
133,175,169, 19                 <jp>
140 data133,173,132,172,169,25
5,199,172,208,  2,198,173,199,
174,208,  2,198                 <ga>
150 data175,177,174,145,172,16
5,175,201,  8,208,232,165,174,
201,156,208,226                 <lf>
160 data162,  1,134,174,169,
8,133,175,177,172,201, 25,208,
 21, 32,102,  1                 <eg>
170 data177,172,170, 32,102,
1,177,172,145,174, 32,113,  1,
202,208,248,240                 <od>
180 data  5,145,174, 32,113,
1, 32,102,  1,208,219,169, 55,
133,  1, 88, 76                 <jl>
190 data  5,  8,230,172,208,
6,230,173,165,173,201, 19, 96,
230,174,208,  2                 <dn>
200 data230,175, 96, 25,  4,
0,169,  5,141, 32,208,169, 14,
141, 33,208, 76                 <oj>
210 data 26,  8,  0,  3,  0, 1
3,226, 17,108,  6,173, 18,  8,
133,253,173, 19                 <op>
220 data  8,133,254,173, 20,
8,133,251,173, 21,  8,133,252,
 56,173, 22,  8                 <pf>
230 data237, 20,  8,133,249,17
3, 23,  8,237, 21,  8,133,250,
 32,231,255,169                 <gj>
240 data 15,162,  8,168, 32,18
6,255,169,  0, 32,189,255, 32,
192,255, 56,165                 <ma>
250 data249,233, 34,133,249,16
5,250,233,  0,133,250,144,  4,
169, 34,208, 12                 <al>
260 data165,249, 73,255,133,24
8,169, 34,229,248,240, 85,133,
248,162, 15, 32                 <jm>
270 data201,255,169, 77, 32,21
0,255,169, 45, 32,210,255,169,
 87, 32,210,255                 <gd>
280 data165,253, 32,210,255,16
5,254, 32,210,255,165,248, 32,
210,255,160,  0                 <ig>
290 data177,251, 32,210,255,20
0,196,248,208,246, 32,204,255,
192, 34,144, 29                 <nf>
300 data 24,165,251,105, 34,13
3,251,165,252,105,  0,133,252,
 24,165,253,105                 <jd>
310 data 34,133,253,165,254,10
5,  0,133,254, 76, 82,  8,162,
 15, 32,201,255                 <od>
320 data169, 77, 32,210,255,16
9, 45, 32,210,255,169, 69, 32,
210,255,173, 24                 <gh>
330 data  8, 32,210,255,173, 2
5,  1, 25,  8, 32,210,255, 32,
204,255, 32, 18                 <cf>
340 data 11,147, 17,  5, 14, 2
5, 12, 32,198,193,211,212, 45,
194,193,195,203                 <ah>
350 data213,208, 32,214, 51, 4
6, 52, 17, 17, 13, 25,  7, 32,
 40,215, 41, 32                 <ac>
360 data 49, 57, 56, 55, 32, 6
6, 89, 32,210, 65, 76, 70, 32,
211, 69, 89, 70                 <bd>
370 data 65, 82, 84, 72,  5, 1
7, 13, 25,  3,  5, 25,  6, 32,
 40,195, 41, 32                 <pp>
380 data 49, 57, 56, 55, 32, 6
6, 89, 32,195,193, 32,214, 69,
 82, 76, 65, 71                 <gg>
390 data 83, 32,199, 77, 66,20
0, 25,  6, 32, 25,  7,  5,146,
 32, 18, 11, 13                 <kn>
400 data 17,209, 85, 69, 76, 7
6, 68, 73, 83, 75, 69, 84, 84,
 69, 32, 69, 73                 <da>
410 data 78, 76, 69, 71, 69, 7
8,  0, 32,200, 10,169,165, 32,
 56, 11,169,  0                 <df>
420 data162, 16,133,  3,134,
4, 32,106, 11,240, 31,  8,160,
  0,145,  3,230                 <dh>
430 data  3, 40, 48, 12, 32,10
6, 11,145,  3,200,208,248,230,
  4,208,229, 41                 <lc>
440 data 31,170,202,134,  6, 7
6, 97, 10, 32,106, 11, 16,  7,
 41, 63,133,  5                 <co>
450 data 76,144,  9, 32,161, 1
0, 32, 18, 11, 13, 17,218, 73,
 69, 76, 68, 73                 <fi>
460 data 83, 75, 69, 84, 84, 6
```

```
9, 32, 69, 73, 78, 76, 69, 71,
 69, 78, 32,  0                    <ab>
470 data 32,200, 10, 32, 56, 1
1,169,  0,133,  3,169, 16,133,
  4,160,  0, 32                    <dm>
480 data106, 11,208, 23,177,
3,  8, 32, 56, 11,230,  3, 40,
 48,240,177,  3                   <fi>
490 data 32, 56, 11,200,208,24
8,230,  4,208,228, 16, 15, 41,
 63,133,  5,165                   <cm>
500 data  3,166,  4,133, 48,13
4, 49, 76,239,  9,201,  3,208,
  6, 32,161, 10                   <hm>
510 data 76,100,  9,201,  9,20
8, 11,165, 48,166, 49,133,  3,
134,  4, 76,239                   <pe>
520 data  9,201, 36,208,  3, 7
6,156, 11, 32,161, 10, 32, 18,
 11, 13, 17,211                   <ki>
530 data 67, 72, 82, 69, 73, 6
6, 45,198, 69, 72, 76, 69, 82,
 32,211, 80, 85                   <nd>
540 data 82, 32,  0,165,  5, 3
2,171, 10, 76,156, 11,169, 54,
133,  1, 32, 18                   <jk>
550 data 11, 13, 17,204, 69, 8
3, 69, 45,198, 69, 72, 76, 69,
 82, 32,211, 80                   <ac>
560 data 85, 82, 32,  5,  5,
0,165,  5, 32,171, 10, 32, 18,
 11, 32,211, 69                   <bd>
570 data 67, 84, 79, 82, 32,
0,165,  6, 32,171, 10,169, 13,
 32,210,255,169                   <da>
580 data 52,133,  1,120, 76,14
4,  9,169, 54,133,  1,169, 27,
141, 17,208, 96                   <gl>
590 data248,170,240,  8,169,
0, 24,105,  1,202,208,250,216,
 72, 25,  4, 74                   <gb>
600 data 32,193, 10,104, 41, 1
5,  9, 48, 76,210,255, 88,238,
 32,208, 32,228                   <ko>
610 data255,208,  3, 76,201, 1
0, 76,  3, 11, 25, 44,  0,169,
 11,141, 17,208                   <am>
620 data160, 32,202,208,253,13
6,208,250,120, 96,104,133, 53,
104,133, 54,160                   <ac>
630 data  0,230, 53,208,  2,23
0, 54,177, 53,240, 11, 32,210,
255,230, 53,208                   <nm>
640 data245,230, 54,208,241,23
0, 53,208,  2,230, 54,108, 53,
  0,133,107,169                   <fg>
650 data 53,133,  1,169, 11,14
1,  0,221,173,  0,221, 16,251,
 169,  3,141,  0                  <fe>
660 data221,162,  4,169,  3, 7
0,107,106, 70,107,106, 74, 74,
234,141,  0,221                   <ae>
670 data202,208,239,162,  1,20
2,208,253,169, 52,133,  1, 96,
169, 53,133,  1                   <ep>
680 data169, 11,141,  0,221,17
3,  0,221, 16,251,169,  3,141,
  0,221,162,  5                   <kf>
690 data202,234,208,252,162,
4,173,  0,221, 10,  8, 10, 38,
107, 40, 38,107                   <oi>
700 data202,208,242,169, 52,13
3,  1,165,107, 73,255, 96, 32,
161, 10, 32, 18                   <gl>
710 data 11, 13, 17, 32, 32,20
3, 79, 80, 73, 69, 32, 70, 69,
 82, 84, 73, 71                   <cm>
720 data 32, 33, 32, 32, 13,
0, 32,200, 10, 32,161, 10, 76,
233,  8, 25,  0                   <mi>
730 data255, 25, 63,255,165,
0, 41,  6,240, 17,201,  2,208,
  3, 76, 26,  3                   <cg>
740 data201,  4,208,  3, 76,19
7,  5, 76, 78,  4, 76,158,253,
165, 68, 25,  5                   <fl>
750 data 74,170,189, 74,  4,13
3, 12,165,119,208,  4,230,119,
165, 67,133, 10                   <kh>
760 data169,  5,133,  9,169, 9
0,133, 75,169, 82,133, 36, 32,
 86,245, 80,254                   <gj>
770 data184,173,  1, 28,197, 3
6,240,  9,198, 75,208,239,169,
 10, 76,105,249                   <nn>
780 data162,  0, 80,254,184,17
3,  1, 28,149, 37,232,224,  7,
144,243, 32,151                   <in>
790 data244,165, 22, 69, 23, 6
9, 24, 69, 25,  1, 25, 69, 26,
240,  7,198,  9                   <im>
800 data208,192, 76, 30,244,16
5, 24,197,  6,240,  3, 76, 11,
244,133, 34,165                   <kl>
810 data 22,166, 23,133, 18,13
4, 19,166, 25,  1, 25,232,232,
228, 67,144,  2                   <hg>
820 data162,  0,169,  2,133, 4
9, 76, 61,  4,202,208, 21, 76,
 25,  1, 25,  4                   <fc>
830 data134,  7,134, 25,  1, 2
5,138, 69, 24, 69, 23, 69, 22,
133, 26, 32, 52                   <mn>
840 data249,162, 90, 32, 86,24
5, 80,254,184,173,  1, 28,217,
 36,  0,208,218                   <nl>
```

```
850 data200,192,  8,208,240, 3
2, 86,245, 80,254,184,173,  1,
 28,145, 48,200                  <ak>
860 data208,245,160,186, 80,25
4,184,173,  1, 28,153,  0,  1,
200,208,244, 32                  <ai>
870 data224,248,165, 56,197, 7
1,208, 45, 32,233,245,197, 58,
240, 11,166,  7                  <gh>
880 data189, 80,  1,208, 35,16
9,  5,208, 33,230, 25,  1, 25,
165, 25,  1, 25                  <ph>
890 data 32, 94,  5,185,  0,
2, 32, 94,  5,200,208,247,173,
  0, 28, 73,  8                  <go>
900 data141,  0, 28,169,255, 4
4,169,132,166,  7,  9,128,157,
 80,  1,  8,160                  <ib>
910 data  0,140,254,  2, 40, 1
6,  7,198, 10,208,  3, 76,158,
253,138, 24,101                  <hb>
920 data 12,197, 67,144,  2,22
9, 67,170,189, 80,  1, 16,  5,
232,138, 76, 54                  <jm>
930 data  4, 76,162,  3,  9,
9, 10, 11,173, 12, 28, 41, 31,
  9,192,141, 12                  <ij>
940 data 28,169,255,141,  3, 2
8,169, 85,141,  1, 28,165, 68,
 25,  5, 74,170                  <lc>
950 data189,190,  5,133, 12,16
0,  0,132, 10,162,  0,165, 57,
153,  0,  2,200                  <id>
960 data200,165, 10,153,  0,
2,200,165,  6,153,  0,  2,200,
165, 19,153,  0                  <fa>
970 data  2,200,165, 18,153,
0,  2,200,169, 15,153,  0,  2,
200,153,  0,  2                  <pe>
980 data200,185,250,  1, 89,25
1,  1, 89,252,  1, 89,253,  1,
153,249,  1,230                  <mb>
990 data 10,165, 10,197, 67,14
4,192,152, 72,169,  2,133, 49,
 32, 48,254,104                  <jn>
1000 data168,136,185,  0,  2,1
53, 69,  2,136,208,247,173,  0
,  2,141, 69,  2                 <kg>
1010 data 32,245,253,169,  0,1
33, 50,169,255,141,  1, 28,162
,  5, 80,254,184                 <hj>
1020 data202,208,250,162, 10,1
64, 50, 80,254,184,185,  0,  2
,141,  1, 28,200                 <ge>
1030 data202,208,243,162,  9,
80,254,184,169, 85,141,  1, 28
,202,208,245,169                 <cn>
1040 data255,162,  5, 80,254,1
84,141,  1, 28,202,208,247,169
, 85, 80,254,184                 <ak>
1050 data141,  1, 28,162,188,
80,254,184,169,221,141,  1, 28
,232,208,245,160                 <df>
1060 data  0, 80,254,184,169,2
21,141,  1, 28,136,208,245,169
, 85,166, 12, 80                 <of>
1070 data254,184,141,  1, 28,2
02,208,247,165, 50, 24,105, 10
,133, 50,198, 10                 <ag>
1080 data208,142, 80,254,184,
80,254,184, 32,  0,254,169,  0
,141,254,  2,133                 <ok>
1090 data 80,169,  1, 76,105,2
49,133,  8, 44,  0, 24, 16,251
,169, 16,141,  0                 <pn>
1100 data 24, 44,  0, 24, 48,2
51,162,  4,169,  0,  6,  8, 42
, 10,  6,  8, 42                 <kb>
1110 data 10,141,  0, 24,202,2
08,240,162,  1,202,208,  0,234
,234,169, 15,141                 <oe>
1120 data  0, 24, 96, 44,  0,
24, 16,251,169, 16,141,  0, 24
, 44,  0, 24, 48                 <eg>
1130 data251,162,  4,202,208,2
53,142,  0, 24,162,  4,173,  0
, 24, 74,  8, 74                 <bh>
1140 data 74,102,  8, 40,102,
 8,202,208,241,169, 15,141,  0
, 24,165,  8, 96                 <fk>
1150 data  8, 11, 16,  7, 76,
93,  6,165, 67,133, 10, 32,  0
,254,169,  0, 32                 <hc>
1160 data 94,  5,234, 32,142,
 5, 48,235,170,202,134,  7,169
,  2,133, 49,160                 <kj>
1170 data  0, 32,142,  5,153,
 0,  2,200,208,247, 32,233,245
,133, 58, 32,143                 <jh>
1180 data247,173,254,  2,133,1
24,169,  0,141,254,  2,133, 80
,165, 15,240,  3                 <bd>
1190 data 76,172,  7, 32, 16,2
45,162,  9, 80,254,184,202,208
,250,169,255,141                 <jl>
1200 data  3, 28,173, 12, 28,
41, 31,  9,192,141, 12, 28,169
,255,162,  5,141                 <jn>
1210 data  1, 28,184, 80,254,1
84,202,208,250,160,187,185,  0
,  1, 80,254,184                 <he>
1220 data141,  1, 28,200,208,2
44,165,124,141,254,  2,177, 48
, 80,254,184,141                 <nj>
1230 data  1, 28,200,208,245,
80,254, 32,  0,254,173,  0, 28
```

```
, 73,  8,141,  0                    <eh>
1240 data 28,169,  0,141,254,
 2,198, 10,240,  3, 76,204,  5
, 76,158,253,120                    <pi>
1250 data169,  7,133, 71,162,
 0,202,208,253,169,  1,133,  6
,169,176, 32,163                    <gl>
1260 data  7,169,224, 32,163,
 7, 32,142,  5,165,  6,133, 13
,169,  0, 32, 94                    <hi>
1270 data  5,165,  6,  9,128,
32, 94,  5, 32,152,  7,169,  0
,133,120,133,131                    <hj>
1280 data133,119,133, 16,169,2
26, 32,163,  7,201,  2,144, 22
,165, 16,208, 18                    <la>
1290 data169,192, 32,163,  7,1
69,176, 32,163,  7,230, 16,169
,  0,133,120,133                   <mf>
1300 data131,162,  0,134,  9,1
89, 80,  1,201,255,208,  8,232
,228, 67,208,244                    <jl>
1310 data 76,240,  6,164,120,1
85,219,254,240, 10,169,  0,157
, 80,  1,230,  9                    <gn>
1320 data 76,201,  6,134,  7,2
32,138,  9,128, 32, 94,  5,166
,  7, 76,201,  6                   <in>
1330 data165,  9,240, 23,133,1
19,165,131, 56,249,219,254,133
,131,185,219,254                    <op>
1340 data 88, 32,118,214,120,2
30,120, 76,159,  6,165,131, 88
, 32,118,214,120                    <fb>
1350 data230,  6,165,  6,201,
12,240, 11,201, 23,240,  7,201
, 36,240,  3, 76                    <ba>
1360 data134,  6,169,  0, 32,
94,  5,169,  1, 32, 94,  5,234
, 32,142,  5,165                    <hl>
1370 data 13,133,  6,169,176,
32,163,  7,165,  6,  9,128, 32
, 94,  5,169,  5                    <mj>
1380 data133, 14,169,  9, 32,
94,  5,169,230, 32,163,  7,169
,228,162,  0,134                    <ja>
1390 data 15, 32,163,  7,201,
 2,176, 16,169,  9, 32, 94,  5
,230, 15,169,228                    <dd>
1400 data 32,163,  7,201,  2,1
44,  9,198, 14,208,213,169,  5
, 76,146,  7,230                    <gn>
1410 data  6,165,  6,201, 12,2
40, 11,201, 23,240,  7,201, 36
,240, 11, 76, 60                    <em>
1420 data  7,169,  3, 32, 94,
 5, 76,127,  6, 32, 94,  5, 76
,108,  6,169,  0                    <df>
1430 data162, 20,157, 80,  1,2
02, 16,250, 96,133,  0, 88,165
,  0, 48,252,120                    <kl>
1440 data 96, 32, 10,245,160,1
87,185,  0,  1, 80,254,184, 77
,  1, 28,208, 26                    <jj>
1450 data200,208,242,165,124,1
41,254,  2,177, 48, 80,254,184
, 77,  1, 28,208                    <jk>
1460 data  8,200,192,253,208,2
41, 76, 80,  6,169,  0,141,254
,  2, 76,197,246                    <bf>
1470 data255,255,  0,  0,  1,
25,  5,  0,  1, 32                  <pn>
1480 aa=2061:open1,8,1,"fastba
ckup.1541"                          <oc>
1490 print#1,chr$(aa-int(aa/25
6)*256);                            <gh>
1500 print#1,chr$(aa/256);          <pf>
1510 fori=0to2308                   <ai>
1520 print#1,chr$(peek(aa+i));      <mf>
1530 next:close1:end                <eg>
```

BÖRSE

Bin umgestiegen! Verk. VC-20 m. 32-27-KB-Erw. u. Spielkass. f. 120,– DM. Su. möglichst billig Prog. aller Art f. C64/128 auf Disk. Joachim Schwanter, Keilbergweg 5, 8225 Traunreut

---

Suche Tauschpartner für Spiele, Anwend.-Prog. C16 u. Plus4. Nur Disk.! Listen bitte an: Burkhard Kitter, Gottlieb-Kellner-Str. 4, 3500 Kassel

---

Suche Software f. C16 (64K Mem). Keine Spiele. Hans Bialas, Hasenbühlweg 44, 79 Ulm 12

*** C16 *** C16 ***
Suche f. C16/116/Plus4 einen Tauschpartner. Ich habe ca. 100 Prog. Info bei Thomas Braun, Lange Gewanne 58, 8900 Augsburg. Antw. garantiert.

Habe „halben" Plus4 erstanden! Suche kompl. Gehäuse m. Tastatur, Platine egal. Nehme auch Datasette. Gebe ab CW-Special (5/87). Hubert Mackenberg, Schmiedebogen 22, 4780 Lippstadt 15

C16/116/+4. Ich suche Tauschpartner f. Prog. auf Kass. Habe selbst den C+4. Prog.-Liste an Peter Leimann, Horserstr. 185, 4650 Gelsenkirchen-Buer. Ich beantworte auf jeden Fall jeden Brief. Ihr solltet wenigstens 5 Prog. haben.

Verkaufe Verkaufe Verkaufe C64, 1541, Speedos, MPS-801, Disketten, 2 Joysticks, 3 Diskboxen, Zeitungen, Data, 1000 Blatt, Preise: VB. Meldet Euch bei: Christoph, Tel. 07941/61853

Verk. 100 Games f. 20,–. Nur Disk.! Für wen? C16/Plus4 natürlich. Schickt einen 20-DM-Schein an Thore Kirks, Olbersstr. 6N, 2800 Bremen 44. Nur Überraschungspaket

---

Ich biete 7,– DM f. alle C128-Sonderhefte, die Ihr auftreiben könnt. Weiter 3,– f. die Hefte 1–5/87 v. CW. Außerdem suche ich ein Modem f. C128. Angebote bitte an: Fabian Dohmes, Umgebung Bonn, 0228/643290

TEXTER 128

Das Programm „Texter 128" unterstützt alle diejenigen, die beruflich oder privat mit Texten zu tun haben. Es erlaubt, Texte in den unterschiedlichsten Formen für Briefe, Notitzen, geschäftliche Korrespondenzen udw. darzustellen. Damit nicht versehentlich über den Rand des Briefbogens geschrieben wird, kann die Zeilenlänge eingestellt werden. Durch eine Tonanzeige wird der Schreiber rechtzeitig auf das Ende der Zeile hingewiesen. Darüber hinaus kann zur Analogie der Schreibmaschinentastatur automatisch der Buchstabe „Z" mit „Y" ausgetauscht werden. Sollten Texte berichtigt oder gelöscht werden, so kann dies während der Texteingabe oder nach Fertigstellung des Schriftstückes erfolgen. Erfaßte Texte lassen sich über Bildschirm jederzeit anzeigen oder mittels Drucker ausdrucken.
Sollte es notwendig sein, Texte abzuspeichern, damit sie später wieder verfügbar gemacht werden können, so steht dafür ein umfangreicher Diskettenservice (Texte speichern/laden, anzeigen, Directory, Diskettenname oder ID ändern, Diskette formatieren, Dateien umbenennen oder löschen) zur Verfügung.
Durch ein einfaches und übersichtliches Menü (farblich unterlegt) wird der Bediener durch das Leistungsspektrum von „Texter 128" geführt.
Dennoch sollte beachtet werden, daß der Editier-Modus durch Drücken der Tasten „Commodore & M" verlassen werden kann und daß das Codewort, falls der gespeicherte Text mit einem solchen geschützt wurde, nicht vergessen werden darf.
Das Programm kann jederzeit mit der Run-Stop-Taste unterbrochen werden.

(Eric Steinerberger

```
10 rem texter ============128 <cj>
20 rem (p) commodore welt   -- <jp>
30 rem ====================== <ci>
40 rem (c) by eric          -- <id>
50 rem        steinerberger == <am>
60 rem                      == <ef>
70 rem version 7.0 80 z/ascii= <ch>
80 rem 128 pc+floppy+drucker== <mh>
90 rem ====================== <ge>

95 gosub 60000                <oc>
100 trap1510                  <ac>
110 a=peek(215):if not a=128 t
hen color0,1:color4,1:printcl$
c4$c4$c4$wh$"bitte 80-zeichen
bildschirm einschalten!":end  <cp>
120 print chr$(14)            <io>
130 dim a$(1000):dim g$(80):d=
72                            <pm>
140 dim c(80):dim d$(1000):di
m e(1000)                     <fp>
150 dim b(80) :a$(1)=zu$:fast <bk>
160 printcl$re$left$(qr$,21)rn
$"
    "                         <bo>
170 printleft$(qr$,21)rn$"  "r
f$left$(qr$,6)ye$"80 Zeichen-T
exter      "re$;rn$"  "rf$    <op>
180 printleft$(qr$,21)rn$"  "r
f$left$(qr$,6)pu$zn$zn$zn$zn$z
n$zn$zn$zn$zn$zn$zn$zn$zn$zn$z
n$zn$zn$"      "re$rn$"  "rf$ <nk>
190 printleft$(qr$,21)rn$"  "r
f$left$(qr$,29)re$rn$"  "rf$  <ia>
200 printleft$(qr$,21)rn$"  "r
f$c3$cy$"(c)"wh$"  by  Eric St
einerberger "re$;rn$"  "rf$   <bg>
210 printleft$(qr$,21)rn$"  "r
f$left$(qr$,29)re$rn$"  "rf$  <je>
220 printleft$(qr$,21)rn$"  "r
f$left$(qr$,7)lg$"in  April  1
987"cy$"       "re$rn$"  "rf$ <nb>
230 printleft$(qr$,21)rn$"
                          "
rf$                           <lc>
240 printwh$c4$c4$"
                Maximale Text
laenge:"                      <dp>
250 printc4$"
       1000 Zeilen zu 78 Zeic
hen!"                         <dp>
260 printc4$"
         === 78.000 Zeichen =
==":getkey g$                 <nn>
270 printcl$c4$c4$ye$"80 Zeich
en Texter"                    <ob>
280 printwh$zn$zn$zn$zn$zn$zn$
zn$zn$zn$zn$zn$zn$zn$zn$zn$zn$
zn$cy$c4$c4$                  <ph>
290 print"          "ye$"1."cy
$"  editieren"                <ac>
300 print"          "ye$"2."cy
$"  Text speichern"           <nk>
310 print"          "ye$"3."cy
$"  Text nachladen"           <mc>
320 print"          "ye$"4."cy
$"  Fehler berichtigen"       <oe>
330 print"          "ye$"5."cy
$"  Directory anzeigen"       <bl>
340 print"          "ye$"6."cy
$"  Text drucken"             <hb>
350 print"          "ye$"7."cy
$"  Text listen"              <dp>
360 print"          "ye$"8."cy
$"  Disk Kommandos"           <ik>
370 print"          "ye$"9."cy
$"  Text loeschen"            <nh>
380 print"          "ye$"0."cy
$"  Rand einstellen           <di>
390 getkey g$                 <im>
400 if g$="1" then 500        <ok>
410 if g$="2" then 910        <pk>
420 if g$="3" then 1190       <dg>
430 if g$="4" then 1550       <eb>
440 if g$="5" then printcl$;ch
r$(142):slow:directory:getkey
g$:print chr$(14):fast:goto270 <if>
450 if g$="6" then 780        <cl>
460 if g$="7" then printc4$+c4
$:goto1460                    <jo>
470 if g$="8" then 1800       <gl>
480 if g$="9" then 2100       <ha>
490 if g$="0" then2210:else390 <bo>
500 if q1=1 then 520          <cd>
510 printc4$"in den Schreibmas
chinentastaturmodus wechseln (
y und z vertauscht) j/n ?":get
keyg$:ifg$<>"j"then520:else pr
intc4$ye$"einen Momment bitte!
";:q1=1:goto2240              <ea>
520 if not a$(1)=zu$ then for
x=1 to a:a$(x)="":e(x)=1:next:
a=0:a$(1)="":else a$(1)="":e(x
)=1                           <gb>
530 printc4$wh$" editieren"   <oi>
540 print" "ye$zn$zn$zn$zn$zn$
zn$zn$zn$cy$c4$               <ee>
550 b=0:printrn$wh$"<"rf$c1$cy
$;                            <hf>
560 for x=1 to 1000           <fd>
570 if x/10 = int(x/10) then p
rintc1$+wh$;x;c1$+"te Zeile---
------------------------------
------------------------------
----"+cy$:printrn$wh$"<"rf$c1$
cy$;                          <na>
```

```
580 getkey g$(b):if g$(b)=c4$
or g$(b)=c2$ or g$(b)=c3$ or g
$(b)=c1$ or g$(b)=he$ or g$(b)
=c1$ then580                    <eb>
590 if g$(b)=chr$(13) then pri
nt" ":goto740                   <oj>
600 if g$(b)=chr$(20) and b=0
then 580                        <cg>
610 if b = 77 and g$(b) <> chr
$(20) then580                   <ca>
620 print g$(b);                <ea>
630 if g$(b)=chr$(34) then pri
nt chr$(34);c1$+" "+c1$;        <cp>
640 if g$(b)=chr$(20) then b=b
-1:goto580                      <fa>
650 if b=d then play"we"        <ni>
660 if g$(b)=z6$ then printc1$
+" ":goto680                    <km>
670 if g$(b)=za$ then printc1$
+" ":goto680:else710            <gi>
680 a=x-1                       <nb>
690 for x=1 to a:if a$(x)=chr$
(13) or a$(x)="" then a$(x)="
":next:else next                <ac>
700 goto270                     <oa>
710 if g$(b)=":" or g$(b)="," 
then e(x)=2                     <mk>
720 printrn$wh$"<"rf$cy$c1$;    <mb>
730 b=b+1:goto580               <en>
740 if g$(1)=chr$(13) then a$(
x)=" ":e(x)=2:goto760           <kd>
750 for c=0 to b-1:a$(x)=a$(x)
+g$(c):next                     <mf>
760 if not e(x)=2 then e(x)=1   <hh>
770 b=0:printrn$wh$"<"rf$c1$cy
$;:next:print" ":printc4$ye$"T
EXTSPEICHER"s2$"VOLL !":getkey
 g$:goto270                     <bh>
780 if a$(1)=zu$ then printc4$
+c4$+ye$+"NOCH"+s2$+"KEIN"+s2$
+"TEXT"+s2$+"EINGEGEBEN !":get
key g$:goto270                  <fm>
790 printc4$"(a)lles drucken o
der (Z)eilenweise ?":getkey g$
:if g$="a" then f=1:g=a:goto85
0                               <ch>
800 if g$="z" then810 :else pr
intc2$c2$c2$:goto790            <hp>
810 printc4$"von Zeile ";:inpu
tf                              <nb>
820 print"bis Zeile"s2$;:input
g                               <nf>
830 if f<0 or f>a or not f=int
(f) thenprintc4$+ye$+"falsche
Eingabe":getkey g$:goto270      <kk>
840 if g<f or g>a or not g=int
(g) thenprintc4$+ye$+"falsche
Eingabe":getkey g$:goto270      <mo>
850 printc4$ye$"drucken laeuft
 !"cy$                          <dn>
860 open 7,4:cmd7               <kd>
870 for x=f to g                <cf>
880 print chr$(17);a$(x):next   <jb>
890 print#7," ":close7          <mn>
900 printc4$wh$"fertig !"cy$:g
etkey g$:printc2$c2$:goto270    <ae>
910 if a$(1)=zu$ then printc4$
+ye$+"NOCH"+s2$+"KEIN"+s2$+"TE
XT"+s2$+"EINGEGEBEN !":getkey
g$:goto270                      <jj>
920 printc4$"Name der Datei ";
:inputb$:if len(b$) > 16 then
printc4$ye$"? FILE"s2$"TO"s2$"
LONG":getkey g$:goto270         <lf>
930 if b$="" then printc2$+ye$
+"? MISSING FILENAME"+cy$+c2$+
c2$:getkey g$:goto910           <gf>
940 print b$;" mit einem Code
schuetzen (J/N) ?":getkey g$:i
f g$ <> "j" then970             <ni>
950 printc4$"Code (6 Zeichen)
";:inputc$:printc4$c4$:if len(
c$) > 6 then printc4$ye$"CODE"
s2$"ZU"s2$"LANG":getkey g$:got
o270                            <km>
960 if len(c$) < 6 then printc
4$ye$"CODE"s2$"ZU"s2$"KURZ":ge
tkey g$:goto270:else990         <mp>
970 for x=1 to a:d$(x)="":next  <gl>
980 c$="000":printc4$           <dc>
990 for x=1 to a                <ic>
1000 if e(x) = 1 then1090       <li>
1010 if a$(x)=" " or a$(x)=""
then d$(x)=zg$:goto1080         <pd>
1020 b=len(a$(x))               <bf>
1030 for y=1 to b               <kn>
1040 x$=mid$(a$(x),y,1)         <en>
1050 if x$=":" then x$=za$      <oj>
1060 if x$="," then x$=zi$      <on>
1070 d$(x)=d$(x)+x$             <cm>
1080 next                       <lo>
1090 if e(x) <> 2 then d$(x)=a
$(x)                            <mn>
1100 printc2$cy$"Bearbeite Zei
le"wh$;x:next:printc4$          <ac>
1110 for x=1 to a:if d$(x)=""
then d$(x)=" ":next             <db>
1120 open2,8,2,"§0:"+b$+",s,w"  <bb>
1130 print#2,a;chr$(13);c$      <po>
1140 for x=1 to a               <bj>
1150 printc2$cy$"speicher Zeil
e"ye$;x                         <fj>
1160 print#2,d$(x);chr$(13);e(
x)                              <hd>
1170 next:close2                <ce>
1180 printc4$wh$"fertig !"cy$:
```

```
getkey g$:goto270                                   <bi>
1190 printc4$"Name der Datei "
;:inputb$:if len(b$) > 16 then
 printc4$ye$"? FILE"s2$"TO"s2$
"LONG":getkey g$:goto270                            <me>
1200 if b$="" then printc2$+ye
$+"? MISSING FILENAME"+cy$+c2$
+c2$:getkey g$:goto1190:else p
rintc4$                                             <oi>
1210 for x=1 to a:a$(x)="":nex
t                                                   <fj>
1220 a=0:open 2,8,2,b$+",s,r"                       <df>
1230 input#2,a,c$                                   <ii>
1240 for x=1 to a                                   <hn>
1250 input#2,d$(x),e(x)                             <pd>
1260 printc2$cy$"lade Zeile"ye
$;x                                                 <ik>
1270 next                                           <hm>
1280 close2                                         <nh>
1290 if not a>0 then printc4$y
e$"? FILE NOT FOUND":getkey g$
:goto270                                            <km>
1300 printc4$:for x=1 to a:a$(
x)="":next                                          <cc>
1310 for x=1 to a                                   <me>
1320 if e(x) = 1 then1410                           <pg>
1330 if d$(x)=zg$ then a$(x)="
 ":goto1400                                         <hn>
1340 b=len(d$(x))                                   <fk>
1350 for y=1 to b                                   <oo>
1360 x$=mid$(d$(x),y,1)                             <jb>
1370 if x$=za$ then x$=":"                          <ck>
1380 if x$=zi$ then x$=","                          <co>
1390 a$(x)=a$(x)+x$                                 <gh>
1400 next                                           <pp>
1410 if e(x) <> 2 then a$(x)=d
$(x)                                                <ao>
1420 printc2$cy$"Bearbeite Zei
le"wh$;x:next                                       <nf>
1430 if not c$="000" then 1440
:else printwh$c4$"fertig":getk
ey g$:goto270                                       <nb>
1440 printc4$;b$;" ist gesiche
rt!";:input"  code ";b$                             <kc>
1450 if c$=b$ then 270:else pr
intc4$ye$rn$"CODE"s2$"ERROR":g
etkey g$:for x=1 to a:a$(x)="
":next:a$(1)=zu$:a=0:c$="000":
goto270                                             <lc>
1460 if a$(1)=zu$ then printye
$+"NOCH"+s2$+"KEIN"+s2$+"TEXT"
+s2$+"EINGEGEBEN"+s2$+"!":getk
ey g$:goto270                                       <ln>
1470 for x=1 to a                                   <ge>
1480 if x < 10 then printwh$;x
;c1$cy$;a$(x):goto1500                              <lp>
1490 printc1$wh$;x;c1$cy$;a$(x
):if x/22=int(x/22) then getke
y g$                                                <df>
1500 next :getkey g$:goto270                        <po>
1510 if er= 5 then close 7:pri
ntc4$ye$"? DEVICE NOT PRESENT"
:getkey g$:resume270                                <mp>
1520 if er=23 then print:print
ye$c4$"?STRING"s2$"TO"s2$"LONG
":getkey g$:resume270                               <in>
1530 if er=24 then print:print
c4$ye$"? FILE DATA ERROR":slee
p5:resume270                                        <jb>
1540 if er=30 then printc1$+cy
$+c4$+c4$+c4$+"Wirklich Schlus
s? Alle Daten werden geloescht
! (j/n)":getkey g$:if g$ <>"j"
 then resume270:else scnclr:pr
int chr$(142);:clr:end                              <pn>
1550 printc4$c4$"Berichtigen v
on Zeile NR. ";:inputf:if f <
1 or f > 1000 or not f=int(f)
then printc4$+ye$+"Falsche Ein
gabe !"+cy$:getkey g$:goto270                       <pe>
1560 print"Berichtigen bis Zei
le NR. ";c:if c < 1 or c > 100
0 or not c=int(c) or c < f the
n printc4$+ye$+"Falsche Eingab
e !"+cy$;:inputgetkey g$:goto2
70:else print:b=0                                   <jh>
1570 if c>a then for x=a+1 to
c:a$(x)=" ":next:a=c                                <ih>
1580 for x=f to c                                   <ok>
1590 print c1$ye$;x;c1$wh$" ";
a$(x);cy$                                           <oa>
1600 printrn$wh$"<"rf$cy$c1$;                       <df>
1610 getkey g$(b):if g$(b)=c4$
 or g$(b)=c2$ or g$(b)=c3$ or
g$(b)=c1$ then1610                                  <fc>
1620 if g$(0)=chr$(13) then pr
int a$(x):next                                      <aj>
1630 if g$(b)=chr$(13) then pr
int" ":goto1760                                     <da>
1640 if g$(b)=chr$(20) and b=0
 then1610                                           <gf>
1650 if b = 77 and g$(b) <> ch
r$(20) then1610                                     <fp>
1660 print g$(b);                                   <fe>
1670 if g$(b)=chr$(34) then pr
int chr$(34);c1$+" "+c1$;                           <ed>
1680 if g$(b)=chr$(20) then b=
b-1:goto1610                                        <ip>
1690 if g$(b)=z6$ then printc1
$+" ":goto1700:else1730                             <lk>
1700 a=x-1                                          <nb>
1710 for x=1 to a:if a$(x)=chr
$(13) or a$(x)="" then a$(x)=s
2$:e(x)=2:next:else next                            <hd>
1720 goto270                                        <oa>
1730 if g$(b)=":" or g$(b)=","
```

