

CK

Die User Zeitung

mit Sonderteil für Commodore,

Atari, Sinclair, TI 99/4A

und Schneider

Nr. 10/11 3. Jahrgang

Computer Kontakt

Sinclair

- ★ Spectrum: Spielepokes
- ★ ZX-81: Kopierschutz
- ★ QL-Special

Commodore

- ★ Grafik-Kurs: VIC-II
- ★ Mailbox, Marke Eigenbau
- ★ Spiel: Eastern Fight
- ★ Amiga: Pro & Contra

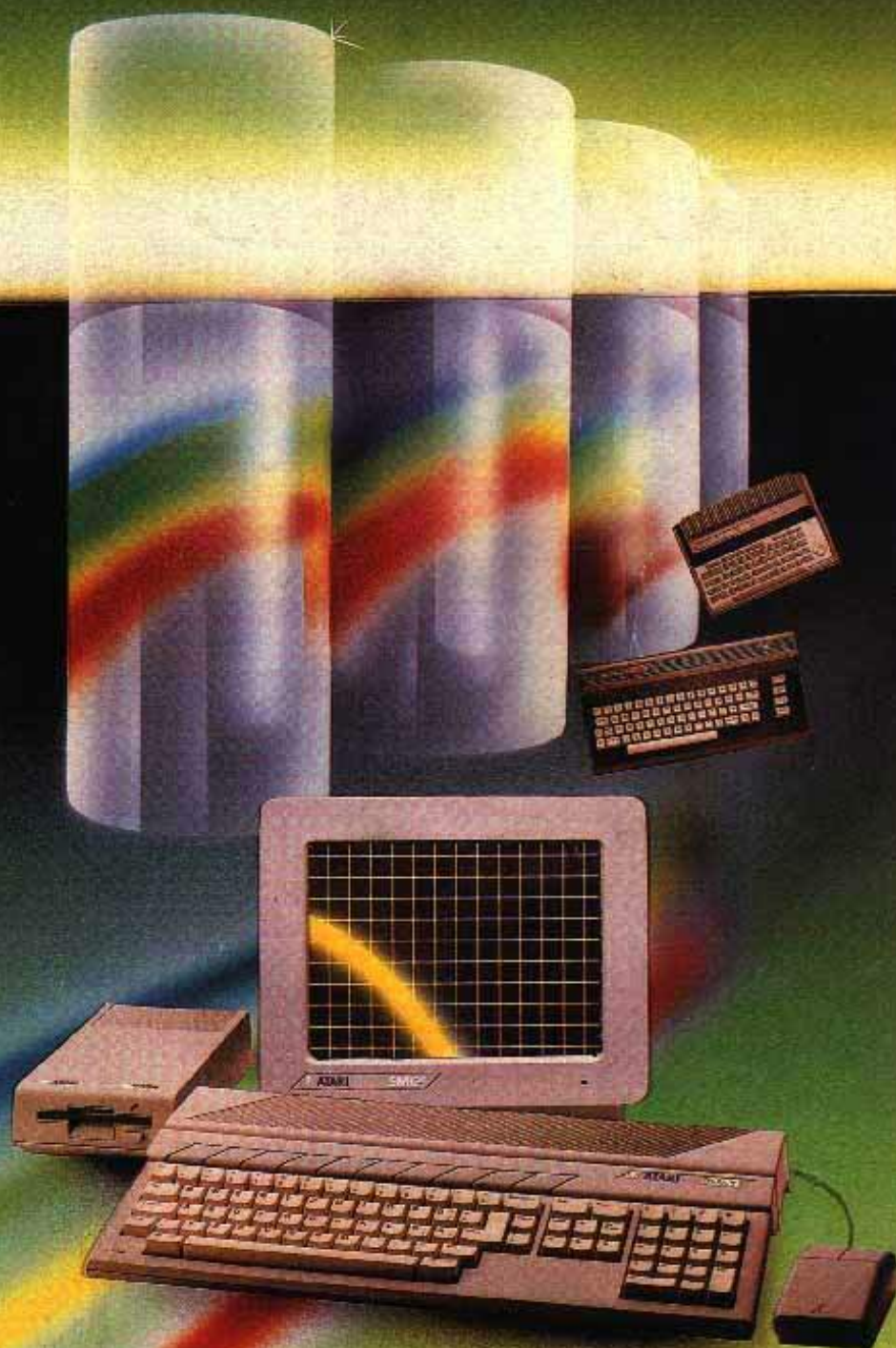
Atari

- ★ Topprogramm:
Micro Mon - MS-Monitor
- ★ Spiele:
Hungrier Guff, Puzzler
- ★ Maschinensprachekurs
- ★ ST: GFA-Basic

TI 99/4A

- ★ TI-Writer-Zwilling
- ★ Monopoly 2. Teil
- ★ Buchstabenmix

Atari ST
Sonderseiten



ATARI ST

ASSEMBLER-PRAXIS AUF ATARI ST

Roland Löhrl

...ein Altmeister der Assembleranwendung, Herausgeber des Mikrocomputer Magazins MICRO MAG, veröffentlicht bei te-wi seine souveräne Darstellung der Assemblerprogrammierung auf ATARI STs.

Erklärt Grundlagen:

Begriffe und Werkzeuge der Assemblerprogrammierung...erforderliche Systemkenntnisse...systembezogene Erläuterung der 68000er Befehlsfunktionen.

Zeigt Anwendungen:

Hantieren mit Assemblern: Aufruf von Assemblern; Steuern ihrer Optionen über Direktiven; Stellungnahme zu realen ATARI-ST-Assemblern.

Arbeiten in der ATARI-ST-Programmierungsumgebung: Textprogramme zur Programmentwicklung; ein Editor; ein Parser; das Betriebssystem; BIOS-Funktionen; BIOS-Toolbox; GEMDOS Toolkit; das erweiterte XBIOS.

Anwenden des Befehlssatzes in Musterprogrammen für: E/A-Routinen, Rekursionen, dez/bin Rechenarten, Stackverwaltung, Adressverwaltung, Entscheidungen, Schleifenkonstrukte, Unterprogramme, nummerierte Traps, Bedienen von Interfacebausteinen, Texterkennung, Textverarbeitung, Tastaturdekodierung, memory dumps, Floppy-Tests/Funktionen, serielle RS232-Datenübertragung usw.

Entwickelt Hilfsprogramme:

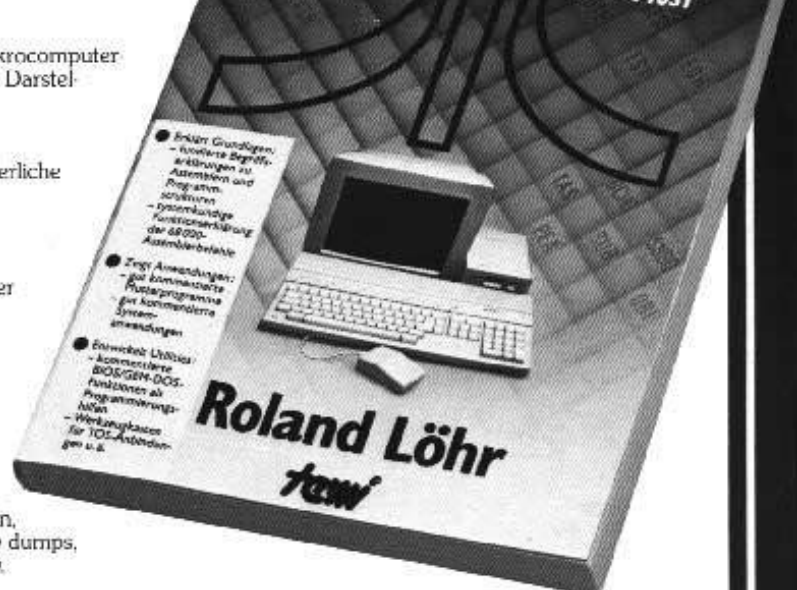
BIOS-Toolbox; GEMDOS-Toolkits; ein Editor; ein Parser; Arbeiten mit Toolkits. Die Programme des Buchs sind auf Diskette vom Autor erhältlich.

Ein Fachtext in klarer Sprache mit leserfreundlichem Druckbild, guter Bildokumentation und umfangreichen Listings von Musterprogrammen (auf Diskette beim Autor erhältlich).

ca. 300 Seiten, Softcover, DM 59,-

ASSEMBLER-PRAXIS AUF ATARI ST

ATARI 260ST, ATARI 520ST, ATARI 1040ST



te-wi Verlag GmbH
Theo-Prosel-Weg 1
8000 München 40

Weitere te-wi-Bücher



NEU

DAS „C“-BUCH

(Herold/Unger)
Ein „C“-Kurs der Industrie. Für sämtliche C-Konstrukte. Über 100 Beispiele. Anspruchsvoll in Text/Bildmaterial, ca. 500 Seiten, Softcover, DM 79,-

UNIX

(Yates/Thomas) US-Standardwerk der UNIX-Promoter Yates. Eine sachkundige Übersicht und Einführung in die Anwendung, 550 Seiten, Softcover, DM 79,-



LOGO - Jeder kann programmieren

(Daniel Watt)
Buch des Jahres in den USA.
Best-rezensiert von Pädagogen und deutschen Kultusministerien. Ein bildreicher Führer durch u.a. ATARI's LOGO. Von Papert's Schüler D. Watt. 384 Seiten, A4, DM 59,-



M68000 FAMILIE, 2 Bd.

Hill/Nausch, ges. 968 Seiten
Einzige Motorola-authentische Darstellung von CPU-68000-Architektur, Programmierung, Systemaufbauten. Behandelt alle 68000-Bausteine sowie 68020, 68881. Bd 1, Grundlagen + Architektur, 568 Seiten, DM 79,-
Bd 2, Anwendung und Bausteine, 400 Seiten, DM 69,-



UMWELTDYNAMIK

30 Programme für kybernetische Umwelterfahrungen auf allen BASIC Rechnern. Das Buch enthält beides: Ein Programmsystem zur Simulation eigener Problemformulierungen und 29 kommentierte Modellbeispiele wie Baumsterben, Heizungsbedarf, Nahrungsketten usw. Prospekt anfordern. Von Hartmut Bossel, 480 Seiten, Softcover, DM 59,-



Mein ATARI Computer

Best-rezensiertes Standardwerk deutscher ATARI User Groups. Kompakte ATARI 100-/800-System/Peripheriebeschreibung. Von Poole/McNiff/Cook, 560 Seiten, Softcover, DM 59,-

Sprühende Ideen mit ATARI-GRAPHIK
Fröhlicher Lehrstoff in Geometrie und Farbenlehre eines amerikanischen Lehrers mit ATARI-Graphikmöglichkeiten. Von Tom Rowley, 224 Seiten, Softcover, DM 49,-



6502 - Programmieren in Assembler
Dieses Buch behandelt ausführlich die Assemblersprachen-Programmierung für den weitverbreiteten Mikroprozessor 6502. Lance Leventhal, 704 Seiten, Softcover, DM 59,-

Noch im Programm: Einführung in die Mikrocomputer-Technik, DM 66,-
Computer für Kinder, ATARI, DM 29,80



Liebe Leser,

diesmal finden Sie gleich auf der nächsten Seite unseren kleinen Fragebogen. Hier wollen wir wieder mal wissen, wer die CK liest. Um ein wirklich repräsentatives Ergebnis zu erhalten, sollten möglichst viele Leser mitmachen. Und zu gewinnen gibt es natürlich

auch wieder einige wertvolle Preise. Also, sofort ausfüllen und einsenden.