```
 then e(x)=2                                    <mk>
1740 printrn$wh$"<"rf$cy$c1$;                   <mb>
1750 b=b+1:goto1610                             <hi>
1760 a$(x)="":if not e(x)=2 th                  <kb>
en e(x)=1
1770 for y=0 to b-1:a$(x)=a$(x                  <pb>
)+g$(y):next
1780 b=0:printrn$wh$"<"rf$cy$c                  <eh>
1$;
1790 next:printc1$" ":printc4$                  <dl>
wh$"O.K.":getkey g$:goto270
1800 printc4$c4$wh$"1.  "cy$"D                  <gb>
iskette formatieren"
1810 printwh$"2.  "cy$"Datei l                  <ff>
oeschen"
1820 printwh$"3.  "cy$"Disk ID                  <ek>
 veraendern"
1830 printwh$"4.  "cy$"Disk Na                  <gj>
me veraendern"
1840 printwh$"5.  "cy$"Datei-N                  <pa>
ame aendern"
1850 printwh$"6.  "cy$"Menue"                   <ed>
1860 getkey g$                                  <fa>
1870 if g$="1" then1920                         <of>
1880 if g$="2" then2020                         <oi>
1890 if g$="3" or g$="4" then
2350                                            <do>
1900 if g$="5" then2490                         <ak>
1910 if g$="6" then 270:else18
60                                              <ln>
1920 f$="":g$="":printc4$"Disk
etten-Name ";:inputf$                           <jb>
1930 input"Disketten ID    ";g$                 <li>
1940 if len(f$) >16 or len(g$)
>2 then printc4$c4$ye$"? FILE"
s2$"TO"s2$"LONG":getkey g$:got
o270
1950 if f$="" then f$="
          "                                     <jh>
1960 if g$="" then g$="  "                      <kd>
1970 printc4$"          "ye$"j"                 <ib>
1980 printc2$cy$"sicher ";:inp
uth$                                            <bn>
1990 if not h$="j" then270                      <om>
2000 open 15,8,15,"n:"+f$+","+
g$:close15                                      <en>
2010 printc4$wh$"fertig!  Nohm
al?":getkey g$:if g$="j" then
scnclr:goto1800:else270                         <oo>
2020 g$="":printc4$c4$"Name de
r Scratchdatei ";:inputg$                       <gj>
2030 if g$="" then printc4$+ye
$+"? MISSING"+s2$+"FILENAME":g
etkey g$:goto270                                <am>
2040 if len(g$) > 16 then prin
tc4$ye$"? FILE"s2$"TO"s2$"LONG
"cy$:getkey g$:goto270                          <gj>
2050 printye$"          j"                      <pd>
2060 printc2$cy$"sicher ";:inp
utb$                                            <gi>
2070 if b$="j" then 2080:else2
70                                              <gi>
2080 open 15,8,15,"s:"+g$:clos
e15                                             <hf>
2090 printc4$wh$"fertig !  wei
ter (j/n) ?"cy$:getkey g$:if g
$="j" then scnclr:goto1800:els
e270                                            <ki>
2100 if a$(1)=zu$ then printc4
$+ye$+"NOCH"+s2$+"KEIN"+s2$+"T
EXT"+s2$+"EINGEGEBEN"+s2$+"!":
getkey g$:goto270                               <if>
2110 printc4$c4$"(a)lles loesc
gen oder (z)eilenweise ?":getk
ey g$:if g$="a" then2160:else2
120                                             <fd>
2120 printc4$"von Zeile ";:inp
utx:print"bis Zeile ";y:if y>x
 or not y=int(y) or not x= int
(y) then printye$+c4$+"das geh
t nicht!";:inputgetkey g$:goto
270                                             <fj>
2130 printc4$ye$"sicher (j/n)
?":getkey g$:if g$="j" then214
0:else270                                       <go>
2140 for z=x to y:a$(z)="":e(z
)=1:next                                        <ep>
2150 printwh$c4$"O.K.":getkey
g$:goto270                                      <dk>
2160 printc4$"sicher (j/n) ? "
wh$;:getkey g$:if g$ <> "j" th
en print"N":sleep1:goto270:els
e print"J"                                      <dm>
2170 for x=1 to a                               <cd>
2180 if x=1 then a$(x)=zu$:got
o2200                                           <bd>
2190 a$(x)=" ":e(x)=1                           <kf>
2200 next:a=0:printc4$wh$"fert
ig"cy$:getkey g$:goto270                        <ba>
2210 printc4$"Warnton Bei (Zei
chen NR. 5-77 ) ";:inputf                       <na>
2220 if f<6 or f>77 or not f=i
nt(f) then printc4$+c4$+ye$+"f
alsche Eingabe":getkey g$:goto
270                                             <jf>
2230 d=f:printc4$wh$"O.K.":get
key g$:goto270                                  <aa>
2240 bank 15                                    <co>
2250 for i=0 to 90*4                            <gl>
2260 poke 6656+i,peek(64128+i)                  <he>
2270 next                                       <gi>
2280 poke 2757,peek(2757) or 1
28                                              <ll>
2290 poke 6656+12,89:poke 6656
+25,90                                          <ga>
2300 poke 6656+89+12,89+32                      <eg>
```

```
2310 poke 6656+89+25,90+32          <en>
2320 poke 830,0:poke831,26          <ki>
2330 poke 832,89:poke833,26         <ph>
2340 printwh$"  O.K.":goto520       <bn>
2350 open15,8,15:print#15,"i"+
dv$:open1,8,3,"#":print#15,"u1
:3"dv",18,0"                        <ff>
2360 if g$="4" then 2420            <na>
2370 printc4$"Neue Disk-ID (5
Zeichen) ";:inputni$                <jg>
2380 if len(ni$) >5 then print
:printye$"? FILE"s2$"TO"s2$"LO
NG"s2$:printleft$(qu$,5)cy$:ge
tkey g$:goto2370                    <ai>
2390 ni$=left$(ni$+"      ",5)      <op>
2400 print#15,"b-p:3,162":prin
t#1,ni$;:print#15,"u2:3"dv",18
,0                                  <dg>
2410 i$="i"+mid$(str$(dv),2):p
rint#15,i$:goto2460                 <fb>
2420 printc4$"Neuer Disk-Name
(16 Zeichen)";:inputnd$             <fl>
2430 if len(nd$) > 16 then pri
nt:printye$"? FILE"s2$"TO"s2$"
ONG":getkey g$:printleft$(qu$,
5)cy$:goto2420                      <oo>
2440 nd$=left$(nd$+"
      ",16)                         <ej>
2450 print#15,"b-p:3,144":prin
t#1,nd$;:print#15,"u2:3"dv",18
,0":goto2410                        <ap>
2460 printc4$wh$"O.K.   weiter
 (j/n) ?":getkey g$                 <dm>
2470 poke198,2:poke631,13:poke
632,13                              <dn>
2480 if g$="j" then scnclr:pri
ntc4$c4$:goto1800:else270           <ll>
2490 printc4$c4$"alter Dateina
me ";:inputg$                       <ll>
2500 input"neuer Dateiname ";f
$                                   <mc>
2510 if len(g$) > 16 or len(f$
) > 16 then printc4$ye$"? FILE
"s2$"TO"s2$"LONG":getkey g$:go
to270                               <mh>
2520 open 15,8,15,"r:"+f$+"=0:
"+g$                                <jg>
2530 printc4$wh$"fertig    wei
ter (j/n) ?":getkey g$:if g$="
j" then scnclr:printc4$c4$:got
o1800:else270                       <ij>
60000 rem nachspann ==========      <nf>
60010 rem farb-/steuercodes         <ka>
60020 wh$=chr$(005):c4$=chr$(0
17)                                 <lh>
60030 rn$=chr$(018):he$=chr$(0
19)                                 <no>
60040 re$=chr$(028):c3$=chr$(0
29)                                 <mk>
60050 c2$=chr$(145):rf$=chr$(1
46)                                 <ne>
60060 cl$=chr$(147):lg$=chr$(1
53)                                 <pd>
60070 pu$=chr$(156):c1$=chr$(1
57)                                 <pi>
60080 ye$=chr$(158):cy$=chr$(1
59)                                 <ch>
60090 rem zeichensatz/graphik       <la>
60100 s2$=chr$(160):z6$=chr$(1
67)                                 <ao>
60110 za$=chr$(171):zg$=chr$(1
77)                                 <ec>
60120 zi$=chr$(179):zn$=chr$(1
84)                                 <gb>
60130 zu$=chr$(191)                 <ae>
60140 rem zeichenfolgen             <cg>
60150 for q=1 to 40                 <jn>
60160 qu$=qu$+c2$:qr$=qr$+c3$       <dc>
60170 next q                        <mi>
60180 return                        <in>

ready.
```

BÖRSE

Suche Spiele, Anwend.-Prog. u. Listings (C64). Angebote an: Frank Stuckstedte, In der Helle 15, 4840 Rheda-Wiedenbrück. Abgelehnte Angebote sende ich zurück!

Hallo, C16/Plus4-Freunde! Wer will Prog. auf Disk. tauschen? Listen an: Matthias Franke, Schlebacher Str. 11, 5308 Rheinbach-Merzbach, Tel. 02226/13252

Achtung VC20Achtung VC20 Software sehr günstig zu bekommen, z.B. Prog. schon ab 0,40 DM. Infos bei F. Segovia gg. 1 DM anfordern! Nordwalderstr. 83, 4407 Emsdetten. Tausche auch!

An alle C16-Besitzer: Besitze Prog. aus anderen Comp.-Zeitschr. Schickt mir eine Kass. m. Euren Prog. u. Ihr bekommt die gleiche Anzahl von Prog. 100 % von mir zurück. Schreibt an: Sandro Sali, Am Kraftwerk 36, 4600 Dortmund 50

Suche Commodore-Drucker MPS 801 o. 803, voll funkt.-fäh.! H.-J. Wallrabe-Adams, Gerh.-Hauptmann-Ring 176a, 6000 Frankfurt, T. 579809

C64: Chicken Chase (C) 8,–; Operation Fireball (C) 9,–; Fighting Warrior 18,–; Thing on a Spring (C) 18,–; Ultima 1 (D) 70,–; Turbo 50,–; Game Killer 50,–. Keine Kopien, nur Orig., nur je 1 x vorhanden! Fred Martschin, Reherweg 5a, 3258 Aerzen

---

Suche Software f. C16 u. Plus4! Außerdem suche ich Kontakt zu Computerclubs. Angebote bitte an: Jörn Drescher, Lütjenburger Str. 69, 2300 Kiel 14

** C16/C116/Plus4 **
30 Spiele 20 DM (z.B. Karateschule, Champ., Dreikampf usw.). 15 Prog. 10 DM. Schickt Geldschein an: D. Wink, Hinter d. Mauer Nord 35, 3530 Warburg/Westf. Die Kass. kommt sofort!

** C128 ** C128 ** C128 **
Einsteiger u. Software u. Anwenderprog. in Richtung Architektur, Massenberechnung, Ausschreibung, Preisspiegel usw. f. C128 u. 1571. An Lothar Bischeldorfer, An der Steig 8, 7750 Konstanz, Tel. 07531/33777

TELEFONBUCH

BESCHREIBUNG

Es mag dies das einhundertste Adreß-Programm für den Amiga sein. In jeder Dateibeschreibung wird anhand einer Adreßliste eine Beispiel-Datei erklärt. Trotzdem habe ich mir ein eigenes Programm geschrieben, weil kein Programm meine Forderungen erfüllte. Es mag universelle Programme wie Superbase und andere geben, die die gleiche Darstellung ermöglichen, doch ist mir die Einrichtung und Dateipflege zu aufwendig und umständlich. Außerdem haben diese Programme auch einen relativ hohen Anschaffungspreis.
Meine Forderungen an das Programm waren:

1.) Es mußten neben der privaten Telefonnummer auch die geschäftliche Nummer angegeben werden können.
2.) Ein Feld sollte für Geburtstagsdaten zur Verfügung stehen.
3.) Es sollte auf jedem Blatt eine Überschrift mit Seitenangabe sein.
4.) Die Anfangsbuchstaben mußten herausgestellt sein.
5.) Der Zeilenabstand sollte etwas verringert sein.
6.) Für eine gezielte Korrektur sollte die Datensatz-Nummer ausgedruckt sein.
7.) Es sollte der Ausdruck nach einem bestimmten Personenkreis möglich sein.

Der Programmaufbau ist wieder ähnlich wie beim Autokostenprogramm. Bei der Übernahme des Programms sollte jeder Anwender seine eigenen Codes verwenden. Sie stehen in den Labels: SubCode und CodeMaske.
Nach dem Programmstart befindet man sich im Auswahlmenü. Nun zur Erklärung der einzelnen Punkte.

1 Datei einlesen

Ist eine Datei bereits vorhanden, wird diese zuerst eingelesen.

2 Neuer Eintrag

Man trägt in die entsprechenden Felder Name, Vorname, Straße und Hausnummer, PLZ und Ort, evtl. Geburtstag, die private und geschäftliche Telefonnummer, sowie den Zuordnungscode ein.
Bei Übereinstimmung mit vorhergehendem Namen braucht nur RETURN gedrückt zu werden.

3 Daten speichern

Die Daten werden als sequentielle File unter dem Namen "Tel.Datei" gespeichert.

4 Daten ändern/löschen

Mit den Pfeiltasten wird wieder geblättert:
links 1 Satz zurück
rechts 1 Satz vor
oben 10 Satz vor
unten 10 Satz zurück
mit "l" wird ein Satz gelöscht,
mit "a" leitet man eine Änderung ein,
mit "m" gelangt man wieder ins Menü.

5 Daten sortieren

Nach jeder Korrektur bzw. Neueingabe sollten die Daten neu nach dem Familiennamen sortiert werden.

6 Ausgabe Drucker

Man wird zuerst gefragt, ob alle Daten oder nach Code gedruckt werden soll. Bei Druck nach Code können auch mehrere Codes einfach aneinandergereiht werden. Nun wird die Überschrift mit Seitennummer geschrieben.
Es folgen die Einträge, sortiert nach Alphabet. Der Anfangsbuchstabe ist jeweils beim ersten Datensatz nach links gerückt. Dann kommt die laufende Nummer, Name, Vorname, Geburtstag, Codierung, Tel. privat und in der 2. Zeile Straße, PLZ und Ort und geschäftliche Telefonnummer und eine Trennlinie zum nächsten Satz.
Nach je 26 Ausgaben ist eine Seite voll und es geht mit dem nächsten Blatt weiter.

7 Ausgabe Bildschirm

Die Ausgabe erfolgt am Bildschirm. Mit "m" kommt man auch vorzeitig nach jeder Bildschirmseite zurück ins Auswahlmenü.

8 Ende

Beendet das Programm .

LISTE DER WICHTIGSTEN VARIABLEN

Variable	Erklärung	
sa$	kompletter Datensatz	
nf	Anzahl der Einträge	
c	Flag bei Ausdruck nach Code	
Code$(0)	Codestring bei Auswahl Gesamt	
Code$(n)	Codestring bei Auswahl (2 Stellen)	
a%	Anzahl der Auswahl-Codes	
ab	Anfangsbuchstabe	
fc	Anfangsbuchstabe der vorherigen Zeile	
zz%	Zeilenzähler	
Aufbau des Datensatzes Sa$:	Na$ Name	mid$(sa$(n),1,22)
	Vn$ Vorname	mid$(sa$(n),23,16)
	St$ Straße	mid$(sa$(n),40,22)
	Ot$ Plz und Ort	mid$(sa$(n),63,25)
	Gt$ Geburtstag	mid$(sa$(n),89,8)
	Tp$ Tel. privat	mid$(sa$(n),98,14)
	Tg$ Tel. gesch.	mid$(sa$(n),123,14)
	Co$ Codierung	mid$(sa$(n),138,1)

Erforderliche Hardware: Amiga 500/1000/2000, Drucker
Erforderliche Software: Amiga-Basic
Ziel des Programms: Ausgabe einer Adreß- und Telefon-Liste sortiert nach Alphabet

LISTING

```
' Telefonbuch ============Amiga
' (c) by COMMODORE WELT        ==
' ============================
' (p) by Franz Bayerstadler    ==
'         Gernlinden           ==
' Amiga 500/1000/2000          ==
' AmigaBasic, Drucker          ==
' ============================

Vorbereitung:
  CLEAR,32000,2000
  l1$="
" 'Leerstring
  ue3$=STRING$(79,45)
   '79 Zchn -
  ue4$=STRING$(79,61)
   '79 Zchn =
  DIM sa$(200) , Code$(12)
  PALETTE 0,0,0,0          'schwarz
   Color 0 Hintergrund schwarz
  PALETTE 1,1,1,1!         'weiss
   Color 1
  PALETTE 2,1,1,.08        'gelb
   Color 2
  PALETTE 3,.3,.5,1        'dunkelbl
au Color 3
  WIDTH "scrn:",80         'Bildschi
rmbreite

Menue:
  BEEP
  LINE (10,1)-(615,185),3,b
  LINE (20,5)-(605,180),3,b
  LINE (30,10)-(595,175),3,b
  LINE (40,15)-(585,170),3,b
  LINE (50,20)-(575,165),3,b
  LINE (60,25)-(565,160),1,b
  LINE (160,7)-(465,22),0,bf     '
Ausschnitt fuer Ueberschrift oben

  LINE (160,166)-(465,182),0,bf '
Ausschnitt fuer Statusanzeige unt
en
  LOCATE 2,33:COLOR 1:PRINT  "Tel
efonbuch"
  LOCATE 3,21:COLOR 2:PRINT " ein
 Programm von Franz Bayerstadler
"
  LOCATE 9,12:  COLOR 1:PRINT "1"
;:COLOR 3:PRINT "  Datei einlesen
"
  LOCATE 9,43:  COLOR 1:PRINT "2"
;:COLOR 3:PRINT "  neuer Eintrag"

  LOCATE 11,12: COLOR 1:PRINT "3"
;:COLOR 3:PRINT "  Daten speicher
n"
  LOCATE 11,43: COLOR 1:PRINT "4"
;:COLOR 3:PRINT "  Daten andern /
 loschen"
  LOCATE 13,12: COLOR 1:PRINT "5"
;:COLOR 3:PRINT "  Daten sortiere
n"
  LOCATE 13,43: COLOR 1:PRINT "6"
;:COLOR 3:PRINT "  Ausgabe Drucke
r"
  LOCATE 15,12: COLOR 1:PRINT "7"
;:COLOR 3:PRINT "  Ausgabe Bildsc
hirm"
  LOCATE 15,43: COLOR 1:PRINT "8"
;:COLOR 3:PRINT "  Ende"

Waehlen:
 a$=INKEY$:IF a$="" GOTO Waehlen
  IF a$< "1" OR a$ >"8" THEN GOTO
 Waehlen
   IF a$="1" THEN GOTO Lesen
    IF a$="2" THEN GOTO Eingeben
     IF a$="3" THEN GOTO Speicher
n
      IF a$="4" THEN GOTO Editor
       IF a$="5" THEN GOTO Sort
        IF a$="6" THEN GOTO Druck
en
         IF a$="7" THEN GOTO Bild
schirm
          IF a$="8" THEN GOTO Bee
nden

Lesen:
  LINE (61,26)-(564,159),0,bf
   OPEN "i",#2,"Tel.Datei"
    INPUT#2,nf
     LOCATE 22,28: COLOR 2:PRINT
 "Tel.Datei gefunden"
      FOR n = 1 TO nf
       LOCATE 22,48: PRINT n
        INPUT#2,sa$(n)
         NEXT n
          CLOSE#2
           GOTO Menue

Speichern:
 LINE (61,26)-(564,159),0,bf
  LOCATE 22,27: COLOR 1:PRINT "Si
nd Sie sicher M = Menu" :GOSUB Ta
ste
   OPEN "Tel.Datei" FOR OUTPUT AS
#2
    LOCATE 22,26: PRINT " Tel.Dat
ei gespeichert    "
     WRITE#2,nf
      FOR n = 1 TO nf
       LOCATE 22,49: PRINT n
```

LISTING

```
          WRITE#2,sa$(n)
           NEXT n
            CLOSE#2
             GOTO Menue

Eingeben:
 nf=nf+1:n=nf
      ' Zahler Anzahl Datensatze
 CLS
   COLOR 2
    GOSUB CodeMaske
     GOSUB AnschrMaske
      GOSUB EingabeMaske
       GOSUB SubRichtig :CLS
        GOSUB Uebernahme
         GOTO Menue

Sort:
 LINE (61,26)-(564,159),0,bf
  LOCATE 22,32: COLOR 1:PRINT "Ich sortiere ..."
 FOR n = 1 TO nf-1
  FOR m = n+1 TO nf
   IF LEFT$(sa$(n),25) > LEFT$(sa$(m),25) THEN SWAP sa$(n),sa$(m)
    NEXT m
     LOCATE 22,49:PRINT n:NEXT n
      LOCATE 22,27: PRINT USING"###";n;:PRINT " Datensatze sortiert    "
       GOSUB Taste
       GOTO Menue

Editor:
 CLS
  LOCATE 12,15:INPUT "Ab welchem Datensatz wollen Sie blattern ",n

   IF n>nf AND n<1 GOTO Editor
    CLS
     GOSUB CodeMaske
      COLOR 1
       LINE (5,165)-(609,185),2,b

        LOCATE 22,3:PRINT "Cursor rechts - links = 1  Satz vor oder zuruck    A = Andern L = Loschen "
         LOCATE 23,3:PRINT "Cursor oben    - unten = 10 Satz vor oder zuruck                M = Menu";


 Blaettern:
  COLOR 2
   GOSUB Entpacken
     GOSUB AnschrMaske
 Entscheide:
 a$=INKEY$:IF a$="" GOTO Entscheide
   IF ASC(a$)-31 THEN n=n-1:IF n<1 THEN n=nf ELSE GOTO Blaettern' 1 rueck
   IF ASC(a$)-30 THEN n=n+1:IF n>nf THEN n=1  ELSE GOTO Blaettern' 1 vor
   IF ASC(a$)=28 THEN n=n+10:IF n>nf THEN n=nf ELSE GOTO Blaettern' 10 vor
   IF ASC(a$)=29 THEN n=n-10:IF n< 1 THEN n=1  ELSE GOTO Blaettern' 10 rueck
   IF a$="l" THEN GOTO Loeschen
   IF a$="a" THEN GOTO Aendern
   IF a$="m" THEN COLOR 3:CLS: GOTO Menue
    GOTO Blaettern
Loeschen:
 a=n
  FOR x=n TO nf-1: SWAP sa$(x),sa$(x+1):NEXT:nf=nf-1
   LOCATE 21,27: PRINT USING"###";a;:PRINT ". Datensatz geloscht "

    GOTO Blaettern
Aendern:
 GOSUB EingabeMaske
  GOSUB Uebernahme
   IF n<nf THEN n=n+1 ELSE n=1
    GOTO Blaettern

Drucken:
CLOSE 1
OPEN "PRT:" FOR OUTPUT AS 1
 CLS
  IF nf <1 THEN CLOSE 1:LOCATE 22,26:PRINT "Keine Datensatze vorhanden": GOTO Menue
   PRINT#1,CHR$(27)"A0z";
   '  1/8 pro Linie
    GOSUB SubWahl
     ue1$="T E L E F O N L I S T E      Seite"
      ue2$="<  NR NAME / ANSCHRIFT                          GEB.-TAG   CODE       TELEFON"
       zz%=0: m=1: n=1
        GOSUB SubUeberschrift

Schleife1:
 IF n> nf THEN GOTO Menue
  IF a%>0 THEN FOR x=1 TO a%: IF Code$(x)<> MID$(sa$(n),122,2) THE
```

```
 NEXT x : n=n+1 :GOTO Schleife1
C     IF zz% <26 THEN GOTO Label3 EL
  SE zz%=0
C
C Label3:
C  ab$=MID$( sa$(n),1,1)
                  ' = Anfangsbuchstab
  e
C     IF ab$<>fc$ THEN PRINT#1,ab$;"
   "; ELSE PRINT#1,"  ";
C     PRINT#1,USING"###";n;:PRINT#1,
  " ";                  'lfd.Nr
C     PRINT#1,MID$( sa$(n),1,22);" ";
                       'Name
C     PRINT#1,MID$( sa$(n),23,16);" "
  ;                    'Vorname
C     PRINT#1,MID$( sa$(n),86,8);" ";
                       'GebTag
C     PRINT#1,MID$( sa$(n),122,2);" "
  ;"Privat ";          'Code
C     PRINT#1,MID$( sa$(n),94,14)
                       'Tel privat
C     ' nun Druck 2.Zeile
C     PRINT#1,"       ";
C     PRINT#1,MID$( sa$(n),39,22);" "
  ;                    'Stra3e
C     PRINT#1,MID$( sa$(n),61,25);"
    Gesch. ";          'PLZ Ort
C     PRINT#1,MID$( sa$(n),108,14)
                       'Tel geschaft
C     PRINT#1,ue3$
                       'Strich
C     fc$=ab$
                      'Ubergabe Anfang
  sbuchstabe
C     n=n+1:zz%=zz%+1:
         'pos.Zaher + Zeilenzahler +
  1
C     IF zz%>25 THEN FOR j = 1 TO 13
  : PRINT#1,: NEXT:m=m+1:zz%=0:GOSU
  B SubUeberschrift
C     IF n> nf THEN CLOSE 1: GOTO Me
  nue ELSE GOTO Schleife1
C
C
C
C Bildschirm:
C  CLS
C  FOR n = 1 TO nf
C   GOSUB Entpacken
C    IF x =>7 THEN x=0: GOSUB Taste
   :CLS
C      PRINT USING"####";n;:PRINT "
  ";na$;" ";vn$;" ";gt$;" ";co$;"
  Priv ";tp$
C       PRINT  SPC(5);st$;" ";ot$;"
       Gsch ";tg$
C         PRINT  ue3$
C          x=x+1
C  NEXT
C   GOSUB Taste :CLS
C    GOTO Menue
C
C Beenden:
C  END
C
C SubRichtig:
C  COLOR 2
C   Label1:
C    LOCATE 23,23: PRINT "Alle Anga
  ben richtig ? (J/N/M)";
C     a$=INKEY$:IF a$="" GOTO Label
  1
C      IF a$ = "m"THEN nf=nf-1: CLS
  :COLOR 3: GOTO Menue
C       IF a$ = "n"THEN nf=nf-1: GO
  TO Eingeben
C        COLOR 3
C         RETURN
C
C SubWahl:
C  COLOR 2
C   Label2:
C    LOCATE 22,21:PRINT "Ausgabe nu
  r nach Code oder Alles ?  (C/A)"
C     a$=INKEY$:IF a$="" GOTO Label
  2
C      IF a$ <> "c" AND a$ <>"a" TH
  EN GOTO Label2
C       IF a$="c" THEN GOSUB SubCod
  e
C        COLOR 3 :CLS :LOCATE 11,33
  :PRINT "Ich drucke ..."
C         RETURN
C
C SubCode:
C   CLS
C   COLOR 1
C   LOCATE 5,19:  PRINT "fu Familie
    Uschi            ff Familie  Fran
  z"
C   LOCATE 7,19:  PRINT "gu Geschaf
  t Uschi            gf Geschaft Fran
  z"
C   LOCATE 9,19:  PRINT "hs Haus Be
  rl.Str             fr Freunde"
C   LOCATE 11,19: PRINT "co Compute
  r                  am Amter"
C   LOCATE 13,19: PRINT "ba Banken
                     ei Einkauf"
C   LOCATE 15,19: PRINT "lo Lokale
                     so Sonstiges"
C   COLOR 2
C   LOCATE 22,27:INPUT "Bitte Codes
```

LISTING

```
eingeben ",Code$(0)
C    a%=LEN(Code$(0)) :a%=a%/2
                          'Anzahl der
  Codes
C     p=0
C    FOR n= 1 TO a%
C     Code$(n) = MID$(Code$(0),p+1,2)
  : p=p+2
C      NEXT n
C       RETURN
C
C SubUeberschrift:
C    PRINT#1,ue4$
C     PRINT#1,CHR$(27)"Ä6w";      ' Bre
  itschrift ein
C      PRINT#1,SPC(3);ue1$;m
C       PRINT#1,CHR$(27)+"Ä5w";      '
  Breitschrift aus
C        PRINT#1,ue3$
C         PRINT#1,ue2$
C          PRINT#1,ue4$
C           RETURN
C
C CodeMaske:
C       LOCATE 3,4:  PRINT "Codierung
  :
     Eintrag Nr.";:PRINT  USING"###"
  ;n
C    COLOR 1
C    LOCATE 5,4:  PRINT "fu Familie U
  schi                              .
  .....................<-"
C    LOCATE 6,4:  PRINT "ff Familie F
  ranz"
C    LOCATE 7,4:  PRINT "gf Geschaft
  Franz                             .
  ...............<-"
C    LOCATE 8,4:  PRINT "gu Geschaft
  Uschi"
C    LOCATE 9,4:  PRINT "hs Haus Berl
  .Str                              .
  .....................<-"
C    LOCATE 10,4: PRINT "fr Freunde"
C    LOCATE 11,4: PRINT "co Computer
                                    .
  ... .....................<-"
C    LOCATE 12,4: PRINT "am Amter"
C    LOCATE 13,4: PRINT "ba Banken
                                      T
  T.MM.JJ<-"
C    LOCATE 14,4: PRINT "ei Einkauf"
C    LOCATE 15,4: PRINT "lo Lokale
                                    .
  ...............<-"
C    LOCATE 16,4: PRINT "az Arzte"
C    LOCATE 17,4: PRINT "hw Handwerke
  r"
C    LOCATE 18,4: PRINT "nb Nachbarn"

C    LOCATE 19,4: PRINT "so Sonstiges
  "
C    LOCATE 17,48:PRINT ".....
  ..<-"
C    LOCATE 19,48:PRINT "..<-"
C      LINE (5,10)-(170,160),2,b
C      RETURN
C AnschrMaske:
C     LOCATE 3,59:  PRINT USING"###";
  n
C     COLOR 3
C     LOCATE 5,26: PRINT "Name
                  ";:COLOR 2:PRINT na$:C
  OLOR 3
C     LOCATE 7,26: PRINT "Vorname
                  ";:COLOR 2:PRINT vn$:C
  OLOR 3
C     LOCATE 9,26: PRINT "Stra3e Nr.
                  ";:COLOR 2:PRINT st$:C
  OLOR 3
C     LOCATE 11,26:PRINT "PLZ  Ort
                  ";:COLOR 2:PRINT ot$:C
  OLOR 3
C     LOCATE 13,26:PRINT "Geburtstag
                  ";:COLOR 2:PRINT gt$:C
  OLOR 3
C     LOCATE 15,26:PRINT "Telefon pri
  vat             ";:COLOR 2:PRINT tp$:C
  OLOR 3
C     LOCATE 17,26:PRINT "Telefon ges
  chaftlich  ";:COLOR 2:PRINT tg$:C
  OLOR 3
C     LOCATE 19,26:PRINT "Code
                  ";:COLOR 2:PRINT co$:C
  OLOR 3
C      LINE (185,10)-(609,160),2,b
C      RETURN
C EingabeMaske:
C     COLOR 3
C     LOCATE 5,48: LINE INPUT "",na1$
  :IF na1$<>"" THEN na$=na1$
C     LOCATE 7,48: LINE INPUT "",vn1$
  :IF vn1$<>"" THEN vn$=vn1$
C     LOCATE 9,48: LINE INPUT "",st1$
  :IF st1$<>"" THEN st$=st1$
C     LOCATE 11,48:LINE INPUT "",ot1$
  :IF ot1$<>"" THEN ot$=ot1$
C     LOCATE 13,48:LINE INPUT "",gt1$
  :IF gt1$<>"" THEN gt$=gt1$
C     LOCATE 15,48:LINE INPUT "",tp1$
  :IF tp1$<>"" THEN tp$=tp1$
C     LOCATE 17,48:LINE INPUT "",tg1$
  :IF tg1$<>"" THEN tg$=tg1$
C     LOCATE 19,48:LINE INPUT "",co1$
  :IF co1$<>"" THEN co$=co1$
```

20/64/128

Das unabhängige Commodore-Magazin

KOMMT REGELMÄSSIG ZU IHNEN INS HAUS

Finden Sie Ihre COMMODORE-WELT nicht am Kiosk? Weil sie schon ausverkauft ist? Oder „Ihr" Kiosk nicht beliefert wurde? Kein Problem! Für ganze 60 DM liefern wir Ihnen per Post zwölf Hefte ins Haus (Ausland 80 DM). Einfach den Bestellschein ausschneiden – fotokopieren oder abschreiben, in einen Briefumschlag und ab per Post (Achtung: Porto nicht vergessen). COMMODORE-WELT kommt dann pünktlich ins Haus.



WICHTIGE RECHTLICHE GARANTIE!
Sie können diesen Abo-Auftrag binnen einer Woche nach Eingang der Abo-Bestätigung durch den Verlag widerrufen – Postkarte genügt. Ansonsten läuft dieser Auftrag jeweils für zwölf Ausgaben, wenn ihm nicht vier Wochen vor Ablauf widersprochen wird, weiter.

NUTZEN SIE UNSEREN BEQUEMEN POSTSERVICE

DAS SONDERANGEBOT: PRIVATE KLEINANZEIGEN KOSTENLOS!

Das bietet Ihnen COMMODORE-WELT: KLEINANZEIGEN SIND KOSTENLOSE FÜR PRIVATANBIETER! Suchen Sie etwas, haben Sie etwas zu verkaufen, zu tauschen, wollen Sie einen Club gründen? Coupon ausfüllen, auf Postkarte kleben oder in Briefumschlag stecken und abschicken. So einfach geht das. Wollen Sie das Heft nicht zerschneiden, können Sie den Coupon auch fotokopieren. Oder einfach den Anzeigentext uns so schicken, auf Postkarte oder im Brief. Aber bitte mit Druckbuchstaben oder in Schreibmaschinenschrift!
Und: Einschließlich Ihrer Adresse und/oder Telefonnummer sollten acht Zeilen à 28 Anschläge nicht überschritten werden.
ACHTUNG: WICHTIGER HINWEIS!
Wir veröffentlichen nur Kleinanzeigen privater Inserenten, keine gewerblichen Anzeigen. Die kosten pro Millimeter DM 5.00 plus Mehrwertsteuer!
Wir versenden für Privat-Inserenten keine Beleg-Exemplare!
Chiffre-Anzeigen sind nicht gestattet! Wir behalten uns vor, Anzeigen, die gegen rechtliche, sittliche oder sonstige Gebote verstoßen, abzulehnen!
Anzeigenabdruck in der Reihenfolge ihres Eingangs, kein Rechtsanspruch auf den Abdruck in der nächsten Ausgabe!
Die Insertion ist nicht vom Kauf des Heftes abhängig! Wir behalten uns vor, Anzeigen, die nicht zum Themenkreis des Heftes – Computer – gehören, nicht abzudrucken oder sie nur insoweit zu berücksichtigen, wie es der Umfang des kostenlosen Anzeigenteils zuläßt.

ANZEIGENSERVICE

Die große Börse für jeden Zweck in der CBM REVUE / COMMODORE-WELT. Kostenlos für Privat-Inserenten. Spottbillig für gewerbliche Anbieter. Einfach Coupon ausschneiden, fotokopieren o.ä., ausfüllen und ab die Post – Freimachen nicht vergessen! – Unsere Adresse steht auf dem Coupon, ebenso die Preise für gewerbliche Anbieter! Achtung! Wir weisen ausdrücklich darauf hin, daß wir offensichtlich gewerbliche Anzeigen nicht kostenlos veröffentlichen und uns jedweden Abdruck kostenloser Anzeigen vorbehalten müssen, insbesondere, wenn deren Inhalt nicht CBM-typisch ist oder gegen geltendes Recht verstößt. Private Chiffreanzeigen werden nicht aufgenommen. Für Privatanbieter: maximal acht Zeilen à 28 Anschläge. Für gewerbliche Anbieter: 5 DM p. mm.

COMMODORE WELT
ANZEIGENABTLG 9/87
POSTFACH 1161
D-8044 UNTERSCHLEISSHEIM

Name _____
Vorname _____
Straße/Hausnr. _____
Plz/Ort _____

ABO-SERVICE-KARTE

9/87 **COUPON**

Ja, ich möchte von Ihrem Angebot Gebrauch machen.
Bitte senden Sie mir bis auf Widerruf ab sofort jeweils die nächsten zwölf Ausgaben an untenstehende Anschrift. Wenn ich nicht vier Wochen vor Ablauf kündige, läuft diese Abmachung automatisch weiter.

Ich nehme zur Kenntnis, daß die Belieferung erst beginnt, wenn die Abo-Gebühr dem Verlag zugegangen ist.

Name _____
Vorname _____
Straße/Hausnr. _____
Plz/Ort _____

Ich bezahle:
☐ per beiliegendem Verrechnungsscheck
☐ gegen Rechnung
☐ bargeldlos per Bankeinzug von meinem Konto

bei (Bank) und Ort _____
Kontonummer _____
Bankleitzahl _____
(steht auf jedem Kontoauszug)
Unterschrift _____

Von meinem Widerspruchsrecht habe ich Kenntnis genommen.

COMMODORE WELT
ABO-SERVICE 9/87
POSTFACH 1161
D-8044
UNTERSCHLEISSHEIM

Unterschrift _____ 9/87

PROGRAMMSERVICE

Hiermit bestelle ich in Kenntnis Ihrer Verkaufsbedingungen die Listings dieses Heftes auf

☐ Kassette 9/87 (10 DM)

☐ Diskette 9/87 (25) ohne Amiga-Programme
☐ AMIGA-DISK: (25 DM) auf 3,5

Ich zahle:
Zutreffendes bitte ankreuzen!

per beigefügtem Scheck ( ) Bar ( )

Gegen Bankabbuchung am Versandtag ( )

Meine Bank (mit Ortsname) _____ 9/87

Meine Kontonummer _____

Meine Bankleitzahl _____ (steht auf jedem Bankauszug) _____

Vorname _____ Nachname _____

Str./Nr. _____ Plz/Ort _____

Verkaufsbedingungen: Lieferung nur gegen Vorkasse oder Bankabbuchung. Keine Nachnahme. Umtausch bei Nichtfunktionieren.

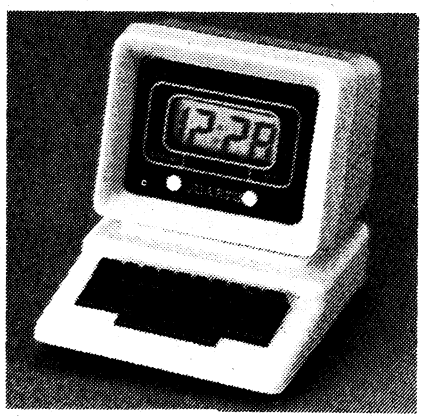
Unterschrift _____

Bitte ausschneiden und einsenden an

COMMODORE WELT
KASSETTENSERVICE 9/87
POSTFACH 1161
D-8044 UNTERSCHLEISSHEIM

LESER WERBEN LESER

GEWINNEN SIE EINE COMPUTER-UHR! Und zusätzlich eventuell noch ein großes Commodore-Buch. Oder ein Paket Disketten. ODER AUCH EINEN COMMODORE-DRUCKER – ODER EINE DISKETTENSTATION! Wie? Sie werben einen Abonnenten. Dann haben Sie auf jeden Fall schon die Computer-Uhr gewonnen. Zusätzlich verlosen wir unter allen, die mitmachen, jeden Monat vier weitere wertvolle Preise. Und alle sechs Monate gibt es einen Hauptpreis unter allen Abo-Werbern zu gewinnen. Also: Mitmachen. Mitgewinnen.



Herrn/Frau _____

_____

Straße/Hausnr. _____

_____

Plz/Ort _____

_____

Ja, ich mache mit beim Abo-Wettbewerb. Ich habe als neuen Abonnenten der COMMODORE WELT geworben.

Der neue Abonnent war bisher noch nicht Bezieher dieser Zeitschrift.
Als Prämie erhalte ich nach Eingang des Abo-Entgeltes auf jeden Fall eine Computer-Uhr, wie abgebildet, und nehme zusätzlich noch an der Verlosung des Monats sowie der halbjährlichen Hauptpreise teil. Mir ist bekannt, daß der Rechtsweg bei den Verlosungen ausgeschlossen ist.
Meinen Preis senden Sie an

Name _____

Straße/Hsnr. _____

Plz/Ort _____

(Bitte ausschneiden und zusammen mit der Abo-Bestellkarte links einsenden!) 9/87

COMMODORE DISC Nr. 3

COMMODORE DISC 64/128

Nr. ●
DM 19,80
ÖS 168,–
SFR19,80

Unverbindliche Preisempfehlung

8 Programme für Commodore-Computer

Spielautomat 128 PC

Text 64 Lawine C 64

Diskeditor, Yathzee C 64

DAS 1. COMMODORE MAGAZIN MIT PROGRAMM-DISKETTE

SIE IST DA!

DIE ERSTE ZEITSCHRIFT MIT PROGRAMM-DISKETTE IM HEFT!

Spion, Mathehilfe C 64

Biorhythmus 128 PC

Alle Programme auf Disc im Heft