Ein kleiner Fehler hat sich bei unserem letzten Logical eingeschlichen. Überall wo noch "Stein" steht, hätte es "Immel" heißen müssen. Wir werden aber alle eingeschickten Lösungen berücksichtigen, da es ja keine richtige Lösung gegeben hat. So hat jeder die gleichen Chancen, einen Preis zu gewinnen.

Nicht viel Neues können wir den Sinclair-Fans berichten, obwohl sich hier so langsam etwas tun müsste. Zwar gibt es in England den neuen Spectrum mit Cassettenlaufwerk, aber hier bei uns ist noch kein Vertrieb in Sicht, nachdem Schneider in Türkheim abgelehnt hat. Wir werden den Sinclair-Markt aber weiter beobachten und entsprechend berichten, wenn sich was tut.

Inhaltlich berücksichtigt die CK ja auch den ZX 81 und den QL. Wer ein solches Gerät hat, sollte mal nachschauen, ob er uns Programme zum Abdruck zuschicken kann. Hier ist durch die geringe Zahl der User die Einsendungsquote relativ gering.

Das gleiche gilt auch für die Atari-ST-User. Auch hier suchen wir noch gute freie Mitarbeiter. Und warum sollte nicht einmal ein ZX-81-, QL- oder ST-Programm zum "Topprogramm des Monats" werden?

Viel Spaß bis zum nächsten Mal
Ihr

Thomas Eberle, Chefredakteur

Jupitersoft GdbR

SVI 318/328 und VZ 200-Laser
210-310 Software ab 10 DM

Adventures, Action,
Anwender, Grafik
und Dateiprogramme.

**Viel Software stark
im Preis reduziert.**

Kostenloser Katalog bei

**Robert Goth
Jupitersoft GdbR**

Höttingerstr. 34b
8836 Ellingen
Tel. (091 41) 81 74

NEU FÜR C 64 zum Einführungspreis

Super-Eprom-Brenner "BURNY"
Brennt bis 27512, 25-er Reihe, EPROMs,
liest PROMs u. ROMs, insgesamt 38 versch.
Typen, inkl. Software auf Disk **DM 159,-**

Super-Eprom-Karte "ESB"
Bis 64 K, Autostart, Menüführung **DM 39,95**

Floppy-Speeder "QUICK-DOS"
Liest 200 Blocks in ca. 5 Sek., Speed-DOS
kompatibel, inkl. Kopier-Disk **DM 149,-**
Einbau-Service **DM 25,-**

"POWER-CARTRIDGE"
Das Super Modul **DM 159,-**

Alle Artikel inkl. aust. deutscher Anleitung,
Versand zuzügl. Porto u. Verpackung.

E EDV-Zubehör-Studio
Z Warbruckerstraße 2-4
S 4100 Duisburg
Telefon 02 03 / 40 21 36

Brandheiße Knüllerpreise

TI-99/4 A
CPS 88 Peripheriebox mit 1 Disketten-
laufwerk CSDO + 10 Disketten 1399,-
Externe 32 K-Erweiterung 199,-
Ext. 32 K-Erweiterung + Contronics 269,-
Extended Basic II Plus 279,-
Mini Memory (orig. TI) 169,-
Editor/Assembler (32 K notwendig) 129,-
Grafiktablett Superketch + Dig Day
+ Defender + Statistik 179,-
Sprachsynthesizer 175,-
Fernbedienung (orig. TI) 69,-
Alpiner, Chisholm Trail, Othello je 29,-
Defender, Dig Day, Car Wars, Moon Mine,
Munch Man, Invaders je 39,-
Buck Rogers, Chess, Shamus je 69,-
+ Preisauswahl an Hard- und Software
+ Bücherei!

Schneider
CPC 6128 mit Grinmonitor 899,-
CPC 6128 mit Palmmonitor 1540,-
Epsondrucker LX 85, anschlussfertig 879,-
Epsondrucker FX 85, anschlussfertig 1239,-

Atari
500 XL 179,-, 130 XE 339,-, 1050 399,-
800 XL + 1050 599,-, 130 XE + 1050 729,-
Epsondrucker LX 85 + Wissmann-Interf. 1049,-
Epsondrucker FX 85 + Wissmann-Interf. 1419,-

Commodore
Commodore AMIGA 1000 3249,-
256 K-Erweiterung für AMIGA 289,-
Commodore 128 + Floppy VC 1571 1399,-
Commodore C 128 D 1399,-
Farbmonitor 1901 mit RGB-Kabel 929,-
Farbmonitor Commodore 1201 575,-
Epsondrucker LX 85 + Görtz-Interface
8428 + Verbindungskabel 999,-
eto. = FX 85 1399,-
eto. = FX 800 1799,-
Epsondrucker LX 90, anschlussfertig 749,-
Sterndrucker NL-10, anschlussfertig 899,-
Commodore-Drucker MPS 803 399,-
Commodore Plus 4 299,-, VC 1551 329,-
Plus 4 + Floppy-Disk VC 1551 569,-
Akustikkoppler Dataphon S 21 d
+ Kabel + Terminalprogramm 299,-

Versandkostenpauschale (Warenwert bis DM
1000,-/darüber): Vorauskasse (DM 8,-/20,-), Nach-
nahme (DM 11,20/23,20), Ausland (DM 18,-/30,-).
Lieferung nur gegen Vorauskasse oder per NN;
Ausland nur Vorauskasse. Preisliste (Computertyp
angeben) gegen Zusendung eines Freumschlags.

CSV RIEGERT

Schloßhofstr. 5
7324 Reichenberghausen
Tel.: (071 61) 528 89

**Die nächste Ausgabe
»Computer-Kontakt«
erscheint am 24.11.1986**

Preis ausschreiben

Eigentlich ist es kein richtiges Preisausschreiben, sondern eine Fragebogenaktion, aber auf jeden Fall gibt es etwas zu gewinnen. Da wären als Preise ein MSX-Computer, Bücher im Wert von 200.- DM und Software aus unserem Programmservice.

Wissen wollen wir wieder mal etwas über unsere Leser. Wer liest Computer Kontakt, wie alt ist unser typischer Leser und welche Zeitschriften liest er sonst noch. Das wäre eigentlich schon alles. Wer Bedenken wegen dem Datenschutz hat, den können wir beruhigen. Wir wollen dies nur zu unserer Information wissen. Wichtig ist, daß möglichst viele Leser mitmachen, denn nur dann erhalten wir ein repräsentatives Ergebnis.

Und das sind die Preise:

1. Preis

Ein Philips VG 8010 (MSX) mit Z 80-Prozessor (3,57 MHz), 48 KRAM, davon 32 KRAM Basic, 32 Sprites, 16 Farben, Auflösung 256 x 192 Bildschirmpunkte.

2. Preis

Bücher im Wert von DM 80.-.

3.-10. Preis

Je ein Buch

11.-20. Preis

Software aus dem CK-Programmservice

Teilnahmebedingungen:

Es dürfen alle mitmachen. Jeder, der uns einen ausgefüllten Fragebogen zuschickt, nimmt an der Verlosung teil. Unsere Anschrift ist: Verlag Rätz-Eberle, Postfach 1640, 7518 Bretten.

Einsendeschluß ist der 24.11.86. Der Rechtsweg ist ausgeschlossen.

Fragebogen

Name, Anschrift:

Ich besitze folgende Geräte:

Computer: Floppy:

Drucker: Sonstiges:

Ich lese folgende Zeitschriften: selten manchmal regelmäßig

Computer Persönlich	☐	☐	☐
Chip	☐	☐	☐
Homecomputer	☐	☐	☐
Compute mit	☐	☐	☐
Happy Computer	☐	☐	☐
64er	☐	☐	☐
Run	☐	☐	☐
Computronic	☐	☐	☐
ST	☐	☐	☐
TI-Revue	☐	☐	☐
Data-Welt	☐	☐	☐

Ich lese Computer-Kontakt

☐ selten ☐ manchmal ☐ regelmäßig

INHALTSVERZEICHNIS

RUBRIKEN

Vorwort	3
Topprogramm des Monats	5
Clubnachrichten	6
Buchversand	39
CK-Programmservice	62
Kleinanzeigen	117
Fundgrube, Inserentenverzeichnis, Impressum	122

USER-CLUB SINCLAIR

ZX-Spectrum Maschinensprachekurs	10
Assemblertips für den Spectrum	13
Spielpokas	16
Jahrmarkt	20
Schreibschrift	21
Hisoft C für Heimcomputer	23
Discovery-Diskettenverzeichnis	24
Schnellere Apfelmännchengrafik	26
Beliebig viele Nachkommastellen	27
Hardcopy mit dem LPRINT-II-Interface	28
Der Colour Pen Graph EB 50	30
ZX 81-Kopierschutzprogramm	33
DNS	33
Grafik	34
QL: Was nicht im Handbuch steht	35
Das BEEP-Kommando des QL	37

COMMODORE-NEWS

Der Profi-C-Compiler	41
Grafikkurs C 64, Teil 2	42
Das RP-System	45
Mailbox, Marke Eigenbau	46
Das Versatile-Interface-System	47
Der SYBEX-Star-Painter	48
Eastern Fight	51
Drucker STAR NL 10	54
Der Amiga: Pro und Contra	60

ATARI

320 KByte für den Atari 800XL	64
Neues für die Atari Spielekiste	65
Peters Assemblerecke	66
Der hungrige Goff	68
Slow Motion	70
Kurs: 6502 Maschinensprache, Teil 2	72
Spiel zum Abtippen: Puzzler	74
Karteiverwaltung	77
Disc-Collector	79
Das MIDI-Disk-Programm für den 800XL	82
MicroMon	87

ATARI ST

3D Flying Ace	91
Textomat ST	94
GfA-Basic	96
Softwaretest: Cards	99

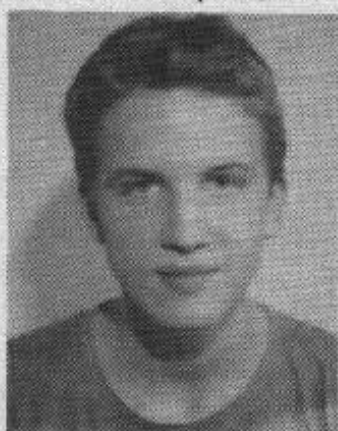
TI 99/4A

Messebericht: TI 99/4A in Perfektion	100
TMS 9900 Assembler auf dem TI	102
Der Zwillingbruder des TI-Writers	102
Def-Maker	104
Buchstabenmix	105
Schlüssel	110
Katalog für Maschinenspracheprogramme	115

Toppogramm des Monats: "MikroMon" für den Atari XL

MicroMon ist ein Monitor für den Atari 600/800 XL, mit dem man recht komfortabel Programme in Assembler schreiben kann. Er arbeitet parallel zu Basic, wodurch nach Belieben zwischen Basic und Maschinensprache gewechselt werden kann.

Der Autor dieses Programms, Arno Welzel aus Augsburg, ist 17 Jahre alt und hat die Computerei mit einem ZX81 begonnen. Anschließend folgte ein VC 20 und dann der Atari 800 XL. Seine Hauptinteressen liegen im Bereich der Hardware und Maschinensprache. Demnächst will er einen C64 mit seinem Atari über eine selbstentwickelte Schnittstelle koppeln.