```
C     RETURN
C
C Uebernahme:
C    na$=na$+ll$: na$=LEFT$(na$,22):
     sa$(n)=na$
C    vn$=vn$+ll$: vn$=LEFT$(vn$,16):
     sa$(n)=sa$(n)+vn$
C    st$=st$+ll$: st$=LEFT$(st$,22):
     sa$(n)=sa$(n)+st$
C    ot$=ot$+ll$: ot$=LEFT$(ot$,25):
     sa$(n)=sa$(n)+ot$
C    gt$=gt$+ll$: gt$=LEFT$(gt$,8):
     sa$(n)=sa$(n)+gt$
C    tp$=tp$+ll$: tp$=LEFT$(tp$,14):
     sa$(n)=sa$(n)+tp$
C    tg$=tg$+ll$: tg$=LEFT$(tg$,14):
     sa$(n)=sa$(n)+tg$
C    co$=co$+ll$: co$=LEFT$(co$,2):
     sa$(n)=sa$(n)+co$
C     RETURN
C
C Entpacken:
C  na$=MID$(sa$(n),1,22):vn$=MID$(s
   a$(n),23,16):st$=MID$(sa$(n),39,2
   2)
C    ot$=MID$(sa$(n),61,25):gt$=MID$
   (sa$(n),86,8):tp$=MID$(sa$(n),94,
   14)
C     tg$=MID$(sa$(n),108,14):co$=MI
   D$(sa$(n),122,2)
C      RETURN
C
C Taste:
C  COLOR 2 :BEEP
C   Label4:
C    LOCATE 23,27: PRINT "bitte ein
   e Taste drucken";
C      a$=INKEY$:IF a$="" GOTO Label
   4
C       IF a$="m" THEN CLS : COLOR 3
   : GOTO Menue
C        COLOR 3
C         RETURN
C
```



SENSOR

Der Computer gibt eine Tonfolge vor, die über Druck auf die Funktionstasten nachgespielt werden muß. Wird dies fehlerfrei gemacht, fügt der Computer eine weitere Note an die Folge. Die Noten werden zum einen akustisch und zum andern graphisch in unterschiedlichen Farbtönen auf dem Bildschirm dargestellt.
Fehler macht man
– wenn man den falschen Ton spielt,
– wenn man eine andere Taste als F1-HELP oder ESC betätigt,
– oder wenn man zu lange zögert.

Am Anfang wird die Spielstärke auf einer Art Thermometer eingestellt. Dann beginnt das Spiel auf Tastendruck. Mit ESC kann man das Spiel vorzeitig beenden.

Durch Training mit diesem Programm kann man seine Konzentrationsfähigkeit und sein Gedächtnis steigern.

Also, auf nach dem Motto: "C-16 für Grips statt Kalk!" Viel Spaß!!

Zum Programm

Zeilennr.	Kommentar
120 - 130	Funktionstasten werden mit ASCII-Code belegt
150 - 280	Titelbild mit Schwierigkeitsskale
200 - 400	Bewegung des Balkens mittels der Cursor-Tasten
410	Bestimmung des Schwierigkeitsgrades
420 - 460	Warten auf Start
470	Rücksetzen der Variablen
480 - 490	Spielbildschirmgestaltung
510	Erzeugung einer Zufallszahl
520 - 550	Umwandlung der Zahl in einen String
560	Ankettung der Note an den Hauptstring, Ausdruck der Länge des Hauptstrings, also der Nummer des Durchgangs
570 - 700	Der Hauptstring wird in Notenwerte umgewandelt und hörbar gemacht. Das Unterprogramm druckt den Balken im entsprechenden Farbton an die jeweilige Stelle
710 - 990	Abfrage der Notenfolge
740 - 750	Abfrage der Tastertur
760 - 780	Wer zu lang zögert, fliegt 'raus'!
790	Bei ESC wird das Spiel beendet
800	Weder ESC noch F1-HELP gedrückt –– Ende
810	Die aktuelle Note wird bestimmt
820 - 960	Die betätigte Note wird gespielt und per Leuchtbalken dargestellt

Bitte lesen Sie weiter auf Seite 45

```
10 rem sensor============c16=  <hd>
20 rem (p) 7/86 commodore welt <pj>
30 rem ======================= <ci>
40 rem (c) 1986 by             <lo>
50 rem      jan lichtenberg    <mb>
60 rem                         <ml>
70 rem basic v3.5              <cd>
80 rem c16/116/plus4           <ph>
90 rem ======================= <ge>
100 gosub 1410                 <lh>
110 trap1270                   <ap>
120 key1,chr$(133):key2,chr$(1
37)                            <mb>
130 key3,chr$(134):key8,chr$(1
40)                            <mo>
140 printchr$(14):printchr$(8)
:color0,1:color1,6:vol5        <hb>
150 rem *** schwierigkeit **** <bn>
160 scnclr:print:printtab(16)"
Sensor":printtab(16)"======":p
rint                           <ik>
170 char,5,5,"Schwierigkeitsgr
ad"                            <hp>
180 char,5,6,zm$+zm$+zm$+zm$+z
m$+zm$+zm$+zm$+zm$+zm$+zm$+zm$
+zm$+zm$+zm$+zm$+zm$+zm$       <bd>
190 char,25,9,"Benuetzten Sie" <ol>
200 char,25,10,"bitte die Tas-
":char,25,11,"ten              <ih>
210 char,25,13,rn$+"CURSOR auf
"+rf$                          <nd>
220 char,25,15,rn$+"CURSOR run
ter"+rf$                       <od>
230 char,25,17,"Bestaetigen Si
e"                             <gj>
240 char,25,18,"bitte mit"     <mk>
250 char,25,20,rn$+"RETURN"+rf
$                              <bp>
260 char,9,9,ze$+" sehr schwer
"                              <ik>
270 char,9,20,zm$+" sehr leich
t"                             <lc>
280 char,2,24,rn$+"(C) 1986 by
 Jan Lichtenberg, Bremen"      <ld>
290 l=1                        <fc>
300 for c=20 to 20-l+1 step -1
:char,6,c,rn$+bl$+bl$+rf$:next <bh>
310 getkey a$                  <dg>
320 if a$=chr$(13) then 410    <jl>
330 if a$=chr$(145)then 360    <np>
340 if a$=chr$(17) then 380    <lj>
350 goto310                    <hm>
360 l=l+1:if l=12 then l=l-1:g
oto310                         <ni>
370 goto300                    <ip>
380 l=l-1:if l=0  then l=l+1:g
oto310                         <lj>
390 char,6,20-l,bl$+bl$        <ii>
400 goto310                    <ko>
410 l=int(1/l*5000)            <dk>
420 rem ***** spielstart ***** <od>
430 scnclr:print:printtab(16)"
Sensor":printtab(16)"======":p
rint                           <jj>
440 char,10,12,"Bitte "+rn$+"R
ETURN"+rf$+" fuer Start"       <on>
450 getkey a$:if a$<>chr$(13)
then 450                       <am>
460 char,10,12,left$(bk$,25)   <fc>
470 b$="":e=0                  <je>
480 char,2,5,"F1":char,2,9,"F2
":char,2,13,"F3"               <jn>
490 char,2,17,"HELP":char,25,2
2,"Durchgang:"                 <fa>
500 rem *** ton bestimmen **** <dp>
510 b=int(rnd(1)*4)+1          <pb>
520 if b=1 then c$="1"         <op>
530 if b=2 then c$="2"         <pl>
540 if b=3 then c$="3"         <ah>
550 if b=4 then c$="4"         <bd>
560 b$=b$+c$:char,36,22,str$(l
en(b$))                        <ll>
570 rem ** melodie spielen  ** <ic>
580 for i=1to len(b$)          <bb>
590 b=val(mid$(b$,i,1))        <kn>
600 rem ** notenwerte erkennen <jo>
610 on b goto 620,630,640,650  <kk>
620 p=5:t=770:goto660          <oc>
630 p=9:t=810:goto660          <ol>
640 p=13:t=854:goto660         <ci>
650 p=17:t=881                 <nh>
660 gosub1300                  <oh>
670 sound 1,t,int(l/80)        <lo>
680 for q=1tol/4:next          <om>
690 gosub1360                  <al>
700 next i                     <ij>
710 rem ***** abfrage ******** <gb>
720 for x=1to len (b$)         <km>
730 char,12,12,f1$+"??????????
????"+fo$:poke239,0            <ek>
740 for i=1 to l/1.5:get a$:if
 a$<>"" then 790               <bn>
750 next i                     <ll>
760 sound1,300,30:char, 8,5,"S
ie haben zu lange gewartet !"  <pl>
770 for m=1to1000:next m       <ce>
780 goto1070                   <ga>
790 a=asc(a$):if a=27 then 107
0                              <bi>
800 if a<>133 and a<>137 and a
<>134 and a<>140 then 1130     <bk>
810 b=val(mid$(b$,x,1))        <jj>
820 rem * tastaturcode wandeln <im>
830 if a=133 then a=1          <cb>
840 if a=137 then a=2          <da>
850 if a=134 then a=3          <di>
```

```
860 if a=140 then a=4                <ea>
870 rem ** notenwerte erkennen       <kn>
880 on a goto 890,900,910,920        <mb>
890 p=5:t=770:goto930                <pb>
900 p=9:t=810:goto930                <pk>
910 p=13:t=854:goto930               <dh>
920 p=17:t=881                       <og>
930 gosub1300                        <pg>
940 sound1,t,int(1/80)               <mn>
950 for j=1to 1/4:next               <pe>
960 gosub 1360                       <bk>
970 if a<>b then 1170                <kp>
980 char,12,12,left$(bk$,14)         <fm>
990 next x                           <ll>
1000 if len(b$)=29 then 1030         <hi>
1010 for y=1to 600:next y            <ab>
1020 goto510                         <bo>
1030 scnclr:print:print:print:
print:print"Bravo, Du hast mic
h geschafft !!"                      <lc>
1040 print:print"Du hast 29 Du
rchgaenge gemacht !!!"               <hb>
1050 gosub1200:gosub1240             <pm>
1060 goto160                         <ei>
1070 scnclr:print:print:print:
print                                <gg>
1080 print"Du hast ganze";len(
b$)-1;" Durchgaenge geschafft.
"                                    <dn>
1090 if len(b$)<11 then print"
Du musst aber noch etwas ueben
.":goto1110                          <ic>
1100 print"Du bist ja wirklich
 ein Konzentrations-":print"ku
enstler !":gosub 1200               <lb>
1110 gosub1240                       <ko>
1120 goto160                         <ie>
1130 scnclr:print:print:print:
print:print"Jetzt hast du eine
 falsche Taste"                      <kl>
1140 print"gedrueckt ! Deshalb
 ist das Spiel zu-":print"ende
"                                    <fl>
1150 for y=1to1500:next              <gf>
1160 goto1070                        <nn>
1170 scnclr:print:print:print:
print:printtab(8)"Das war fals
ch."                                 <ag>
1180 sound 1,300,30                  <pj>
1190 for i= 1to 1500:next i:go
to 1070                              <fb>
1200 for i=1to10                     <gg>
1210 for y= 1000 to 800 step-8       <nd>
1220 sound 1,y,1:next y,i            <dp>
1230 return                          <ga>
1240 print:printrn$"Noch ein S
pielchen ?"rf$:getkey a$             <cd>
1250 if a$="j" then return:els
e if a$<>"n"thenprintc2$c2$;:g
oto1240                              <hh>
1260 print:print:print"na dann
 bis bald.tschuess !!"               <pg>
1270 printchr$(9):printchr$(14
2):end                               <ca>
1280 return                          <jd>
1290 rem * balken ausdrucken *       <je>
1300 f=8-b                           <dg>
1310 color1,3,f                      <am>
1320 for t1=pto p+3                  <ag>
1330 char,6,t1,rn$+left$(bk$,2
0)+rf$                               <ic>
1340 next:color1,6:return            <dp>
1350 rem ** balken loeschen **       <ia>
1360 for t1= pto p+3                 <co>
1370 char1,6,t1,left$(bk$,20)        <ah>
1380 next:return                     <ld>
1390 rem nachspann ==========        <dl>
1400 rem farbcodes/steuercodes       <ak>
1410 rn$=chr$(018):fl$=chr$(13
0)                                   <pm>
1420 fo$=chr$(132):c2$=chr$(14
5)                                   <ob>
1430 rf$=chr$(146):bl$=" "           <jo>
1440 bk$=bl$:fori=1to5:bk$=bk$
+bk$:next                            <np>
1450 rem zeichensatz/graphik         <ll>
1460 ze$=chr$(175):zm$=chr$(18
3)                                   <eo>
1470 return                          <fb>
1480 rem 12277 bytes memory          <lp>
1490 rem 03808 bytes program         <ai>
1500 rem 00161 bytes variables       <ii>
1510 rem 00000 bytes arrays          <md>
1520 rem 00383 bytes strings         <dd>
1530 rem 07925 bytes fre(0)          <hk>
1540 rem 00000 bytes fre(0)          <go>
1550 rem ====================        <ke>
```

Fortsetzung von Seite 43

SENSOR

970	Prüfung, ob die betätigte Note falsch ist
1000	Mehr als 25 Durchgänge — Spitze!
1030 - 1060	Wer mehr als 25 Durchgänge geschafft hat, wird hier gelobt
1070 - 1120	Auswertung der Leistung
1130 - 1190	Fehleranalyse
1200 - 1230	Neuer Spielstart oder Spielende
1100 - 1170	Hier werden die Balken gedruckt und wieder gelöscht

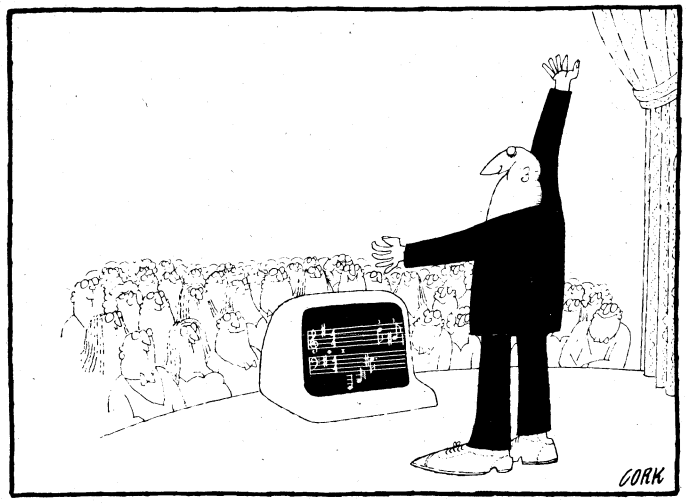
Außerdem ist das Programm mit einigen REM-Kommentarzeilen versehen, so daß es sehr leicht ist, den Programmaufbau zu verstehen.

```
10 rem skeet===============c16 <fc>
20 rem (p) 7/87 commodore welt <pk>
30 rem ======================== <ci>
40 rem (c) hubert muelbacher  <on>
50 rem                        <mb>
60 rem                        <ml>
70 rem basic v3.5             <cd>
80 rem c16/116/plus4          <ph>
90 rem ======================== <ge>
100 poke55,0:poke56,48:clr    <gl>
110 gosub 1230                <mb>
120 color 0,1:color4,1:scnclr <ma>
130 data -40,-39,1,41,40,39,-1
,-41                          <lj>
140 fori=1to8:readx:ax(i)=x:ne
xt                            <gb>
150 x=15:y=7                  <ic>
160 get a$:if a$=" " then 280 <ah>
170 color1,x,y:color4,(x+1),y <ff>
180 printrn$"DDDDDDDDDDDDDDDDs
DkDeDeDtDDDDDDDDDDDDDDDD"     <jn>
190 y=y-1   :ify= 1then 210   <dj>
200 goto 160                  <oi>
210 color1,x,y:color4,(x+1),y <hn>
220 printrn$"DDDDDDDDDDDDDDDDs
DkDeDeDtDDDDDDDDDDDDDDDD"     <mf>
230 y=y+1:if y=8 then 250     <gl>
240 goto 210                  <am>
250 y=7:x=x-1:ifx=1 then x=15 <ml>
260 goto160                   <cf>
270 getkey a$:list -199:end   <ol>
280 color1,1:vol 8            <nl>
290 poke 65298,peek(65298)and2
51                            <fm>
300 poke 65299,peek(65299)and3
or48                          <bp>
310 for i = 0 to 17 : read a  <pb>
320 poke819+i,a:next:sys819   <ba>
330 for za=12808 to 13148 step
 8                            <kc>
340 for i=0to7:read a         <no>
350 if a=999 then 430         <mg>
360 poke za+i,a:nexti :nextza <fh>
370 data162,0,189,0,208,157,0,
48,189,0,209,157,0,49,202,208,
241,96                        <cl>
380 data 0, 60, 102, 90, 90, 1
02, 60, 0                     <fi>
390 data 24, 0, 24, 165, 165,
24, 0, 24                     <go>
400 data 24, 60, 126, 62, 124,
 126, 24, 60                  <bb>
410 data 8, 66, 16, 132, 33, 8
, 66, 16                      <ge>
420 data 999,999,999,999,999,9
99,999,999                    <le>
430 color0,14,5               <li>
440 s=3811:p=200:mu=62:h=410:m
t=-1                          <ib>
450 scnclr                    <kl>
460 color4,1                  <ef>
470 color1,1                  <em>
480 printhe$rn$" tauben"t;c1$"
..."                          <oh>
490 printhe$rn$left$(qr$,12)"h
ighscore"hc;c1$"....."        <na>
500 printhe$rn$left$(qr$,28)"s
core"sc;c1$"....."            <pk>
510 char1,0,1,rn$+"..........
.........................."   <nh>
520 for wa=3952 to 4071       <bi>
530 pokewa,67:poke(wa-1024),53
:next                         <fi>
540 goto1160                  <gp>
550 ax=joy(1)                 <me>
560 if ax>127 then 780        <oh>
570 ifax<>0thenri=ax(ax):goto6
00                            <me>
580 pokes,66                  <mi>
590 goto 640                  <hd>
600 ifpeek((s+ri))<>32then630 <ok>
610 s=s+ri                    <ab>
620 pokes,66:poke(s-ri),32    <dm>
630 ifpeek((s+ri))=65then610  <fk>
640 poket,32                  <po>
650 if t>0 then 720           <im>
660 rl=int(rnd(1)*10)+1:ta=ta+
1:sa=0                        <lc>
670 if ta>10then ta=0         <pa>
680 printrn$he$left$(qr$,7)ta;
c1$" "                        <kh>
690 if ta=0then 1160          <cf>
700 l=int(rnd(1)*10)+10       <bh>
710 ifrl<6thentr=-39:t=3*rl+39
09:elsetr=-41:t=3*rl+3921     <nj>
720 d=d+1                     <oh>
730 if d>landd<(l+10) then tr=
40                            <fj>
740 if d>(l+9)then t=0:d=0:got
o550                          <ia>
750 t=t+tr                    <jk>
760 poket,65                  <hm>
770 goto 550                  <ci>
780 sa=sa+1:sb=sb+1           <lf>
790 ifsa>2then640             <fm>
800 ifpeek(s)<>65then900      <cd>
810 poke s,68                 <lb>
820 p=p-(d*5):if p<=0then p=10 <fo>
830 if p>200 then p=200       <jk>
840 sc=sc+p                   <cd>
850 sound3,853,15:d=0:t=0:p=20
0:at=at+1                     <ep>
860 printrn$he$left$(qr$,33)sc <hf>
870 ifsc>hcthen hc=sc         <ea>
880 printrn$he$left$(qr$,21)hc <hl>
890 pokes,32:goto550          <ff>
```

```
900 sound3,999,10                                   <pi>
910 goto 550                                        <le>
920 color0,2,1                                      <gn>
930 printhe$cy$left$(qd$,8)lef
t$(qr$,7)"DDDDDDDDDDDDDDDDDDDD
DDDDDD"                                             <fe>
940 printcy$left$(qr$,7)"D"pu$
"AAAAAAAAAAAAAAAAAAAAAAA"cy$"
D"                                                  <ea>
950 printcy$left$(qr$,7)"D"pu$
"A.....A"cy$"
D"                                                  <ki>
960 printcy$left$(qr$,7)"D"pu$
"A.."ye$"g a m e....o v e r.."
pu$"A"cy$"D"                                        <jh>
970 printcy$left$(qr$,7)"D"pu$
"A.....A"cy$"
D""                                                 <no>
980 printcy$left$(qr$,7)"D"pu$
"AAAAAAAAAAAAAAAAAAAAAAA"cy$"
D"                                                  <gi>
990 printcy$left$(qr$,7)"DDDDD
DDDDDDDDDDDDDDDDDDD"bk$                             <fk>
1000 printoe$c4$left$(qr$,7)".
....neues spiel? j/n"bk$                            <pi>
1010 for tn= 1000 to 400 step
- 10                                                <hi>
1020 sound1,tn,3:sound2,(tn+1)
,3:next                                             <jc>
1030 geta$:ifa$="j"thenat=0:mt
=0:ro=0:sc=0:goto430                                <bo>
1040 if a$="n" then1060                             <lg>
1050 goto 1030                                      <gl>
1060 printlg$c4$left$(qr$,15)"
schade!!!"                                          <ne>
1070 poke65290,160                                  <jc>
1080 fora=0to7                                      <ll>
1090 poke65287,a                                    <fg>
1100 next a                                         <bd>
1110 fora=7to0 step -1                              <fo>
1120 poke65287,a                                    <he>
1130 next a                                         <db>
1140 poke 65287,8                                   <hp>
1150 goto 1080                                      <ne>
1160 if at<mtthen 920                               <ec>
1170 at=0:mt=mt+1:ifmt>10then
mt=10                                               <oj>
1180 ro=ro+1                                        <gp>
1190 printbk$rn$he$rn$left$(qd
$,23)left$(qr$,6)"..runde.."ro
;c1$".....min.."mt;c1$".."                          <jk>
1200 for tn =500 to  900 step
 8                                                  <dm>
1210 sound 1,tn,2:sound2,(tn+1
),3:next                                            <fa>
1220 goto550                                        <ol>
1230 rem nachspann ==========                       <jk>
1240 rem farbcodes/steuercodes                      <gj>
1250 c4$=chr$(017):rn$=chr$(01
8)                                                  <ee>
1260 he$=chr$(019):c3$=chr$(02
9)                                                  <do>
1270 oe$=chr$(129):bk$=chr$(14
4)                                                  <gg>
1280 lg$=chr$(153):pu$=chr$(15
6)                                                  <ii>
1290 c1$=chr$(157):ye$=chr$(15
8)                                                  <hc>
1300 cy$=chr$(159)                                  <ec>
1310 rem ******* zeichenfolgen                      <jj>
1320 for q=1 to 40                                  <ok>
1330 qd$=qd$+c4$:qr$=qr$+c3$                        <fo>
1340 next q                                         <be>
1350 return                                         <nj>
1360 rem ====================                       <of>
1370 rem 12277 bytes memory                         <fb>
1380 rem 03529 bytes program                        <jk>
1390 rem 00273 bytes variables                      <bo>
1400 rem 00062 bytes arrays                         <fn>
1410 rem 00432 bytes strings                        <ma>
1420 rem 00512 bytes zeichen                        <kf>
1430 rem 07469 bytes fre(0)                         <bj>
1440 rem ====================                       <df>
```

SKEET – TONTAUBENSCHIESSEN

Schießen ist keine Kunst, das Treffen dagegen schon, zumal, wenn das zu treffende Objekt einfach nicht stillstehen will.
Absichtlich kurz gehalten, damit es in Kürze abzutippen ist, ist dieses Spiel. Mit dem Joystick in Port 1 haben Sie die Aufgabe, ein Koordinatenkreuz, welches das Visier eines Gewehrs versinnbildlicht, mit der irgendwo aus einer unsichtbaren Abschußvorrichtung hervorsausenden Tontaube in Deckung zu bringen und im richtigen Augenblick auf den Feuerknopf zu drücken. drücken. Das hört sich vielleicht leichter an, als es ist. Es braucht aber viel Übung, bis man dieses meisterhaft beherrscht.



C128 INTERN

Der Zugriff auf den Videochip 8563, der für die 80-Zeichen-Darstellung zuständig ist, kann — auch von Basic aus — überraschende Effekte auslösen. Im letzten Heft hatten wir bereits durch direkten Zugriff auf diesen 80-Zeichen-Manager gezeigt, wie selbst durch ein einfaches Basicprogramm eine Hardcopy-Routine für ASCII-Zeichen programmiert werden kann. Wenn man ein wenig mehr über diesen Baustein weiß, sind aber auch noch andere, teilweise sehr verblüffende Effekte zu programmieren. Damit auch wir uns schlau machen konnten, haben wir verschiedene Unterlagen gewälzt. Dabei fiel uns auf, daß sich die — uns vorliegenden — Unterlagen teilweise widersprechen. Wir richteten uns deshalb vor allem nach den englischen Unterlagen. Da wir aber nicht voraussetzen können, daß jeder C-128-Besitzer englisch versteht, haben wir die Tabelle teilweise eingedeutscht. Wer noch weiter gehen will, als wir es mit diesem Artikel tun, kommt ohne weiterführende Literatur nicht aus und noch mehr Details würden den Rahmen dieses Artikels sprengen.

Im 80-Zeichen-Mode geschieht die Video-Ausgabe (RGBI) über den 8563, der nur über zwei Adressen angesprochen werden kann. Auch der Bildschirmspeicher (16 K-Byte) ist nur indirekt erreichbar, wie Sie in unserer letzten Ausgabe bereits lesen konnten.
Der 8563-Videochip ist mit verschiedenen Registern ausgestattet, die über insgesamt zwei Adressen angesprochen werden. Über die Adresse $D600 erreicht man das Adreß- und über $D601 das Datenregister. Um mit dem Videochip kommunizieren zu können, muß dem Adreßregister erst mitgeteilt werden, welches der internen Register angesprochen werden soll. Dies geschieht von Basic aus dadurch, daß durch den Befehl: POKE DEC("$D600"),x mitgeteilt wird, daß das Register mit der Nummer x gemeint ist.
Der 8563 besitzt 37 interne Register mit den Registernummern 0 bis 36. Jedem dieser Register sind besondere Aufgaben zugeteilt. Siehe hierzu Aufstellung 1.
Wie Sie dieser Aufstellung entnehmen können, sind die in manchen Registern stehenden Werte für verschiedene Punkte zuständig.
Wir können nur andeutungsweise auf die vielfältigen Möglichkeiten der Manipulation von Basic aus eingehen, wollen aber gleich darauf hinweisen, daß nicht alles, was in Maschinensprache realisiert werden kann, auch von Basic aus durchführbar ist.
Wie Sie aber den vielen Programmbeispielen entnehmen können, ist es trotzdem jede Menge.

EXPERIMENTIEREN IST ANGESAGT!

Einige der Programmbeispiele entstanden bei uns durch Experimente, nachdem wir mit dem Leseprogramm VDCREAD (Programm 1) die Inhalte der verschiedenen Register ausgelesen hatten.

```
100 REM VDCREAD
110 REM PROGRAMM ZUM AUSLESEN
120 REM DER REGISTER DES VDC 8563
130 REM
140 R0=DEC("D600"):R1=R0+1
150 INPUT"REGISTER-NR: ";RN
160 POKE R0,RN
170 PRINT PEEK(R1)
180 GOTO 150
```

Dieses Programm dient zum Auslesen der VDC-Register-Inhalte.

```
100 REM EFFEKTE 1
110 REM NUR ZUR DEMONSTRATION WAS DURCH
120 REM BEEINFLUSSUNG DER VDC-REGISTER
130 REM GETRIEBEN WERDEN KANN
140 REM
150 R0=DEC("D600"):R1=R0+1
160 FOR I =0 TO 15:POKE R0,9:POKE R1,I
170 FOR J = 1 TO 200:NEXT J
180 NEXT I
190 POKE R0,9:POKE R1,7
```

Kurz aber effektvoll — abtippen und laufen lassen.

```
100 REM AUFZUMITTE
110 R0=DEC("D600"):R1=R0+1
120 FOR I = 1 TO 80:A$=A$+"*":NEXT I
130 PRINTCHR$(147):FORI=1 TO 20:PRINTA$:NEXTI
140 :
150 FOR I = 90 TO 7 STEP-1
160 POKE R0,34:POKE R1,92-I:POKE R0,35:POKE R1,I
170 NEXT I
180 :
190 REM RUECKSTELLUNG
200 POKE R0,35:POKER1,100
210 POKE R0,34:POKER1,125
```

Diese Methode der VDC-Manipulation eignet sich gut für Spiele.

```
100 REM SCHRAEGSCREEN2
110 R0=DEC("D600"):R1=R0+1
130 POKE R0,1:POKER1,79
140 POKE R0,2:POKER1,110
150 :
160 GETX$:IFX$="" THEN 160
170 IF X$="S" THEN STOP
180 :
200 POKE R0,1:POKE R1,80
210 POKE R0,2:POKE R1,102
```

Falls Sie mit schrägem Listing arbeiten wollen drücken Sie bitte 'S'.

```
100 REM ZAUF-AB
110 R0=DEC("D600"):R1=R0+1:CS$=CHR$(147)
120 :
130 PRINTCS$"COMMODORE-WELT":PRINT:INPUT"VERZOEGERUNG :";V
140 :
150 FOR I = 1 TO 20
160 PRINT"****************************************"
170 NEXT I
180 :
190 FOR I = 25 TO 1 STEP -1:POKE R0,6:POKE R1,I:GOSUB 240:NEXT I
200 :
210 FOR I = 1 TO 25:POKE R0,6:POKE R1,I:GOSUB 240:NEXT I
220 END
230 REM ---------- VERZOEGERUNG ----------
240 FOR J = 1 TO V:NEXT J:RETURN
```

Lassen Sie Zeilen verschwinden und wieder auftauchen. Wählen Sie anfangs kurze Verzögerungszeiten.

```
100 REM ZITTERN
110 R0=DEC("D600"):R1=R0+1:CS$=CHR$(147)
120 PRINTCS$"COMMODORE-WELT
130 PRINT:PRINT"HEUT' HAB ICH ABER WIEDER ANGST!
140 PRINT:PRINT"SIEHST DU, WIE ICH ZITTERE? (ICH ARBEITE IM INTERLACE-MODUS!)
150 PRINT:POKE R0,8:POKER1,1:REM INTERLACE-MODUS EIN
160 PRINT:PRINT"WILLST DU MIR DIE ANGST NEHMEN ? (J = JA  N=NEIN)
170 :
180 GETX$:IFX$="" THEN 180
190 IF X$="N" THEN PRINT"GUT, DANN HAB' ICH EBEN WEITER ANGST!":END
200 IF X$="J" THEN PRINT"DANKE DAS IST WIRKLICH NETT VON DIR!":GOTO 250
210 PRINT"DA MUSS MAN JA ANGST HABEN, DU KANNST JA NICHT EINMAL LESEN"
220 PRINT"ES SOLLTE 'J' ODER 'N' GEDRUECKT WERDEN UND NICHT: "X$:GOTO 160
230 :
240 REM RUECKSTELLUNG
250 POKE R0,8:POKE R1,0
```

Interlace-Mode — nicht auf allen Bildschirmen gut sichtbar.

```
100 REM HAMMER
110 PRINTCHR$(147):FORI=1 TO 20
120 PRINT"**********************
130 NEXT I
140 PRINT"ZUM BEENDEN 'E' DRUECKEN
150 R0=DEC("D600"):R1=R0+1
160 FOR I = 32 TO 26 STEP-4
170 POKE R0,7:POKER1,I
180 FOR J=1 TO 20:NEXT J
190 NEXT I
200 FOR I = 32 TO 37
210 POKE R0,7:POKER1,I
220 FOR J=1 TO 20:NEXT J
230 NEXT I
240 GETX$:IF X$<>"E" THEN 160
250 :
260 POKE R0,7:POKER1,32
```

Wirkt wie ein Hammer — abtippen, laufen lassen und ansehen.

```
100 REM SCHRAEGSCREEN
110 R0=DEC("D600"):R1=R0+1
130 POKE R0,1:POKER1,79
140 POKE R0,2:POKER1,80
150 :
160 GETX$:IFX$="" THEN 160
170 IF X$="S" THEN STOP
180 :
200 POKE R0,1:POKE R1,80
210 POKE R0,2:POKE R1,102
```

Nochmal ein schräger Bildschirm — diesmal anders.

```
100 REM PIXRADIER
110 REM VERAENDERUNG DER DARGESTELLTEN ZEICHEN PRO ZEILE
120 :
130 R0=DEC("D600"):R1=R0+1:X=9:C$=CHR$(147):WS=50:AZ=15
140 Z$="**********************************************"
150 PRINT C$:FOR I=1 TO AZ:PRINT Z$:NEXT I
160 :
170 PRINT"BITTE DIE SPACE-TASTE DRUECKEN!
180 GETX$:IFX$<> CHR$(32) THEN 180
190 :
200 FOR I = 0 TO X:GOSUB 310:REM SCHLEIFE ZUR AENDERUNG DER PIXELZEILEN
210 GOSUB 320:REM VERZOEGERUNG
220 NEXT I:REM NAECHSTER SCHLEIFENDURCHLAUF
230 :
240 PRINT"MOMENT ES GEHT NOCH WEITER  -  NUN ALLES ANDERS HERUM!
250 :
260 FOR I = X TO 0 STEP-1:GOSUB 310:GOSUB 320:NEXT I
270 :
280 POKE R0,23:POKE R1,9:END
290 :
300 REM ------------ UNTERPROGRAMME ------------------------------
310 POKE R0,23:POKE R1,I:RETURN:AENDERUNG DER PIXELZEILEN
320 FOR J=1 TO WS:NEXT J:RETURN:REM VERZOEGERUNGSSCHLEIFE
```

Pixelzeilen verschwinden und tauchen wieder auf.

Die von uns ausgelesenen Werte finden Sie ebenfalls in Tabelle 1. Aufgrund der widersprüchlichen Angaben und auch deshalb, weil wir noch nicht alles nachgeprüft haben, können wir noch keine genauen Details bringen. Mit Details meinen wir in diesem Falle, welche Bits (in den verschiedenen Registern) für was zuständig sind. Eine derartige Überprüfung muß aber (um wirklich sicher zu gehen) auf Maschinensprache-Ebene geschehen und das nimmt sehr viel Zeit in Anspruch. Wer also weiter gehen will, als wir es in diesem Artikel tun, kommt um weiterführende Literatur und auch die Programmierung in Maschinensprache nicht herum. Artikel über die Programmierung in Assembler (Maschinensprache) sind aber eigene Beiträge, verzeihen Sie deshalb, wenn wir es in dieser Artikelserie nicht berücksichtigen können. Zum allgemeinen Verständnis könnten die Angaben in der Tabelle aber schon ausreichen.
Zwar konnten wir aufgrund der ausgelesenen Werte vieles verstehen, einiges blieb aber auch noch im Dunkeln! Trotzdem sind mit den Programmen Beispiele auf den Bildschirm zu zaubern, die manchen verblüffen werden. Und das, obwohl (wir erwähnten es schon) in manchen Publikationen zu lesen ist, die Befehle PEEK, POKE und WAIT können für Zugriffe auf den 8563-Videochip nicht benutzt werden! Wir beweisen das Gegenteil! Wie Sie durch die Programmbeispiele selbst feststellen können, ist auch mittels Basic vieles möglich. Durch eigene Experimente können Sie noch vieles andere herausfinden und noch viel schlimmere Effekte auf den Bildschirm bringen, als wir es tun.
Benutzen Sie in den Beispielprogrammen ruhig einmal andere Werte. Aber bevor Sie ein Programm mit geänderten Werten starten, speichern Sie es sicherheitshalber ab! Wenn Sie mit dem Bildschirm und den Eingaben nach einem Programmlauf nicht mehr klarkommen, dann führen Sie am besten einen Neustart durch.

ZU DEN PROGRAMMEN

Es wäre nun wirklich zu umfangreich, auf alle Demonstrationsprogramme, die wir hier abdrukken, genauestens einzugehen. Sie sollen nur als Anregung für eigene Unterprogramme dienen. Damit der Einstieg aber nicht zu schwer wird, hier die Erläuterung zum Programm VDCREAD (Programm 1). Die Zeilen bis 130 enthalten lediglich Bemerkungen. In Zeile 140 werden den beiden Variablen R0 und R1 die Adreßwerte mitgeteilt. In Zeile 150 erfolgt die Abfrage nach der Register-Nummer, dessen Inhalt Sie abfragen wollen. In Zeile 160 wird die gewählte Register-Nummer dem Adreßregister mitgeteilt. Durch Zeile 170 wird der Inhalt des ausgewählten Registers auf den Bildschirm ausgegeben. Durch Zeile 180 fährt das Programm fort. Sie sehen, das Basiswissen, das man zur Beeinflussung des VDC-Chips benötigt, ist gar nicht so umfangreich. Komplizierter wird es aber dann, wenn man noch tiefer einsteigt. Trotzdem bleibt die Vorgehensweise aber immer gleich. Dem Adreßregister die interne Registernummer mitgeteilt und dann das Datenregister ($D601) ausgelesen oder hineingeschrieben. Das ist alles! Sinnvollerweise aber setzen Sie bei Ihren Experimenten (vor einem Programmlauf) im Programmende immer die Werte, die Sie aus dem zu manipulierenden Register ausgelesen oder unserer Aufstellung entnommen haben.