1000 DM Honorar

In der CK gibt es das Toplisting des Monats. Das läuft dann so, daß wir hier in der Redaktion von allen Programmeinsendungen das beste Programm herausuchen und in der CK als Toplisting des Monats abdrucken. Der Autor dieses Programmes erhält dann als Honorar 1000.- DM.

Mitmachen können alle Programmierer mit den Geräten C64, Atari, Sinclair, TI 99/4A. Ein Listing muß nicht unbedingt beiliegen, falls der Autor des Programms noch keinen Drucker hat. Werden Programme abgedruckt, die nicht zum Toplisting des Monats gewählt wurden, erhält der Autor dafür ganz normal das übliche Honorar. Beim Toplisting des Monats ist das Honorar für den Abdruck in den 1000.- DM schon enthalten. Mit der Einsendung seines Programmes erklärt sich jeder Autor mit den einzelnen Punkten im Text "Bei uns können Sie mitmachen" einverstanden (siehe Seite 4 gegenüber).

Deshalb Leute aufgepaßt: Bei uns kann man Geld verdienen. Die Chancen für einen Abdruck, oder gar Gewinner des Toplistings des Monats zu werden, stehen bei uns immer gut.

QL Saison bei JEPOSOF

● QL (engl.) mit 640 K Speicher **DM 920.00**

QL (engl.) mit 128 K Speicher **DM 450.00**

Vorführung und Selbstabholung der Ware nur nach tel. Vereinbarung

● evtl. Leasing (Mietkauf)

Abdeckhaube für QL **DM 24.90**

Abdeckhaube für Spectrum **DM 19.90**

MICRODRIVE CARTRIDGE STORAGE BOX

100 Holds 20 cartridges

DM 18.50 **4 Cartridges** **DM 30.-**

Cartridge-Box mit 20 Cartridges **DM 158.50**

Multiface One, neue Version **DM 156.00**

Multiform-Diskettenbox für 3" und 3 1/2"-Disketten **DM 24.90**

Brandneu: Spectrum-Software

Der Stern des Druiden (deutsches Adventure) **DM 64.30**
lieferbar ab ca. Mitte Oktober

JEPOSOF

Kruppstraße 9, 4040 Neuss 21
persönlich erreichbar: 021 07/8184

Di-Fr. ca. 17-22 Uhr und Sa. 9-16 Uhr,
ansonsten Bestellungen 24 h auf
Anrufbeantworter möglich.

Abo-Bestellschein

Ich möchte Computer-Kontakt in Zukunft regelmäßig zugeschickt bekommen. Meine Abo-Bestellung gilt ab der nächsten Ausgabe. Die Abodauer beträgt 6 Ausgaben, also ein Jahr und kann bis spätestens 4 Wochen vor Aboende wieder gekündigt werden. Der Abonnementpreis beträgt 33.- DM einschließlich Mehrwertsteuer und Versandkosten. Für Bestellungen aus dem Ausland wird es aber nur ein wenig teurer: Hier kostet das Abo 37.50.- DM.

Name/Vorname

Straße

PLZ

Ort

Ich bezahle wie folgt:

Ich bestelle ab Ausgabe:

☐ Scheck liegt bei

☐ Vorauskasse auf Postscheckkonto Karlsruhe Nr. 43423-756

Mir ist bekannt, daß ich diese Bestellung innerhalb 8 Tagen widerrufen kann und bestätige dies mit meiner Unterschrift. (Dieses Widerrufsrecht ist per Gesetz vorgeschrieben.)

Datum/Unterschrift

Diesen Bestellschein ausschneiden oder fotokopieren und an Computer-Kontakt, Postfach 1550, 7518 Bretten schicken.

NEU!! Der Katalog Nr. 6

Auf 112 Seiten Hardware, Software und Bücher
für AMSTRAD/SCHNEIDER CPC 464,664,6128,JOYCE
COMMODORE 64 UND 128, SPECTRUM/SPECTRUM PLUS

Katalog gegen 1.88 DM in Briefmarken anfordern bei:

THOMAS WAGNER
SOFTWAREVERSAND

T. WAGNER SOFTWAREVERSAND P.O. BOX 112243, D-8900 AUGSBURG 1



USER-CLUBS

Atari-Clubs

Wir suchen Mitglieder für unseren Atari-Computerclub. Es gibt eine Computerzeitschrift mit vielen Infos, Programme zum Abtippen, ein Leserforum und vieles mehr. Die Clubzeitschrift erscheint regelmäßig im Abstand von zwei Monaten und wird frei Haus geliefert. Wir beantworten alle Fragen und veröffentlichen die interessantesten in unserer Zeitschrift. Wer Probleme oder Fragen zum Atari XL/XE hat, ist bei uns zum aufgehoben. Wir vermitteln Tips und Tricks zum vereinfachten Programmieren. Der Clubbeitrag beträgt 1.50 DM pro Monat. Man kann jederzeit ein- oder austreten.

Key-Soft Computerclub
Ralf Merx
Ringstraße 13/A
5100 Aachen

Ich möchte einen Atari-Club gründen, dessen Mitglieder sich nur mit den Systemen 400/600/800/XL/130XE befassen. Gegenseitige Hilfe der Mitglieder ist bei Problemlösungen selbstverständlich; ein Clubbei-

trag wird prinzipiell nicht erhoben, da sowieso schon alles teuer genug ist und der Spaß am Computern nicht am Geld scheitern soll. Etwaige Kosten für Disketten oder Porto müssen aber von jedem selbst getragen werden. Natürlich wird auch eine eigene Clubzeitschrift herausgegeben. Diese kostet keinen Pfennig und wird hauptsächlich Listings und Informationen enthalten. Wer also Interesse hat, kann mir jederzeit schreiben.

Atari Computerclub
Jörg Schubert
Dillener Straße 56 C
1000 Berlin 33

Der Atari-User-Club Cuxhaven, der mittlerweile seit einem halben Jahr besteht, will mit möglichst vielen Atari-XL-Usern eine Verbindung per Post aufbauen. Mitmachen können nicht nur User aus dem Raum Cuxhaven, sondern auch solche aus dem ganzen Bundesgebiet und Europa (wir pflegen den Kontakt zu einem Atari-Club in der CSSR). Jeder, der mitmachen will, bekommt monatlich eine Clubzeitschrift zugeschickt, in der Tips, Tricks,

Softwaretests usw. abgedruckt werden. Der Clubbeitrag, in dem natürlich auch unser "Problemlösungsservice" enthalten ist, beträgt 10.- DM im Vierteljahr. Weitere Infos bei

A.U.C.C.
Höfenweg 24
2190 Cuxhaven 1.

Der A&C (Atari & Commodore Club) bietet ein monatliches Magazin, in dem man für 2.- DM Anzeigen aufgeben kann und das Tricks, Tips, News und Adventures enthält. Der Beitrag liegt bei 20.- DM.

Alexander Schneider
Wilhelm-Maybach-Str. 8
6832 Hockenheim
Tel. 06205/14210

Ich suche Kontakt zu Usern im Raum Kaiserslautern, aber auch im ganzen Bundesgebiet.

Alexander Reinwarth
Rosenstraße 42
6750 Kaiserslautern

Ich suche Kontakt zu Spectrum-Usern im Raum Münster.

Carsten Brönstrup
Tannenweg 18
4544 Ladbergen

Kontakt gesucht!

Briefkontakte zu deutschen C-64-Anwendern sowie Bücher, Software und Zeitschriften sucht der Pole Waldemar Sadowski. Wer mit ihm in

Kontakt treten möchte, sollte möglichst in Englisch schreiben.

Waldemar Sadowski
Ul. Pokoju 11 n 29
42-200 Czystochowa
Polen

Andreas Baro sucht Leute oder Clubs im Raum Hamburg, die Interesse an einem Erfahrungsaustausch über den C64 haben.

Andreas Baro
Brödermannsweg 44
2000 Hamburg 61

Ich suche Kontakt zu einem Atari-Club oder zu einzelnen Atari-Usern im Raum Mainz Wiesbaden/Frankfurt.

Rainer Schmidt
Kolpingstr. 9
6095 Gustavsborg 1

Wer will noch bei meinem Atari-Club mitmachen? Kein Beitrag, 600/800 XL, 130 XE.

Wolf Bartha
Altenberndstraße 3
4930 Detmold

Als erfahrener Atari-Besitzer suche ich Kontakt zu XL/XE-Usern im Raum Darmstadt zum Programmieren in 6502 Basic und Pascal.

Bern Nowotka
Am Fuchsbau 25
6102 Pfungstadt

Ich suche Kontakt zu einem Atari-Club in Stuttgart oder Leinfelden.

Matthias Wunder
Lehenbühlstraße 54
7253 Renningen

Ich würde gern einen Club in Duisburg gründen. Wer macht mit?

Oliver Montag
Karl-Albert-Straße 22
4100 Duisburg 12

DATENKASSETTEN

Deutsches Markenband
alle Größen von C0-C83 lieferbar,
z.B. C10 ab DM 0,60-DM 0,94
Copy Service.
Laufend Sonderangebote,
interessant auch für
Wiederverkäufer.
Preisliste sofort anfordern.

3,5" Atari 520/260
Preis auf Anfrage
5 1/4" HC/PC Preis auf Anfrage
10 Disketten 5,25" HC/PC
in Plastikbox (transparent o.
farbig) Preis auf Anfrage
Preislisten ab 50 Stück
5,25" Diskettenkopien auf
Anfrage

DISKETTEN



Holschuh Tapes
Kellenstr. 67, 6140 Bensheim
Tel. 06251/62565

★ RTS ★

TI 99/4A und MSX-Software

★ TI 99/4A Software ★

Simmerspiele 19.- DM
Doppel-Pack I
(Strumpel/Amazona) 20.- DM
Doppel-Pack II
(007/Heint) 25.- DM
Sammelbox
(4 Programme) 25.- DM
Carry
(Adventure Disk) 39.- DM

★ MSX Software ★

Nador 25.- DM
Quastel 29.- DM
Filter 29.- DM
Cobra, die Mission 34.- DM
Gorbu 29.- DM

Infos bei: Roland Toonen Software, Pf. 31, 4178 Kevelaer 1

EXTENDED ROM

EXTENDED ROM V 3.5 FÜR SPECTRUM
(C) R. HERVENS HARD & SOFTWARE

> Backup-Kopien über NMI
> Sonderfunktionen (menuegesteuert)
- Totalreset
- jederzeit zurück ins BASIC
- kompl. Type-Headeranzeige
- Kopierfunktion
- Renumber
> Verkürzte M Drive-Befehle
> Schöne Schrifttafel
> 100% kompatibel zum Original ROM
> Spezielle Entprellschaltung
für den NMI-Tester (100% Funks)
> ausführliche deutsche Einbau-
und Bedienungsanleitung

PREIS: ROM Tester mit Schaltung
nur 45.- (w. Einbau 75.-)
Bestellungen & Infos:
Reinhold Merkens, Gillerstr. 51,
4049 Ronsbockerkirchen
(Tel.: 02183/5394) ab 18.00 Uhr

Ich suche Kontakt zu Leuten (auch Urlauber) in der Gegend des Dümmer Sees zwecks Erfahrungsaustausch.

Michael Beise
Dielingen 454
4995 Sternwede 2

Ich suche Kontakt zu Usern und Userclubs im Raum Osnabrück, Bielefeld und Melle.

Holger Hagemann
Hannoversche Straße 68
4520 Melle 10

Ich bin ein österreichischer Spectrum-Besitzer und suche einen österreichischen Club oder auch Kontakt zu Usern.

Herrmann Aigner
Markt 187
A-3324 Euratsfeld

Ich suche einen User-Club für den QL.

Spamer
Amselweg 7
6300 Gießen

Ich suche Mitglieder für meinen 64er Club. Alle zwei Monate gibt es abwechselnd eine

Clubzeitung oder eine Clubdiskette/Cassette mit Tips, Tricks, Programmen, News, Spielerpokers usw.. Der Beitrag beträgt im 1. Jahr 30.- DM (25.- DM Clubbeitrag und 5.- DM Aufnahmegebühr).

Club der 64er Fische
Georg Wenk jun.
Waldhofweg 7
8240 Berchtesgaden

Jetzt gibt es auch in Lübeck eine Mailbox! Endlich wird die Computerwüste Schleswig-Holstein durch eine Oase bereichert. Wir sind auf Atari 8-Bit spezialisiert, haben aber auch viele Infos für andere Rechner. Wir bieten z.B. Public Domain Software, Bootsmarkt, großes Mailboxnummernverzeichnis und vieles, vieles mehr.

Charly-Mailbox (by Cube-Soft)
offen von 20-07 Uhr
Tel. 04 51/3 16 42
300 Baud, 8 Datenbits, 1 Stoppbit, keine Parität

Wer sucht noch alte CK-Hefte?

Alle neuen Leser haben bei uns die Möglichkeit, die zurückliegenden Hefte nachzubestellen. Die Ausgaben von 1984 sind nicht mehr lieferbar. Bestellt wird mit untenstehendem Bestellschein. Die Lieferung erfolgt aber nur gegen Vorkasse in Form von Briefmarken oder gegen Scheck.

Bestellschein für CK-Hefte

Ich möchte folgende CK-Hefte bestellen:

- | | |
|---|-------|
| Ex. Heft Januar '85 (4,50 DM) | |
| Ex. Heft März (4,50 DM) | |
| Ex. Heft April (4,50 DM) | |
| Ex. Heft Mai (4,50 DM) | |
| Ex. Heft Juni (4,50 DM) | |
| Ex. Heft Juli (4,50 DM) | |
| Ex. Heft August-September (4,50 DM) | |
| Ex. Heft Oktober (4,50 DM) | |
| Ex. Heft November (4,50 DM) | |
| Ex. Heft Dezember-Januar '85/'86 (5,50 DM) | |
| Ex. Heft Februar-März (5,50 DM) | |
| Ex. Heft April-Mai (5,50 DM) | |
| Ex. Heft Juni-Juli (5,50 DM) | |
| Ex. Heft August-September (5,50 DM) | |
| Versandkosten (1-2 Hefte 1,40 DM,
2-4 Hefte 2,00 DM, 5-15 Hefte 3,00 DM) | |

Summe

Meine Anschrift:

Den Bestellschein einsenden an den Verlag Ratz-Eberle, Postfach 1640, 7518 Bretten.

Verlangen Sie mehr!

Rainer Fischer
Datenverarbeitung mit dem Sinclair QL



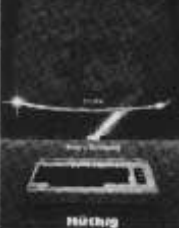
Rainer Fischer

Datenverarbeitung mit dem Sinclair QL

1986, 198 S., kart., DM 42,-
ISBN 3-7785-1325-7

Das Buch zeigt zunächst die entsprechenden Möglichkeiten von Super-BASIC und führt den Leser dann in die Handhabung der Dateiverwaltungssprache von QL-ARCHIVE ein. Der Leser erhält dabei wichtige Hinweise für die Erstellung wohlstrukturierter Programme.

Hajo Lemcke
Assembler mit dem C-64



Hajo Lemcke

Assembler mit dem C-64

1985, 227 S., 22 Abb., 6 Tab., kart., DM 38,-
ISBN 3-7785-1091-6

In der Maschinensprache nutzen Sie die Fähigkeiten Ihres Computers erst richtig aus. Diese zweibändige Einführung verhilft Ihnen dazu. Jeder Befehl bzw. jede Gruppe wird mit einem lauffähigen Beispiel vorgestellt, das sich später in große Programme einbauen läßt.

Thomas Tai
Cracker, Hacker, Datensammler



Thomas Tai

Cracker, Hacker, Datensammler

1986, 96 S., kart., DM 24,-
ISBN 3-7785-1236-6

Die Piraterie der meist jugendlichen Kopierer steht im Mittelpunkt dieser mit vielen authentischen Fällen versehenen Schrift, die in einer Art Bestandsaufnahme des Software(schwarz)marktes beide Seiten zu Wort kommen läßt.

BESTELLCOUPON

einsenden an: Dr. Alfred Hüthig Verlag,
Im Weiher 10, 6900 Heidelberg

Titel

Name, Vorname

Straße, Nr.

PLZ, Ort

Datum, Unterschrift



Hüthig

USER CLUB

sinclair®

Sinclair ist ein eingetragenes Warenzeichen der Sinclair Ltd.

Hallo Freunde!

Jetzt kommt wieder die Zeit der kurzen Tage und langen Nächte, naturgemäß also eine Zeit, in der man sich wieder mehr mit seinem Computer beschäftigen kann. Der Sinclair-Teil der vorliegenden neuen Ausgabe der CK bietet diesmal auch etwas mehr für die Freunde des QL. So soll es auch in Zukunft sein, wobei es keine feste Aufteilung gibt, wie viele Seiten auf welchen Computer entfallen. Ich bin sicher, daß die Spectrum-Benutzer damit einverstanden sind.

Jetzt aber zum Inhalt. Die Spielfreaks unter den Spectrum-Benutzern bekommen in dieser und in der nächsten Ausgabe einen besonderen Leckerbissen serviert. Es handelt sich um G-Force, ein tolles Actionprogramm, das komplett in Maschinensprache geschrieben wurde. Da das Programm sehr

umfangreich ist, mußten wir leider zwei Teile daraus machen. Spielen könnt ihr dieses Programm also erst nach der nächsten Ausgabe. Die Wartezeit lohnt sich aber auf jeden Fall.

QL-Befehle, die nicht jeder kennt, zeigt R. W. Gerling in seinem Artikel. Vom gleichen Autor stammt auch das QL-Listing, das sich mit dem Sound beschäftigt. Darüber hinaus haben wir auf den rund 30 Seiten des Sinclair-Teils wieder Neuigkeiten, Informationen sowie Tips und Tricks zusammengetragen, um jedem Leser etwas zu bieten. Schreibt doch mal, was ihr von dem neuen Konzept der CK haltet. Auch für Anregungen sind wir immer dankbar.

Viel Freude an der neuen CK wünscht wie immer Euer

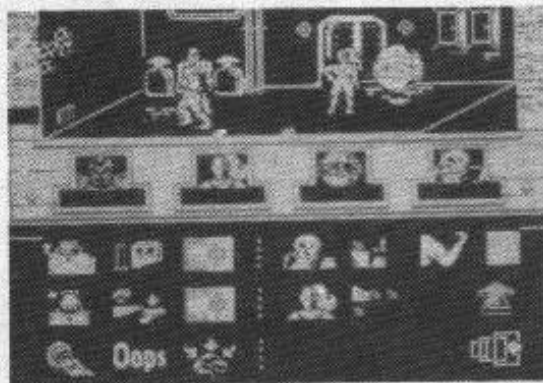
Rolf Knorre

Enigma Force

Lang erwartet wurde der Nachfolger bzw. die Fortsetzung von Shadowfire. Unter dem Namen "Enigma Force" ist das Programm jetzt endlich auf dem Markt. Die Handlung knüpft direkt an Shadowfire an. Nach der geglückten Verhaftung von General Zoff soll dieser von den vier Mitgliedern des Enigma-Teams in einem Raumschiff zum Planeten des Imperators gebracht werden. Der General läßt das Raumschiff je-

doch auf dem Heimatplaneten von Syyk abstürzen. Dort führt Syyks Rasse einen Krieg gegen reptilienartige Lebewesen, die zu General Zoffs Armee gehören. Nach der Bruchlandung kann der General fliehen und versucht im Einflußbereich der Reptilien ein raumtuchtiges Raumschiff zu finden.

Hier beginnt nun die mehrteilige Aufgabe des Enigma-Teams. Sie müssen den Führer



Prima Grafik

der Insektenwesen finden, ihn zu ihrem Verbündeten machen. General Zoff wieder einfangen und das letzte flugfähige Raumschiff im Einflußbereich der Reptilien entdecken. Diese Aufgaben hören sich einfach an. Wer jedoch schon Shadowfire gespielt hat, weiß, welche Schwierigkeiten noch auftauchen können.

Im Gegensatz zu früher sind die Figuren nicht mehr durch Symbole innerhalb des Labyrinths gekennzeichnet. Die verwendete Grafik kommt diesmal der Technik des Zeichentrickfilms sehr nahe. In der rechten unteren Ecke des jeweiligen Raumes befindet sich die Kamera in einer etwas überhöhten Position. Man sieht jeweils den Raum, in dem sich das gerade gesteuerte Enigma-Mitglied aufhält. Der Bildhintergrund ist immer schwarz. Die Gegenstände und Figuren werden in der Farbe des Teammitglieds angezeigt, z.B. gelb wenn Syyk gesteuert wird.

Beginnt man das Spiel, sollten zuerst die Teammitglieder mit Waffen, Munition und anderen nützlichen Gegenständen

ausgerüstet werden. Feindliche Truppen tauchen auf und müssen bekämpft werden. Der richtige Weg ist zu finden und die Enigma-Mitglieder sind entsprechend ihrer Fähigkeiten einzusetzen.

Für alle Shadowfire-Freunde stellt dieses Nachfolge-Programm ein Muß dar für die anderen User ein Erlebnis. Die Grafik der selbstständig handelnden Figuren, z.B. in einem Kampf, lohnt das Zusehen. Wenn auch gewöhnungsbedürftig ist die Steuerung der Figuren, jedoch nicht zu schwierig. Das Spiel kann nicht direkt einer Kategorie zugeordnet werden, da sowohl Arcade- wie auch Adventure-Teile zusammen dieses Spiel ausmachen und interessant halten. Diese Zeichentrickfilm-Grafik sollte man sich nicht entgehen lassen.

Bezugsquelle:
Speedysoft
37 Church Road
UK-London
SW13 9HQ
United Kingdom
Preis:

£ 9.95

Horst Müller

Cosmic Wartoad

Der englische Cassetteneinleger zu diesem Programm bietet eine so verschrobene Geschichte, daß ich gar nicht erst versuchen möchte, sie hier wiederzugeben. Wichtig ist nur, daß zur Zeit irgendwo der Kampf der kosmischen Kriegskröten tobt und der Spieler in diesen einbezogen wird. Auf dem Bildschirm wird dann die ganze Angelegenheit schon klarer. Das erste Bild zeigt ein Raster mit 8x8 Feldern. Diese 64 Felder stellen Zeittunnel dar. Der Spieler muß sie nach und nach aufsuchen, um eine bestimmte Spielebene zu erreichen, wo er unterschiedliche Aufgaben zu lösen hat.