```
100 REM AUWEI
110 R0=DEC("D600"):R1=R0+1
120 PRINT CHR$(147)"COMMODORE-WELT":PRINT
130 PRINT"MANCHMAL SIEHT ES JA SCHLIMM AUS,
140 PRINT"WENN DURCH MANIPULATION DER
150 PRINT"8563-REGISTER, DER BILDSCHIRM
160 PRINT"QUASI ZUSAMMENBRICHT"
170 PRINT:PRINT"DER PROGRAMMAUTOR
180 :
190 FOR J = 1 TO 2
200 FOR I = 0 TO 80:POKE R0,1:POKE R1,I:NEXT I
210 FOR I = 40 TO 1 STEP -1:POKE R0,1:POKE R1,I:NEXT I
220 FOR I = 0 TO 80:POKE R0,1:POKE R1,I:NEXT I
230 NEXT J
240 :
250 POKE R0,1:POKE R1,80
```

Sieht wirklich schlimm auf dem Bildschirm aus.

```
100 REM CRAZYSCREEN
110 R0=DEC("D600"):R1=R0+1:CS$=CHR$(147)
120 PRINT CS$"DIESES PROGRAMM DER COMMODORE-WELT BEEINFLUSST DEN
130 PRINT"8563-VIDEO-CHIP DES C128 IN VERSCHIEDENER WEISE
140 :
150 PRINT:PRINT"SCHIEBEN WIR DEN TEXT ERST EINMAL LANGSAM NACH RECHTS ..->
160 VZ=20:FOR I = 102 TO 0 STEP-1:GOSUB 360:GOSUB 330:NEXT I
170 :
180 PRINT TAB(15)"ETWAS SCHNELLER WIEDER ZURUECK, ALSO AB NACH LINKS ..<-
190 VZ=10:FOR I = 0 TO 102:GOSUB 360:GOSUB 330:NEXT I:PRINT
200 :
210 PRINT"DAS GANZE NUN NOCH SCHNELLER ..RECHTS... ->":GOSUB 340
220 VZ=5:FOR I = 102 TO 0 STEP-1:GOSUB 360:NEXT I:I=102:GOSUB 360:PRINT
230 PRINT"START VON DER 'NULLSTELLUNG' AUS, AB NACH LINKS ...<- ...":GOSUB 340
240 FOR I = 0 TO 102:POKE R0,2:POKE R1,I:NEXT I:PRINT
250 PRINT"NUN SCHMEISSEN WIR ALLES DURCHEINANDER, BAUEN AUF UND WIEDER AB!"
260 GOSUB 340:FOR I = 0 TO 80:GOSUB 370:NEXT I
270 FOR I = 80 TO 1 STEP -1:GOSUB 370:NEXT I
280 VZ=10:FOR I=1 TO 80:GOSUB 360:GOSUB 330:GOSUB 370:GOSUB 330:NEXT I
290 :
300 REM KORREKTUR AUF KORREKTE WERTE UND ENDE
310 POKE R0,1:POKE R1,80:POKE R0,2:POKE R1,102:END
320 REM ------------ UNTERPROGRAMME ----------------------------------
330 FOR J = 1 TO VZ:NEXT J:RETURN:VERZOEGERUNGSSCHLEIFE 1
340 PRINT:PRINT"BITTE DIE SPACE-TASTE DRUECKEN!"
350 GETX$:IFX$<>CHR$(32) THEN 350:ELSE RETURN
360 POKE R0,2:POKE R1,I:RETURN
370 POKE R0,1:POKE R1,I:RETURN
```

Mehrere Beeinflussungen gleichzeitig — sieht stark aus.

Spaß mit den Programmen und vor allem auch mit Ihren eigenen Experimenten.

ZEILENNUMMERN AUTOMATISCH ERZEUGT

Auch wenn die Übersichtlichkeit durch die Vergabe von Zehnerschritten (manche Computer-Magazine halten sich daran!) doch ziemlich verbessert ist, die Tipparbeit bleibt halt doch. Lassen Sie das künftig Ihren C 64 erledigen! Das folgende, kleine Programm erweitert das Basic 2.0 um den Befehl "AUTO", wie ihn beispielsweise der C 16, Plus/4 oder der C 128 kennen. Er wird vor der Abtipparbeit mit der gewünschten Schrittweite (z.B. 10) eingegeben, und nach jeder eingetippten Listingzeile, die mit der RETURN-Taste vom Computer übernommen wurde, erscheint am nächsten Zeilenanfang die neue Zeilennummer, so daß Ihnen diese Arbeit schon mal abgenommen ist und Sie gleich mit dem Listingtext weitermachen können.

Das Programm „Autonumber.64", das Sie nach dem Eintippen sofort abspeichern sollten, initialisiert nach "RUN" den neuen Befehl "A" (von "AUTO") für den C 64 durch ein "SYS 49269" und löscht sich selbst. Der Basicspeicher ist für die Eingabe Ihres Listings frei, das Sie nun abtippen wollen. Sind die Zeilennummern z.B. in 5er-Schritten im Heft vorgegeben, so lautet der Befehl:

A 5 (anschließend drücken Sie die <RETURN>-Taste!)

Beginnt nun das Listing mit der Zeile 100, so geben Sie diese Zeile ein. Nach <RETURN> erscheint dann in der nächsten Zeile die Zahl "105", in der Sie dann sofort weitertippen können. Falls es einmal erforderlich sein sollte, können Sie vorher diese vorgegebene Zeilenzahl auch verändern, z.B. in "108", die von Ihnen angegebene Schrittweite wird jedoch beibehalten, bis Sie sie ändern. (Als nächste Zeilenzahl würde dann eben "113" erscheinen!). Der "Auto"-Modus läßt sich ganz einfach wieder ausschalten: Drücken Sie die RETURN-Taste, nachdem eine neue Zeilennummer auf dem Bildschirm erschienen ist.

Das Listing finden Sie a.S. 67!

AUFSTELLUNG 1:

Register #	Zweck	Inhalt
00	Gesamt-Zeichenanzal pro Zeile	127
01	Dargestellte Zeichen pro Zeile	80
02	Synronisationsposition horizontal	102
03	Vertikale/horizontale Sync-Breite	73
04	Rasterzeilen vertikal (gesamt)	38
05	Justierung vertikal (total)	224
06	Anzahl der dargestellten Rasterzeilen	25
07	Synchronisationsposition vertikal	32
08	Modus-Kontrolle (Interlace-Mode)	252
09	Gesamtzeichen vertikal	231
10	Erste Cursorzeile/Cursor-Modus	160
11	Letzte Cursorzeile	231
12	Start Bildwiederholspeicher (high)	0
13	Start Bildwiederholspeicher (low)	0
14	Cursorposition (high)	x
15	Cursorposition (low)	x
16	Lichtgriffelposition vertikal	(34)
17	Lichtgriffelposition horizontal	(102)
18	Update-Adresse high	x
19	Update-Adresse low	x
20	Start Attributspeicher high	8
21	Start Attributspeicher low	0
22	Zeichendarstellung horizontal und Zeichenanzahl gesamt (horizontal)	120
23	Zeichendarstellung vertikal	232
24	Smooth-Scrolling vertikal, Zeichen-Blinkrate, revers, Block kopieren	32
25	Smooth-Scrolling horizontal Pixelbreite, Semigrafik, Zeichenattribute, Text-/Grafik-Modus	71
26	Hinter- und Vordergrundfarbe	240
27	Adresse erhöhen pro Zeile	0
28	RAM-Typ, Startadresse Zeichensatz	47
29	Rasterzeilen beim Unterstreichen	231
30	Bytezähler	x
31	Daten	x
32	Block-Quell-Adresse high	15
33	Block-Quell-Adresse	208
34	Anzahl der dargestellten Zeichen Beginn	125
35	Anzahl der dargestellten Zeichen Ende	100
36	Refresh-Zyklen pro Rasterzeile	245

Die in obiger Aufstellung unter Inhalt angegebenen Werte wurden mit dem Programm 1 ausgelesen und stellen in den meisten Fällen den Standardwert dar. Dort wo x steht, erhielten wir bei mehreren Abfragen unterschiedliche Werte.

Aufteilung des 8563-RAM-Bereiches

0000 – 07FF Zeichenanzeigebereich (Bildwiederholspeicher)
0800 – 0FFF Zeichenattribute (Code für Farben, Blinken usw.)
2000 – 3FFF Zeichensatz-Definitionen

KAMPF DEN ERRORS!

Selbst Profiprogrammierern mit langjähriger Computererfahrung unterlaufen Fehler. Zeitraubend ist die Suche nach den „Bugs", wie Errors in der Fachsprache genannt werden. Besonders Anfänger haben mit Errors Schwierigkeiten: Oftmals wissen sie nicht einmal, was die jeweilige Errormeldung des Computers bedeutet. Aus diesem Grund besprechen wir die einzelnen Fehlermeldungen des Basic 3.5 (Commodore C16) und führen auf, wie die Errors zu beheben sind.

Da C16, C64, C128 und PC10 ähnliche Basic-Interpreter haben, ist dieser Kurs auch für Benutzer der genannten Computersysteme wertvoll.

SPEZIELLES FÜR DEN C16–ANWENDER

Der Basic-Interpreter des C16 benutzt zur Fehlerangabe 2 Variablen: ER und EL.
In ER steht der letzte Fehler seit Start des Basic-Programmes, aus EL ist die Zeilennummer zu lesen, in der zuletzt ein Error auftrat. Die nachfolgenden Erklärungen gehen zum einen auf die ER-Nummer des C16, zum anderen auf den Ausgabetext ein, der durch die Befehlsfolgte PRINT ERR$(ER) zu erhalten ist und den andere Basic-Interpreter bei Auftreten eines Fehlers ohne weitere Eingaben ausgeben.

ÜBER CODES, FEHLER UND TRICKS . . .

Errorcode des C16: 01 Text: Too Many Files Error

Öffnet ein Programm mehr als 10 Kanäle vom Rechner zu externen Geräten wie Laufwerk, Modem oder Drucker, gibt der C16 beim 11. Versuch, einen Kanal zu öffnen, diesen Fehler aus.
Gerne entsteht dieser Fehler, wenn man im OPEN-Befehl Variablen benutzt (Beispiel: OPEN A;B;C), die schlecht kontrolliert sind. Am besten prüft man die benutzten Variablen durch einen PRINT A,B,C-Befehl in einer neuen Zeile direkt vor dem OPEN-Befehl, um zu sehen, ob wirklich die geplanten Kanäle geöffnet werden.

Errorcode des C16: 02 Text: File Open Error

Soll der Basic-Interpreter einen Datenkanal öffnen, der bereits geöffnet ist, reagiert er mit diesem Fehler. Meist verursacht ein falscher Sprungbefehl, der die Abarbeitung des Programmes wieder zur OPEN-Anweisung führt, diesen Fehler.
Zweithäufigste Ursache ist, ähnlich wie unter Code 1, die unkontrollierte Nutzung von Variablen im OPEN-Befehl. Am Besten ist es hier, ebenfalls in einer neuen Zeile direkt vor den OPEN-Befehl einen PRINT A,B,C-Befehl zu stellen (s. Code 1).

Errorcode des C16: 03 Texte: File Not Open Error

Greift ein Programm auf externe Geräte wie Drukker, Floppy oder Bildschirm zu, ist zuvor ein Datenkanal hierfür zu öffnen. Gerne ist ein Tippfehler Ursache dieses Fehlers: Entweder in der OPEN-Anweisung oder in einer der folgenden GET#, INPUT# oder PRINT#-Anweisungen. Auch hier kann unkorrekter Inhalt einer benutzten Variablen Ursache sein (selten). Bei Verdacht auf diese Ursache prüft man vor jedem GET#, INPUT# und PRINT#-Befehl die benutzten Variablen auf deren Inhalt und vergleicht mit dem planmäßigen Inhalt.

Fehlercode des C16: 04 Texte: File Not Found Error

Ist am C16 eine Datasette angeschlossen und sucht der Rechenr auf der dort eingelegten Kassette nach Daten, gibt er einen File Not Found-Error am Bandende aus, wenn er die Bandende-Markierung liest. Floppy-Benutzer erhalten den Fehlercode 04 meist nach DLOAD-Befehlen, wenn entweder das einzulesende File falsch geschrieben wurde, die falsche Diskette eingelegt ist (auf der das gewünschte Programm nicht steht) oder der Diskettenhebel nicht verriegelt ist.

Errorcode des C16: 05 Text– Device Not Present Error

Öffnet man einen Datenkanal zu einem nicht angeschlossenen Peripheriegerät oder versucht man, ein File von Laufwerk zu lesen oder zu schreiben, ohne daß die Floppy angeschlossen oder angeschaltet ist, erhält man diesen Fehlercode. Man prüft daher, ob alle Anschlüsse richtig sitzen und ob jedes Gerät Strom erhält.

Fehlercode des C16: 06 Text: Not Input File Error

Um aus einer Datei Daten zu lesen, muß man einen Datenkanal öffnen.
Der dazu verwendete OPEN-Befehl muß OPEN A,B,0 lauten. Das bedeutet: Variablen A und B sind frei wählbar, der dritte angegebene Wert muß 0 (für Lesen) sein. Meist handelt es sich bei diesem Error um einen Tippfehler.

Errorcode des C16: 07 Text: Not Output File Error

Die Erklärung dieses Fehlers ähnelt Fehlercode 06: Will man aus einer Floppy-Datei Daten lesen, muß der dritte Wert der OPEN-Anweisung 1 oder 2 sein, um dem Interpreter klar zu machen: Aus dieser Datei soll gelesen werden. Benutzt man den Wert 2 und Datasette, schreibt der Commodore automatisch nach Abspeichern aller Daten die Bandendemarkierung EOT auf Kassette.

Errorcode des C16: 08 Text: Missing File Name Error

Lädt man ein Programm von Diskette (DLOAD) oder transferiert Daten mit anderen Befehlen (DSAVE, R, W, . . .) zwischen Rechnern und Laufwerk, ist stets der Filename anzugeben. Fehlt der Name der Datei oder des Programmfiles, gibt der Basic-Interpreter den Missing File Name-Error aus.

Errorcode des C16: 09 Text: Illegal Device Number Error

Öffnet man einen Datenkanal auf den Bildschirm und versucht, von dort eine Datei auszulesen, gibt der Rechner diesen Fehler aus. Denselben Error erhalten Sie ebenfalls, wenn Daten vom Drucker eingelesen werden sollen. Derartige „Unmöglichkeiten" stammen meist aus Tippfehlern.

Errorcode des C16: 10 Text: Next Without For Error

In einer FOR-NEXT-Schleife (Beispiel: FOR I=1to50:PRINT I:NEXT) führt der Rechner sämtliche Befehle zwischen FOR und NEXT sooft aus, wie in einer Zählvariablen angegeben. Stößt der Computer jedoch auf ein NEXT, ohne im FOR-NEXT-Modus zu sein, gibt er den NEXT-WITHOUT FOR-Error

aus. Meist liegt der Fehler darin, daß eine Variable verwechselt wurde. (Beispiel: FOR I=1to50: NEXT A)
Eine weitere Möglichkeit: Der Rechner wird durch einen GOTO- oder GOSUB-Befehl an eine Stelle des Programmes geschickt, an der er unvorbereitet auf ein NEXT stößt.

Errorcode des C16: 11
Text: Syntax Error
Häufigster aller Fehler. Jedesmal, wenn der Computer eine Anweisung absolut nicht versteht und keine nähere Erklärung für den Fehler finden kann, gibt er Syntax Error aus. Meist ist ein Tippfehler (GOWO, PLINT, etc.) der Grund. An zweiter Stelle steht das Benutzen von Basic-Schlüsselwörtern in Variablen. Die Anweisung: LET WORT$="Hallo" interpretiert der Computer als LET W OR T$ ="Hallo", da er OR als reservierten Befehl behandelt. Aus LET W OR T$ kann der Interpeter allerdings keinen Sinn erkennen: Deshalb der Fehler. Ein weiterer Grund ist oftmals vorheriges, sinnloses Herum-GePOKe im Betriebssystem. Dann hilft nur noch ein Hardware-RESET oder Ausschalten.

Errorcode des C16: 12
Text: Return Without Gosub Error
Ähnelt Fehlercode 10: Next Without For Error. Trifft der Interpreter auf den Befehl RETURN, ohne im GOSUB-Modus zu sein, gibt er den Error-Code 12 aus. Der Grund hierfür liegt in der GOSUB-RETURN-Anweisung: Springt der Interpreter in ein Unterprogramm, „merkt" er sich die Zeilennummer, von der er kam. Trifft er nun auf die Anweisung RETURN, nimmt er sich die „gemerkte" Zeilennummer und bearbeitet das Programm ab der folgenden Zeile weiter.
Ist allerdings keine Zeilennummer im Gedächtnis, gibt es auch keine Rücksprungzeilennummer: Also Error. Meist tritt der Fehler auf, wenn der Basic-Interpreter nach Abarbeiten des Hauptprogrammes ein folgendes Unterprogramm bearbeitet, anstatt auf eine END- oder STOP-Anweisung zu stoßen. Manchmal führt eine Verwechslung der Befehle GOSUB und GOTO zu dem Fehler. Bei GOTO gibt es kein RETURN.
Selten ist der Error eine Folge des Befehles POP. POP löscht die letzten Rücksprungszeilennummer im Gedächtnis des Rechners. Das Kurzprogramm
10 GOSUB 100
20 END
100 POP
110 RETURN
endet mit RETURN WITHOUT GOSUB ERROR in 110, da in Zeile 100 Befehl POP die gemerkte Zeilennummer 10 (siehe 10 GOSUB 100) löscht.

Errorcode des C16: 13
Text: Out Of Data Error
Eine beliebte Art, Maschinenspracheroutinen in Basic-Programme einzufügen, sind DATA-Anweisungen. Die Zahlenwerte des Unterprogrammes stehen in langen Kolonnen nach dem DATA-Befehl (Beispiel: 100 DATA 10,23,22,6) und werden durch eine FOR-NEXT-Schleife und READ ausgelesen, bevor der Interpreter sie per POKE in den Speicher schreibt (Beispiel:
10 FOR I=768to770
20 READ A: POKE I,A
30 NEXT I
90 DATA 1,2,3)
Manche Unterprogramme sind so lang, daß beim Abtippen der Kolonnen, ein Wert vergessen wird. Versucht das Programm in einem solchen Fall alle vorgesehenen Werte einzulesen, gibt es den OUT OF DATA ERROR. Denn die letzte einzulesende Zahl gibt es nicht. Damit ist auch gleich die Fehlerursache beschrieben. Beste Lösung ist, alle DATA's im eingetippten Programm auf Vollständigkeit zu prüfen. Seltenere Ursache ist, daß der Interpreter versehentlich zweimal in die READ-Schleife gerät, also ein zweites Mal versucht, alle Daten einzulesen. (Im obigen Beispiel wäre das mit der zusätzlichen Zeile:
40 GOTO 10 ermöglicht.)
Um dieses Problem zu lösen, genügt in manchen Fällen der RESTORE-Befehl vor der FOR-READ-NEXT-Schleife.

Errorcode des C16: 14
Text: Illegal Quantity Error
Der C16 gibt Fehlercode 14 beispielsweise bei den Anweisungen PRINT A$(−1) oder POKE 257,0 aus. Der Grund: Matrizen nehmen immer positive Werte inklusive 0 an, in eine Speicherstelle des Computers passen maximal die Werte 0−255. Anweisungen mit Werten jenseits aller Möglichkeitsgrenzen führen zu ILLEGAL QUANTITY ERRORS. Meist sind diese Errors auf Tippfehler zurückzuführen, wobei der Tippfehler durchaus in einer Formel versteckt liegen kann. (Beispiel:
10 A=5
20 B=80
30 C=A*B/Fehler: Ursprünglich sollte hier C=A+B stehen
40 POKE 768,C)

Errorcode des C16: 15
Text: Overflow Error
Jeder Rechner hat Bearbeitungsgrenzen. Im Commodore 16 ist die größte Zahl 1.701411833E+38 und die kleinste −1.701411833E+37. Zahlen außerhalb dieses Bereiches führen zu OVERFLOW ERROR, egal ob die Zahl in einer Variablendefinition übergeben wird oder in einer Formel entsteht. Meist ist der Fehler Folge einer falsch eingegebenen Formel oder unkontrollierter Nutzung von Variablen in Formeln.

Errorcode des C16: 16
Text: Out Of Memory Error
C16/116-Nutzer ohne Speichererweiterung schlagen sich oft mit diesem Fehler herum. Im Programm selbst ist kein Fehler. Der Grund liegt im fehlenden Speicherplatz des Systems.
Gibt man ein Programm ein und füllt dabei den gesamten freien Basic-Speicher auf, kommt es bei der nächsten Programmzeile nach RETURN-Taste zum OUT OF MEMORY ERROR. In diesem Fall muß das Programm in 2 oder mehr Teilen eingetippt und auf Diskette gespeichert werden. Ein Programmteil lädt bei Bedarf weitere Folgeteile von Diskette nach. Weitere Gründe für Fehlercode 16 sind: Mehr als 10 FOR-NEXT-Schleifen sollten zur gleichen Zeit eröffnet werden. Meist liegt der Fehler in einem unkorrekt eingegebenen GOTO- oder GOSUB-Befehl, der den Interpreter immer und immer wieder an eine FOR-Programmstelle schickt. Dieselbe Zahlenbeschränkung ist bei DEF FN-Anweisungen zu beachten. Sind Programme bereits voll eingetippt und erscheint der Fehler erst nach Start des Programmes durch RUN, überlasten sehr wahrscheinlich Variablen den Speicher. Hier hilft die Funktion Variable =FRE(0). FRE(0) löscht alle nicht weiter benötigten Variablen und schafft somit Platz für neue Definitionen. Oftmals läuft ein Programm schon seit Minuten, bis der Error erscheint. Hier gilt dasselbe: FRE(0) muß her.
Eine weitere Möglichkeit ist, von vorne herein möglichst wenige Variable zu verwenden und bei FOR-NEXT-Schleifen möglichst oft dieselbe Varia-

ble zu nutzen (I hat sich eingebürgert).

Fehlercode des C16: 17 Text: Undef'd Statement Error

Springt der Interpreter zu einer Zeile, die noch nicht eingegeben wurde, gibt es UNDEF'D STATEMENT ERROR.
Beispiel:
10 GOTO 100
20 END
führt zu UNDEF'D STATEMENT ERROR in Zeile 10.
Dasselbe gilt für die Basic-Anweisungen GOSUB Zeilennummer oder RUN Zeilennummer, falls Zeilennummer nicht vorhanden. Oft wollen Computer-Benutzer Programme starten, von denen sie gerade ein Drittel oder die Hälfte eingegeben haben. Tritt in solchen Fällen dieser Fehler auf, suchte der Interpreter eine Zeilennummer, die noch nicht eingegeben war. Bei eigenen Programmen sind meist Tippfehler entweder im GOTO/GOSUB/RUN-Befehl oder in der Zielzeilennummer die Ursache. Manchmal spielt das eigene Gedächtnis einen Streich, indem es eine Zeilennummer vorgaukelt, die nicht benutzt wird.

Errorcode des C16: 18 Text: Bad Subscript Error

Nutzen Programmierer Index-Variablen (Bsp. A(3), B(1,5)) ohne vorher alle Variablenfelder zu dimensionieren oder sind Matrizen zu klein dimensioniert, gibt der 16er diesen Fehler aus. Grundsätzlich gilt: Matrizen mit 11 Feldern (0 bis 10) müssen nicht DIMensioniert werden (DIM-Befehl-Beispiel: DIM A(20), B(10,15))
Weist man den Variablenmatrizen über 10 Werte zu, ist die entsprechende Variable vorher zu dimensionieren.
Beispiel:
10 DIM A(30): Für 30 freie Variablen
20 FOR I=1 TO 30
30 A(I)=I
40 NEXT
Nach RUN erhält jede Variable den Wert ihrer Stellung. A(3) ist 3, A(23) ist 23.
Gibt man nun im Direktmodus:
LET A(31)=31 ein, gibt der Computer den BAD SUBSCRIPT ERROR aus. Oftmals entsteht der Fehler aus Folge einer Ausweitung der Programmkapazität. Im obigen Beispiel wäre das die Ausweitung der FOR-NEXT-Schleife auf über 30 Einheiten. Oft ist in „verbesserten" Versionen der ursprüngliche DIM-Befehl noch derselbe, was in der FOR-NEXT-Schleife zu Fehlern führt. Am Besten ist, Matrizen etwas größer als notwendig zu DIMensionieren. Voraussetzung hierfür ist genügend Speicherplatz, denn sonst gibt es bald einen OUT OF MEMORY ERROR!

Errorcode des C16: 19 Text: Redim'd Array Error

Manche Programmierer versuchen, Matrizen erneut zu DIMensionieren, weil die vorherige DIMensionierung zu gering war. Aus Variablenverwaltungsgründen geht das beim C16 nicht. Er gibt Fehlercode 19 aus. Es ist deshalb zu vermeiden, gleiche Variablenfelder mehrfach zu dimensionieren. Besser ist, von Anfang an größere Dimensionen anzustreben. Nicht genutzte Variablen einer Matrix benötigen sehr wenig Speicherplatz! Soll Auf Biegen und Brechen dasselbe Variablenfeld noch einmal DIMensioniert werden, muß vorher ein CLR durchgeführt werden. CLR löscht ALLE Variablen; sowohl Matrizen als auch String- und sonstige Variablen.

Errorcode des C16: 20 Text: Division By Zero Error

Beim Versuch, irgendeine Zahl durch 0 zu teilen, entsteht dieser Fehler. Selten liegt es daran, daß direkt eine Variable oder Zahl durch 0 geteilt wird. Viel öfter versucht der Interpreter, eine Variable durch eine Zahl oder Variable zu teilen:
C=B/A.
Hat A in diesem Fall den Wert 0, ist die Rechnung, so wollen es die Mathematiker, nicht durchführbar. Um diese Fehler zu vermeiden, prüft man die Variable A vor der Division auf ihren Wert und überspringt die Division, wenn A=0 ist. Beispiel hierfür:
10 IF A=0 THEN PRINT "A war 0!":GOTO 30
20 C=B/A
30 PRINT C

Errorcode des C16: 21 Text: Illegal Direct Error

Im C16-Basic gibt es zwei Modi: Den Direkteingabemodus und den Programmmodus. Der Direkteingabemodus ist zur Programmierung und direkten Kommunikation mit dem Betriebssystem des Rechners gedacht. Er benötigt keine Zeilennummer. Im Gegensatz zum Programmmodus, den der Computer dadurch erkennt, daß der Anwender vor der Befehlsfolge eine Zahl (Zeilennummer) schreibt. Einige Befehle funktionieren nur im Direkt- andere nur im Programmmodus. Gibt man beispielsweise
INPUT "Eingabe"; A$
im Direktmodus, also ohne Zeilennummer ein, erscheint der ILLEGAL DIRECT ERROR. Dasselbe gilt für GET. Umgekehrt sind beispielsweise die Befehle DLOAD und DSAVE im Programmmodus nur in abgeänderter Form oder mit weiteren Funktionen versehen zu benutzen.

Errorcode des C16: 22 Text: Type Mismatch Error

Der C16-Interpreter unterscheidet zwischen 3 Variablenarten:
Stringvariablen wie A$ oder B$(3), Integervariablen wie A% oder B%(3), Zahlenvariablen wie A oder B(3).
In Stringvariablen stehen Buchstaben und/oder Zahlenkombinationen wie: "0012AS", "ttgr", "0000","Hallo".
In Integervariablen stehen geradzahlige Werte wie 12,10034 oder 0. Bei dem Befehl LET A% =12.234 behält die Variable A% nur die Werte bis zur Kommastelle, im obigen Beispiel den Wert 12. Zahlenvariablen nehmen alle Werte, ob geradzahlig oder mit Komma, an. In der Verarbeitung zwischen String- und Zahlen-/Integervariablen gibt es große Unterschiede: Zwar sind Strings genauso wie Zahlen miteinander addierbar, jedoch ist es unmöglich, Buchstaben einer Zahlenvariable zuzuweisen wie A="ABER" oder BALL="HER MIT".
Die Befehle RIGHT$, LEFT$, MID$ (siehe Handbuch) sind nur in Verbindung mit String-Variablen ohne Fehler anwendbar. Um Zahlenvariablen in Stringvariablen umzuwandeln, gibt es den STR$-Befehl. Umgekehrt erhalten Anwender durch den VAL-Befehl alle Zahlen eines Strings bis zum ersten Buchstaben herausgefiltert. Meist ist Unachtsamkeit beziehungsweise Verwechslung zwischen einzelnen Variablentypen Ursache des Fehlers. Wichtigster Unterschied und weitverbreitetster Fehler sind Hochkommazeichen ("), die entweder vergessen oder unsinnig hinzugefügt wurden.

Errorcode des C16: 23 Text: String Too Long Error

Durch den 8 Bit-Prozessor bedingt, ist die Stringverwaltung des C16 auf bis zu maximal 255 Zeichen pro Stringvariable ausgelegt. Versucht man, einem String mehr als 255 Zeichen zuzuordnen, egal ob in einer LET-Anweisung oder durch Addition verschiedener Strings miteinander, gibt der C16 STRING TOO LONG ERROR aus. Derselbe Fehler tritt auf, wenn Programmierer versuchen, eine

Basic-Zeile mit mehr als 80 Zeichen (beim C16 entsprechend 2 Bildschirmzeilen) einzugeben. Sehr oft stehen Listings-Abtipper vor diesem Problem: Da stehen tatsächlich mehr als 80 Zeichen in einer Zeile. Der Grund: Im Commodore-Basic sind Befehle wie INPUT oder READ auch in Kurzform (Klein i groß N "iN" für INPUT) einzugeben. Nach LIST zeigt der Computer die Befehle jedoch ausgeschrieben. Bei INPUT sind damit 3 Zeichen zu sparen. Aus diesem Grund sind bei überlangen Zeilen in Programmlistings stets alle Basic-Befehle in Kurzform einzugeben. Eine Tabelle mit Kurzschreibweisen ist dem Handbuch angegliedert.
Letzte Ursache für diesen Fehler: Gibt der Anwender bei Abfragen durch INPUT mehr als 80 Zeichen ein, erscheint ebenfalls Fehlercode 23.

Errorcode des C16: 24
Text: File Data Error

Dieser Fehler entsteht, wenn falsch abgespeicherte Daten von Kassette wieder in den Computer geladen werden sollen. Sind Daten beispielsweise nicht korrekt durch CHR$(13) entsprechend RETURN-Taste getrennt, kommt es zum Überladen des Stringspeichers nach 255 Zeichen. Dasselbe gilt beim Einlesen von Daten von Diskettendateien.

NEWS RUND UM COMMODORE

NOCH SCHLAUERER FUCHS?

Scanntronik, Zorneding, bringt bis Herbst eine neue, verbesserte Version des Grafik- und Druckertolls „Printfox" auf den Markt. Er soll das Modul-Version (für den Expansionsport des C 64) mit Installations-Diskette verkauft werden. Da uns eine Testversion zur Verfügung gestellt werden soll, werden wir darüber zu gegebener Zeit berichten.

BILDSCHIRM-TEXT FÜR DEN C 64 UND 128 PC

Commodore Deutschland will anläßlich der Internationalen Funkausstellung Berlin das bereits angekündigte BTX-Modul für den C 64 und 128 in Zusammenarbeit mit der Fa. Siemens vorstellen. Siemens ist für das Innenleben dieses Moduls verantwortlich, Commodore stellt das Exterieur. Als Erstverkaufsauflage sind 10 000 Stück geplant. Der Endpreis soll knapp unter DM 400,– liegen.

EPROM IM AUSTAUSCH

Die Fa. Rolle-Kommunikationstechnik, München, die erst vor kurzem mit ihrem universell kompatiblen Druckerinterface für Furore sorgte, hat dies bereits wieder verbessert: Im Austausch wird jetzt ein neues Eprom angeboten, das die Qualität von 8-Nadelgrafik-Ausdrucken noch verbessern soll, da der Nadelabstand eines 8-Nadel-Druckers sich von dem, der mit 24 Nadeln druckt, merkbar unterscheidet. (1/72 Zoll gegenüber 1/60). Außerdem werden die Abstände der Grafikdruckzeilen neu berechnet, was der Grafikdichte des Bildes zugute kommt. Das neue Eprom kann problemlos ausgetauscht werden.

SPIELE-KLASSIKER

Mitte September will das Software-Haus „Activision" vier schon ältere Spiele von Lucasfilm Games auf *einer* Diskette oder Kassette auf den Markt bringen, da noch immer eine nennenswerte Nachfrage bestehen soll: Ballblazer, Rescue on Fractulus, The Eidolon, Koronis Rift.
Kosten soll dieser „Lucasfilm Games-4-Greatest"-Sampler in der Kassettenversion DM 34,95, auf Disk DM 49,95.

STRATEGIE-KRIEGSSPIELE

Ebenfalls von „Activision" kommen in Kürze zwei sogenannte „Strategie-Spiele" auf den Markt: High Frontier und Guadecanal. Das erste beschäftigt sich mit SDI-(Strategie Defence Initiative), der wohl meistumstrittensten Verteidigungspolitik unserer Zeit, das andere bezieht sich auf die Schlacht im Jahr 1942 um „Guadecanal-Island" zwischen Japan und USA. Obwohl wir den effektiven Wert solcher Spiele für den Benutzer in Frage stellen, werden wir nach ausführlichem Test darüber berichten.
Die bekannte amerikanische Softwarefirma Epyx (Winter Games, Summer Games), kündigte auf einer Pressekonferenz in München einige neue Produkte an. In Kürze werden die neuen Games, die sich nahtlos in die bekannte Sportspiel-Reihe von Epyx einfügen, auch in Deutschland erhältlich sein. Unsere Bildschirmfotos zeigen das wohl interessanteste Produkt „California Games". In gewohnter Bedienungsart, wie sie ja schon von Summer- und Winter Games bekannt ist, kann sich der Sport-Fan jetzt in typisch amerikanischen Sportarten messen: Surfen, Rollschuhlaufen, Skateboard, BMX-Rennen und noch weitere Disziplinen erwarten den Spieler. Epyx bliebt auch hier wieder bei den Joystick-mordenden Bewegungen zum Aktivieren der Figuren, die Zubehörindustrie kann sich also ebenfalls auf das Erscheinen von California Games freuen. Ebenfalls neu sind zwei weitere Produkte, die unter dem Namen „Street Sports" erscheinen. Zur Zeit existieren zwei Street Sports Versionen: Basketball und Baseball.



Noch eine weitere interessante Meldung kommt von Epyx: Dort wurde Mitte Mai David S. Morse zum „Chief Executive Officer", also als oberster „Boss" des Unternehmens (außer dem Präsidenten), berufen. Morse war der Gründer der damaligen Firma Amiga Computer, die ja bekanntlich 1984 von Commodore aufgekauft wurde. Es ist also zu hoffen, daß durch Morse einige neue Innovationen in Richtung Amiga-Software stattfinden (schließlich war er Mitentwickler des Amiga-Systemes).

EIN COMPUTER WAS IST DAS?

Wer alle Möglichkeiten seines Computers voll ausschöpfen möchte, bemerkt sehr schnell, daß er mit Basic nicht sehr weit kommt. Weiterführende Aufsätze in Computerzeitschriften setzen oft bereits soviel an Grundlagenwissen voraus, daß man nur frustriert aufgeben kann. In unserer neuen Serie schließen wir nun diese Lücke und zeigen, wie ein Computer funktioniert. Sie erfahren Wissenswertes über seine Bausteine, und lernen, ihn mittels Monitor, Assembler, Compiler und Interpreter zu programmieren. Am Ende des Kurses steht Ihnen ein Programmiersystem zur Verfügung, mit welchem Sie jeden Computer, unabhängig davon, welche CPU er besitzt, oder ob es nun ein Homecomputer oder PC ist, mit Komfort programmieren können.

Wie ein Computer aussieht, dürfte fast jedem, jedenfalls aber unseren Lesern, bekannt sein. Darüber, was ein Computer eigentlich seinem Wesen nach ist, herrschen viele, oft falsche, Vorstellungen. Manche meinen, ein Computer sei gescheit oder klug, weil er so vieles kann, manche betrachten ihn auch fast wie ein intelligentes Wesen.
Wenn ein Computer aber intelligent ist, dann jedenfalls nur in dem Sinne, wie man manche Bücher auch als schlau bezeichnet. Nicht das Papier des Buches ist schlau, oder die Druckerschwärze, sondern höchstens der Autor, der mit Einfallsreichtum und Ausdauer seine Ideen in lesbare Form gebracht hat. Die Information in einem Buch ist dafür gedacht, vom Auge eines Menschen wahrgenommen zu werden. Die Datentechnik tut sich dagegen mit Daten, die auf Magnetträger gespeichert sind, wesentlich leichter.



DER DATENRECORDER ALS STEUERGERÄT

Normalerweise werden die Signale eines Kassettenrecorders an einen Lautsprecher gesandt, der sie dann als Sprache oder auch als Musik für das menschliche Ohr hörbar macht. Jedoch sind auch andere Verwendungsarten denkbar. Als Datasette für einen Computer ist der Recorder vielen bekannt. Wir könnten ihn auch als Massenspeicher für eine elektronische Schreibmaschine benützen, oder Text auf einen Bildschirm ausgeben. Wir könnten ein sogenanntes automatisches Klavier mit ihm bedienen, oder die Bewegungen eines Greifarmes mit ihm steuern.

WAS MACHT DEN COMPUTER ZUM COMPUTER?

Ein Kassettenrecorder oder auch ein Filmprojektor haben etwas gemeinsam. Sie spulen ihre Aufzeichnungen ab, und zwar ganz genauso, wie diese aufgenommen wurden. Wenn wir uns nun vorstellen, daß ein Greifarm vielleicht nur ein paar Bewegungsmuster benötigt, um verschiedene Aufgaben zu erledigen, daß aber so ein Bewegungsmuster oft aus einer Unzahl kleinster Einzelbewegungen zusammengesetzt ist, die alle gesteuert werden müssen, so wird uns vielleicht bewußt, was ein Computer einem Datenabspielgerät voraus hat, nämlich die Selbststeuerung.

DIE ERFORDERLICHE ERGÄNZUNG

Auf einem Recorder, der sich selbst steuern kann, braucht nun nicht jedesmal eine ganze Steuersequenz für ein bestimmtes Bewegungsmuster abgespeichert sein, sondern lediglich die Befehlsfolge, die notwendig ist, um das Band auf die Stelle zu spulen, wo sich die entsprechende Sequenz befindet. Allerdings gibt es eine Schwierigkeit. In dem Augenblick nämlich, wenn das Band beginnt, sich zurückzuspulen, wird das Steuerprogramm für den Rückspulvorgang unterbrochen und das Unternehmen ist mißlungen. Abhilfe schüfe ein zweiter Recorder, der dem ersten diese Aufgabe abnähme.

MACRO- UND MICROPROGRAMM

Natürlich ist es etwas aufwendig, einen Recorder zu steuern. Mit einer Musikbox braucht man nur zwei Tasten zu drükken und schon ist die richtige Platte angewählt. Anstelle eines zweiten Recorders stellen wir uns daher besser eine Musikbox mit vielen Platten vor. Ein Code auf dem Band reicht nun aus, eine ganze Codesequenz anzusteuern. Ein Macrocode, sozusagen, generiert eine Folge von Microcodes. Die Programme auf den Platten nennen wir daher „Microprogramme“ und das Programm auf dem Band „Macroprogramm“.

DER BEFEHLSZYKLUS

Haben wir bei einem Datenabspielgerät einfach das Band abspielen lassen können, so hat nun eine gesonderte Regelung zu erfolgen. Macroprogramm und Microprogramme müssen gegenseitig Rücksicht nehmen. Erst wenn das gewählte Microprogramm abgearbeitet ist, darf mit dem Macroprogramm weitergefahren werden. Die Steuerung dieses Vorganges übernimmt nun ein Microprogramm, das folgendes tut:

1 Code lesen
2 Code in Befehlsregister ablegen
3 Microroutinenaufruf
4 Weiter mit 1

Drei Microprogramme erlauben die Bandsteuerung. Der Befehl "JMP" (Jump), gefolgt vom gewünschten Zählerstand, verstellt das Band. Vorgänge, die stets gleich bleiben, sich also ständig in derselben Weise wiederholen, lassen sich automatisieren. Mit dem JMP-Befehl lassen sich prima Wiederholungen realisieren:

```
:BEGIN  .........
        .........
        .........
        .........
        .........
        JMP BEGIN
```

Der Befehl "CALL" verstellt ebenfalls das Band. Zusätzlich merkt er sich aber vorher den alten Zählerstand. Mit einem weiteren Befehl "RET" kann nach Ausführung einer Routine an anderer Stelle wieder die ursprüngliche Bandposition hergestellt werden:

```
:BEGIN  CALL GETIN
        CALL BSOUT
        JMP BEGIN

:GETIN  ............
        ............
        ............
        RET

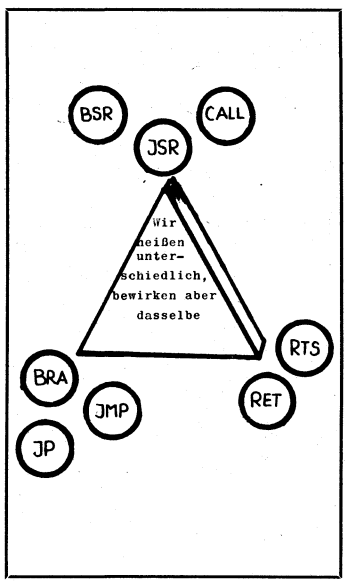
:BSOUT  ............
        ............
        ............
        RET
```

Wenn die hier angedeutete Routine GETIN dafür sorgt, daß ein Zeichen von der Tastatur eingelesen wird, und wenn wir weiterhin annehmen, daß BSOUT die Bildschirmausgabe bewirkt, so sorgt unser Programm nun dafür, daß jeder Tastendruck auf den Bildschirm geschrieben wird.

ASSEMBLER-MNEMONICS UND LABELS

In unseren Programmbeispielen verwendeten wir keine Computerdaten, sondern Namen, denn Zahlen sind nichtssagend, besonders, wenn unterschiedliche Computer unterschiedliche Zahlen für dieselbe Routine benützen. Ein Begriff dagegen ist weit griffiger. Assembler-Mnemonics sind abgekürzte Begriffe, mit denen üblicherweise Maschinenbefehle bezeichnet werden. Leider gibt es kleine Abweichungen in der Begriffsgebung bei unterschiedlicher Assembler- und Bausteinentwicklern, wie Sie es folgender Gegenüberstellung entnehmen können:

CPUs	68000	8086	Z80	6502 /	BASIC
	BRA	JMP	JP	JMP	GOTO
	BSR	CALL	CALL	JSR	GOSUB
	RTS	RET	RET	RTS	RETURN



Basic ist kein Microprozessor, sondern eine Programmiersprache, jedoch auch hier sind die besprochenen Routinen vertreten.
Ein Assembler ist ein Programm, das ein als Text in Assembler-Mnemonics geschriebenes Programm in die Daten übersetzt, die der Computer versteht. Im Assembler Programm brauchen nicht die Adressen (Zählerstände) angegeben werden, auf welche nun der CALL oder JMP springen soll, das wäre manchmal eine mühsame Rechnerei. Wenn man nachträglich ein paar Zeilen einfügen möchte, verschöben sich alle nachfolgenden Adressen und man müßte diese alle neu berechnen.
Beim Assembler braucht man nur Marken (Labels), zumeist Namen am linken Rand und mit Doppelpunkt zusätzlich besonders kenntlich gemacht. Um den Rest brauchen wir uns dann nicht mehr zu kümmern, denn das Berechnen der Adressen besorgt der Assembler.
Mit den besprochenen Routinen bringt ein Computer erstaunliche Kunststücke fertig. Anstatt das Band zu verstellen, könnten wir uns unsere Routine auch anders definieren, z.B. es sollen die Bänder gewechselt werden.

WO BLEIBT NUN DIE SOGENANNTE INTELLIGENZ?

Von einer Intelligenz des Computers war bislang noch nichts bemerkbar. Worin äußert sich Intelligenz? Doch darin, daß man sich richtig in verschiedenen Situationen verhalten kann, wenn man z.B. die richtigen Antworten weiß, so mögen manche meinen.
Ein Computer nun hat die Möglichkeit, auf unterschiedliche Situationen unterschiedlich zu reagieren. Ob allerdings richtig, das hängt nicht von einer Intelligenz des Computers, sondern von der des Programmierers ab.

BEDINGTE SPRÜNGE

Mehrere Microroutinen, die "BRANCH"-Routinen, ermöglichen den bedingten Sprung. Ob ein Sprung stattfindet, hängt nun vom Zustand eines gewissen Registers, dem sogenannten Statusregister, ab. Solch ein Statusregister beinhaltet verschiedene "Flags", die sogenannten Statusflags. Wir stellen uns am besten das Statusregister als Schalterleiste vor, wobei die Flags die Schalter darstellen. Die Schalter können nun auf "Ein" oder auf "Aus", bzw. auf 1 oder 0 stehen. Angenommen, es soll ein Sprung erfolgen, wenn Flag0 den Wert 1 hat, so erreichen wir es mit:
BF0 : Branch if Flag0=1
Sollte Flag0 den Wert 0 haben, so wird der Sprung nicht ausgeführt, sondern einfach mit dem darauffolgenden Befehl fortgefahren.
Folgender Befehl würde das Gegenteil bewirken, nämlich einen Sprung, wenn Flag0=0 und Unterlassung, wenn Flag0=1:
BNF0: Branch if not Flag0=1

Hierzu ein Beispiel:
Die Macroroutine "MESSEN" mißt die Raumtemperatur. Als Ergebnis dieser Messung werden Flag0 auf 1 gesetzt, wenn es zu kühl ist. Wenn nicht, so habe Flag0 den Wert 0. Wenn es zu kühl ist, soll geheizt werden, wenn nicht, dann ist die Heizung auf "Aus" zu setzen. Die Routine "HEIZEN" besorgt das Heizen, die Routine "KUEHLEN" das Gegenteil. Unser Programm, das nun heizen oder kühlen soll, sieht nun folgendermaßen aus:

```
:REGEL0 CALL MESSEN
        BF0    REGEL1
        CALL KUEHLEN
        JMP    REGEL0
:REGEL1 CALL HEIZEN
        JMP    REGEL0
```

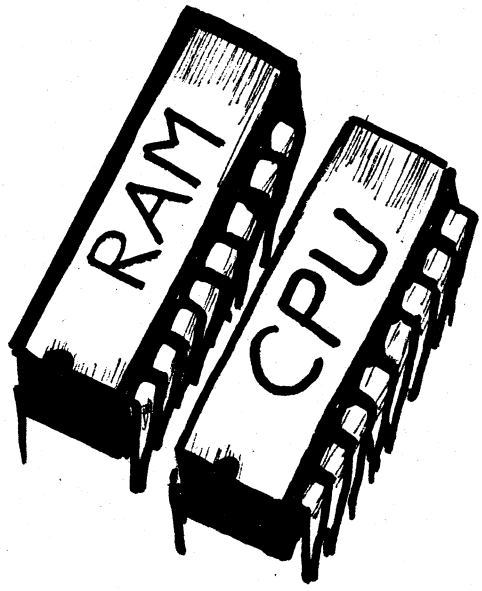
Genauso richtig könnte man auch schreiben:

```
:REGEL0 CALL MESSEN
        BNF0 REGEL1
        CALL HEIZEN
        JMP    REGEL0
:REGEL1 CALL KUEHLEN
        JMP    REGEL0
```

CHIPS STATT GRIPS

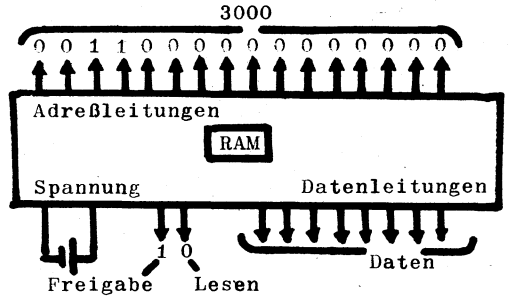
Das, was einen Computer ausmacht, hat nicht mit Grips zu tun, mit einfachen Geräten, wie z.B.

einem Kassettenrecorder und einem musikboxähnlichem Apparat wäre er bereits zu verwirklichen. Jedoch was den Computer besonders auszeichnet ist seine Schnelligkeit, mit der er Programme ausführen und Datenzugriffe vornehmen kann. Daher wollen wir unsere etwas antiquiert anmutenden Vorstellungen beiseite legen und uns mit den Bausteinen auseinandersetzen, aus denen ein Computer tatsächlich besteht, den sogenannten Chips. Uns interessiert nun nicht, wie diese im Detail hergestellt sind, sondern das Funktionsprinzip. Wir lassen nun RAM und CPU die von Kassette und Platten einnehmen.

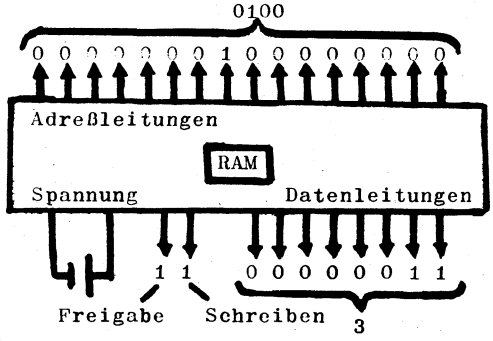


RANDOM ACCESS MEMORY

Ein Kassettenrecorder hat einige Nachteile. Er ist relativ langsam. Technisch aufwendig wäre wohl eine exakte Bandansteuerung zu realisieren, mit welcher jede beliebige Stelle exakt angesteuert werden könnte. Das Vor- und Zurückspulen nimmt erhebliche Zeit in Anspruch. Ein Band ist eben mehr dafür gedacht, der Reihe nach gelesen zu werden. Ein Speicherbaustein, der einen wahlfreien Zugriff erlaubt, ist das Optimale. Speicher mit wahlfreiem Zugriff heißt in englisch "RANDOM ACCESS MEMORY", abgekürzt auch RAM genannt. An so einem Baustein sitzen nun verschiedene Leitungen, nämlich Stromversorgung, Adreßleitungen, eine oder mehrere Datenleitungen, und wenigstens zwei Steuerleitungen, nämlich eine für die Wahl von Lesen oder Schreiben und eine für das Freigabesignal. Um nun Daten aus dem RAM zu lesen oder auch hineinzuschreiben, brauchen nur entsprechende Spannungen an den Leitungen anzuliegen und schon ist der Vorgang möglich.



So wird der Inhalt von Speicherzelle $3000 gelesen.



So wird die Zahl 3 in die Speicherzelle $0100 geschrieben.

An die Stelle unserer Musikbox tritt die CPU, die Zentraleinheit, um die sich alles dreht, auch oft als Herz des Computers bezeichnet. Da das RAM, wenn es nicht abgefragt wird, von selbst überhaupt nichts tut, die CPU aber ständig – wir erinnern uns an den ablaufenden Befehlszyklus – aktiv ist, sieht man sie als den wichtigsten Teil des Systems an. Genausogut könnte man aber auch argumentieren, daß die Aktionen der CPU ja durch die im RAM stehenden Codes gesteuert werden, daß deshalb das Programm als geistiges Element die Seele der ganzen Sache sei.
Die Anweisungen, die im RAM stehen, werden eben von der CPU ausgeführt. Da wir uns nicht in philosophische Spekulationen verlieren wollen, ob nun dem Geist oder der Materie die führende Rolle zukommt, oder wie es in der Computerterminologie heißt, der Soft- oder der Hardware, oder gar, ob nun das Huhn oder das Ei früher war, sehen wir uns die CPU etwas näher an. Rein äußerlich gleicht sie dem RAM. Auch die CPU hat Adreßleitungen, Datenleitungen, Steuerleitungen und eine Stromversorgung. Der Unterschied zum RAM offenbart sich nicht in den sichtbaren als vielmehr in den unsichtbaren Dingen. Was die CPU haben muß, ergibt sich aus dem, welche Aufgaben sie zu erfüllen hat. Jedenfalls muß aber auch die CPU verschiedene Speicher, die in diesem Falle Register genannt werden, besitzen.
Diese Register dienen aber nicht dazu, einfach massenweise Daten irgendwo abzulegen, wie dies beim RAM der Fall war, sondern werden für spezielle Steuerungsaufgaben der CPU benötigt.

BEFEHLSZÄHLER BEFEHLSREGISTER

Da zum Lesen von Daten aus dem RAM deren Speicheradresse benötigt wird, muß die CPU sich ständig merken, wo sie den nächsten Befehl zu suchen hat.
Hierzu dient der Befehlszähler. Der Befehlscode, welcher nun gelesen werden kann, wandert in das Befehlsregister.
Der Befehlszyklus, den wir bereits besprochen haben, sieht nun so aus:

1 Befehlszählerinhalt auf Adreßbus
2 Lese/Schreibleitung auf Lesen
3 Freigabesignal senden
4 Datenbus an Befehlsregister
5 Befehlszähler erhöhen
6 Befehlsaufruf
7 weiter mit 1

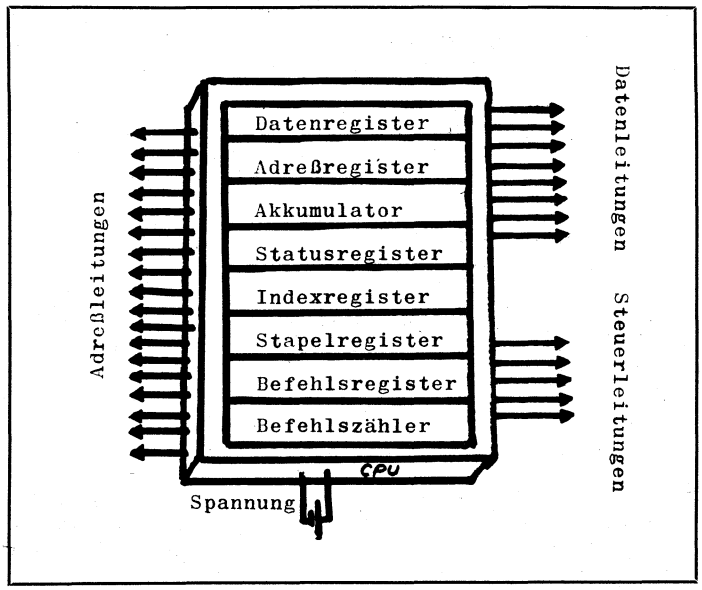
Die Daten- und Adreßleitungen nennt man Daten- bzw. Adreßbus. Der Befehlszähler wird um 1 erhöht, damit er nun auf den nächsten Befehl zeigt. Wird ein solcher Befehl abgearbeitet, so kann der Befehlszählerinhalt unter Umständen währenddessen auch weiter verändert werden, weil dem Befehlscode vielleicht weitere Parameter folgen.

SO WIRD GEJUMPT

1 Befehlszähler auf Adreßbus
2 L/S-Leitung auf Lesen
3 Freigabesignal
4 Datenbus an Daten register
5 Befehlszähler erhöhen
6 Befehlszähler auf Adreßbus
7 Datenregister in Befehlszähler H
8 L/S-Leitung auf Lesen
9 Freigabesignal
10 Datenbus auf Befehlszähler L
11 Rückkehr zum Befehlszyklus

Der Befehlszähler bekommt somit die dem Befehlscode folgende Sprungadresse übermittelt. Der anschließend folgende Befehlszyklus liest nun an der neuen im Befehlszähler vermerkten Adresse weiter.

Bei einem CALL ist die Rücksprungadresse zu vermerken. Mit einem normalen Register ist es hierbei nicht getan. Angenommen, es erfolgt der Aufruf einer Maschinenroutine durch CALL: Ein Rücksprung durch RET wäre dann ohne weiteres möglich. Was aber dann, wenn innerhalb der aufgerufenen Routine ein weiterer Routinenaufruf erfolgen würde? Die ursprüngliche Rücksprungadresse würde durch die neue überschrieben werden und



ein Rücksprung in das anfänglich aufrufende Programm könnte nicht mehr erfolgen. Die Lösung bietet ein sogenannter Kellerspeicher, auch Stapel oder Stack genannt. Meistens wird hierzu ein RAM-Bereich verwendet. Ein Register, das Stapelregister, auch Stapelzeiger genannt, beinhaltet eine Adresse, in welche der nächste zu merkende Wert geschrieben werden soll. Nachdem ein solches geschehen ist, wird das Stapelregister entweder entsprechend erhöht oder erniedrigt, je nachdem, ob man sich zu einem Stapel- oder Kellerspeicher entschlossen hat. Dort kann dann ein erneuter Wert gespeichert werden und sukzessive weitere. Erfolgt ein RET (Return), so wird der Stapel einfach wieder entsprechend abgehoben und der Rücksprungwert wieder in den Befehlszähler gebracht.

ROUTINENAUFRUF DURCH CALL

1 Stapelregister erniedrigen
2 Stapelregister auf Adreßbus
3 Befehlszähler H auf Datenregister H
4 Befehlszähler L auf Datenregister L
5 Datenregister erhöhen
6 Datenregister erhöhen
7 Datenregister H auf Datenbus
8 L/S-Leitung auf Schreiben
9 Freigabesignal
10 Stapelregister erniedrigen
11 Stapelregister auf Adreßbus
12 Datenregister L auf Datenbus
13 L/S-Leitung auf Schreiben
14 Freigabesignal
15 weiter wie bei JMP

Bei den meisten CPUs liegt ein Kellerspeicher vor, weswegen wir in unserem Beispiel den Stapelzähler erniedrigt haben.

RÜCKSPRUNG DURCH RET

1 Stapelregister auf Adreßbus
2 Lesesignal setzen
3 Freigabesignal setzen
4 Datenbus an Befehlszähler L
5 Stapelregister erhöhen
6 Stapelregister auf Adreßbus
7 Lesesignal setzen
8 Freigabesignal setzen
9 Datenbus an Befehlszähler H
10 Stapelregister erhöhen
11 Rückkehr zu Befehlszyklus

DAS STATUSREGISTER

Das Statusregister enthält mehrere Statusflags. Jedes dieser Flags erfüllt eine spezielle Aufgabe. Wir haben sie einfach einmal durchnumeriert. In Wirklichkeit hat jedes einen besonderen Namen. Dieser kann aber erst in der nächsten Folge besprochen werden, weshalb wir vorerst bei unserer Terminologie bleiben. Zu den Branchbefehlen braucht vorerst einmal nichts mehr gesagt werden. Statt dessen machen wir mit den bisherigen Elementen unserer Sprache einige Übungen, welche wir auch an unserem Rechner ausprobieren können.

BEISPIEL: EINGABESCHLEIFE

```
Assembler
    ORG      $3000
    GETIN EQU  $FFE4
    BSOUTEQU  $FFD2

:BEGIN CALL  GETIN
       CALL  BSOUT
       JMP   BEGIN
```

ORG $3000 sagt dem Assembler, daß er die Adressen so berechnen soll, daß das Programm an Adresse $3000 lauffähig ist. Diese Adresse wurde willkürlich gewählt, wir hätten auch $2000 nehmen können, oder eine krumme Zahl wie z.B. $24AF.
Die Routinen GETIN, welche eine Tastaturabfrage bewirkt, liegt bei unseren Commodore-Computern an Adresse $FFE4, die für die Zeichenausga-

be verantwortliche Routine an Adresse $FFD2. Mit der EQU-Anweisung wird dem Assembler gesagt, daß er, sofern er auf die solcherart definierte Konstante stößt, dafür den entsprechenden Wert einsetzen soll. Die Konstantendefinition durch EQU bietet den Vorteil, daß man nur die Definition am Programmanfang abzuändern braucht, wenn bei einem anderen Computersystem die bereits vorhandenen Maschinenroutinen anders im Adreßraum positioniert sind. :BEGIN ist eine Labeldefinition. Da vorher noch kein einziger zu übersetzender Code stand, weist der Assembler dieser Marke den Wert $3000, welches unser Programm ist, zu.

MASCHINENMONITOR C16/C128

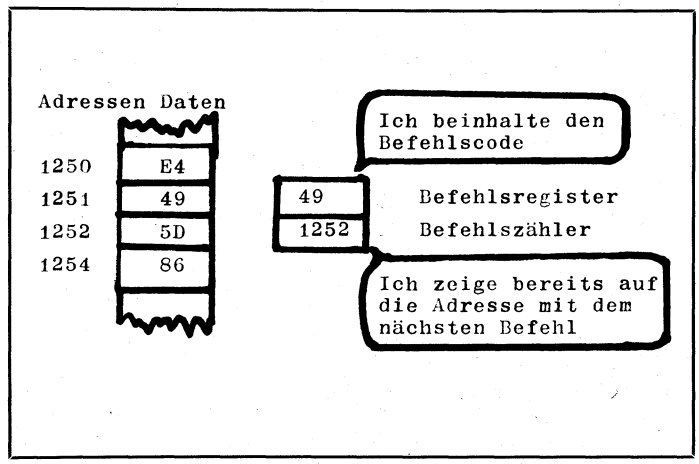
Ein Maschinenmonitor ist nicht so komfortabel. Die Adressen, die einmal geschrieben sind, sind geschrieben. Soll das Programm an eine andere Stelle kommen, so sind alle Adressen dementsprechend umzuschreiben. In unserem Falle wäre dies allerdings nur die JMP-Adresse. Lägen die Systemroutinen an anderer Stelle, so müßte man sie überall, wo ein solcher Aufruf erfolgt, abändern. Bei einem längeren Programm kann dies mit einigem Aufwand verbunden sein. Nun das Monitorprogramm:

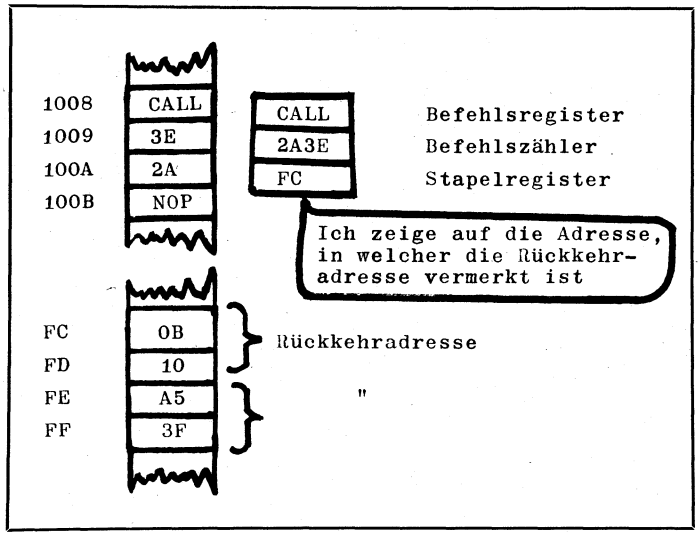
```
. 3000 JSR  $FFE4
. 3003 JSR  $FFD2
. 3006 JMP  $3000
```

Links sieht man die Speicheradressen, rechts die richtigen Assemblermnemonics des 6502-Prozessors, gefolgt von den Adreßangaben. Auf die Hexzahlendarstellung haben wir verzichtet. Wir holen das aber schnell nach, wobei wir gleich die Dezimalzahlendarstellung beifügen.

```
. 3000 20 E4 FF /
       032 228 255
. 3003 32 D2 FF /
       032 210 255
. 3006 4C 00 30 /
       076 000 048
```

Die Hexzahlendarstellung befindet sich normalerweise bei unserem Maschinensprachmonitor zwischen der Adreßangabe und den Mnemonics und wird vom Monitor selbsttätig erzeugt.
Ging beim Monitor gegenüber dem Assembler





schon einiges an Lesbarkeit verloren, da man nicht vielsagende Begriffe vor sich hat, sondern sich Adressen merken muß, so tut man sich bei der reinen Zahlendarstellung besonders schwer, da man nun auch die Operationscodes im Kopfe haben muß. Ganz schlimm wird es aber, wenn die Daten als Datazeilen in Basicprogrammen vorliegen, da sich hier gar nicht mehr unterscheiden läßt, was nun Operationscode und was nun Daten oder Sprungadressen sind.

```
10 for i=1288 to 12296
20 reada:pokei,a:next
30 sys 12288
40 data 32,228,255,32,
   210,255,76,0,48
```

Dieses Basicprogramm lädt und startet nun das gewünschte Maschinenprogramm. Wenn Sie einen C16/116/Plus4 oder einen C64 oder 128 besitzen, so sollten Sie es ruhig einmal ausprobieren.

BEISPIEL: VERTEILER

In einem Programmzyklus erfolgt eine Messung. Als Ergebnis derselben, werden dem Computer 4 verschiedene Zustände mittels Flag0 und Flag1 signalisiert. Der Computer hat daraufhin eine von 4 verschiedenen Routinen auszuführen, wie in nachfolgender Tabelle dargestellt.

Flag1	Flag0	Routine
0	0	ROUTINE0
0	1	ROUTINE1
1	0	ROUTINE2
1	1	ROUTINE3

Nach der Ausführung der entsprechenden Routine soll wieder die nächste Messung erfolgen usw.

```
:BEGIN   CALL   MESSUNG
         BF1    VERT10
         BF0    VERT01
         CALL   ROUTINE0
         JMP    BEGIN
:VERT01  CALL   ROUTINE1
         JMP    BEGIN
:VERT10  BF0    VERT11
         CALL   ROUTINE2
         JMP    BEGIN
:VERT11  CALL   ROUTINE3
         JMP    BEGIN

:BEGIN   CALL   MESSUNG
         CALL   VERTEIL
         JMP    BEGIN
```

Übersichtlicher und kürzer, sowohl datenmäßig als auch zeitlich, ließ sich das Programm folgendermaßen schreiben:

```
:VERTEIL BF1    VERT10
         BF0    VERT01
         JMP    ROUTINE1
:VERT01  JMP    ROUTINE2
:VERT10  BF0    VERT11
         JMP    ROUTINE3
         JMP    ROUTINE4
```

Da in der VERTEIL-Routine keine neue Rücksprungadresse auf den Stapel kommt, wird nach Ausführung der gewünschten Routine gleich wieder in die anfängliche Programmschleife zurückgesprungen.

AUSBLICK

Wir sind nun am Ende der ersten Folge angelangt und haben das Prinzip des Computers verstanden. In der nächsten Folge besprechen wir den ganzen Befehlssatz eines Microcomputers und haben so den Grundstein zur erfolgreichen Maschinenprogrammierung.

Alfons Mittelmeyer

BÖRSE

C128/64! SPRACH-LERN-PROG. GESUCHT!
1.) Italienisch u. Spanisch f. Anfänger u. Fortgeschrittene: Vokabeln u. Grammatik. 2.) Schul-Prog. Englisch, 3.–5. Lernjahr. Gute u. günst. Angebote an: A. Kilches-Wienstein, Elisabethplatz 2, 8000 München 40

Verkaufe C16/64KB, Datasette, Zubehör f. 120 DM VB. Außerdem zugehörige Software in rauhen Mengen. Preis VB. Auch einzeln erhältlich bei Joachim Oswald, Bäumlesweg 19, 7039 Weil 1, Tel. 07157/63457

Suche f. C64: 4 gewinnt; Squash; Donald Duck's Playground; Frankie Goes to Hollywood; Renaissance u. Computer Cluedo. Bitte schreibt an: Urte Gerhardt, Quellenweg 20, 5330 Königswinter 41

Suche dBase II (auch ohne Beschr.) f. C128 im Tausch gg. andere CP/M-Software f. C128. Tel. 0541/21561

** C128 **** C128 **
Suche Kontakt zu anderen C128-Usern im Raum HSK/SO. Suche Programme nur C128 od. CP/M-Modus. Liste an Haiko Scholl, Weidenstr. 15, 5778 Meschede

---

Ich habe C16/64K m. Floppy 1551 u. ca. 130 Games (Mercenary, Winter Olympiade ...) Will umsteigen auf 128 D. Wer tauscht mit mir? Angebote an Bernhard Bierek, Am Unteren Weinberg 5, 8403 Oberndorf

---

Top-Games zu verkaufen. Über 70! Für C16 u. C16/64K (nur Disk.). Liste gg. 0,90 DM. Suche Hardcopy (HiRes) f. MCS-801, evtl. Tausch. Frank Bauer, Seeor 2, 8261 Polling, Tel. 08631/14508

Top-Games f. C16/Plus4. Verk. Orig.-Spiele-Kass, spottbillig! Z.B. Kickstart, Berks u.v.m. Weiterhin div. Zeitschr. ab 01/85. Liste gg. 80 Pf. bei Manfred Knippel, Bergstr. 2, 8751 Hausen

---

Lotto-Ass-C128/80 Z. Schnelle Erstellung u. Auswertung v. 63 Systemen m. Gewinngarantien einschl. aller VEW-Systeme. Disk. 30 DM. Alle Lotto-Zahlen ab 1955 f. eigene Ausw. Disk. m. Beispiel 10 DM.Peter Hielscher, Am Wall 22, 4401 Saerbeck

---

C16/116/Plus 4: 20 Spiele 10 DM, Tape u. Disk. z.B. Kampf um Rom, Golf, Tennis, Submarine, Cowboy, Tank Wars, Cave u.v.a. od. 50 Pf. f. eines. Schickt einen 10-DM-Schein u. 80 Pf. Rückporto an Uwe Heyer, Hellweg 27, 43 Essen 14

Wer repariert meinen C128 günstig? Im Großraum Hamburg od. Marburg. Achim Kramer, Unnastr. 9–15, 2000 HH 20, Tel. 040/497413 od. Steffenberg, Tel. 06464/7047. PS. Wohne im Stud.-Wohnheim, deshalb öfter versuchen (Zi. 102)

Verk. C16 m. 64K, Datasette, Turbo3-Joystick, Adapter, 44 Orig.-Spiele u. Grafikbuch C16. Ca. 500 DM. Tel. 0911/361182

*** COMMODORE 128 USER ***
Sucht zuverlässige(n) Partner (in) zwecks Austausch v. Programmen od. Ideen, egal ob Basic od. Assembler. Markus J. Reiser, Am Bahndamm 5, FRG-7890 Tiengen/Hochrhein

---

Suche Physik-Programm: Simulation eines Zusammenstoßes zwischen zwei ungleichen Kugeln, elastisch, eben, nicht zentrisch. Verkaufe: HF-Video-Interface für den Anschluß eines Fernsehers an Atari ST = 120 DM. Tel. 09732/4297

Achtung!
Suche f. den C64 gute Computerumsetzung der Spiele Skat, Mühle u. Dame. Angebote an: Wolfgang Liebers, Langer Kamp 170, 2000 Norderstedt

---

Verk. C16 m. eingebauter 64K-Erw., Floppy 1551, Datasette, Joystick, ca. 50 Disk. m. Spielen. Preis: gg. Gebot (VS). Hubert Mühlbacher 08681/772

Hallo, C16-Plus4-Besitzer: Verk. 7 Spielemappen u. 3 Seiten voll Spielepokes (mehrere Leben usw.) f. nur 10,–. 10-DM-Schein an: Jens Schröder, Osterholzer Möhlendamm 55, 2800 Bremen 44

Suche Tauschpartner f. C16/116/+4, Spiele u. Anwender. Ralf Hofschröer, Ringstr. 3, 4450 Lingen. Ich beantworte alle Briefe. Bitte nur Tape. Schickt Eure Listen. Auch 64K-Spiele erwünscht. Ich warte auf Briefe.

Hallo C16/116/Plus4-User! Suche Drucker f. Plus4. Zahle 20 DM (mehr ist nicht drin) u. Porto-Geb. Schreibt an: Michael Hennemann, Schulstr. 41, 2300 Klausdorf/Schw. Bitte keine defekten Geräte! Danke!

C64**The Happy Hackers**
C128. Wir bieten Euch: über 280 Prog., Clubzeitung, große Gewinnmögl., verbilligte Angebote, über 500 Pokes u. noch vieles mehr!! Infos bei der Clubadresse: Kasselerstr. 39, 3501 Fuldabrück 1

Verkaufe Programme f. C64, C128 u. Amiga sehr preisgünstig. Liste f. 80-Pf.-Briefm. anfordern bei: Heinz Prangenberg, Carlo-Schmid-Str. 11a, 534 Bad Hennef 6

Kaufe das Modul Magic Formel u. Abdeckhaube f. Floppy 1571. Bitte sendet od. ruft an: Cliff Reinicke, Dorststr. 22, 3171 Ribbesbüttel, Tel. 05374/4900. Bin zw. 12 u. 20 Uhr erreichbar.

Tausche alle VC-20-Prog. v. GV-16K. Besonders suche ich 16K-Prog. Frank Thiel, Hauptstr. 54, 5231 Mudenbach, Tel. 02688/633

Computer Club Hup & Co.: c/o Thorsten Hup, Postfach 1274, Bramstr. 21, 4690 Herne 1. Computer: C16 C64 C128. Clubbeitrag pro Monat: 5,– DM. Clubzeitung, Softw.-Bibliothek, Einkaufsprozente u. vieles mehr. Kostenloser Katalog ist gg. 1,10 DM Rückporto erhältl.

Alle Grafiken aus Printmaster, Übersicht u. Erklärung. Info f. 80-Pf.-Briefm. Suche Astrologie- u. Bio-Rhythmus-Prog., biete Software im Tausch, Tips u.Tricks, Super-Pokes. Info gg. Rückporto Karl-Heinz Münter, Hochheimstr. 13, 5162 Niederzier 2, Tel. 02428/2933

Suche Plus4-User, die nicht nur spielen wollen, zw. Kooperation * Suche Floppy 1551,ca. 200,– DM * Biete C116, 1531 Citizen-RL-Drucker u. Lernkass. f. 250,– DM. E. Oestreich, Am Rain 11, 6423 Wartenberg 1, Tel. 06641/7372 zw. 15 u. 17 Uhr!

Suche Steckmodule, Spiele u. 8KRAM-Speichererw. f. VC-20. Angebote bitte an P. Reinhard, Dorfstr. 37, CH-8542 Wiesendangen. PS. Zahle guten Preis!

Suche Empire Input Nr. 8/86. Verk. Profi-Painter, Laser, Basic, Easy-Finance III, alles orig. Suche Monopoly u. andere Handels- u. Strategie-Simulationen. Alles f. C64, versteht sich. Karl-Heinz Münter, Hochheimstr. 13, 5162 Niederzier 2

Wir suchen Tauschpartner f. C64 (nur Tape). Listen an: E. Peters, Voßstr. 13 od.: C. Hennings, Paulstr. 16 in 2427 Malente

---

Suche Tauschpartner f. C16/Plus4 nur auf Disk. Listen an Harald Schneider, Valenciennerstr. 259, 5160 Düren. PS. Kaufe auch, wenn kein Tausch möglich ist. Suche auch Prog. aus Commodore Welt.

Verkaufe für C16/Plus4 Centr.-Karte u. Betriebssystem zum Anschl. der CBM 600/700 Tastatur. Brenne auch Eproms. Verkaufe od. tausche Spiele. Suche defekte C16/Plus4 zu günstigen Preisen. Tel. 040/7125694

Toto-Ass-C128/80Z. Schnelle Erstellung u. Auswertung von mehr als 60 Systemen m. Gewinngarantien einschl. aller Toto-VEW-Systeme. Disk. 30,– DM. P. Hielscher, Am Wall 22, 4401 Saerbeck

---

* Der Super-Commodore-Club *
Wer bietet Euch mehr als 220 Prog., Clubzeitung, monatl. Preisrätsel, Sammelbestell., über 350 Pokes, Preisvergleiche zw. 150 Firmen? Adr.: Boris Bouchon, Lärchenweg 5, 3501 Bergshausen

---

Wer digitalisiert mir Bilder, möglichst Farbe, s/w geht auch, f. C64, abgelegt auf Koala- od. Scanner-Format. Zahle pro Disk. 20,– DM. Bitte melden bei Marcus Ahrens, Am Quälberg 6a, 3170 Gifhorn

---

Suche Tauschpartner für C16/+4-Software!! Auch auf Disk. Meldet Euch bei Sven Kressner, Tel. 02378/4465 od. Bredde 6, 5758 Fröndeberg-Ardey

---

C16, C64, C128, Plus4, Amiga, PC. AN ALLE! Werdet Mitglied im DEHOCA, dem größten Userclub in Deutschland, nach ADAC-Vorbild. Wir bieten Rabatte, Clubzeitung und, und, und. Info bei: DEHOCA, Postf. 1430, 3062 Bückeburg

---

BÖRSE

C16/Plus4-DFÜ! C16/Plus4-DFÜ! Ich möchte mit dem Plus 4 in die DFÜ einsteigen! Wer kann mir weiterhelfen? Brauche Infos über Hardware, Software, Modems, Akustikkoppler, Mailboxen. Hans-Peter Lehmann, Guttenbrunnweg 1, 7950 Biberach, Tel. 07351/76503

Suche Epson-kompatib. Drucker od. Okimate 20, GP700A/VC. Alle C64-anschlußfertig, suche Printfox, Character-Fox u. Basar. Angebote an Gottfried Roeffen, Eduard-Heinkes-Pl. 4, 4154 Tönisvorst 2

Tausche Computerzeitschriften im Wert v. 500,– DM gg. MPS 801 od. ähnlichen (Alter egal): 64er, Happy, RUN, Chip, Data Welt, CW, 68000er, CP, teils Sonderhefte). Wer Interesse hat, Info bei Stefan Bels, 0581/74793

Wer hat dt. Handbuch f. Commodore Drucker VIC 1525/GP 100VC Seikosha. Möchte es, gg. Bezahlung, kopieren. Rückgabe garantiert. Klaus Funk, Mörikestr. 1, 7952 Oggelshausen

---

Suche Tauschpartner f. C64. Spiele u. Programme usw. Enrico Quellen, Alte Str. 13, 2743 Selsingen

---

Tausche 400 C16-Programme!! Gg. alles, was mit dem VC20 zusammenhängt. Kaufe auch preiswertes Zubehör. Suche besonders def. VC20 od. Plus4. Nicht zögern, sofort los! Bitte wenden an: A. Balgar, Fortkamp 7, 4250 Bottrop

---

Suche Hardware aller Art; suche Software aller Art, egal ob defekt od. wie alt. Bin Bastler u. Schüler, drum sollte es billig sein, toll wäre es geschenkt. Schreibt an: Lutz Martschin, Reherweg 5a, 3258 Aerzen

---

Wer schenkt od. verkauft armem Schüler HF-Modulator f. VC20, Netzteil f. VC20 u. Floppy 1541. Mögl. billig od. umsonst. Tel. 05361/61788, Danny Ringel, Schulenburgallee 11, 3180 Wolfsburg 1

Verk. gut erhaltene Computer-Zeitschr. z.B. Commodore Welt, Compute mit, ct, Computronic u.v.a.! Ferner SONDERHEFTE! Frei. ab 19 Uhr. Till Franzmann, Tel. 06132/8168

Plotter zu verkaufen. Panasonic KX W 08 GK RS 232C über eingebaute Tastatur auch als Schreibmasch. m. Grafik-Funktionen wie neu! Mit Verp. u. Reservestiften f. 320,– DM unter 0202/640774. Neupreis? Na ja!

ACHTUNG! Suche Tauschpartner f. +4. Suche außerdem folgende Prog.: PASCAL & BASIC-COMPILER. Holger Scherer, Jungstr. 25, 6550 Bad Kreuznach

---

C16/116/Plus4Anwender: Der Verein speziell für uns! Mit Zeitschrift u. Software-Bibliothek. Info beim hrc e.V., Bauerland 15, 48 Bielefeld 1

Verk. Input 64 v. 3/85 bis 1/87, 21 Disk. u. 11 Kass. m. Begleitheften mögl. kompl. gg. Gebot. The Pan, Orig.-Disk. 25,–. Suche zuverl. Tauschpartner f. C64, C16, Plus4. J. Trettin, Wittinger Str. 159 A, 3100 Celle

---

COMMODORE USER CLUB GERMANY. Gegründet: 1.7.87. Für wen: Alle Commodores. Beitrag: 10,–/Monat. Leistung: Info, Rabatte, Software, Tips u. Tricks. C.U.C.G., c/o Lutz Martschin, Reherweg 5a, 3258 Aerzen

---

Verkaufe 400 Prog. f. C16 nur 30,– DM (Disk.). Verkaufe viele Prog. f. VC20 nur 15,– (Disk.). Nicht lange zögern, sondern sofort schreiben!!! SW1520. Bitte an: A. Balgar, Fortkamp 7, 4250 Bottrop

---

Verk. Computer-Zeitschr., alle neuwertig. 86 u. 87 Ausgaben, ca. 150 St. 10 St. f. 20,– plus 3,– Porto. Liste gg. 1,30 DM Rückporto. Evtl. auch im Tausch gg. Feuerwehrfahrzeuge im Maßstab 1:87 v. Preiser, Herpa usw. Joachim Reichard, Laubacher Str. 3a, 4020 Mettmann

---

Gratisinfo für C64/C128-User erhältlich bei Fr. Neuper, 8473 Pfreimd, Postfach 72

Suche VC20 Programme bis 32KByte. Uwe Hoppe, Adolfstr. 59, 6208 Bad Schwalbach

---

C16 ACHTUNG! C16 Suche orig. Spielkassetten, zahle bis zu 15,– DM. Schickt Eure Listen an: Sascha Dieter Leidel, Waldstr. 63, 4100 Duisburg 1

C64 C128 VC20 User Achtung Lichtgriffel mit Programmen und dt. Anleitung nur DM 49, Versand gegen Scheck/Nachn. Fordern Sie unseren kostenlosen Commodorezubehörprospekt an. Firma Klaus Schißlbauer, Postfach 1171C, 8458 Sulzbach, Tel.: 09661/6592 bis 21 Uhr. Sonderaktion: Legen Sie Ihrer Prospektanforderung DM 10,– bei. Sie erhalten 10 Superprogramme auf Kass. od. Disk für C64/C128.

Wettersatelliten-Programme m. allem Zubehör zu kaufen gesucht f. C128D. Angebote an Norbert Wulf, Fettberg 16, 2300 Kiel 14

Plus4, Floppy 1551, Datasette, Joystick, Programme, Zeitschr. m. Prog., 1A-Zustand wg. Computerwechsel günst abzugeben. H. Dingelstädt, Wilhelmshavener Str. 30, 8510 Fürth, tel. 0911/7591427 ab 18.30 Uhr

---

** Zahle bis zu 10,– DM ** Für kaputte Joysticks aller Art bezahle ich je nach Neupreis u. Schaden bis zu 10,– DM. Schickt sie einfach an: Boris Bouchon, Lärchenweg 5, 3501 Fuldabrück 1. Suche auch defekte Datasetten!

Für C64-User:
Mit PROLOG 64 in die „künstl. Intelligenz" einsteigen!
PROLOG 64 kostet 131,– + 5,– Vers.
Bestellung: Brainware GmbH
Gustav-Meyer-Allee 25
1000 Berlin 65
030/4633048

Reste aus Systemwechsel: C64. Dela-Hardcopymodul 30,– DM, GCS-Crackerrom 55,– DM. Beide m. dt. Anl. Tel. 0208/436241 (Volker)

Verkaufe Copy-Prog. u. Spiele ab 1,– DM. Suche Tauschpartner f. C16/+4/64. Listen an Thorsten Esser, Schützenstr. 2, 4980 Bünde 1.

Verk. Super-Prog. f. C16/Plus4 z.B. Turbo-Tape, Turbo-Text, Einkommensteuer, Kaufmann, Adreß 16/64 KB, alle auf Tape od. Disk. f. 20,– DM bar od. Scheck. Spiele: Winter Olympics, Street Olympics, Water Grandprix, Sky 1941: f. 20,– bar bei: Jörg Hildebrandt, Schlenkenbrink 25, 4973 Vlotho-Uffeln

---

Suche Lernprogramme aus allen Bereichen (Mathe, Chemie, Erdkunde, Latein usw.) Angebote an Markus Roeffen, Eduard-Heinkes-Platz 4, 4154 Tönisvorst 2

Textomat C128 m. Handb. 50,–; Datamat C128 m. Handb. 50,–; Finanzgenie 64 m. Handbuch 30,–; G. Klein-Uebbing, Eschenweg 8, 4290 Bocholt, Tel. 02871/39072 ab 17 Uhr

---

Suche: Handbuch f. Plus4, Stromversorgung f. Plus 4 u. VC20, Gehäuse f. C16 u. VC20. Angebote an: Helmut Marksteiner, Schwand 23, 7822 Häusern, Tel. 07672/4262

Hallo, Commodore 16 Freaks! Ich verkaufe meine 64K-Steck-Erw. m. noch 1/2-jähriger Garantie f. 60,– DM. Nur hinten einstecken. Ruft an od. schreibt: C. Ostermann, Sulenbecker Weg 3a, 5882 Meinerzhagen, Tel. 02354/2467! Suche Spiele!

CBM-600-700er Serie. Wer hat f. diesen Rechner Anwendersoftw. zu Hobbypreisen zu verkaufen? Weiter suche ich 64/128er Anwendersoftw. zu kaufen od. zu tauschen. Tel. 06201/33860, Klaus Günter Derksen, Am Tannenbuckel 18, 6943 Birkenau

---

Verkaufe C16 m. 64KB, Handb., Datasette, Turbo-Tape, Turbo-Base, Basic-Kurs, Joystick-Adapter, C16-Sonderhefte, Data-Becker-Buch VB 350,– DM. An Jürgen Dittgen, Douvermannstr. 1, 4100 Duisburg 18, Tel. 0203/497560

Wer verschenkt an armen Schüler C64 m. Datas. u. Drucker? Nur funktionstüchtig! Kann höchstens 450,– DM bieten. Angebote an Rainer Fuchs, Kurze Str. 3a, 3993 Esterwegen, Tel. 05955/444

Suche Public-Domain-Software f. Plus4. Listen u. Probe-Disk. an Viktor Jürgens, Prinzenstr. 131, 1220 Eckernförde

---

Verkaufe Plus4 m. Turbo, Datasette u. 20 Prog.; 1551 m. 200 Prog.; Farbmon. 1802; Akustikkoppler sowie entspr. Bücher (Wert 300,–) u. Zeitschr. (m. 4 Spez.-Ausgaben) gg. Höchstgebot od. einzeln. Tel. 02941/62046

BÖRSE

COMMODORE MCS 6500 Hardware- u. Programmier-Handbücher (dt.) u. Zeitschrift „Funktechnik", Jahrg. 1955–1964, gebunden gg. Gebot abzugeben. G. Kinder, Wilh.-Leuschner-Str. 34, 2120 Lüneburg

---

Für C16: 64K-Erw.-Einbau ohne Löten – einschl. Einbauanleitung: 50,– DM (bei Nachn. Postgeb.) f. VC20: 16K-Erw. (schaltbar): 30,– DM. E. Krause, Kiebitzstr. 8, 2949 Waddewarden, Tel. 04461/3705

Verk. DREAM GIRLS – ein pikantes Adventure, Super-Bilder, 4 Disk.-Seiten, dt., C64, f. nur 35,– DM. Marcus Ahrens, Am Quälberg 6 a, 3170 Gifhorn

---

Wer hat eine dt. Bedienungsanleitung zum „Gamemaker" v. Activision od. eine Übersetzung? Klaus Sommer, Kulmbacher Str. 16, 8670 Hof/Saale, Tel. 09281/62510

Computerfan sucht dringend C64, Plus4 od. ähnl. Typ u. zugeh. Hardware gg. Wertausgleich nach Wunsch! Erfahrungsaustausch erwünscht. Zuschriften bitte über Wolke, Pichelsdorfer Str. 59, 1000 Berlin 20

HALT! Suche Tauschpartner f. C64 Top-Games. Habe z.B. World-Games, Silent Service usw. Schreibt an: Jörg Jungkind, Neckartal 92, 7210 Rottweil

---

Wer hat alte CW u. CWS-Hefte! Bitte schickt mir alles, was Ihr entbehren könnt (C16/P4). Auch User- u. Spielelistings. Danke im voraus! A. Horn, Stettnerstr. 36, 8000 München 90

Übersetze preiswert engl. Anleitungen f. Computerspiele. U. Jürgens, St. Anton-Str. 199, 4150 Krefeld, Tel. 02151/777966

Schreib doch mal!!! Computerclub Ruhrgebiet c/o AWO Jugendwerk, Pferdemarkt 7, 4300 Essen 1. Mitgliederinfo u. Prog.-Liste (PDS) gg. Rückporto

---

Verkaufe: Orig.-Softw. f. C16/+4 z.B. Quiwi, Austrospeed +4, weit unter Neupreis u. C16/64K, Datasette 200,– DM. Tauschpartner gesucht (Disk.) Marc Dörr, Brückenstr. 57, 5130 Geilenkirchen, Tel. 02453/1275

C64 Simons-Basic-Modul u. engl. Handbuch 35,– DM. C64 Macro-Assembler, Disk. u. engl. Handbuch 25,– DM. Gerd Engels, Loyerender Weg 25, 2900 Oldenburg

---

Hallo, PRINTFOX-FANS! Habe etliche neue Zeichensätze u. Grafiken z. Selbstkostenpreis f. C64! Info kostenlos bei: Bettina Stockhausen, Kennwort „C-64", Marktstr. 4, 5480 Remagen 1

Welcher nette Mensch schenkt mir die Commodore-Welt-Ausgabe Nr. 8/86? Schickt es mir bitte zu! Thomas Langmaak, Pommernring 7D, 2400 Lübeck 14

Suche Kopierprogramme aller Art (Tape-Disk./Disk.-Disk.). Übernehme Porto. Zahle 5,– DM pro Prog. (nur Disk.) Brauche auch guten Basic-Compiler. Ruft gleich an. Ralf Schmidt (02774/51180) nach 15 Uhr

---

Achtung Achtung Achtung suche Tauschpartner(in) f. C64/128. Bitte keine Anfänger. Schickt Eure Listen an Siegfried Braun, Zehntwiesenstr. 35, 7505 Ettlingen. Antwort 100 %. Tel. 07243/30357

---

* Suche Frau * zum Programmieren/Computern und mehr. Von 25–38 Jahre, max. 1,65 m. Amiga 1000 vorhanden. Schreibe an: Harry Mehne, Händelstr. 29, 5060 Berg.-Gladbach

---

VC 20. Suche defekten VC 20 (Gehäuse, Tastatur, RAM-Erw.), sowie Textverarb. Textomat v. Data Becker o.ä. f. Schreibmaschine „Privileg 3000" (Gabriele 8008); Beschr. f. Superexpander. An D. Braun, Lupinenstr. 8, 7000 Stuttgart 70

C64 – C16 – Plus4 – Lernprg. Techn. Mathe + Schulanwend. + Grafik zu reellem Preis. Cass/Disk z.B. Bruchrechn. Vokab. Geometrie, Zahnr. Festigk. Hydr. E-Techn. Katalog 1 DM Briefm.-Comp. Typ angeben! Softvers. A. Ristau Peetzweg 9, 3320 Salzgitter 1

Verk. f. C16 u.Plus4: Orig.-Software zum halben NP u. div. Prog.-Sammlungen à 20 Prog. zu je 10,– DM. Info unter Tel. 0421/491238 od. Brief an: Jens Schröder, Osterholzer Möhlendamm 55, 7800 Bremen 44

Suche f. VC-20 eine 16K-Erw. (VC-1111) v. Commodore. Zahle bis zu 35 DM. Schreibt an: Frank Thiel, Hauptstr. 54, 5231 Mudenbach, Tel. 02688/633

Kaufe Top-software f. C16 wie z.B. Future Knight. Angebote ab 20 Uhr. An Tel. 08381/3891 od. Armin Eiband, Bahnhofstr. 9, 8999 Heimenkirch

---

Verk. Plus4, Floppy 1551, Drucker 803, Datasette 1531, Turbo Plus, Joystick, Monitor grün, Softw., Lit., Floppys, 2 Mon. jung, kompl. f. 700,– DM. Wg. Wechsel auf IBM. Lutz Bienzeisler, Tel. 0911/706792

Verk. C16/64K, Floppy, Datasette, ca. 150 Games, 2 Joy, 3 Bücher, zahlreiche Zeitschr. Alles 1.000,– DM: B. Bierek, Am Unteren Weinberg 5, 8403 Oberndorf

---

Achtung C16/Plus4 Besitzer! Neuwert. Floppy 1551 (1/2 Jahr alt) wg. Systemwechsel f. * 299,99 * abzugeben! Tel. 0211/292921 ab 19 Uhr

---

C16 *** C16 *** C16 *** Hallo, Jungs! Ich besitze den C16 u. will m. Gleichgesinnten Softw. u. Informationen austauschen. Gründung eines Clubs möglich. Wer mitmachen will, schreibt an: Rainer Lerbs, Robert-Bunsen-Str. 64, 6090 Rüsselsheim

---

Die ersten 10, die mir ihre VC-20-Liste schicken, bekommen 5 Prog. gratis! Frank Thiel, Hauptstr. 54, 5231 Mudenbach

---

Achtung! Suche Composite Monitor! Stephan Schlamminger, Waldstr. 21, 8429 Ihrlerstein

---

Verkaufe Plus4, Datasette, Joystick, Bücher/Handbücher, Lerne Basic, P4 Tips u. Tricks sowie Orig.-Softw. (Mercenary, Space Pilot, Graphic Designer, Rockman u.v.a.) nur 350,– DM. M. Wiegratz, Karnaperstr. 1A, 4650 Gelsenkirchen-Horst, Tel. 0209/53810

Verk. Commodore Plus4 m. Datasette, Joyst., Adapter u. über 100 Spiele (u.a. Ace, Pluspaket, Winterolympiade, viele 64K-Spiele). Preis: nur 300,– DM VB! Tel. 07351/76503. Neuwertig! Wenig gebraucht!

Verkaufe f. C16! 20 Spiele- u. 20 Anwend.-Prog. f. je 20,– DM! Top-Anwender-Prog.! Info gg. Rückumschlag. Oder schickt 20-DM-Schein (Prog. nur auf Tape) an Klaus Spengler, Burgunder-Str. 9, 4600 Dortmund 30

Verk. Komplettanlage: Plus4, Floppy 1551, super Farbmon. f. sagenhafte 830,– DM (NP: über 1.600,– DM). Bei Selbstabholung: 790,– DM u. sehr viel Zubehör. M. Beckmann, Haselweg 1, 8028 Taufkirchen, Tel. 089/6124964

Suche Drucker f. Plus4 (nicht defekt), zahle bis 100,– DM. Verk. u. tausche auch Spiele. Ab 50 Pf. Schreibt mit Rückporto an Uwe Heyer, Hellweg 27, 4300 Essen 14, 100 % Antwort

---

Verkaufe Commodore 128 m. Floppy 1541, Diskbox, Joystick, 40 Disk., Disk.-Locher, massig Games, alles zus. f. lächerliche 1.099,– DM. Tel. 06245/3746 (Martin)

Wer verkauft mir seinen Plus4? Angebote an: Karl-Heinz Schön, Bergstr. 14, 7015 Korntal 1. Verk. Orig.-Softw., ca. 30 versch. Prog. auf Kass., Liste nur gg. Rückporto

---

Suche Calc/Plus sowie Script/Plus f. C16/64K. Tel. 04731/4360

VC-20 ** VC-20 ** VC-20 Suche Plotter VC 1520 u. m. DIL-Schalter adressierbare 8KB-Erw. f. VC-20. Angebote an: Andreas Fröhling, Ulmenstr. 25 a, 2940 Wilhelmshaven

---

Verkaufe: Monitor 1901, Monitor-TV-Tuner, Datasette 1530, Kass., GEOS, Lightpen m. Modul, 21 Zeitschr., Turbo Tape m. Disk., 2 Lernbücher, 4 Disk., 2 Joy.; alles orig.-verp. Preise nach VB. Tel. 07524/6886

++ Apfelmann Plus 4 ++ Suche Apfelmann-Prog. f. 64K, möglichst m. Masch.-Routine. Zoom-Möglichkeit wie beim C64 und Ausgabe auf Drucker wäre schön! Hajo Stengert, Ehrenmalstr. 15, 4130 Moers 2, Tel. 02841/61321

Verkaufe f. C16: 20 Spiele u. Anwend.-Prog. f. je 10,– auf Tape. Schickt einen 10-DM-Schein an C. Schnoor, Lausanner Str. 129, 2800 Bremen 44.

C16/116 PLUS4 — Menue

Menuesteuerung

Ein benutzerfreundliches Programm wird über Menues gesteuert. Die Erstellung solcher Menues und die dabei nötige Tastenabfrage macht Anfängern oft Mühe. Dabei ist es ganz einfach:

1. Alle Auswahlmöglichkeiten auf dem Bildschirm ausgeben,
2. abfragen, ob eine der angegebenen Tasten gedrückt wurde,
3. entsprechend der gedrückten Taste im Programm fortfahren.

C16/116 PLUS4 — Floppy-Tricks

Zugriffszeit verkürzen

Mit folgender Zeile, welche ins Programm eingebaut wird, kann die Zugriffszeit der Floppy verkürzt werden, der Lesekopf wird schneller bewegt.

```
10 open1,8,15,"m-w"+chr$(7)+chr$(28)+chr$(1)+chr$
(15)
```

C16/116 PLUS4 — REM-Zeilen-Marker

REM-Zeilen invers darstellen

Programme werden durch REM-Zeilen besonders übersichtlich. Will man diese zusätzlich hervorheben, so ist es sinnvoll, den Text revers darzustellen.
1. Methode: Setzt man nach REM ein Anführungszeichen, so kann man mit der CTRL-Taste die meisten Buchstaben revers schreiben.
2. Methode: Steht nach dem REM-Token (143) im Speicher eine 18, so werden alle restlichen Zeichen der Zeile revers.

C16/116 PLUS4 — LIST-Befehl

Tricks zum LIST-Befehl:

Abkürzung für LIST: LI (L/SHIFT I)

list	listet alle Zeilen
list 10	listet Zeile 10
list 10-	listet alle Zeilen ab 10
list -10	listet alle Zeilen bis 10
list 10-20	listet Zeilen 10-20

Listing auf den Drucker:
open1,4:cmd1:list (RETURN)
print#1:close1 (RETURN)
(mit open128,4:cmd128... doppelter Zeilenabstand)

C16/116 PLUS4 — POKE 1343,x

Maximale Länge des Tastaturpuffers

Kann durch einpoken eines Wertes zwischen 0 und 17 verändert werden.

POKE 1343,0: keine Eingabe von Tastatur mehr möglich
POKE 1343,10: normal, Tastaturpuffer geht von 1319-1328
POKE 1343,14: erweitert, Speicherstellen 1329-1333 werden überschrieben (werden nur beim Einschalten benötigt).

C16/116 PLUS4 — Strings sortieren

Sortieren von Zeichenketten alphabetisch
Die logischen Operatoren <=> funktionieren auch beim Vergleich von Zeichenketten. Z.B. ist "aab" > "aaa". Dies kann man zum Ordnen der Daten verwenden.

Zum Einlesen der Daten von Diskette siehe seq-Files.

C16/116 PLUS4 — Startadresse

Mit diesem Programm kann man die Startadresse eines auf der Diskette befindlichen Programms ermitteln:

```
2 input "programmname";p$
3 open1,8,2,p$+",p,r":get#1,sl$:get#1,sh$:close1
4 sl=asc(sl$+chr$(0))
5 sh=asc(sh$+chr$(0))
6 print"startadresse"=sl+256*sh
```

C16/116 PLUS4 — SEQ-Dateien

Sequentielle Dateien

Will man größere Datenmengen verarbeiten, so ist es sinnvoll, diese als sequentielle File abzuspeichern und dann wieder einzulesen.

DATA-Zeilen zu sequent. File:

```
10 open8,8,8,"daten,s,w"
20 fori=1 to 50: read a,b,c
30 print#8,a;",";b;",";c
40 next:close8
```

Das Komma zwischen den Daten ist wichtig, da sie sonst nicht richtig mit INPUT# gelesen werden.

Hinweis:

Die Anwendung ist sinnvoll bei Zugriff auf viele kleine Files. Je größer das File, umso weniger bringt es. Bei 1k sind es ca. 7 %.

Allgemeingültiges Beispiel:
Texte, Sprungadressen und g$ anpassen. Nur die Zeichen (Zahlen oder Buchstaben) in g$ werden akzeptiert.

```
1 g$="abc":rem gueltige zeichen
2 printchr$(147)
3 m$(1)="Menuepunkt 1                    a"
4 m$(2)="Menuepunkt 2                    b"
5 m$(3)="Menuepunkt 3                    c"
10 for i=1 to 3:char,10,5+i,m$(i):next
11 getkeya$
13 on instr(g$,a$)+1 goto 2,20,30,40:rem
prg-fortsetzung
```

Listing ohne Zeilennummer:
poke22,35:list:poke22,25 (RETURN)

Listing als seq-File of Floppy:
(für Textverarbeitungssystem wie SCRIPT/PLUS)
open8,8,8,"name,s,w":cmd8:list (RETURN)
print#8:close8 (RETURN)

Listing in Farbe:
Orem":*** (RETURN)
(für *Steuerzeichen, z.B.
CTRL/I,CTRL/h,SHIFT/CLEAR)
Poke4103,13 (RETURN)
Bei LIST werden alle Steuerzeichen ausgeführt.

Dieses Programm erledigt die Arbeit automatisch. Einfach an das Programm hinten anhängen und starten. Nach REM sollte ein Leerzeichen stehen, dieses wird in den Revers-Kode (18) umgewandelt. Wieder normal mit ze=32.

```
63000 ze=18:rem-zeilen markieren *****
63001 for i=peek(43)+256*peek(44)+4 to peek(45)+256*
peek(46)
63002 ifpeek(i)=Otheni=i+5
63003 ifpeek(i)=143thenpokei+1,ze
63004 next
```

Sortieren der Strings alphabetisch:

```
30 for i=0 to n:a$=d$(i):
40 for k=i to n:b$=d$(k)
50 ifa$>b$thend$(i)=b$:d$(k)=a$:a$=b$
60 next:next
70 for i=0 to n:printd$(i):next
```

Die Zeilen 30–50 kann man zu einer Zeile zusammenfassen, dann wird es etwas schneller.

Eine Abschaltung der Tastatur ist in Spielen oft erwünscht (Joystick, zu frühe Eingabe ...)
Eine Erweiterung auf 19 ist möglich, wenn kein Kassettenbetrieb erfolgt.

Man kann Zeile 30 auch so schreiben:
30 print#8,a:print#8,b:print#8,c

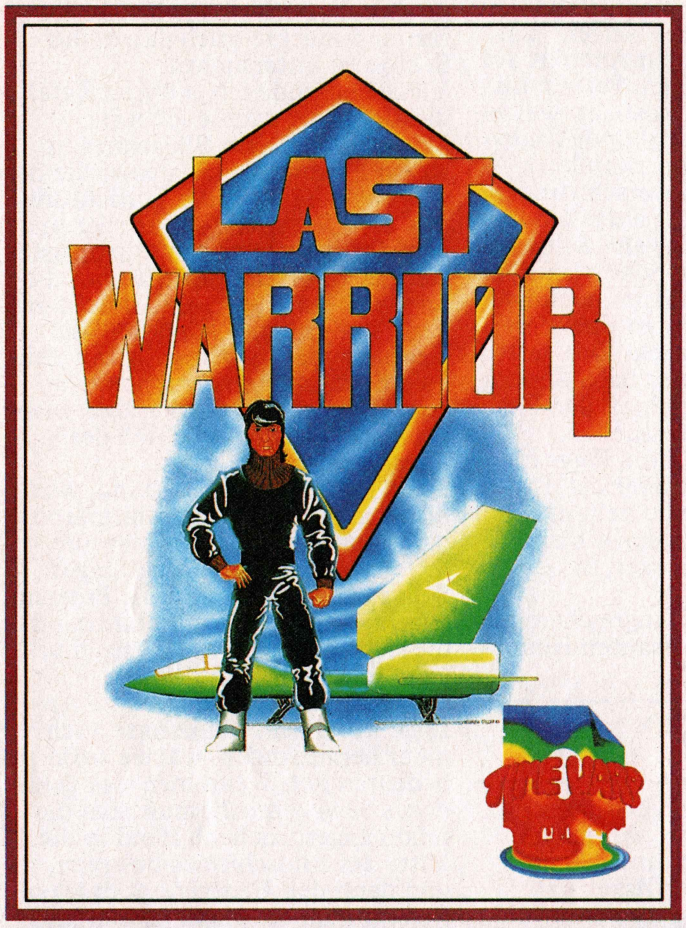
Einlesen der Daten von Diskette:

```
100 dim a(50),b(50),c(50)
110 open8,8,8,"daten,s,r"
120 for i=0 to 50
130 input#8,a,b,c :next
```

Anwendung:

Bei den meisten Maschinenprogrammen ist die SYS-Adresse zum Starten identisch mit der Adresse, bei der es abgespeichert wird. Erhält man als Startadresse 4097, so handelt es sich um ein Programm, das wie ein Basicprogramm zu behandeln ist.

NUN ERBT MAL SCHÖN



Vor einiger Zeit machte ein von der französischen Firma Infrogrames erstelltes Iconadventure mit dem Namen Inheritance (Deutsch: Die Erbschaft) auf sich aufmerksam.
Bei diesem Game schlüpfte der Spieler in die Rolle von Peter Stone, einem Taugenichts, der bei fast allen Menschen auf dieser Welt Schulden hat (bei mir leider nicht!).
Sein Leben ist nicht das angenehmste, wenn man bedenkt, daß er in ständiger Flucht vor seinen Gläubigern ist. Doch sein unruhiges Leben soll noch aufregender werden. Eine seiner reichen Tanten segnet das Zeitliche. Prompt erhält Peter ein Telegramm, in dem er lesen kann, daß besagte Tante ihn zum Universalerben eingesetzt hat. Nach einem Freudentanz fängt er an, das Kleingedruckte zu lesen. Die Erbschaft ist abhängig von einer Bedingung: Peter muß in einer Nacht eine Million Dollar in einem Spielcasino im Zockerparadies Las Vegas gewinnen. Sollte ihm das gelingen, erhält er die Erbschaft seiner reichen Tante, mit der alle seine Sorgen beseitigt wären.

DIE URKUNDE IST DIE FORTSETZUNG

Die Erbschaft bestand aus 3 Spielteilen. Im ersten Teil mußte man versuchen, sich in seiner näheren Umgebung für sein Abenteuer auszurüsten. Im zweiten Teil galt es, zum Flugplatz und schließlich nach Las Vegas zu gelangen, und im dritten Teil ging es dann um den Gewinn der Million.
Das Abenteuerspiel zeichnete sich durch eine Cursorsteuerung aus, mit der man einige auf dem Bildschirm dargestellte Objekte anklikken konnte, um dann Vorgänge ausführen zu können, zum Beispiel einen Türknauf.
Durch Drücken auf den Feuerknopf öffnet sich die Tür dann.
Die Erbschaft erfreute sich damals großer Beliebtheit, und so ist es auch nicht weiter verwunderlich, daß mit „Die Urkunde“ (Die Erbschaft II), eine Fortsetzung auf den Markt gebracht wurde.
Die Urkunde fängt genau da an, wo die Erbschaft aufgehört hat. Peter hat eine Million Dollar im Spielcasi-



Zwei Spiele – Die Urkunde und The Last Warrior auf dem CW-Prüfstand

INFOGRAMES

no gewonnen, genau so, wie seine Tante das in ihrem Testament verlangt hat. Doch Peter stellen sich neue Probleme: Er findet seine Geburtsurkunde nicht mehr. Plötzlich tauchen viele „falsche Neffen" auf, die ebenso glauben, ein Anrecht auf die Erbschaft zu haben. Peter muß aus diesem Grunde, koste es, was es wolle, seine Geburtsurkunde finden. Doch leider ist diese Urkunde auf Nimmerwiedersehen verschwunden. Beinahe die Hoffnung aufgebend, erinnert er sich an das alte Schloß eines Verwandten in Schottland, der den Familienstammbaum irgendwo versteckt hält. Leider ist dieser Verwandte auch vor einiger Zeit gestorben, und so muß Peter nach Schottland reisen, um im Schloß den Familienstammbaum zu suchen.
Ihre Aufgabe ist es nun, Peter im Schloß mit Joystick oder der Tastatur bei seiner Suche zu helfen. Strengen Sie sich an, denn sonst bleibt Peter der arme Schlucker, der er schon lange ist.
Soviel zur etwas umfangreicheren, aber durchaus interessanten Hintergrundstory.

GESPANNT AUF DEN NACHFOLGER . . .

Da ich das sehr gute Vorgängerspiel Die Erbschaft kannte, war ich auf den Nachfolger gespannt. Ich lud voller Erwartung die Diskette in meinen 64er ein und harrte auf die Dinge, die da kommen mögen. Zu Beginn des Spiels befand ich mich vor dem Eingang des Schlosses. Diese Grafik ist ein wirklicher Leckerbissen, aber ich sollte keine Zeit haben, sie lange zu bewundern. Plötzlich öffnete sich die Tür, und der Butler des Hauses erschien auf der Bildfläche. Nachdem ich die Space Taste gedrückt hatte, fing er auch schon an zu sprechen. Leider nicht mit Sprachausgabe. Das Sprechen äußerte sich durch Sprechblasen, in denen seine Kommunikationsversuche durch Text ausgedrückt wurden. Er sagte: „Willkommen Mister Stone, Sie werden erwartet."
„Stone?" Wieso Stone, ich heiße doch Borgmeier. Egal, ich schlüpfte in eine neue Rolle. Nun forderte der Butler mich auf, ihm zu folgen. Er führte mich durch einige Räume des Schlosses, um mir mein Zimmer zeigen zu können. Mir fiel sofort die exzellente Grafik auf. Wenn wir durch eine Tür gingen, sah man auf dem Bildschirm wie langsam die Tür näherkam und schließlich aufging. Als wir nun endlich in meinem Zimmer angekommen waren, ließ der Butler mich einfach in meiner Unterkunft stehen und sagte, ich solle warten, der Notar komme gleich wegen der Formalitäten. Da ich mich nie irgendwelchen Anweisungen gerne füge, ließ ich meine Koffer stehen und machte mich daran, das Schloß zu untersuchen.
Vielleicht finde ich ja durch Zufall die Urkunde, dachte ich mir.
Man sieht auf dem Bildschirm in jedem Zimmer des Schlosses die hervorragend dargestellte Einrichtung und eine Hand, die der Spieler mit dem Joystick oder den Cursortasten steuern kann. Berührt man mit dieser Hand ein Objekt und drückt den Feuerknopf, dann kann man die Gegenstände bzw. ihren Inhalt nehmen oder anderen Personen, wie zum Beispiel dem Butler, dem man bei seiner Suche im Schloß des öfteren begegnet, geben.
Trägt man einen Gegenstand, wird dieser am unteren Bildschirmrand dargestellt. Man kann leider nur 4 Gegenstände auf einmal tragen.
Sehr rätselhaft war mir die Prozentanzeige am linken Bildschirmrand, die sich von 99% abwärts nach unten zählte. In der Anleitung schweigt man sich über diese Anzeige leider aus. Ich konnte auch nicht herausfinden, was sie wohl zu bedeuten hat. Wenn man sich geradeaus bewegen will, muß man die schon angesprochene Hand in die Mitte des Bildschirmes bewegen, und dann den Feuerknopf drücken. Bewegt man die Hand an den linken bzw. rechten Bildschirmrand, vollzieht man eine Vierteldrehung, und man bekommt ein anderes Bild zu sehen, auf dem sich wieder neue Gegenstände befinden.

. . . UND SEHR ENTTÄUSCHT

Ich machte mich auch zu Beginn des Spiels wieder mit der mir schon aus der Erbschaft bekannten Steuerung vertraut. Ich lief durch zahlreiche Räume und untersuchte an die 20 Möbelstücke. Aber nichts geschah. Ich konnte keine Gegenstände, wie mir in der Anleitung versprochen, nehmen und mit mir herumtragen. Ebenso hatte ich das Gefühl, bei meiner Erkundungstour im Schloß immer die selben Räume zu sehen. Ich machte unzählige Vierteldrehungen, hatte aber immer das Gefühl, im Kreis zu laufen. Die selbe Steuerung, die ich bei der Erbschaft in den allerhöchsten Tönen gelobt hätte, begann ich nun zu verfluchen. Die Programmierer haben bei der Urkunde nicht präzise genug gearbeitet. Ebenso enttäuschend war es, daß ich zunächst keine Gegenstände aufsammeln konnte, obwohl ich fast jedes Möbelstück mit der Hand angrabbelte. Erst nach meinem einständigen Programmtest fand ich in einem Raum einen Fotoapparat.
Durch diesen Fund angespornt, suchte ich weiter. Doch nach einiger Zeit schaltete ich den Computer ab und warf die Diskette in die nächste Ecke.

FAZIT

Die Grafik ist bei „Die Urkunde" hervorragend. Man merkt ihr die Liebe zum Detail an. Auf Sound wurde verzichtet.
Am Anfang ist die Motivation sehr hoch, was vielleicht an der sehr originellen Hintergrundstory liegt. Doch nach einigen Versuchen, Gegenstände aufzusammeln oder Treppen zu steigen, strebte die Motivation bei mir (mathematisch ausgedrückt) gegen Null. Aus der spielerischen Lust wurde Frust. Mein Rat an Sie: Wenn Sie noch ein Spiel aus der Actionadventurekategorie suchen, dann machen Sie einen großen Bogen um „Die Urkunde", wenn Ihnen Ihre Nerven lieb sind. Statt dessen sollten Sie sich lieber den ersten Teil, die Erbschaft, zulegen.

THE WAR

Wir haben das 31. Jahrhundert und befinden uns auf dem Planeten Argon, der momentan von einem bösen Diktator beherrscht wird. Schon viele Aufständische haben versucht, mit ihrem Kampfgleiter den Superbösewicht zu beseitigen. Leider hat das bis jetzt noch nicht so ganz geklappt und keiner kam mehr von dieser Mission zurück. Sie sind nun einer von vielen Anwärtern auf die Kampfpilotenprüfung und der Computer hält ausgerechnet Sie für den letzten halbwegs geeigneten Kampfpiloten, der den Diktator vernichten könnte. Also nehmen Sie, tapfer wie Sie sind, den Auftrag an, denn es ist die letzte Chance der Rebellen, ihr Land vor dem endgültigen Untergang zu bewahren. So etwa lautet die Hintergrundgeschichte zum Spiel „The Last Warrior". Nicht sehr einfallsreich, aber doch stimmungsvoll. Das Ziel des Spiels ist es also, eine feindliche Basis anzufliegen und dort zum Zweikampf gegen den

LAST RIOR

Diktator anzutreten. Allerdings wird Ihnen der Weg dorthin nicht leicht gemacht. Anscheinend hat der Diktator Angst vorm letzten Kämpfer und schickt ihm allerlei Kleinzeug an Kampfgleitern und Jägern entgegen. Einige dieser feindlichen Einheiten kommen dem kinoerfahrenen Spieler ziemlich bekannt vor. Da hat sich schon manchmal ein Jäger von Darth Vaders Geschwader daruntergemischt.
Ist man im Spiel schon weit gekommen, so hat man die Möglichkeit, sich in eine Highscore-Liste einzutragen, die auch abgespeichert wird.
Nun zum Spiel. Hat man das Programm geladen, so erscheint zuerst ein Screen mit einer Straße und Bäumen. Nichts weiter. Nach ein paar Sekunden jedoch beginnt ein Text, der die gesamte Geschichte vom Spiel erzählt, 3-dimensional die Straße entlang in die Tiefe zu ziehen. Diese Art, eine Story zu erzählen ist sehr beeindruckend, da sie eine Menge an Effekten mit sich bringt. Ist die Geschichte zu Ende, so erscheint ein Titelbild. Erst danach wird das Hauptprogramm geladen. Und hier erwartet den aktiongespannten Spieler eine herbe Enttäuschung. Zuerst wird eine Karte des Planeten auf dem Bildschirm sichtbar, auf der die eigenen Basen, die allerdings nur mehr Ruinen sind, und das Hauptquartier des Diktators dargestellt werden. Außerdem kann man entscheiden, in welche Richtung man fliegen will.
Erst dann kommt man in den eigentlichen Spielscreen. In ihm wird die ganze Szenerie in 3D-Grafik dargestellt. Wer da aber eine phantastische Darstellung mit Gebirgen usw. erwartet, der liegt gewaltig daneben. Die grafische Darstellung beschränkt sich lediglich auf ein allseits bekanntes riesiges Netz, über dem der Spieler in seinem hypermodernen Jet dahindüst. Das Netz ist und bleibt während des ganzen Spiels eben und schnurgerade. Sogar wenn man die Flugrichtung wechselt, bleibt es eben. Da fragt man sich schon, ob man in einem Fahrzeug oder in einem Flugzeug sitzt. Es ist sehr fein gezeichnet und bewegt sich einigermaßen fließend und schnell. Mit dem Hintergrund verhält es sich nicht anders. Er ist zwar zeichnerisch gut gelungen, aber auch nicht sehr abwechslungsreich. Er besteht nur aus einem Gebirge, das allerdings recht farbenfroh geworden ist, was nichts Negatives bedeuten soll. Das Scrolling des Horizonts ist butterweich und ebenso schnell wie die Animation des 3D-Netzes. Die Sprites sind sehr vielfältig vorhanden, aber nur einfarbig, und lassen leider nicht sehr viel Intelligenz erkennen. Sie fliegen immer nach dem selben System, so daß man schon nach kurzer Übungszeit keine Probleme mehr mit seinen Feinden hat. Die Animation ist allerdings recht gut gelungen. Es geschieht oft, daß ein Fighter gekonnte Pirouetten dreht. Der eigene Kampfflieger ist mit einer Bordkanone ausgestattet. Auf dem Bildschirm ist dazu ein Fadenkreuz zu sehen. Die Geschosse der eigenen und auch der feindlichen Kanonen sind große Kugeln, die nicht sehr überzeugend aussehen.
Das Cockpit Ihres Jets ist mit genau drei Anzeigen ausgerüstet. Dazu gehören natürlich die Punkteanzeige, der Radarschirm und das Schutzschild.
Auch der Sound läßt einiges zu wünschen übrig. Aus dem Lautsprecher tönen nur schwache Schuß- und Explosionsgeräusche. Von Musik und dergleichen ist nichts zu hören. Ihr Raumgleiter kann leider nur über Joystick gesteuert werden. Die Abfrage reagiert allerdings recht schnell und genau und läßt in dieser Beziehung keine Wünsche mehr offen.
Die Motivation des Spiels ist nicht sehr hoch. Schon nach etwa einer Viertelstunde beginnt das Spiel enorm langweilig zu werden, da es ja absolut nichts Neues bietet.

FAZIT

Nach der anfangs sehr guten Aufmachung durch die Präsentation der Hintergrundgeschichte,enttäuscht das Spiel sehr.
Trotz der relativ gut gelungenen Grafik hat das Programm enorme Mängel. Das größte Manko von „The Last Warrior“ ist jedoch seine Eintönigkeit. Schon vor zwei, drei Jahren hat es bessere Spiele der gleichen Art gegeben. Deshalb ist dieses Spiel nur eingefleischten Fans von diesem Genre zu empfehlen. Wer's unbedingt haben will, der kann es bei Eurogold für ca. 40,– DM für die Diskette beziehen. *(mn)*

(TEXT AUF SEITE 50)

ZEILEN-NUMMERN

Warum soll man es sich nicht leichter machen, wenn's geht? Ein nicht unerheblicher Aufwand bei der Eingabe von Listings ist das unvermeidliche Eintippen der Zeilennummern.