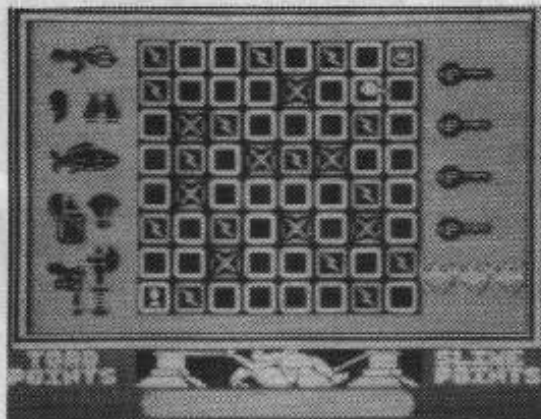
Die Spielfigur ist eine der in der Anleitung beschriebenen

Kampfkroten, also ein dicker bewaffneter Frosch. Bei den Aufgaben handelt es sich in der Regel um kleine Schießspielen. Ist eine Stufe bewältigt, erfolgt ein Rücksprung in das Zeitraster, damit eine weitere Ebene angewählt werden kann.

Leider bietet Cosmic Wartoad weit weniger als man zu Anfang erwartet. Die Grafik ist nicht umwerfend, Sound ist so gut wie keiner vorhanden und der Spielwert ist auch nicht besonders hoch. Insgesamt also ein Programm, das man nicht unbedingt haben muß.

Bezugsquelle: P. West Record System: Spectrum 48K

Rolf Knorre



So fängt es an

User-Clubs

Wir wollen für den ZX Spectrum und dessen User einen Club gründen, der ein monatliches Clubmagazin herausgibt, von Mitgliedern eingeschickte Programme zugänglich macht, alle zwei Monate eine Mitgliederliste veröffentlicht (um Kontakte zwischen den Mitgliedern zu ermöglichen) und schließlich auch versuchen soll, Händlerabatte auszuhandeln.

Daniel Tietze
Nienburger Str. 37
2800 Bremen

Der Sinclair User Club Sandbach ist im "Computerclub für Sinclair, Commodore und Schneider-Computer" in Reit im Winkel aufgegangen. Hier erhalten die Sandbacher weiterhin preiswerte Software (Listings, Programme usw.), die

Clubzeitschrift "top computer news" sowie als Service einen Programmaustausch zwischen den Mitgliedern.

Computerclub für Sinclair, Commodore und Schneider-Computer
Postfach 1109
8216 Reit im Winkel

Vor ein paar Monaten kaufte ich mir zu meinem Sinclair ZX Spectrum + einen Drucker (Seikosha SP-1000 AS, Epson-FX-80-komp. und mit dem Interface 1 über die RS-232-Schnittstelle verbunden). Jetzt habe ich aber einige Mühe mit dem Grafikausdruck. Wer kann mir hier weiterhelfen. Ich würde gern wissen, wie ich z.B. eine Hardcopy erstellen kann oder wie es mir sonst gelingt, meinem Drucker Grafik beizubringen!

Franz Sprecher
Nelkenstraße 1
CH-8645 Jona

Uta Jäkel & A. Klintworth GbR
Hard- & Softwarevertrieb
Marschhorst 2
2732 Klein-Meckelsen
Telefon: 0 42 82 / 56 15

Sinclair ZX Spectrum			
Spectrum plus 48 KB	349.-	Spectrum 128 KB deutsch	459.-
Spectrum 48 KB	259.-	Doppelport-Joyst. Interface	45.-
Saga 1-Tastatur	129.-	Kempston-E-Druckerinterface	175.-
Saga 3-Tastatur	249.-	Multiface One (neue Ausführ.)	159.-
Saga 2001-Tastatur	399.-	Currah Sprachsynthesizer	89.-
Discovery Diskstationen	ab 399.-	ZX Lprint 3 Druckerinterface	175.-
Beta-Disk Controller 5.03	329.-	Beta-Disk Kpl.-Syst. (1MB)	ab 749.-
Beta Basic 3.0	53.-	Art Studio Malprogramm	59.-
Sinclair QL			
QL-englische Ausführung	459.-	Drucker	
QL-deutsche Ausführung	459.-	Seikosha M3 15 Typenraddr.	699.-
QL-Diskcontroller	ab 329.-	Seikosha SP 1000 (A/AS/I)	749.-
QL-Disk-Kpl.-Syst. (1MB)	ab 749.-	Seikosha MP 1300 AI	1799.-
QL-Centronics-Interface	ab 149.-	Citizen 120 D	749.-
QL-Monitore	ab 249.-	Centronics-QLP II Drucker	599.-
		Aufsatztraktor für GLP	89.-
QL-Speichererweiterung 256 KByte mit durchgeführtem Bus	339.-		
QL-Speichererweiterung 512 KByte mit durchgeführtem Bus	439.-		
QL-Giga-Soft-Mouse-Paket incl. leistungsstarker Software	220.-		
CUB-Super-Farbmonitor für QL oder Spectrum 128	1099.-		
QL-Sandy-Super-Diskcontroller + 512 KB + Centronics-Interface	899.-		
QL-Eprom-Erweiterungssteckkarten (ansteckbar an ROM-Port)	ab 19.-		
Microdrive-Cartridges	4 Stk. 32.-, 12 Stk. 93.-, 20 Stk. 150.-		
3.5"-Disketten SSD	1 Stk. 7.90, 10 Stk. 75.-, 50 Stk. 350.-		
3.5"-Disketten DSD	1 Stk. 9.90, 10 Stk. 89.-, 50 Stk. 399.-		
3.5"-Disketten 2D No Name	1 Stk. 8.-, 10 Stk. 57.-, 50 Stk. 275.-		

Lieber Kunde, diese Preise sind 6 Wochen alt, sie können sich daher inzwischen wesentlich geändert haben.

Alle Preise incl. MwSt. zuz. Versandkosten zum Selbstkostenpreis.
Unsere Gesamtpreislise erhalten Sie gegen 2.- DM in Briefmarken!
Selbstabholung nur nach vorheriger telefonischer Terminabsprache!
Versand nur per Nachnahme oder Vorauskasse!

Sonderpreise für Schulen, Sammelbesteller und Clubs!!!

Wetten, daß Sie bei uns preisgünstig einkaufen können?
Wetten, daß wir Ihnen trotzdem Service bieten können?
Wetten, daß wir Sie außerdem auch noch beraten können?

MN Michael Naujoks

LOAD ZX81 (dt.)	DM 39.-	Light Pen	DM 68.-
Vokabeltrainer	DM 39.-	(dk'tr.) mit deutscher Anleitung	
The Art Studio	DM 59.-	3-Kanal Soundsynthesizer	
Beta Basic 3.0	DM 59.-	(dk'tr.) mit deutscher Anleitung und Softwareerweiterung	DM 98.-
Schachbuch		Speechsynthesizer	DM 89.-
Datenbank für Schachpartien mit Datei zum Nachspielen aller 72 Partien des Weltkampfes Karpov gegen Kasparov	DM 49.-	(dk'tronics)	
The Colt	DM 56.-	Joystick-Interface	DM 29.-
Schaltbild-CAD	DM 39.-	SpecDrum	DM 149.-
Tasword III	DM 59.-	T-Stecker	DM 25.-
Tasword Two	DM 29.-	(Currah Microslot)	
Masterfile	DM 39.-	Laser Basic	DM 49.-
Hisoft C	DM 99.-	Laser Compiler	DM 36.-
		Laser Genius	DM 49.-

Hardware Sinclair Software SPECTRUM

Discovery 180 K	DM 448.-	2. Laufwerk 180 K	DM 248.-
Discovery 720 K	DM 848.-	2. Laufwerk 720 K	DM 448.-
Bomb Jack	DM 29.-	Pentagramm	DM 35.-
Rescue on Fractalus	DM 29.-	Theatre Europe	DM 35.-
Ping Pong	DM 31.-	Bounces	DM 36.-
Super Bowl	DM 36.-	Cyberun	DM 39.-
Turbo Esprit	DM 33.-	Spindizzy	DM 38.-
Who Dares Wins II	DM 29.-	Samantha Fox Strip Poker	DM 33.-
Way of the Tiger	DM 36.-	Tomahawk	DM 39.-
Ghosts'n Goblins	DM 29.-	Spitfire 40	DM 39.-

Kostenlosen Katalog S/9 anfordern!

Entwicklung & Vertrieb von
Computer Soft- und Hardware
Rottmannstr. 40, 6900 Heidelberg

Hotline:
(06221) 46885

Maschinensprachekurs für den ZX-Spectrum

Bildschirmaufteilung und neue Befehle

Zuerst wieder eine kleine Zusammenfassung des letzten Teiles. Neben der Funktion und Wirkungsweise des Assemblers, dem wichtigsten Werkzeug des MC-Programmierers, wurden die ersten Befehle erklärt. Das war einmal die Gruppe der Ladebefehle, der INC- und DEC-Inkrementierbefehle sowie der Sprungbefehle. Letztere werden diesmal noch mit einer Erläuterung des Stacks und einem Ausnahmebefehl vervollständigt. Neben anderen neuen Befehlsgruppen wird heute auch auf die Handhabung des Bildschirmspeichers eingegangen. Außer einigen Diagrammen zum besseren Verständnis des Geschriebenen findet sich noch eine ganzseitige Befehlsübersicht aller regulären Z80-Befehle mit den jeweiligen Ausführungszeiten und den Wirkungen der Flags.

Der Stack

Zu den im letzten Teil besprochenen Sprungbefehlen gehört auch der Befehl CALL, der mit dem Stack (=Stapel) arbeitet und zum Aufruf von Unterprogrammen benutzt wird. Dieser Stack befindet sich in einem Teil des Speichers, normalerweise nach dem Basic-Programm- und Variablenbereich, wobei das 16-Bit-Register SP (Stackpointer) die Adresse des obersten Elements enthält. Auf diesem Stapel kann man nun mit dem Befehl PUSH ein Registerpaar ablegen oder mit POP das oberste Stapелеlement holen. Dies geht folgendermaßen vor sich: Wenn z.B. mit dem Befehl PUSH DE der Inhalt des DE-Registerpaares auf den Stapel gelegt wird, werden die zwei Speicherbytes über SP mit dem Wert von DE geladen (DE selbst bleibt unverändert). Danach wird SP um zwei erhöht, damit er auf den nun neuen letzten Eintrag zeigt. Beim Holen eines Wertes vom Stack mit POP wird das Verfahren einfach andersherum abgewickelt.

Da der Z80 nicht weiß, von welchem Registerpaar ein abgelegter Wert stammt, könnte man natürlich auch die mit PUSH DE abgelegte Zahl mit POP HL in das HL-Registerpaar laden.

Mit diesen zwei neuen Befehlen wird nun der Stapel direkt vom Programmierer manipuliert. Aber auch mit dem Sprungbefehl CALL wird der Stapel verändert. Ruft man mit CALL ein Unterprogramm auf, wird im gleichen Moment das Sprungziel in den PC (Programcounter) geladen und die Adresse des Befehles hinter dem CALL auf den Stack gelegt. Bei einem RET, dem Rücksprung von einem Unterprogramm, wird nun einfach das oberste Element vom Stapel geholt und in den PC geladen, damit die Systemeinheit wieder an der alten Stelle weitermachen kann.