```
Auto-Number C 64                    - 1 -


10 rem automatische                 <ca>
11 rem zeilennummerierung beim      <bm>
12 rem c 64 : sys 49259             <io>
13 rem commodore-welt/h.b.          <lp>
14 dimhx(75)                        <pb>
15 fori=0to9:hx(48+i)=i:hx(65+
i)=i+10:next                        <gi>
16 fori=49152to49269:readmc$:h
i=asc(left$(mc$,1)):lo=asc(rig
ht$(mc$,1))                         <dp>
17 dz=16*hx(hi)+hx(lo):pokei,d
z:next                              <mm>
18 printchr$(147)"zeilen-autom
atik aktiviert."                    <ch>
19 print"befehl: a <schrittwei
te> z.b.'a 10'"                     <bc>
20 print"ausschalten mit <retu
rn>."                               <hm>
21 sys49259:new                     <jm>
22 rem auto.64 mc-daten             <jp>
23 data20,73,00,c9,41,f0,06,20      <bj>
24 data79,00,4c,e7,e7,20,73,00      <dg>
25 datad0,04,a2,0a,d0,03,20,9e      <ed>
26 datab7,8e,47,c0,a9,29,8d,04      <gk>
27 data03,a9,c0,8d,05,03,4c,ae      <gd>
28 datae7,60,48,c9,a1,d0,39,88      <gd>
29 datab1,7a,d0,0d,a9,7c,8d,04      <hl>
30 data03,a9,a5,8d,05,03,4c,74      <eo>
31 datae4,a5,14,a6,15,18,69,0a      <em>
32 data90,01,e8,85,63,86,62,a2      <dn>
33 data90,38,20,49,bc,20,df,bd      <hc>
34 data20,87,b4,20,82,b7,84,c6      <ek>
35 datab1,22,99,77,02,88,10,f8      <ec>
36 data4c,7c,a5,a9,00,8d,08,03      <gd>
37 datae9,c0,8d,09,03,60            <bp>
```

BÖRSE

VC-20. Suche: Floppy für VC 20 für 200,– DM und einen Drucker für 200,– DM. Verkaufe: Joystick für 10,– DM und Datasette für 100,– DM. Ihre Zuschriften senden Sie bitte an: Martin Stahmer, Fr.-Ebert-Damm 26, 2000 Hamburg 70

C 16 Einsteiger sucht Info zu Software und Programmen. Wie kann ich C 20 Listings für C 16 umschreiben? Sind Peek und Poke bei C 20 und C 16 gleich? Kann andere Software für C 16 verwendet werden? Bin über Info dankbar. W. Koch, Röttgen 133 5600 Wuppertal 1

EINES FÜR ALLE

DAS UNIVERSELLE DRUCKER-INTERFACE

Die Wahl des Druckers fällt oft leicht, da jeder genaue Vorstellung zum Kauf mitbringt. Ein wenig schwerer wird's da schon bei der Auswahl des richtigen Interfaces und vor allem: bei der Programmierung dessen Funktionen, wenn's mal den Drucker mit dem Commodore-Computer verbindet.



Da bleibt gar nichts anderes, als das Druckerhandbuch und die Anleitung des soeben im Computershop erworbenen Interfaces zu studieren. Welche Sekundäradresse beispielsweise bewirkt Klein-/Großschrift auf dem Drucker? Welche Steuerzeichen sind nötig, um den grafikfähigen Drucker mit 24 Nadeln anstatt mit nur acht drucken zu lassen? (Voraussetzung ist natürlich, daß Ihr Drucker diese Möglichkeit vorgesehen hat.)

Nun, ab sofort könnte Ihnen ein hilfreicher "Alleskönner" zur Seite stehen, dem diese Probleme wie ein Kinderspiel erscheinen: Das "Printerface" von RTK, München, seit kurzem als eine überarbeitete Version des PCB C 128 Interface DZ 300 auf dem Markt.

Es wird über den seriellen Bus am Computer und mit dem Centronics-Stecker am Drucker angeschlossen, so wie's für ein Interface üblich ist.

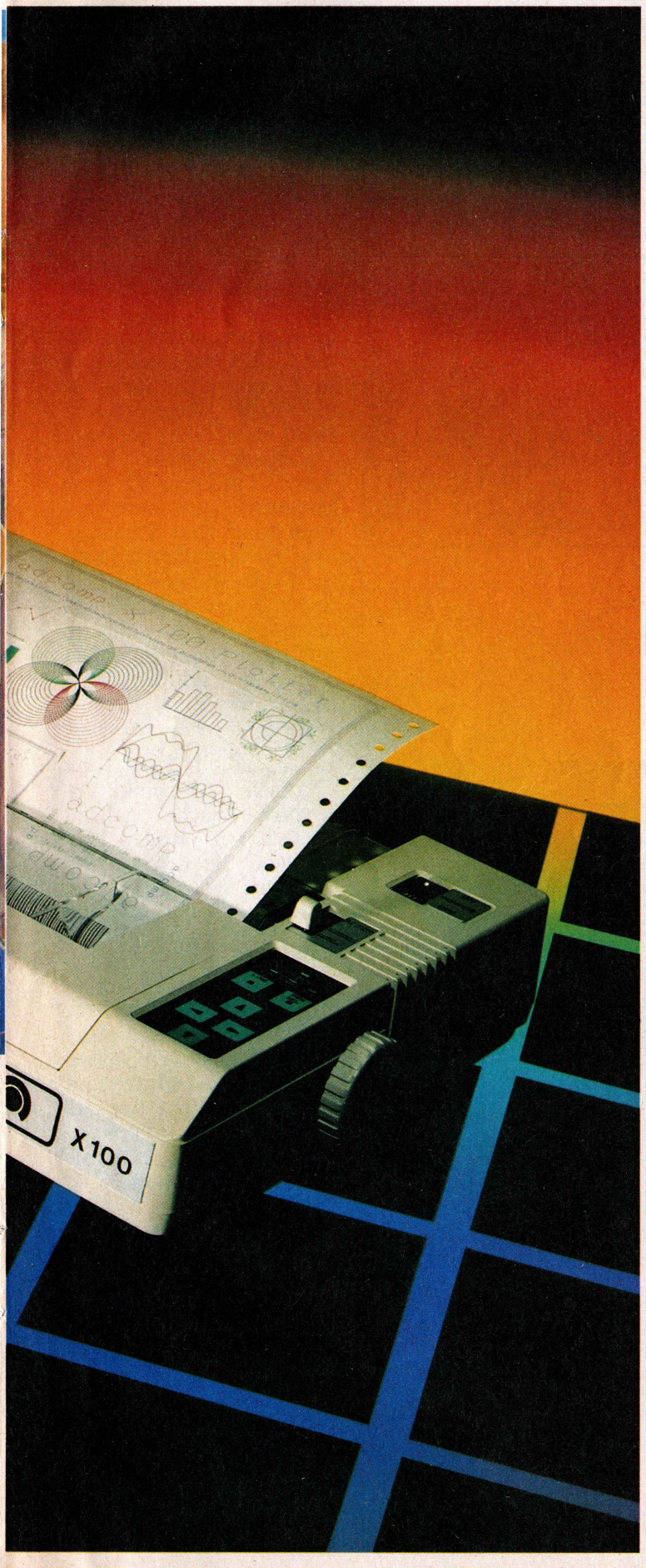
GEEIGNET FÜR DIE GÄNGIGSTEN COMMODORE-HOMECOMPUTER

Die grundsätzlichen Pluspunkte wurden natürlich beibehalten:
Das "Printerface" eignet sich für den C 64 und den C 128 gleichermaßen, falls noch jemand den VC 20 + Drucker zu Hause

PROGRAMM-DISK FÜR DIE DRUCKER-EINSTELLUNG – MIT DIP-SCHALTERN FIXIEREN

stehen hat, wird er seine Texte oder Grafiken ebenso zu Papier bringen können.

Befinden Sie sich im "Commodore-Modus", so wird eine Druckausgabe analog zum Bildschirm erreicht, d.h. ein Klammeraffe ist auch auf dem Druckblatt ein Klammeraffe und kein Paragraphenzeichen oder Umlaut (die betroffenen Zeichen werden jedem bekannt sein!), die DIN-Tastatur des C 128 ist dem "Printerface" inzwischen so vertraut, als wären es die ASCII-Zeichen, ganz wichtig erscheint uns die Möglichkeit, daß auch Typenraddrucker angeschlossen werden können. Mit acht außen auf der Oberseite des Interfaces angebrachten DIP-Schaltern lassen sich die verschiedenen Optionen einstellen, wie Druckerart (Commodore-, Epsonkompatibel), Geräteadresse, ob ASCII- oder DIN-Zeichensatz, sowie die bereits erwähnten Sekundäradressen und letztendlich, ob der



Auto-Line-Feed aus- oder eingeschaltet sein soll.

Schieben Sie nun die dem Hardware-Paket beiliegende Diskette ins Laufwerk und laden Sie das Programm "Einstellung", nach 6 verschiedenen Testausdrucken unter Verwendung jeweils anderer Sekundäradressen werden Sie sehr schnell feststellen, welche für Sie in Frage kommen (nämlich die, die den Probeausdruck so exakt ermöglichen, daß es dem im Anleitungsbeiheft abgedruckten am nächsten kommt). Nun werden die erwähnten DIP-Schalter so eingestellt, wie das Programm "Einstellung" es Ihnen mitteilt. Mal ehrlich, geht's noch einfacher? Das ist nämlich schon eine der Neuerungen des verbesserten Interfaces, daß dieses Anpassungsprogramm vereinfacht und kaum noch "anfängerfeindlich" gestaltet wurde. Ein Wort an die C 128-Benutzer: es ist egal, in welchem Modus Sie das Installationsprogramm laden und starten, da es in reinem Basic 2.0 geschrieben wurde, läuft es in beiden Modi.

EXTERNE STROMVERSORGUNG

Normalerweise wird das "Printerface" nach dem Anschluß über Pin 18 der Centronics-Schnittstelle mit Strom versorgt, das ist jedoch von Drucker zu Drucker verschieden. Sollte das nicht funktionieren, so wird ein ebenfalls mitgeliefertes Kabel mit Verbindung zum Kassettenport Ihres Computers im Gehäuse des Interfaces angeschlossen (das Begleitheft gibt darüber genaue Auskunft). Das Interface bezieht jetzt seine Stromversorgung vom Kassettenport. Durch die vorhin erwähnte unkomplizierte Einstellmöglichkeit per Software hat kaum noch ein exotischer Drucker eine Chance, nicht mit dem "Printerface" zusammenzuarbeiten zu müssen, denn es simuliert praktisch alle gängigen und bekannten Interfaces, von Data Becker über Görlitz bis Merlin. Im übrigen ist auf der letzten Seite der Bedienungsanleitung eine Tabelle mit den Einstellpositionen der DIP-Schalter von 42 (!) verschiedenen Druckertypen abgedruckt. Falls Sie hier Ihren Drucker noch nicht gefunden haben, bleibt noch immer das Programm "Einstellung".

BEIM C 128 GEHT'S SCHNELLER –

Haben Sie das "Printerface" am C 128 angeschlossen, so wird der "Fast Serial"-Modus zur Datenübertragung genutzt und erreicht damit eine Geschwindigkeit von ca. 1300 Zeichen pro Sekunde. Optimal erweist sich in solch einem Fall die Möglichkeit der Zwischenspeicherung von Daten durch "Printerface".

Bei der 8-kByte-Version entspricht das etwa 3, bei der mit 32-kByte-ca. 11 Seiten Text. Praktisch bedeutet das, daß beim Ausdruck eines vergleichsweise langen Schriftstückes keine Wartezeiten für den Compu-

DATENPUFFER IM INTERFACE

ter entstehen. Es können andere Arbeiten erledigt werden, während der Ausdruck über das "Printerface" läuft.

Übrigens ist dieser Puffer immer resident und

muß nicht eigens aufgerufen werden.

Unsere Tests ergaben, daß das Interface problemlos mit CP/M beim C 128 und "Geos" beim C 64 druckermäßig zusammenarbeitete. Vor dem Laden von "Geos" ist das ebenfalls auf der mitgelieferten Einstell-Diskette gespeicherte, kurze Ladeprogramm für "Geos" zu laden, das dem Drucker und der Floppy vor dem Nachladen der Benutzeroberfläche "Geos" wichtige Daten vermittelt.

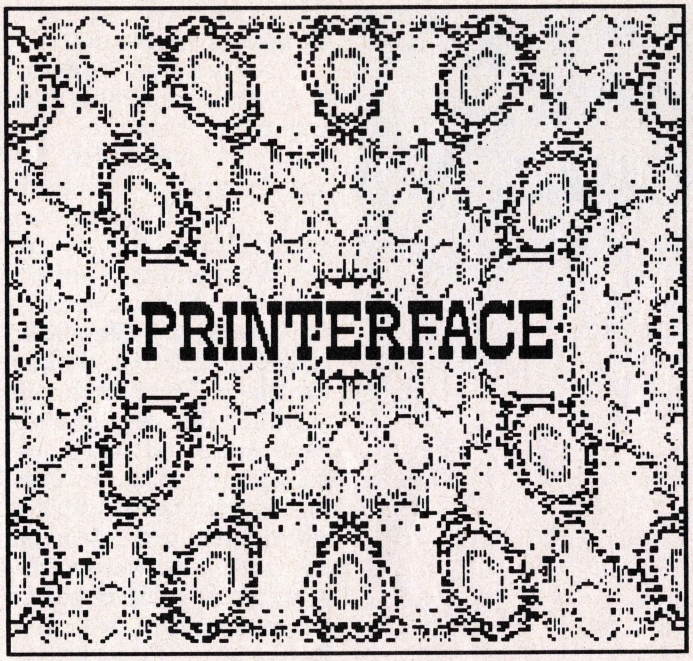
Im Gegensatz zu herkömmlichen Interfaces weist das "Printerface" keine festen Sekundäradressen irgendwelchen Funktionen zu. Es gibt praktisch vier Grundeinstellungen, Sie können aber die Sekundäradressen aller Funktionen frei wählen und bestimmen oder eine Funktion für alle Sekundäradressen festlegen.

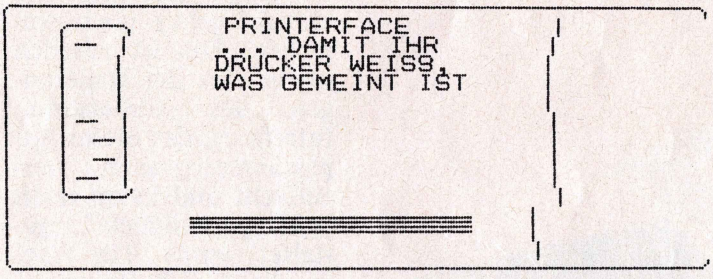
VIER GRUNDEINSTELLUNGEN

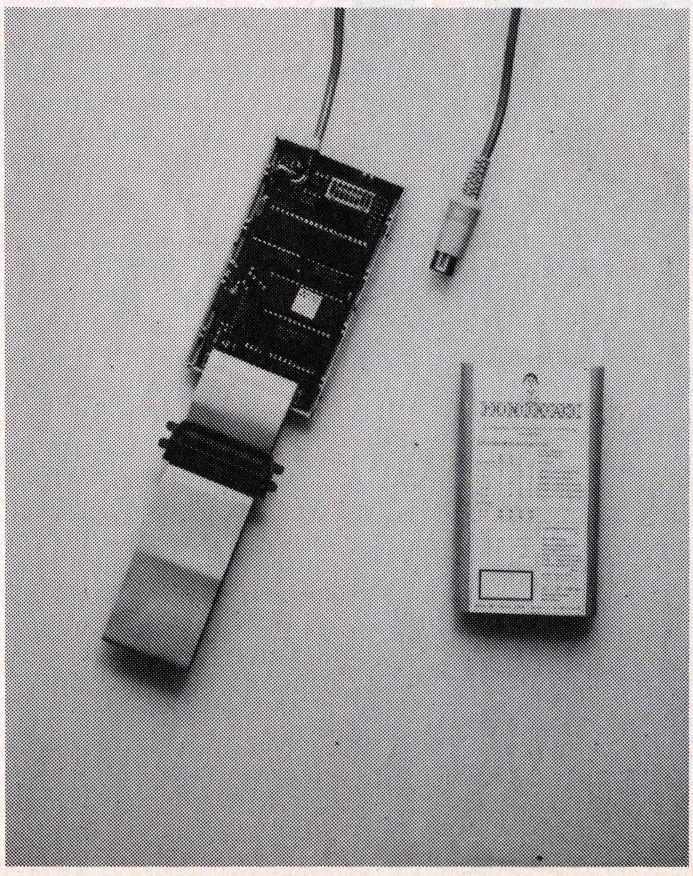
Folgende Grundeinstellungen sind vorgesehen: der Commodore (COM)-, Viza-, Data Becker und der Umlaut-Modus.

Sehr ausführlich ist in der Betriebsanleitung erläutert, welche Sekundäradresse für welche Funktion in Frage kommen. Kleine Beispielroutinen in Basic runden die Erklärungen über die einzelnen Modi ab.

Die Funktion Nr. 5 beispielsweise ist für denjenigen interessant, der seine mit einem Mal- oder Zeichenprogramm erstellten Bilder und Grafiken zu Papier bringen will.







Dieses Programm macht endlich Schluss mit lästigen Drucker-Problemen

Besitzt er nun einen (wie bereits eingangs erwähnt)

GESTOCHEN SCHARFER GRAFIK-AUSDRUCK

24-Nadeldrucker, so erreicht er aber erst dann einen gestochen scharfen und zufriedenstellenden Ausdruck, wenn er den Drucker auch mit 24-Nadeldaten versorgt. Üblich sind 8-Nadeldaten, was bedeutet, daß nur jede 3. Nadel angesprochen wird, und der geplante Ausdruck recht matt wirkt. Funktion Nr. 5 ermöglicht nun eine Umwandlung der Steuercodes für den Grafikdruck in 24-Nadeldaten, so daß jedes Byte der Grafik verdreifacht wird und so praktisch einen gestochen scharfen Schwarz-Weiß-Ausdruck ergibt.

FAZIT:

Dieses "Allround"-Interface macht nahezu Schluß mit lästigen Problemen, die sich aus der Nutzung von Textverarbeitungs- oder Grafiksoftware ergibt, falls sie nicht mit einem komfortablen Druckertreiber ausgestattet ist, bzw. Ihr Drucker darin nicht angegeben ist. Davon ausgehend, daß als letzte (und wichtigste) Neuerung das überarbeitete "Printerface" nunmehr für die 8 kB-Version um DM 50,– reduziert (DM 248,–), für die 32-kb-Version um DM 80,– billiger (DM 298,–) angeboten wird, können wir es aufgrund seiner Leistungsfähigkeit und der ausführlichen Bedienungsanleitung jedem interessierten Leser, empfehlen, vor allem, wenn er sich beim Neukauf eines Druckers für ein Interface entscheiden soll. *(hb)*

WAS IST, WENN SIE IHR TEXT-PROGRAMM ÄNDERN?

Im Laufe der Zeit sind für den C64 immer leistungsfähigere Textprogramme auf dem Markt erschienen, so daß es eine ganze Reihe von C64-Benutzern geben dürfte, die inzwischen auch nicht mehr mit ihrem ersten Programm ihre Texte eintippen.

STATT UMSTÄNDLICHER ARBEIT EIN EINFACHES PROGRAMM NUTZEN

So erging es auch unserem Autor Werner Zempelin, der als früherer Textomat- und späterer Vizawrite-Fan jetzt auf SV-Text schwört. Was aber ist mit den bisherigen Texten, die man ja nicht alle neu eingeben will? Das Textprogramm SV-Text-Plus kennt zwar auch eine Option „Text übernehmen", aber hierbei müssen SEQ -Files vorliegen, sonst nimmt SV dieses File nicht an! Nun liegen die Files von Textomat als auch von Vizawrite nicht als sequentielle, sondern als Programm-Files vor. Was kann man tun, um aus PRG- irgendwie SEQ-Files zu machen?

TEXTPROGRAMM WIRD „ÜBERLISTET"

Zunächst wäre die Möglichkeit zu überprüfen, ob sich das Textprogramm nicht „überlisten" ließe. Wie man weiß, stehen in der Directory vor jedem Titel sowohl Spur- und Sektor-Nummer als auch die Art des Files in Form von Hex-Zahlen. Man könnte nun mit einem Disk-Monitor in die Directory gehen und dort durch Ändern der "82" = PRG in "81" = SEQ ein prinzipiell lesbares File erzeugen. Beim Vizawrite- und Textomat-File klappt dieses Verfahren allerdings nicht so einfach: Beim Einlesen eines derart geänderten Files stürzt SV-Text ab. Eine brauchbare Lösung erreicht man lediglich beim Vizawrite-File. Dort aber auch nur dann, wenn man den ersten Block dieses Textes ausläßst, d.h. überschlägt.

EIN DISK-MONITOR MUSS HER

Zu diesem Zweck geht man mit Hilfe des Disk-Monitors an den Beginn des Textes und schaut nach, wo der nächste Block beginnt. Diese Information steht immer in den ersten beiden Bytes der gerade auf dem Bildschirm befindlichen Seite. Diese beiden Bytes werden nun in der Directory beim Filenamen so geändert, daß das Textfile jetzt praktisch erst beim zweiten Block beginnt.

```
00ff 8211 0055 4d53 4554 5a50 5247 a0a0
a0a0 a0a0 a000 0000 0000 0000 0000 1100
0000 8111 0955 4d53 4554 5a50 5247 2f53
56a0 a0a0 a000 0000 0000 0000 0000 1100
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
track: 18       sector: 01       byte: 00
.?...umsetzprg          ..........f1:read
.....umsetzprg/sv       ..........f2:writ
.............................f3:next
.............................f4:send
.............................f5:bl+1
.............................f6:----
.............................f7:derr
.............................f8:quit
```

Wenn jetzt das File durch SV-Text übernommen werden soll, geschieht das völlig problemlos und ohne Textverlust, weil im ersten Block des Vizawrite-Textes nur allgemeine Informationen abgelegt werden, aber kein Text.

TEXT GEHT NICHT VERLOREN

Ungünstig, weil mühsam, ist allerdings die Tatsache, daß Umlaute und Absatzzeichen nicht mit denen von SV-Text übereinstimmen, so daß mit den Funktionen „Suchen" und „Ersetzen" die Umlaute vertauscht werden müssen. Zusätzlich muß der Text neu justiert werden. Aber die ganzen Prozeduren lohnen in jedem Fall, zumindest bei längeren Texten. Wesentlich einfacher geht die Angelegenheit vor sich, wenn man ein Programm benutzt, das für SV-Text zusätzlich **lieferbar ist:** Mit dem Umsetzprogramm kann man seine mit Vizawrite, Textomat (+), SM-Text oder Wordcraft geschriebenen Dokumente bequem in ein für SV-Text geeignetes Format „übersetzen" lassen, wobei gleichzeitig die Umlaute an die richtige Stelle verschoben werden. Bevor man mit dem Arbeiten beginnen kann, muß man dafür sorgen, daß die Filenamen auf der vorliegenden Textdiskette sich innerhalb der ersten 13 Zeichen unterscheiden, weil das Programm an den letzten drei Stellen zur besseren Unterscheidung immer "/sv" anhängt. Außerdem muß ca. noch einmal soviel Platz auf der Diskette sein, wie die Textfiles schon bisher beanspruchen. Wenn das Umsetzprogramm gestartet ist, meldet es sich mit einem Auswahlmenü, bei dem die Textprogramm-Version eingegeben werden muß. Nach dem Laden von notwendigen Hilfsdateien ist der Titel des Textfiles genau anzugeben. Hierbei darf, und das ist leider in der Anleitung nicht beschrieben, keiner der vom Programm vorgegebenen Punkte bei der Titeleingabe stehen bleiben. Evtl. überzählige Punkte müssen unbedingt durch die SPACE-Taste entfernt werden, da das Programm sonst nicht arbeiten kann und sich die Floppy mit "syntax error" meldet!

TEXTUMWANDLUNG PERFEKT

Der Text wird nun wieder als ein von SV-Text lesbares File ausgegeben und braucht nicht neu eingetippt zu werden.
Ein nützliches Programm, das seinen Preis von DM 48,– wegen der damit erreichten Zeitersparnis auf jeden Fall wert ist.

EPROM BRENNEN KINDER-LEICHT



Es gibt viele EPROM-Programmiergeräte. Meist ist die Arbeit hiermit mühselig und man hat den Eindruck, jede Speicherstelle einzeln „mit dem Lötkolben" einbrennen zu müssen. Anwenderfreundlich kann also nur ein System, bestehend aus Hardware (Programmiergerät) und Software sein, — und auch nur dann, wenn die einzelnen Komponenten optimal aufeinander abgestimmt sind. Die Fa. Roßmöller bietet hierzu eine hervorragende Lösung für den C64 mit dem klangvollen Namen PULSAR. Mit diesem System können alle EPROMs von 1 kByte bis 1 MByte der 25er und 27er Serien und die entsprechenden A-Typen gebrannt werden. Der Aufbau des Systems ist kinderleicht: Die Platine wird einfach auf den USER-Port des C64 gesteckt. Dieser ist durchgeschleift und somit weiterverwendbar. EPROMs können geladen, gespeichert und mit dem integrierten Maschinensprache-Monitor angeschaut und verändert werden. Auch das direkte Brennen von Files von Diskette wird unterstützt.

Die Platine macht einen stabilen Eindruck und ist sauber aufgebaut.
Was ein wenig stört, ist die „offene Bauweise". Sie ist sicherlich etwas für Elektronik-Freaks, aber ein Gehäuse würde die empfindlichen Bauteile eben doch besser schützen. Bei Beachtung einiger Sicherheitsregeln ist jedoch kaum mit Störungen zu rechnen: die Verarbeitung ist eben solide.
Es wird hier nicht auf die grundlegenden Begriffe der EPROM-Brennung und Handhabung eingegangen; beim folgenden Test des PULSAR soll die Arbeit mit einem solchen Gerät im allgemeinen und die mit dem PULSAR im speziellen beschrieben werden.

1.1 Installation

Zum Aufstecken der PULSAR-Karte sollte der Rechner ausgeschaltet sein. Nach dem Aufstecken auf den USER-Port kann der C64 eingeschaltet werden und die mitgelieferte Software von Diskette geladen werden.

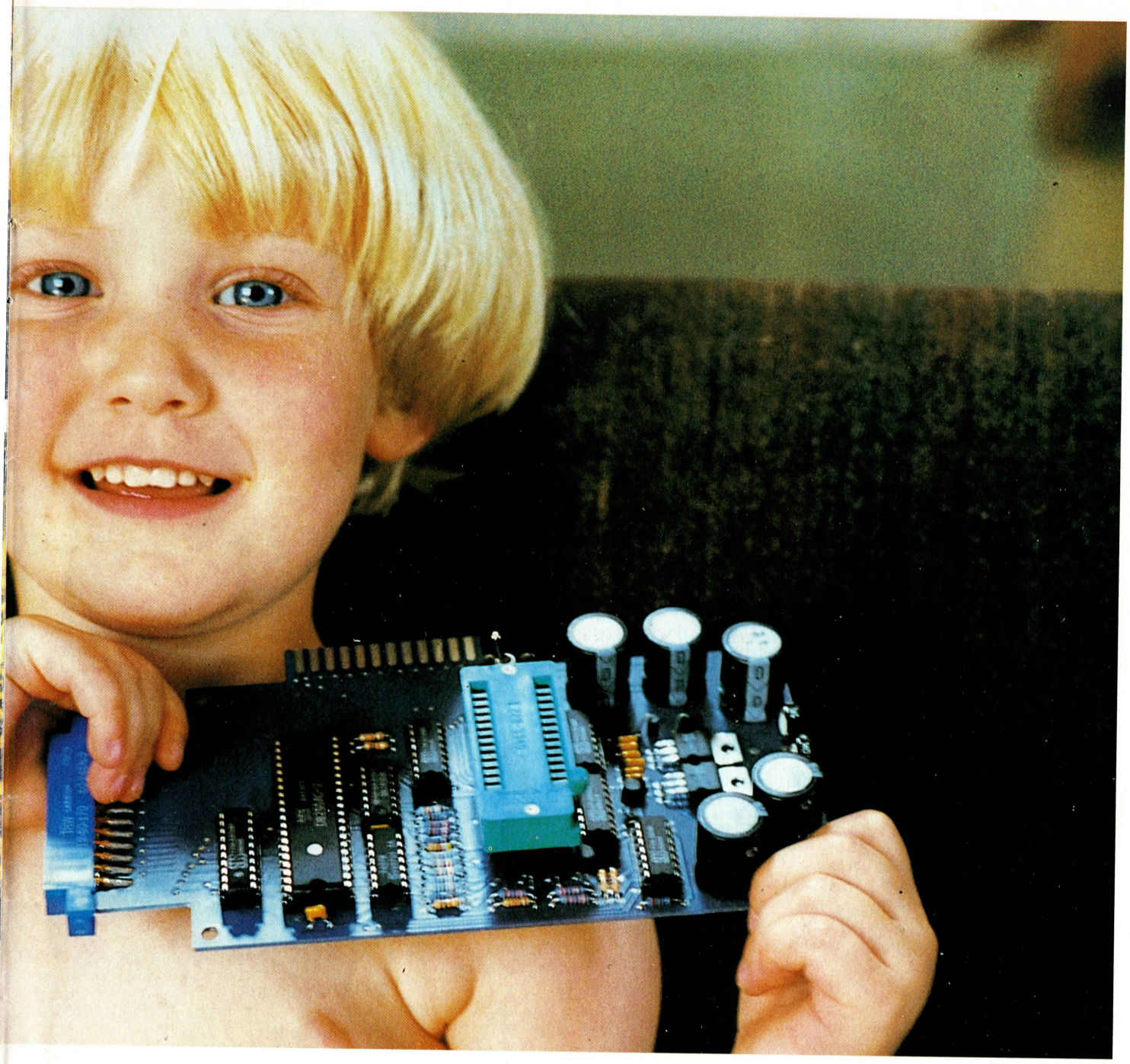
1.2. Inbetriebnahme

Nach dem Einschalten des C64 (C128) kann die PULSAR-Platine ebenfalls eingeschaltet werden.
LOAD"PULSAR*",8 liest das Steuerprogramm von Diskette.
Nach dem Programmstart (RUN) erscheint das Titelbild.
Durch Drücken der SPACE-Taste gelangt man in das Hauptmenü, wobei sich die PULSAR-Software in den Speicherbereich $A000 – $BFFF und der integriere Monitor nach §C000 verschoben hat, damit der untere Speicherbereich des C64 für den Anwender frei bleibt. Am C128 funktioniert PULSAR nur im 64er-Modus.
Nach dem Einlesen der Software und Drücken der Taste SPACE erscheint das Hauptmenü mit allen Eprom-Typen, die zur Programmierung durch PULSAR zur Auswahl stehen: alle EPROMs und 1 KByte bis 1 MByte der 25er und der 27er Serie, die mit einer Versorgungsspannung von +5V auskommen und über nicht mehr als 28 Pole verfügen.
EPROM-Auswahl:
Es ist zu beachten, ob es sich bei dem zu programmierenden EPROM um einen Standard-Typ oder einen A-Typ (niedrigere Brennspannung, siehe Aufdruck auf dem EPROM) handelt. Tip: Wenn man sich nicht sicher ist, welchen Typ man vor sich liegen hat, sollte man den entsprechenden A-Typ auswählen, denn es schadet den Standard-EPROMs nicht, mit einer zu niedrigen Brennspannung programmiert zu werden. Sollte beim Brennen ein Fehler auftreten, obwohl das EPROM korrekt gelöscht war, so handelt es sich wahrscheinlich um ein Standard-EPROM und man kann das gleiche EPROM noch einmal mit der Standard-EPROM-Auswahl überbrennen. Vorsicht: A-Typen würden hierbei zerstört werden.
Nach der Auswahl aus dem Hauptmenü zeigt PULSAR ein weiteres Spektrum seines Könnens: das eigentliche Arbeitsmenü (II) mit dem man laden, speichern, brennen, Module generieren, den Speicher mit dem integrierten Monitor anschauen und verändern kann.

2.0 Grundsätzliches zum Programmieren eines EPROMs mit PULSAR:

2.1 Adreßeingabe

PULSAR benötigt die Information, welcher Adressbereich des EPROMs programmiert werden soll und wo sich im Adreßbereich des Rechners das zu brennende Programm befin-



Pulsar ist nahezu „narrensicher", so daß jedes Kind eigentlich damit arbeiten kann.

det. Grundsätzlich gelten diese Adreßeingaben einschließlich der genannten Endadresse. Beim Abspeichern auf Diskette oder Cassette jedoch verlangt das Betriebssystem des C64 die Angabe der Endadresse +1!
Beispiel: Wenn also ein komplettes EPROM der Klasse 2764 (8 KByte) gebrannt werden soll, muß für die Anfangsadresse (im EPROM) „0" eingegeben und für die Endadresse $1FFF (hexadezimal) oder 8181 (dezimal).

2.2 Automatische Adressvorgabe

In der Regel schlägt das Programm Adressen vor, die mit RETURN übernommen werden können.

2.3 Individuelle Adressvorgabe

Durch einfache Neueingabe mit anschließendem RETURN kann die Adresse verändert werden. Die Eingabe kann wahlweise dezimal oder hexadezimal (mit vorangestellten $-Zeichen, 4stellig) erfolgen.

2.4 Einsetzen des EPROMs

Man sollte sich vorher davon überzeugen, daß die rote LED ausgeschaltet ist und nicht brennt. Solange sie leuchtet, liegt auch die Brennspannung an der Textoolfassung. Ein Einsetzen oder Herausnehmen des EPROMs in diesem Zustand könnte zur Zerstörung des EPROMS führen. Anschließend wird das EPROM

linksbündig in die Textoolfassung eingesetzt. Es ist darauf zu achten, daß die Kerbe nach rechts zeigt. Das EPROM ist nun brennbereit. Durch Starten des Programmes mit RUN beginnt der jeweilige Brenn- bzw. Programmiervorgang.

3.0 Die einzelnen Programmierbefehle

3.1 P = PULSAR programmiert aus dem Speicherinhalt des Rechners

Nach Anwählen des Arbeitsmenüpunktes P erscheint ein weiteres Menü:
L=Langsames Brennen
In diesem Modus brennt PULSAR auch die älteren und gebrauchten EPROMs mit großer Sicherheit.
S=Schnelles Brennen
PULSAR brennt in diesem Modus das EPROM mit schnellen Programmierrhythmen ungleich schneller. Da PULSAR im Gegensatz zu den meisten anderen EPROM-Programmiergeräten der Forderung der EPROM-Hersteller nachkommt, beim schnellen Programmieren die Versorgungsspannung auf 6 V anzuheben, ist auch diese Methode sehr sicher.
Außerdem wird nach dem Programmieren automatisch der Inhalt des EPROMs noch einmal mit dem Speicherinhalt des Rechners verglichen.

3.2. Adressauswahl

3.2.1 Startadresse EPROM

Hier wird nicht abgefragt, in welchem Adreßbereich des Computers dieses EPROM später seinen Dienst versehen wird, sondern ab welcher Adresse im EPROM selbst gebrannt werden soll.
So kann man z.B. in ein 16 KByte EPROM 2 verschiedene 8 KByte-Programme brennen, indem man beim ersten Programmiervorgang die ersten 8 KByte brennt mit der Startadresse 0 (die erste programmierbare Adresse eines EPROMs) und beim 2. Programmiervorgang die zweiten 8 KByte brennt, indem man die Startadresse im EPROM mit S2000 angibt.

3.2.2 Startadresse RAM

Um zu wissen, was in das EPROM gebrannt werden soll, muß PULSAR mitgeteilt werden, an welcher Adresse im Rechner sich dieses Programm im Moment befindet. Der Hersteller empfiehlt, die Programme, die gebrannt werden sollen, ab der Speicheradresse $1000 abzulegen, daher gibt PULSAR hier diesen Wert vor.

3.2.3 Endadresse für EPROM oder RAM

PULSAR benötigt bei diesem Programmpunkt den Hinweis, wie groß das zu brennende Programm ist. Durch Drücken der Taste RETURN übernimmt er sofort die Variante EPROM und gibt automatisch die höchstmögliche Adresse vor.
Bei der Auswahl des Punktes RAM benötigt PULSAR die Angabe der Endadresse des Programmes im Rechner.

3.3 Automatischer Leertest des EPROMs

Bevor nun PULSAR den Brennvorgang beginnt, wird überprüft, ob das zu brennende EPROM gelöscht ist. Sollte dies nicht der Fall sein, kann ein neues EPROM eingesetzt werden. (Achtung: Auf die rote LED achten!) Nach Beendigung des Brennvorganges gelangt man durch Drücken der Taste RETURN ins Hauptmenü zur neuen Anwahl.

4. S = EPROM programmieren „Sonderfälle"

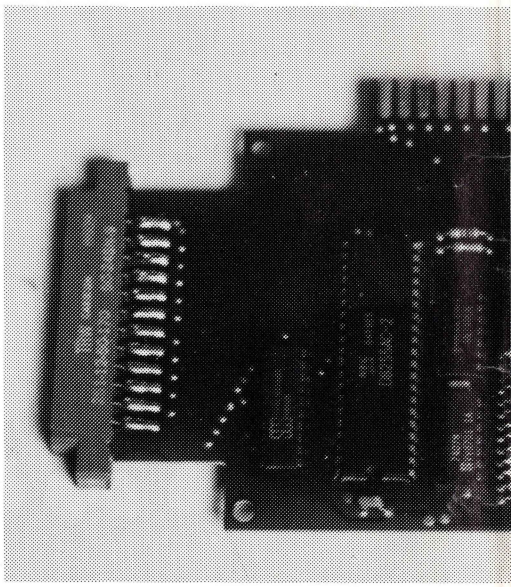
Neben den Standardsfunktionen bietet PULSAR besondere komfortable Möglichkeiten des EPROM-Programmierens. Mit dem Menüpunkt S können einzelne Bytes mit einem binären Editor programmiert werden. Auch gestattet es PULSAR, direkt von der Floppy zu programmieren. Dies ist z.B. bei einem 64 kByte EPROM sinnvoll, da der Inhalt eines solchen EPROMs nicht komplett in den Arbeitsspeicher des C64 geladen werden kann.

E=Einzelbyte
PULSAR bietet nun die Möglichkeit, einzelne Bytes des EPROMs zu brennen. Beispiel: Seriennummer zur EPROM-Erkennung.
PULSAR fragt nach der Startadresse des EPROMs, also nach dem Byte im EPROM, das verändert werden soll. PULSAR gibt zunächst 0 vor, als das erste Byte des EPROMs. Durch einfaches Eintippen der gewünschten Adresse kann diese Vorgabe geändert werden.
Anschließend liest PULSAR den Wert dieser Adresse aus dem EPROM und zeigt sie in der Zeile „Alter Wert" in binärer, hexadezimaler und dezimaler Form an.

F=liest Files direkt von der Diskette
Da der gesamte Inhalt eines 64 KBytes EPROMs nicht komplett in den Arbeitsspeicher geladen werden kann, hat PULSAR die Möglichkeit, direkt von der Diskette ein EPROM zu programmieren. Nach Auswahl des Programmpunktes F fragt PULSAR zunächst, welche Geräteadresse angesteuert werden soll. Hier kann also die entsprechende Device-Nummer angegeben werden, z.B. auch Nummer 1 für Cassette.

Fileauswahl durch F1
Möchte man nun ein File von Diskette laden, gibt man direkt den File-Namen ein oder drückt die F1-Taste des C64. Es erscheint der Name des ersten Files aus dem Directory der Diskette auf dem Monitor. Durch weiteres Drücken der F1-Taste blättert PULSAR die einzelnen Filenamen durch. So kann das gesamte Directory eingesehen werden. Nach Auswahl des gewünschten Files wird diese nach dem Drücken der RETURN-Taste geladen und das EPROM programmiert.

B=Anwählen des Modulgenerators
PULSAR verfügt über einen Modulgenerator, der aus einem Basicpro-



gramm ein Autostartmodulprogramm für den Expansionsport des C64 erzeugt.
Achtung: PULSAR brennt unter diesem Menüpunkt das Programm sofort in das gewählte EPROM.

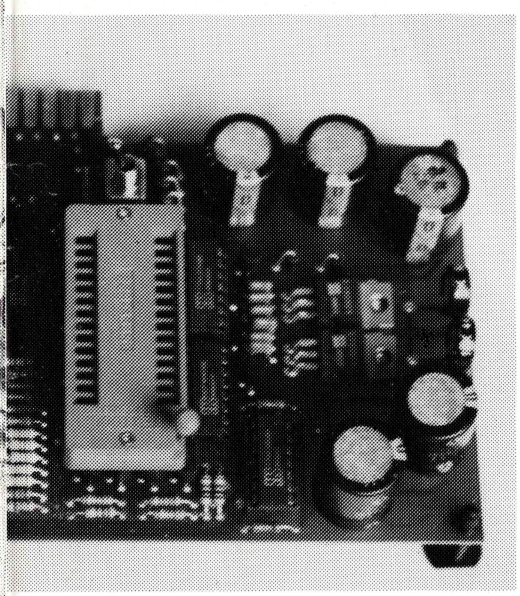
E=Einlesen
PULSAR liest den Inhalt des EPROMs in den Rechner mit automatischem Verify. (Richtigkeitsvergleich). Auf die Frage nach der Startadresse für den Rechner wird am besten $1000 angegeben.

V=Verifikation
Neben seinen automatischen Verifikationskontrollen führt PULSAR Richtigkeitsvergleiche durch. d.h., PULSAR vergleicht den Inhalt des gebrannten EPROMs mit dem Speicherinhalt des C64.

L=Leertest
PULSAR überprüft, ob das zu brennende EPROM leer ist.

B=Basic
Ein im Rechner ab der Speicheradresse $1000 befindliches Basic-Programm wird so verändert, daß es nach dem Brennen als Autostart-Programm am Expansionsport ($8000 – $9FFF) des C64 laufen kann. Dem Basic-Programm wird nämlich ein kleines Maschinenprogramm vorgesetzt, das nach dem Einschalten des Rechners automatisch gestartet wird, und das das eigentliche Basic-Programm an seine ursprüngliche Adresse, nämlich $0801, zurückschiebt und startet.

(–=EPROMauswahl
PULSAR geht zurück ins Hauptmenü zur Auswahl eines anderen EPROMs. Dieser Befehl ist eine permanente Funktion, d.h., in jedem Programmteil bewirkt das Drücken dieser Taste den Rücksprung ins vorangegangene Menü.



Für den Technikfreak sicher ein schöner Anblick, eine Verkleidung wäre jedoch besser.

M=Monitor
PULSAR hat neben seinen Programmierfunktionen noch einen leistungsfähigen Maschinensprachemonitor, der durch Drücken der Taste M angewählt werden kann.

D=Diskettenoperationen
Hier werden die Diskettenfunktionen angesteuert. Bei der Eingabe des Filenamens kann die F1-Taste wie beschrieben verwendet werden. Auch hier ist als Anfangsspeicheradresse günstigerweise $1000 anzugeben. Beim Abspeichern ist darauf zu achten, daß bei der Endadresse immer 1 hinzuaddiert werden muß.

X=Exit
Drücken der Taste X löst einen Reset aus. Die PULSAR-Treibersoftware kann dennoch mit SYS 52485 wieder gestartet werden.

Farbauswahl
Mit den Tasten F3, F5, F7 hat man die Möglichkeit, Rahmen, Hintergrundfarbe und Zeichenfarbe zu verändern.

6. Schlußbemerkung

Das hier beschriebene Programm zeigt deutlich, wie einfach und unkompliziert die Arbeit mit dem PULSAR von der Fa. Roßmöller ist. Die Menüs bieten alle Möglichkeiten, die bei der EPROM-Programmierung notwendig bzw. hilfreich sind. Das ganze System ist leicht zu handhaben. Das knapp 40seitige Handbuch ist gut aufgebaut und führt den Anwender gut durch das PULSAR-Programm und die Welt der EPROM-Brennerei. Alles in Allem: ein bemerkenswertes Gerät.

MONITORBEFEHLE:

\# Umrechnen einer dezimalen Zahl (<65536) in eine hexadezimale Zahl. Ist die dezimale Zahl kleiner 256, so wird sie zusätzlich auch in binärer Form angezeigt.
Beispiel: #223 #12345

S Umwandlung einer 4stelligen hexadezimalen Zahl in eine dezimale; oder einer zweistelligen in eine dezimale und binäre Zahl.
Beispiel: $CD $FFD2

% Umrechnen einer 8stelligen Binärzahl in eine hexadezimale Zahl.
Beispiel: &01011010

? Addition oder Subtraktion zweier 4stelliger Hexadezimalzahlen.
Beispiel: ?FFD2+1111
?ABCD-0EF0

= Beispiel: =1000 F000 vergleicht den Speicher ab der Adresse S1000 mit dem Speicher ab SF000. Eine Ungleichheit wird in Form einer 4stelligen hexadezimalen Adresse angezeigt. (*1)

D Disassemblieren eines Speicherbereichs. Durch Überschreiben des Mnemonics kann ein Maschinenprogramm verändert werden (Direkt-Assembler-Funktion), hierbei muß jedoch das Argument (Adresse, Wert o.ä.) hexadezimal ohne „$"-Zeichen eingegeben werden.
Beispiel: DABCD DE000 E020 (*1,*2)

F Durchsucht den Speicher nach einer maximal 9 Byte langen Zeichenkette in dem angegebenen Speicherbereich.
Beispiel F A9 00 8D 00 C0, 2000 4000 (*1,*2)

G Starten eines Maschinenprogrammes ab der angegebenen Adresse, oder – wenn keine Adresse eingegeben wird – ab der in der Registeranzeige unter PC stehenden Adresse.
Beispiel: G GC000

KA Ausgabe des angegebenen Speicherbereichs in Form von ASCII-Code-Zeichen.
Beispiel: KA1FD2 (*1,*2)

KB Wie KA, jedoch Ausgabe im Bildschirm-Code (*1,*2)

M Ausgabe des angegebenen Speicherbereichs in Form von 8 hexadezimaler 2stelliger Zahlen und dem entsprechenden ASCII-Werten.
Beispiel: MA000 A040 (*1,*2)

O Beispiel: 02000 4000,AA belegt (OCCUPY) den Speicherbereich von $2000 bis $4000 mit dem Wert $AA.

V Beispiel: VE000 FFFF 2000 verschiebt den Speicherinhalt von $E000 – $FFFF nach $2000.

R Zeigt die Register und Flags der 6510 CPU an, mit folgenden Abkürzungen in der Darstellung:

PC	=	Program Counter
SR	=	Status Register
AC	=	Akkumulator
XR	=	X-Register
YR	=	Y-Register
SP	=	Stack-Pointer
N	=	Negativ-Flag
V	=	Overflow-Flag
–	=	nicht benutzt
B	=	Break-Flag
D	=	Dezimal-Flag
I	=	Interrupt-Flag
Z	=	Zero-Flag
C	=	Carry-Flag

X EXIT; beenden des Monitor-Programmes und Rücksprung ins PULSAR-Menü.

*1 bedeutet, daß die Anzeige mit der Taste RUN/STOP abgebrochen werden kann; daß die Bildschirmausgabe durch Drücken einer beliebigen Taste angehalten und wieder fortgesetzt werden kann; daß die Bildschirmausgabe durch Festhalten der SPACE-Taste verlangsamt werden kann.

*2 kennzeichnet Monitorfunktionen, bei denen mit Hilfe des Full-Screen-Editors des C64 (Cursortasten etc..) angezeigte Werte verändert werden können.

CW-SPECIAL

COMMODORE WELT

Sonderheft
Nr. 5/87
DM 14,80
ÖS 124
SFR 14,80

128 SPECIAL

Spiele im Test: Zu Land-zu Wasser- im Weltraum

Lernen Sie Ihren Commodore kennen

Service Tips-Tricks-Kauf-beratung

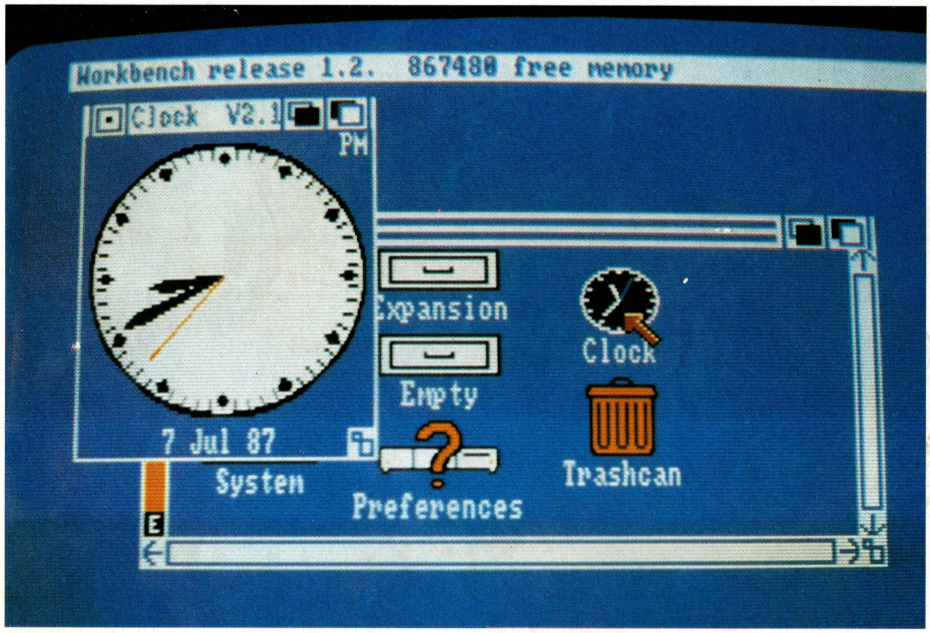
Super-Listings für Ihren 128 PC

Das grosse 128er Special jetzt an jedem gutem Kiosk!

HEFT IM HEFT: Das PC-Magazin CA!

SUPER: 512 KILOBYTE FÜR AMIGA 500

Noch vor einigen Jahren galten 512 KByte bei einem Homecomputer als unerreichbar. Kaum sind jedoch die ersten Geräte da, die diesen Speicherplatz bereitstellen, zeigt sich auch schon, daß die Grenzen sehr eng sind. Die Rede ist natürlich vom Amiga 500, der seinen Siegeszug als Heimcomputer angetreten hat. Wir testeten für den „Kleinsten" in der Amiga-Familie die 512 KByte-Erweiterung mit eingebauter Echtzeituhr von CAS-Computer aus Offenbach.



Nicht nur eine Speicher-Erweiterung, auch noch andere kleine Annehmlichkeiten, wie eine akkugepufferte Uhr, werden mitgeliefert.

Insbesondere die Grafik- und Soundfähigkeiten des Rechners beanspruchen den verfügbaren RAM-Platz. Mangelerscheinungen treten nicht nur bei den Basic-Programmierern auf, die jedes Window in der höchsten Auflösung mehrfarbig gestalten wollen, sondern auch die Profis haben ihre Schwierigkeiten. Der „Aegis Animator" und „Deluxe Music" laufen mit den standardmäßigen 512 KByte nur sehr unzulänglich und fordern vom Amiga den ganzen Einsatz in Form eines Mega-Bytes.
Momentan sind auf dem Markt zwei verschiedene Speichererweiterungen erhältlich, die um die Gunst des Amiga-Users buhlen. Es handelt sich dabei um die Originalkarte von Commodore und um jene Erweiterung von CAS, um die es in diesem Bericht geht. Beide sind für je 298,– DM zu haben, wobei zum Zeitpunkt dieses Berichtes das Testobjekt etwas schneller lieferbar war. Die Leistungsbeschreibung gleicht jener von Commodore aufs Haar. Geboten werden 512 KByte Aufrüstung zusammen mit einer akkuversorgten Echtzeituhr. Auf einer sauber verarbeiteten Platine befinden sich 16 RAM-Bausteine zu je 32 KByte. Die Zugriffszeit von 150 Nano-Sekunden dürfte nur von theoretischem Interesse sein. In der Praxis war nach dem Einbau der Karte keinerlei Verzögerung festzustellen.

KAUM PLATZ IM ERWEITERUNGSSCHACHT

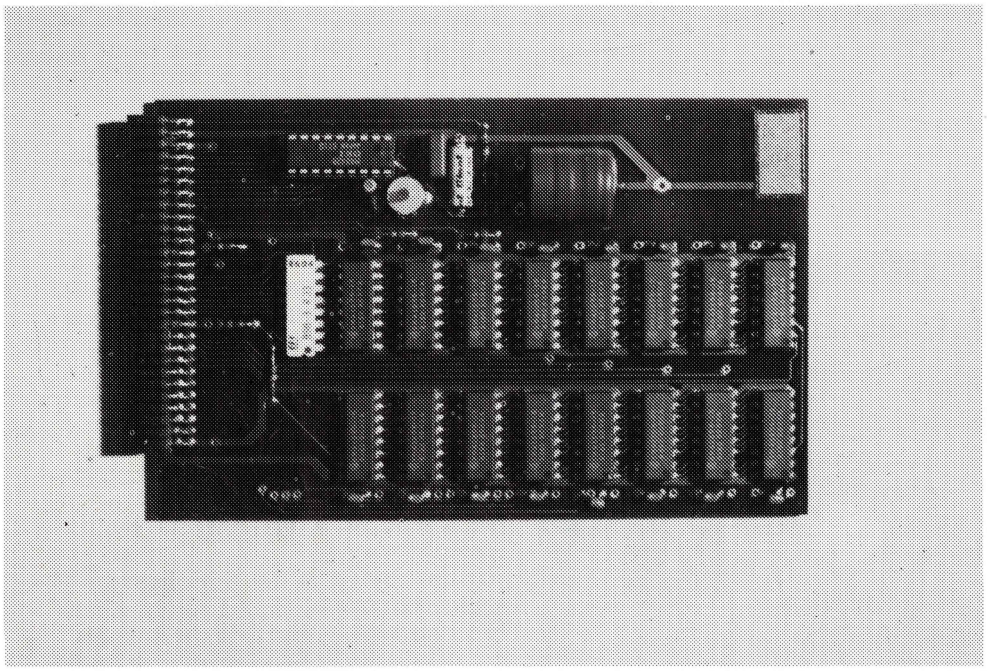
Einige Probleme ergeben sich jedoch aus den herrschenden Platzverhältnissen. Die Chips sind nicht gesockelt. Dies mag neben der üblichen Sparsamkeit noch daran liegen, daß die Karte andernfalls nicht mehr in den Schacht des Amiga 500 passen könnte. Hier geht es nämlich recht eng zu. Während die Commodore-Karte noch in einem stabilen Blechgehäuse eingelötet ist, muß bei der CAS-Karte auch darauf verzichtet werden. Betriebssicherheit wird nur durch die dünne Plastikfolie zwischen Amiga-Platine und Erweiterung garantiert. Die maximale Bauhöhe wird schließlich vom Akku der Echtzeituhr bestimmt. Beim Einschieben der Karte sollte man entsprechend vorsichtig sein.
Ansonsten bietet der 56polige Anschluß genügend Halt. Die eingebaute Erweiterung dürfte sicherlich jeden Transport überstehen, den auch der Amiga ohne Schaden „überlebt". Der Einbau ist denkbar einfach und selbstverständlich ohne Werkzeug durchzuführen. Die Bodenklappe des Amiga wird geöffnet und die Erweiterungskarte seitlich eingeschoben. Der Schacht sollte dann mit etwas Feingefühl wieder geschlossen werden, um den Akku nicht zu beschädigen. Nach dem Anschluß des Rechners meldet sich die Workbench-Zeile mit dem neuen, erweiterten Speicherplatz.

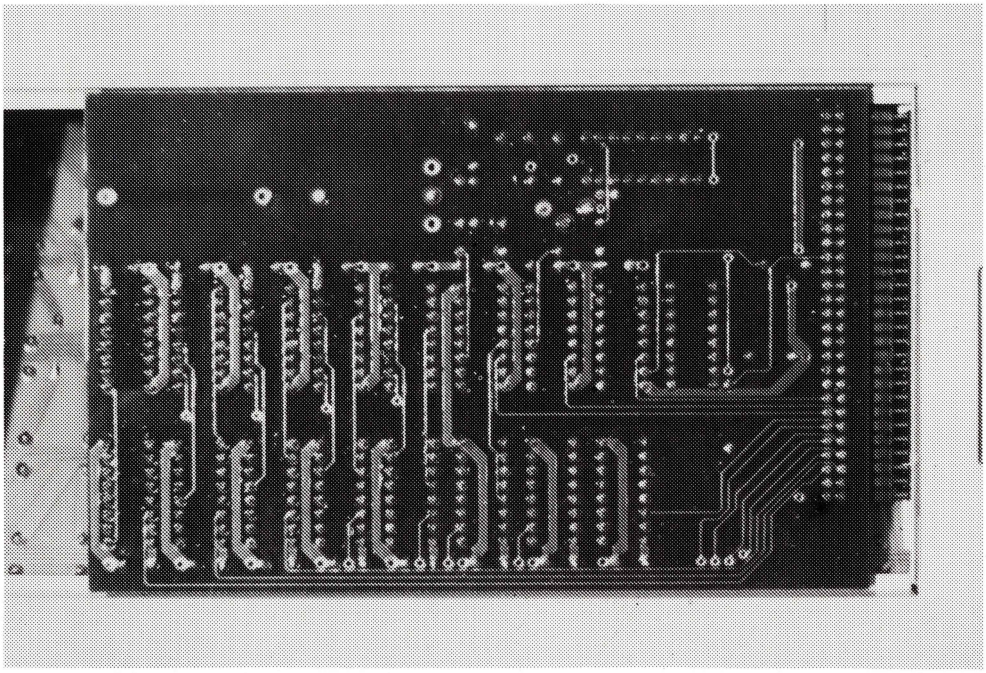
KEIN INSTALLATIONS-HANDBUCH

Die Commodore Welt besaß keinerlei Unterlagen zur Installation. Sollte dies auch beim Einzelkunden der Fall sein, dann beginnen nun einige Schwierigkeiten. Natürlich kann man die Uhr mit dem Preference-Utility einstellen und für den aktuellen Gebrauch nutzen. Schaltet man den Computer jedoch wieder aus, dann ist das Datum verloren. Zwar ist in der Regel die Startup-Sequenz im S-Directory bereits mit der Abfrage der Echtzeit-Uhr ausgestattet, doch diese muß zuerst einmal mit

SPEICHER PLATZ SATT

SETCLOCK OPT SAVE (im CLI) eingestellt werden. Wohlgemerkt, nachdem durch Preference die Zeit eingestellt wurde.
Da mittlerweile eine fast unüberschaubare Anzahl von Workbench-Versionen im Umlauf sind, muß unter Umständen doch noch in der Startup-Sequence der Befehl SETCLOCK OPT LOAD eingefügt werden. Zum anderen darf man nunmehr eine RAM-Disc anlegen und gleich mit Setup aufrufen. Da sich jedoch auch Commodore mit einem einfachen DIN A4-Blatt begnügt, ist es sicherlich möglich, daß sich CAS doch noch zu etwas ähnlichem durchringt.
Mit den nun vorhandenen 1024 Kilobyte läßt es sich vorzüglich arbeiten. Die RAM-Disc läßt sich mit Programmen vollpropfen, die dann in Windeseile gestartet werden. In diesem Falle empfiehlt sich übrigens die Anschaffung eines Utility-Paketes, welches eine Reset-feste RAM-Disk installiert. Dies heißt, daß die Daten, welche in der RAM-Disk abgelegt wurden, auch nach einem Systemabsturz oder Reset beim Booten der Workbench immer noch vorhanden sind.
Der neugewonnene Speicherplatz nutzt besonders bei Arbeiten jedweder Art: Seien es digitalisierte Bilder oder Sounds, beides braucht jeweils immensen Speicherplatz.
Auch bei Grafiken, die mit Malprogrammen erstellt werden, wird in der höchsten Auflösung des Amigas der Speicher knapp. Und nicht zuletzt freuen sich auch die Vielschreiber, wenn in der Textverarbeitung reichlich Speicher zur Texteingabe vorhanden ist. Für all diese Anwender bietet CAS die richtige Lösung.
Die Echtzeituhr stellt bei dieser Erweiterung quasi eine willkommene Dreingabe dar, sie dient neben der recht nützlichen aktuellen Uhrzeit bei der täglichen Arbeit vor allem der Übersicht für erstellte Files.
Bei diesen nämlich wird immer Uhrzeit und Datum der Speicherung mit abgelegt, ein Vorteil, den der Anwender bald nicht mehr missen will.





Achtung beim Einbau der Platine. Es geht etwas knapp zu. Die Verarbeitung ist, das zeigen unsere Bilder, ohne Fehl und Tadel.

FAZIT: KEINE PROBLEME IM GEBRAUCH

Beim Einsatz der Karte tauchten keine weiteren Probleme auf. Sämtliche speicherkritischen Programme, die uns bekannt waren, ließen sich nutzen. Selbst einige Problemfälle, etwa Marble Madness, akzeptierten die neuen Chips ohne Schwierigkeiten. Welche Karte sich der Amiga-Besitzer zulegen sollte, bleibt damit offen. Die Commodore- und die CAS-Karte unterscheiden sich weder im Preis noch im Gebrauch, so daß dem Anwender weiterhin die Qual der Wahl bleibt. Bleibt abzuwarten, wann einer der beiden den Preis senkt.
Einziger Nachteil bleibt das nichtvorhandene Handbuch. Ist es so undenkbar, daß auch Anfänger ohne Hintergrundwissen eine Speichererweiterung kaufen?

SARACEN

Fortsetzung von Seite 6

ner nicht gerade als einfallsreich zu bezeichnenden Melodie, die während des Spiels ununterbrochen vor sich hingedudelt wird. Da bleibt der Griff zum Lautstärkeregler nicht lange aus. Auf effektvolle Hintergrundgeräusche, die gerade bei einem solchen Spiel den Spaß um einiges erhöht hätten, haben die Programmierer vollständig verzichtet. Das Spiel kann, wie oben schon erwähnt, sowohl über die Tastatur als auch über Joystick gesteuert werden. Hierbei funktionieren beide Methoden gleich präzise und schnell, wobei es auf Präzision bei einer derartigen Animation weniger ankommt.

Der Spielwitz läßt im Gegensatz zu „Boulderdash" schnell nach. Natürlich begegnet man neuen Elementen, die den Spielwitz aber kaum erhöhen. Trotzdem konnte unser Spieletester nicht länger als bis zum zweiten Level vor dem Bildschirm gehalten werden.

FAZIT

Saracen ist sehr stark „Boulderdash" nachempfunden, erreicht aber dessen Qualität bei weitem nicht. In Sachen Hintergrund und Spriteanimation wurde leider nicht mit sehr viel Mühe gearbeitet. Besonders in der Motivation kann das Spiel sein Vorbild nicht erreichen. Aus dieser Spielidee hätte man sehr viel mehr machen können. Wer sich Saracen ansehen will, der kann es bei Datasoft für ca. 40 DM für Diskette beziehen.

QUARTET- VIER GEGEN ALLE

Der Trend läßt nicht nach! Es folgt Spielhallenadaption auf Spielhallenadaption. Ein Spiel fügt sich nahtlos in die Reihe seiner Vorgänger ein. Halt! Nicht ganz, denn ein Punkt scheint nachzulassen, und das ist die Qualität der Adaption. So auch bei dem neuesten SEGA-Spiel namens Quartet.
Dieses Spiel braucht keine komplizierte Hintergrundgeschichte, sondern begnügt sich kurz und schmerzlos mit drei Sätzen: Die Weltraumkolonie „06" wurde von bösen Terroristen angegriffen. Es gibt für sie nur eine Rettung, und die heißt Quartet. Nur diese draufgängerischen Vier können dem Unheil die Stirn bieten! Das war die ganze Geschichte des Spiels. Dessen Ziel haben Sie sicher schon erahnt: Mit Ihrer Spielfigur durch die Gegend rennen und alles über den Haufen schießen, was einem begegnet. Kurz gesagt: ballern, ballern und nochmal ballern.
Nun zum Spiel selbst. Während das Spiel lädt, wird der Spieler von einem bemerkenswert schlechten Titelbild erfreut. Durch dieses Bild erhält man einen kleinen Vorgenuß des eigentlichen Spiels, dessen Qualität die des Titelbildes noch bei weitem unterbietet.
Ist das Programm endlich geladen, was seine Zeit dauert, da kein Turbolader eingebaut wurde, so hat man die Möglichkeit, sich von vier Spielfiguren, die alle unterschiedliche Fähigkeiten haben, eine rauszusuchen. Ist das geschehen, so wird der Spielscreen eingeblendet. Und hier geht's erst richtig los.
Im Spiel gibt es einige Objekte, die der eigenen Spielfigur sehr dienlich sein können. So kommt zum Beispiel ein Düsenrucksack vor, mit dem man nicht mehr laufen muß, sondern durch die Lüfte düsen kann. Auch Punktebonusse und bessere Waffen können dadurch erlangt werden.
Die Grafik von Quartet ist allerdings eine ausgewachsene Katastrophe. Bewegt man den Joystick, so beginnt ein Sprite durch die Gegend zu laufen, das nicht kleiner sein könnte. Die Sprites sind für ihre Größe so plump, daß man Mühe hat, die vier verschiedenen Figuren voneinander zu unterscheiden. Das Beste ist aber deren Animation! Das Spielersprite geht nicht etwa durch die Gegend, nein, es rennt, und wie! Die Beine bewegen sich dabei dermaßen schnell, daß man sie fast nicht mehr sehen kann, was beim Tester und bei einigen anderen Redakteuren einen Lachkrampf auslöste. Gleichzeitig bewegen sich die Figuren aber sehr langsam. Normalerweise ist jeder halbwegs begabte Besitzer eines C 64/128 in der Lage, eine bessere Animation zu programmieren. Die anderen (feindlichen) Sprites sind zwar besser gelungen, aber deshalb noch lange nicht gut. Am Ende jedes Levels hat der Spieler gegen ein unidentifizierbares Ungetüm zu kämpfen. Es ist noch nicht klar, ob dieses Sprite beabsichtigt ist, oder ob hier ein Spritefehler vorliegt. Bemerkenswert sind noch die Geschosse der eigenen Waffen, die mehr einem Tennisball ähnlich sehen als Laserstrahlen.
Die Hintergrundgrafik ist zwar grafisch nicht schlecht, aber unendlich eintönig. Ständig scrollt ein langweiliger grauer oder brauner Hintergrund an der Spielfigur vorbei, an dem man sich auch bald sattgesehen hat.
Das Beste an diesem Spiel ist der Sound. Während des Spiels tönt ein recht gut programmiertes Musikstück aus dem Lautsprecher, während von anderen Hintergrundgeräuschen nichts zu hören ist.
Das Spiel kann von einem oder zwei Spielern bestritten werden. Dabei können beide Spieler ausnahmsweise gleichzeitig am Bildschirm umherwandern und Feinde vernichten. Der Spielwitz wird dadurch aber keineswegs gesteigert.
Quartet kann mit dem Jostick oder der Tastatur gespielt werden, wobei sich eins vom anderen nicht unterscheidet. Beide Steuerungen sind gleich präzise und schnell. Im 2-Spieler-Modus müssen allerdings zwei Joysticks verwendet werden.
Die Spielmotivation von Quartet ist erwartungsgemäß gering. Die stupide Ballerei fällt bald auf die Nerven. Da kann auch der 2-Spieler-Modus und die relativ gute Musik nicht mehr helfen.

FAZIT

Die Qualität von Quartet entspricht ungefähr dem Stand von 1982. Durch die schlechte Animation der Sprites, den langweiligen Hintergrund und die stupide Spielidee flaut das Interesse an dem Spiel sehr bald ab. Der Hammer ist allerdings der Preis. Dieses für unsere Zeit relativ unausgereifte Spiel wird zu einem enorm hohen Preis von 59,95 DM für Diskette verkauft. Selbst für ein Billigspiel ist die Qualität von Quartet aber zu niedrig. Wer's sich trotz allem einmal ansehen will, der kann Quartet für den oben genannten Preis bei Ariolasoft beziehen. *(mn)*

VERDIENEN SIE GELD MIT IHREM COMPUTER!

Haben Sie einen Commodore VC 20 oder C 64? Einen 16/116, Plus 4? Oder einen 128? Können Sie programmieren? In Basic oder Maschinensprache? Dann bietet COMMODORE-WELT Ihnen die Möglichkeit, mit diesem Hobby Geld zu verdienen!

Wie? Ganz einfach. Sie senden uns die Programme, die Sie für einen Abdruck als geeignet halten, zusammen mit einer Kurzbeschreibung, aus der auch die verwendete Hardware – eventuelle Erweiterungen – benutzte Peripherie – hervorgehen muß (Schauen Sie sich dazu den Kopf unserer Programmlistings an.)

Benötigt werden: Zwei Listings des Programms sowie eine Datenkassette oder Diskette! Wenn die Redaktion sich überzeugt hat, daß dieses Programm läuft und sich zum Abdruck eignet, zahlen wir Ihnen pro Programm je nach Umfang bis zu DM 300,–!

Sollten Sie keinen Drucker haben, genügt der Datenträger.

Sie erhalten Ihre Kassette/Diskette selbstverständlich zurück, wenn Sie einen ausreichend frankierten Rückumschlag mit Ihrer Adresse beifügen.

Bei der Einsendung müssen Sie mit Ihrer Unterschrift garantieren, daß Sie der alleinige Inhaber der Urheber-Rechte sind! Benutzen Sie bitte anhängendes Formular! (Wir weisen darauf hin, daß auch die Redaktion amerikanische und englische Fachzeitschriften liest und „umgestaltete" Programme ziemlich schnell erkennt).

Um Ihnen die Arbeit zu erleichtern, finden Sie hier ein Formular. Sie können es ausschneiden oder fotokopieren.

Name des Einsenders: _____

Straße/Hausnr./Tel.: _____

Plz/Ort: _____

Hiermit biete ich Ihnen zum Abdruck folgende(s) Programm(e) an:

_____

_____

_____

_____

Benötigte Geräte: _____

_____

Beigefügt ( ) Listings ( ) Kassette ( ) Diskette

Ich versichere, der alleinige Urheber des Programmes zu sein!
Hiermit ermächtige ich die Redaktion, dieses Programm abzudrucken und wirtschaftlich zu verwerten. Sollte es in den Kassetten-Service aufgenommen werden, erhalte ich auch dafür eine entsprechende Vergütung, das Copyright geht insoweit auf den Verlag über.

_____

Rechtsverbindliche Unterschrift

COMMODORE WELT
PROGRAMM-REDAKTION
POSTFACH 1161
D-8044 UNTERSCHLEISSHEIM