Bei einem CALL erfolgt also intern ein PUSH PC+2 und bei einem RET ein POP PC. Bei der Anwendung der eigentlichen Stapelbefehle PUSH und POP ist also darauf zu achten, daß jeweils beide gleich oft verwendet werden, damit nicht eine abgelegte Adresse in ein Register oder eine abgelegte Zahl in den PC geladen wird. Bei geschickter Programmierung kann man diese Eigenschaft des möglichen "Falschladens" aber auch dazu benutzen, den Stack zu manipulieren. Das sei aber nur geübten Programmierern empfohlen, da es bei einem Fehler mit 99% Wahrscheinlichkeit zu einem Absturz kommt.

Zusammenfassend kann man sagen: Auf dem Stack können alle Registerpaare (BC, DE, HL, IX, IY, AF und intern PC) abgelegt bzw. geladen werden. Dies ist z.B. dann sinnvoll, wenn deren Werte vor einem Unterprogrammaufruf zur weiteren Verwendung gerettet werden müssen, weil sie nach

der UP-Beendigung veränderte Inhalte haben.

MC-Programme, die vom Basic aus aufgerufen wurden, werden mit RET beendet und springen wieder ins Basic zurück. Außerdem wird beim Rücksprung ins Basic automatisch der Inhalt des BC-Registerpaares übergeben. Wenn z.B. ein mit LET a=USR... gestartetes MC-Programm wieder im Basic gelandet ist, hat die Variable a den Wert von BC vor dem RET des MC-Programms.

Mit dem Ausnahmebefehl DJNZ wird die Gruppe der Sprungbefehle abgeschlossen. Er vereinigt die zwei Befehle DEC B und JR NZ und ist für kleine Schleifen gedacht, da er schnell in der Ausführung und kurz im Code ist. Leider gibt es diesen Befehl nur für das B-Register, was zur Folge hat, daß das B-Register meist als Schleifenzähler verwendet wird.

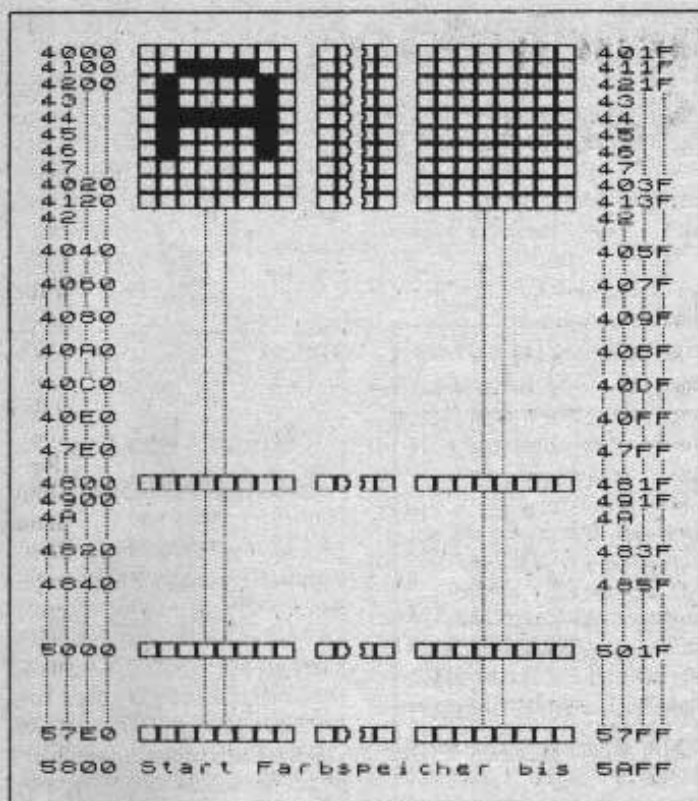
Der Bildschirmspeicher

Wenn wir ein eigenes MC-Programm schreiben, ist ja meist irgend etwas auf dem Bildschirm auszugeben. Dies ist für uns, neben der Tonausgabe, die einzige Kontrolle, was sich in unserem Programm ereignet. Und damit sind wir auch schon beim komplizierten Aufbau des Bildschirmspeichers. Der Bildschirm hat im Speicher immer seinen festen Platz mit fester Länge (sie war beim Vorgänger ZX-81 noch variabel) und ist in zwei Teilen abgelegt. Von 4000h-57FFh ist jeder einzelne Punkt des Bildes als ein Bit eines Bytes definiert. Acht nebeneinanderliegende Punkte sind also in einem Byte integriert. Von 5800h-5AFFh liegen die Farbattribute, wobei jeweils ein Byte die Farbinformation einer PRINT-Stelle (8*8 Pixels) enthält. Die Verschlüsselung der Farbinformationen ist im Sinclair-Handbuch, Kapitel 16, beim Befehl ATTR genauestens erklärt.

Das eigentliche Problem für den Programmierer ist aber nicht die Trennung von Pixels und Farben im Speicher, sondern die Reihenfolge, in der die einzelnen Pixelblöcke im Speicher abgelegt sind. Diese Reihenfolge ist ersichtlich, wenn ein Bild (SCREEN\$) von Band geladen wird: Jeweils ein Drittel wird in 8 mal 8 Zeilen geladen. Nebeneinanderliegende Bild-Bytes befinden sich also auch im Speicher nebeneinander, untereinanderliegende haben aber im Speicher nicht die gleiche Entfernung. Da sich eine Zeile aus 32 Bytes zusammensetzt, müßten die Adressen der darunterliegenden Bytes eigentlich um 32 größer sein. Da aber im Spectrum nicht Zeile für Zeile gelesen wird, sondern eine Reihenfolge herrscht, wie wir sie beim Laden von Band sehen können, ist die Entfernung untereinander im Speicher unterschiedlich.

Zur Veranschaulichung haben wir eine Skizze des Bildschirms abgedruckt, in der jeweils die Adresse des 1. und 32. Bytes einer Pixelzeile angegeben ist. Außerdem sieht man, wie der Buchstabe "A" mit PRINT AT 0,0; "A" dort dargestellt wird. Wie wir selbst etwas auf dem Bildschirm darstellen können, ist erst mit weiteren Befehlen möglich und wird im nächsten Teil der Serie behandelt.

Nach diesem Spectrum-spezifischen Kapitel folgen weitere neue Befehlsgruppen der Maschinensprache. Da wären zuerst die Additions- und Subtraktionsbefehle. Mit ihnen können die oben genannten Register oder auch wie bei den Ladebefehlen 8- oder 16-Bit-Operationen durchgeführt werden (unter Berücksichtigung des Carryflag sogar 9- oder 17-Bit). Bei einer 8- oder 9-Bit-Rechnung ist der eine Hauptoperand immer das A-Register, in dem nach der Berechnung das Er-



Organisation des Bildschirm-RAM

gebnis abgelegt wird. Bei einer 16- oder 17-Bit-Operation stellt das HL-Registerpaar den Hauptoperanden (bei Addition auch IX oder IY möglich) und Ergebnisträger. Die abgedruckte Tabelle zeigt, welche Befehlskombinationen möglich sind.

Der Unterschied zu den Additionsbefehlen ADD - ADC (bei einer Subtraktion SUB - SBC) besteht darin, daß damit jeweils eine Berechnung mit oder ohne Berücksichtigung des Übertragsflags stattfindet. Bei Anwendung der Befehle ADC oder SBC wird nach Ausführung der jeweiligen Rechenoperation noch der Stand des C-Flag mitbewertet und dessen Wert (0 oder 1) zu dem bisherigen Ergebnis dazugezählt (oder bei SBC abgezogen). Dies ist sinnvoll, wenn z.B. nach einer vorherigen Addition der Übertrag in die neue Rechnung mitgenommen werden soll.

Die Logikbefehle

Die drei logischen Befehle sowie der Vergleichsbefehl CP arbeiten jeweils mit 8-Bit-Daten. Bei allen Logikbefehlen ist das A-Register der Hauptoperand, wobei auch hier, mit Ausnahme des CP-Befehls, das Er-

gebnis wieder im A-Register zu finden ist. Bei den logischen Befehlen wird jeweils das 0. bis 7. Bit des A-Registers mit dem 0. bis 7. Bit des Vergleichsbytes mit Hilfe der jeweiligen logischen Verknüpfung verbunden. Das Ergebnis wird wieder im 0. bis 7. Bit des A-Registers abgelegt. Diese drei Befehle lauten folgendermaßen:

AND ...: Wenn beide Bits 1 sind, ist das Ergebnisbit ebenfalls 1, ansonsten 0.

OR ...: Wenn eines oder beide Bits 1 sind, wird das Ergebnisbit 1, ansonsten 0.

XOR ...: Nur wenn eines der beiden Bits 1 ist, wird das Ergebnisbit ebenfalls 1. Wenn beide 0 oder beide 1 sind, wird es 0.

Der letzte Logikbefehl ist der Vergleichsbefehl CP. Mit ihm wird der Inhalt des A-Registers mit dem 2. Operanden verglichen. Danach werden die Flags entsprechend dem Ergebnis gesetzt. Intern führt der Z80 bei dem Befehl CP eine Subtraktion nach dem Muster SUB A, ... durch. Dabei setzt er zwar alle Flags dem Ergebnis nach, schreibt dieses selbst aber nicht in A, sondern läßt alle Arbeitsregister unverändert. Ist nach dem Vergleich z.B. das Z-Flag gesetzt, sind beide Operanden gleich. Wurde das C-Flag gesetzt, ist der 2. Operand größer als das A-Register.

Bei allen vier Logikbefehlen kann der 2. Operand irgendein Register, ein durch (HL) oder (IX+...), (IY+...) adressiertes Speicherbyte oder eine feste Zahl im Befehl sein. Wie bei allen anderen Befehlen mit zwei Operanden bleibt auch in dieser Gruppe der Wert des 2. Operanden während der Befehlsausführung unverändert. Die jeweils beeinflussten Flags sind der Befehlsübersicht zu entnehmen.

Rotations- und Schiebepfehle

Die Rotations- und Schiebepfehle bewirken bei dem angesprochenen Byte, daß sich alle Bits um eine Stelle nach links oder rechts verschieben. Bei einem Rechtsschieben des A-Registers kommt also dessen 7. Bit

an die Stelle des 6. Bit, das 6. an die Stelle des 5. usw. Die Befehle unterscheiden sich aber nicht nur durch die Schiebe-Richtung, sondern auch durch das, was mit dem herausfallenden Bit passiert (im obigen Beispiel das 0. Bit) und was die freige-wordene Stelle ersetzt (im obigen Beispiel das 7. Bit). Dadurch wird der Unterschied zwischen Rotieren und Schieben deutlich. Bei ersterem tritt das herausfallende Bit an die Stelle des freiwerdenden, beim Schieben wird dagegen meist nur eine 0 in das freiwerdende Bit geschrieben, während das herausfallende im Carryflag abgelegt wird. Damit alle Befehle in ihrer Wirkungsweise verständlich werden, haben wir auch hier eine Tabelle mit einem Funktionsmodell des jeweiligen Befehls abgedruckt.

Das Ergebnis nach einer Rotation oder Verschiebung kommt natürlich wieder in das Byte zurück, aus dem auch das Quellbyte kam. Als Operanden kann man, wie schon bei den Logikbefehlen, irgendein Universalregister oder ein durch (HL), (IX+...), (IY+...) adressiertes Speicherbyte einsetzen. Die in der Tabelle vorkommenden Befehle ohne "..." sind in dieser Kombination nur mit dem A-Register möglich und unterscheiden sich nur in ihrer Schreibweise (sie sind ein Wort, z.B. RLCA) von den variablen Befehlen dieser Gattung (z.B. RLC A). Die in den Klammern wiedergegebenen Beispiele stellen zwar verschiedene Befehle dar, haben aber die gleiche Wirkung auf das A-Register. Einziger Unterschied ist die Ausführungszeit. Die Ein-

Art	Befehl	1. Operand	2. Operand
8-Bit ohne Übertrag	ADD	A	A, B, C, D, H, L, R oder (HL), (IX+n), (IY+n) oder direkte Zahl
	SUB		
8-Bit mit Übertrag	ADC	A	A, B, C, D, H, L, R oder (HL), (IX+n), (IY+n) oder direkte Zahl
	SBC		
16-Bit ohne Übertrag	ADD	HL	BC, DE, HL, SP
		IX	BC, DE, IX, SP
		IY	BC, DE, IY, SP
16-Bit mit Übertrag	ADC	HL	BC, DE, HL, SP
	SBC		

Wort-Befehle sind schneller, dafür aber an das A-Register gebunden. Außerdem werden bei den "langen" Befehlen mehr Flags beeinflusst.

Die Schiebepfeile können z.B. für eine Multiplikation oder Division mit 2 verwendet werden (Rechtsschieben =

Operand/2 und Linksschieben = Operand *2).

Die Bitmanipulationsbefehle

Mit den letzten in diesem Teil vorgestellten Befehlen ist nun eine direkte Manipulation ein-

zelner Bits in einem Operanden möglich. Mit diesen Bitmanipulationsbefehlen können einzelne Bits eines Operanden verändert oder abgefragt werden. Sie lauten:

SET: Setzen des x. Bits (auf 1).

RES: Löschen des x. Bits (auf 0).

BIT: Setzen des Z-Flag auf den gleichen Wert wie den des x. Bits, um danach mit einer Abfrage des Z-Flag dessen Inhalt auszuwerten.

Befehl mit Operanden	Flag's SZ H PNC	Dauer	Befehl mit Operanden	Flag's SZ H PNC	Dauer	Befehl mit Operanden	Flag's SZ H PNC	Dauer
ADC A,s	XX X XOX	4-19	INC IY		10	LDD	0 Y0	16
ADC HL,rr1	XX X XOX	15	IND	?Y ? ?1	16	LDDR	0 00	16&21
ADD A,(HL)	XX X XOX	7	INDR	?1 ? ?1	16&21	LDI	0 Y0	16
ADD A,(IX+n)	XX X XOX	19	INI	?Y ? ?1	16	LDIR	0 00	16&21
ADD A,(IY+n)	XX X XOX	19	INIR	?1 ? ?1	16&21	NEG	XX X XIX	8
ADD A,n	XX X XOX	7	JP cl,nn		10	NOP		4
ADD A,r	XX X XOX	4	JP nn		10	OR s	XX 0 X00	4-19
ADD HL,rr1	X 0X	11	JP (HL)		4	OTDR	?1 ? ?1	16&21
ADD IX,rr2	X 0X	15	JP (IX)		8	OTIR	?1 ? ?1	16&21
ADD IY,rr2	X 0X	15	JP (IY)		8	OUT (C),r		12
AND s	XX 1 X00	4-19	JR c2,n		7/12	OUT (n),A		11
BIT b,(HL)	?X 1 ?0	12	JR n		12	OUTD	?Y ? ?1	16
BIT b,(IX+n)	?X 1 ?0	20	LD rr1,(nn)		20	OUTI	?Y ? ?1	16
BIT b,(IY+n)	?X 1 ?0	20	LD rr1,nn		10	POP rr4		10
BIT b,r	?X 1 ?0	8	LD r,n		7	POP IX		14
CALL cl,nn		10/17	LD r,r		4	POP IY		14
CALL nn		17	LD (BC),A		7	PUSH rr4		11
CCF	Y 0X	4	LD (DE),A		7	PUSH IX		15
CP s	XX X XIX	4-19	LD (HL),n		10	PUSH IY		15
CPD	XY X Y1	16	LD (HL),r		7	RES n,m		8-23
CPDR	XY X Y1	16&21	LD r,(IX+n)		19	RET		10
CP1	XY X Y1	16	LD r,(IY+n)		19	RET cl		5/11
CPIR	XY X Y1	16&21	LD (IX+n),n		19	RETI		14
CPL	1 1	4	LD (IY+n),n		19	RETN		14
DAA	XX X X X	4	LD (IX+n),r		19	RL m	XX 0 XOX	8-23
DEC m	XX X X1	4-23	LD (IY+n),r		19	RLA	0 0X	4
DEC rr1		6	LD A,(nn)		13	RLCA	0 0X	4
DEC IX		10	LD (nn),A		13	RLC r	XX 0 XOX	8
DEC IY		10	LD (nn),rr1		20	RLC (HL)	XX 0 XOX	15
DI		4	LD (nn),HL		16	RLC (IX+n)	XX 0 XOX	23
DJNZ n		8/13	LD (nn),IX		20	RLC (IY+n)	XX 0 XOX	23
EI		4	LD (nn),IY		20	RLO	XX 0 X0	18
EX AF,AF'	XXXXXXXX	4	LD A,(BC)		7	RR m	XX 0 XOX	8-23
EX DE,HL		4	LD A,(DE)		7	RRA	0 0X	4
EX (SP),HL		19	LD A,I	XX 0 Y0	9	RRC m	XX 0 XOX	8-23
EX (SP),IX		23	LD I,A		9	RRCA	0 0X	4
EX (SP),IY		23	LD A,R	XX 0 Y0	9	RRD	XX 0 X0	18
EXX		4	LD HL,(nn)		16	RST n		11
HALT		4	LD IX,nn		14	SBC A,s	XX X XIX	4-19
IMO		8	LD IX,(nn)		20	SBC HL,rr1	XX X XIX	15
IMI		8	LD IY,nn		14	SCF	0 01	4
IM2		8	LD IY,(nn)		20	SET n,m		8-23
IN r,(C)	XX X X0	12	LD R,A		9	SLA m	XX 0 XOX	8-23
IN A,(n)		11	LD r,(HL)		7	SRA m	XX 0 XOX	8-23
INC m	XX X X0	4-23	LD SP,HL		6	SRL m	XX 0 XOX	8-23
INC rr1		6	LD SP,IX		10	SUB A,s	XX X XIX	4-19
INC IX		10	LD SP,IY		10	XOR s	XX 0 X00	4-19

Kürzel bei den Befehlsoperanden

r = A, B, C, D, E, H, L
rr1 = BC, DE, HL, SP
rr2 = BC, DE, IX oder IY
rr3 = BC, DE, HL, IX, IY
rr4 = BC, DE, HL, AF
n = 2stellige unmittelbare Daten
nn = 4stellige unmittelbare Daten
= n+r+(HL),(IX+n),(IY+n)
m = wies, nur ohne n
b = Zahlen von 0-7
cl = NZ, Z, NC, C, PO, PE, P, M
c2 = NZ, Z, NC, C
r' = Zweitregistersatz

Kürzel bei den Flagbits

1 = Nach Befehlsende auf 1
0 = Nach Befehlsende auf 0
X = je nach Ergebnis gesetzt
Y = je nach Ergebnis als Sonderfall
? = nicht vorzubestimmen
= bleibt unverändert

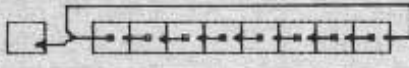
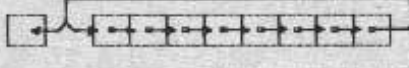
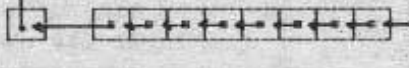
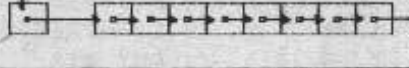
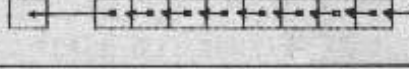
Kürzel bei den Taktdauern

/ = 1. Wert oder 2. Wert
- = 1. Wert bis 2. Wert möglich
& = 1. Wert und x mal 2. Wert

Nach dem Befehl (SET, RES oder BIT) folgt zuerst die Bitnummer (0-7) des gewünschten Bits und danach, mit einem Komma getrennt, der Operand. Dieser kann wieder wie bei der letzten Befehlsgruppe entweder ein Universalregister oder ein adressiertes Speicherbyte sein. So bewirkt z.B. der Befehl SET 04,D, daß das 4. Bit des D-Registers auf 1 gesetzt wird. Da der Befehl BIT nur zum Prüfen eines Bits da ist, folgt danach meist ein bedingter Sprung mit Abfrage des Z-Flag.

Ich glaube, mit den diesmal besprochenen Befehlen und Grundlagen hat sich allen Mitlesern wieder ein großes Experimentierfeld geöffnet. Im nächsten Teil wird neben einigen neuen Befehlen auch ein mit den heutigen Befehlen geschriebenes Programm erklärt, welches die Ermittlung einer Bildschirmpunktadresse vereinfacht.

T. Bertoldo

C-Flag + Bearbeitetes Byte	Befehl	Genaue Bezeichnung
	RLC ... RLCA	Rotiere links zirkular
	RRC ... RRCA	Rotiere rechts zirkular
	RR ... RLA	Rotiere links
	RR ... RRA	Rotiere rechts
	SLA ...	Schiebe links arithm.
	SRA ...	Schiebe rechts arithm.
	SRL ...	Schiebe rechts logisch

Assemblertips für den Spectrum

Teil 10: Adreßunabhängige Maschinenprogramme

Im fünften Teil dieser Serie (CK 1/86) haben wir ein Utility kennengelernt, das Maschinenprogramme an verschiedene Adreßbereiche anpaßt. Selbst entwickelte Maschinencoderoutinen sollten jedoch so ausgelegt werden, daß man auf ein derartiges Anpaßprogramm nicht angewiesen ist, sie also ohne Änderungen in jedem Adreßbereich lauffähig sind. Im folgenden wollen wir sehen, wie dieses Ziel zu erreichen ist.

Dazu führen wir uns noch einmal vor Augen, daß ein normales, längeres Z80-Programm wegen der absoluten Adressen, die im Programmcode vorkommen, im allgemeinen fest an eine Laufadresse gebunden ist. Diese tauchen in Lade- und Sprungbefehlen sowie in UP-Aufrufen auf. Wir müssen nun diese absoluten Adressen aus dem Programm verbannen. Bei LD-Kommandos ist das noch recht einfach. Diejenigen LD-Befehle, welche auf Speicherbereiche zielen, die nicht zum Programm gehören, brauchen natürlich nicht modifiziert zu

werden. Wir müssen nur die Anweisungen angehen, die direkt auf Daten innerhalb des verschiebbaren Maschinencodes zugreifen. Hierfür machen wir uns zunutze, daß die USR-Funktion, dem aufgerufenen Maschinenprogramm die eigene Startadresse im BC-Registerpaar zur Verfügung stellt.

Gleich zu Beginn unseres Programms retten wir den BC-Inhalt an einen sicheren Ort, etwa in eine Basic-Systemvariable, von der wir wissen, daß sie während der Ausführung unseres Programms nicht verändert wird. Jeder Zugriff auf Daten innerhalb des Programms muß so vonstatten gehen, daß zur Adreßbestimmung die Summe aus der geretteten Startadresse und der programmrelativen Adresse (= Abstand zur Startadresse) der Daten gebildet wird. Sofern der Datenbereich klein ist, mag es auch genügen, am Programmstart die Bereichsadresse in das X-Register zu bringen (wieder über den Mechanismus mit der Startadresse in BC und der pro-

grammrelativen Adresse). Alle kritischen LD-Anweisungen sind dann X-indiziert durchzuführen. Zeigt X genau auf die Mitte des Datenbereichs, kommt man an 256 Bytes heran (IX-128 bis IX+127), ohne das Indexregister verändern zu müssen.

Will man jedoch einen JP-Befehl, der eine weite Distanz überbrückt, durch relative Sprünge ersetzen, benötigt man eine ganze Kaskade von JR-Anweisungen, wodurch das Programm aufgebläht und ausgesprochen unübersichtlich und wartungsunfreundlich wird. Ein CALL läßt sich in vergleichbarer Weise gar nicht substituieren. Ich möchte Euch hier zwei unterschiedliche Lösungsvorschläge machen, die beide ihre Vor- und Nachteile besitzen. Das Assemblerlisting zeigt beide Ansätze. Die erste Lösung ist sehr kurz, aber der einzelne Unterprogrammaufruf ist unschön. Das Programm schreibt zu Beginn eine kleine Hilfsroutine in den Bereich MEMBOT. Es ließe sich auch

jeder andere freie Speicherbereich benutzen, die Adresse muß nur für das Programm konstant sein. Jeder UP-Aufruf und jeder absolute Sprung läuft über diese Routine, indem die programmrelative Adresse des Zielpunktes in HL geladen und CALL MEMBOT bzw. JP MEMBOT ausgeführt wird. Die Hilfsroutine berechnet die endgültige Zieladresse und macht einen Sprung dorthin. Diese Methode ist verhältnismäßig schnell, weil pro Sprung nur wenige Kommandos zu durchlaufen sind, weist jedoch einige Nachteile auf. So wird ein Registerpaar benutzt, das damit nicht der Parameterübergabe zur Verfügung steht. Zusätzlich "verbrät" jeder Sprung sechs Bytes. Damit kommt dieses Verfahren vor allem für Programme mit nur wenigen zu ersetzenden JPs/CALLs und für zeitkritische Anwendungen in Frage.

Lösung Nummer 2 ist eleganter und benötigt nur zwei Bytes für einen CALL, der dafür aber leider langsamer ist. Jeder

CALL einer programmeigenen Routine wird durch das Kommando RST \$10, gefolgt von einem Byte, ersetzt. Dieses Folgebyte gibt die Nummer des gewünschten UPs an. In der Reihenfolge der Numerierung sind die programmrelativen Anfangsadressen der UPs in der Liste UPTABL eingetragen. RST \$10 wird normalerweise vom Spectrum-System zur Ausgabe von Zeichen über irgendeinen Strom benutzt. Der Inhalt der Systemvariablen CURCHL gibt die Adresse der Kanalbeschreibung an, die mit dem gerade geöffneten Strom verbunden ist. Die ersten beiden Bytes der Beschreibung enthalten die Adresse der Output-Routine, die zum Kanal gehört. Und eben dorthin verzweigt letztendlich der RST-\$10-Befehl. Unser Programm biegt den Pointer CURCHL so um, daß man durch RST \$10 in der eigenen Routine UPCALL landet.

UPCALL korrigiert zunächst die Register- und Stacksituation, die durch die gerade durchlaufene ROM-Routine etwas durcheinandergeraten

ist, denn die Output-Routinen arbeiten mit dem zweiten Registerset, und HL' muß auf dem Stack zwischengespeichert werden. Sodann liest UPCALL die Nummer des gewünschten UPs, holt sich die zugehörige relative Adresse aus der Liste UPTABL, addiert dazu die Adresse START, um schließlich zu dieser echten UP-Adresse zu verzweigen. Somit wird immer ein CALL zur gewünschten Adresse ausgeführt, kein JP. Man könnte jedoch UPCALL zum Beispiel so erweitern, daß etwa zu den Adressen mit den Nummern von 0 bis 19 immer ein JP und zu den übrigen ein CALL stattfindet, oder man löscht am Zielpunkt einfach die überflüssige Rücksprungsadresse auf dem Stack.

Das Assemblerlisting zeigt einige Beispiel-Ups, um die Vorgehensweise zu verdeutlichen. Insbesondere die Print-Routinen dürften von Interesse sein, da an ihnen deutlich wird, wie man RST \$10 auch weiterhin zur Zeichenausgabe nutzen kann.

Michael Schramm

Listing 1

```
; ***** COMPUTER KONTAKT 1986 *****
; ***** Michael Schramm *****

; * frei verschiebbares Programm
; * einfache, schnelle Version
; * UP-Aufruf speicheraufwendig
```

```
MEMBOT = 23698
OPEN = $1601
```

```
ORG 30000 ;Startadr. beliebig

START LD HL, MEMBOT ;Im Bereich MEMBOT
LD (HL), $C5 ;den unten aufge-
INC HL ;führten Maschinen-
LD (HL), $01 ;code erzeugen.
INC HL ;BC enthält die
LD (HL), C ;Adresse START (das
INC HL ;bringt der USR-
LD (HL), B ;Mechanismus mit
INC HL ;sich).
LD (HL), $09
INC HL
LD (HL), $C1
INC HL
LD (HL), $E9
```

```
TEST LD HL, UP1-START ;Aufruf eines UP
CALL MEMBOT ;mit relativer
LD HL, ENDE-START ;Adreßangabe und
JP MEMBOT ;ein entsprechenden
;der Sprung.
```

```
UP1 LD A, 2 ;Strom 2 öffnen
CALL OPEN ;(Ausgabe auf den
LD A, "A" ;Bildschirm) und
RST $10 ;ein paar Buchsta-
LD A, "B" ;ben drucken als
RST $10 ;"Beweis" dafür,
LD A, $0D ;daß dieses UP
RST $10 ;erreicht wird.
RET
```

```
ENDE LD A, "X" ;Auch hier einige
RST $10 ;Buchstaben
LD A, "Y" ;ausgeben.
RST $10
LD A, $0D
RST $10
RET
```

END

Listing 2

;Die folgende Routine wird im Bereich
;MEMBOT generiert.

```
;
; PUSH BC ;BC-Inhalt retten.
; LD BC, START ;Mit Hilfe der "einge-
; ADD HL, BC ;POKeten" Adresse START
; ;aus der relativen die
; POP BC ;absolute Adr. berechne.
; JP (HL) ;Sprung an die
; ;gewünschte Adresse.
```

; * aufwendigere Version mit sehr kurzen

; * Aufrufmechanismus (2 Byte je Aufruf)

```
ERRSP = 23613
CURCHL = 23633
BREG = 23655
HELPC = 23728
```

```
ORG 40000 ; beliebig
```

```
START LD HL, OFFSET+1-START
ADD HL, BC ;In das Unterprg
LD (HL), C ;UPCALL die akt.
INC HL ;Adreßdifferenz
LD (HL), B ;einschreiben.
LD HL, UPCALL-START
ADD HL, BC ;Adr. UPCALL in
```

```

LD (HELPCH),HL ;HELPCH.
LD HL,HELPCH
LD (CURCHL),HL
LD HL,UPTABL-START
ADD HL,BC
EX DE,HL
LD HL,TABADR+1-START
ADD HL,BC ;Adresse UPTABL
LD (HL),E ;errechnen und
INC HL ;in das UP UP CALL
LD (HL),D ;einschreiben.
RST #10 ;BEEP-UP
DEFB 1 ;aufrufen.

RST #10 ;PRINT-UP.
DEFB 0
DEFT "Test-Text"
DEFB 13 ;CR
DEFB 255 ;Endemarke

RST #10 ;Noch ein BEEP.
DEFB 1
LD SP,(ERRSP) ;Stack löschen und
JP #1B76 ;Sprung in den
;Basic-Interpr.

UPTABL DEFW PRINTST-START ; UP 0
DEFW BEEP-START ; UP 1
DEFW PRINT2-START ; UP 2
DEFW PRINT3-START ; UP 3

UPCALL LD (BREG),A ;Akku merken.
POP HL ;Rücksprung-
;adresse löschen.
POP HL ;Alter Wert f. HL'
EXX ;Wieder 1. Reg.set
EX (SP),HL ;UP-Nummer-Byte
LD A,(HL) ;in den Akku.
INC HL ;Rücksprungadresse
EX (SP),HL ;erhöhen.
PUSH HL ;HL retten.
LD H,0 ;Akkuinhalt ver-
SLA A ;doppeln und in HL
RL H ;übertragen, dann
LD L,A ;A wiederher-
LD A,(BREG) ;stellen.
PUSH DE ;Echte Adresse
TABADR LD DE,0 ;in UPTABL
ADD HL,DE ;errechnen,
LD E,(HL) ;Tabelleneintrag
INC HL ;lesen und
LD D,(HL) ;Offset dazu-

OFFSET LD HL,0 ;zählen.
ADD HL,DE ;Alte Reg.inhalte
POP DE ;wiederherstellen
EX (SP),HL ;und Sprung in das
RET ;gewünschte UP.

PRINTST EX (SP),HL ;Dieses UP
LD A,(HL) ;druckt den
INC HL ;Text, der
EX (SP),HL ;hinter dem
CP 255 ;UP-Aufruf
RET Z ;steht.
RST #10 ;255-Byte ist
DEFB 2 ;Endemarke.
JR PRINTST

BEEP PUSH AF ;Aufruf der
PUSH BC ;BEEP-Routine
PUSH DE ;im ROM.
PUSH HL ;DE bestimmt die
LD DE,100 ;Tonlänge,
LD HL,200 ;HL die
CALL #03B5 ;Tonfrequenz.
POP HL
POP DE
POP BC
POP AF
RET

PRINT2 PUSH AF ;PRINT CHR$ Akku;
LD A,2
JR OPEN

PRINT3 PUSH AF ;LPRINT CHR$ Akku;
LD A,3
OPEN EX (SP),HL
PUSH HL
LD HL,(CURCHL)
PUSH HL
PUSH DE
PUSH BC
CALL #1601 ;OPEN-Routine
POP BC ;im ROM, sie
POP DE ;setzt CURCHL.
POP HL
POP AF
RST #10 ;Zeichen ausgeben.
LD (CURCHL),HL ;Alten Wert von
POP HL ;CURCHL wiederher-
RET ;stellen für den
;UP-Aufrufmecha-
;anismus.

END

```


Tips und Tricks für Spielefreaks

Wir haben wieder mal zahlreiche Unsterblichkeitspokes und Spieletips erhalten, die wir unseren Lesern natürlich nicht vorenthalten wollen. Nachstehend also eine Liste, in der alte und neue Spiele berücksichtigt wurden. Alle Angaben sind aber ohne Gewähr, da wir nicht jeden Poke überprüfen können. Für Neulinge unter den Spectrum-Besitzern, die nicht genau

wissen, was sie damit anfangen sollen, verweisen wir auf das CK-Heft 6-7/86. Dort wird der Einsatz der Unsterblichkeitspokes genau geklärt. Weitere Tips dieser Art haben wir in Heft 1/86 abgedruckt.

Wer Tips und Tricks kennt, die wir bisher noch nicht berücksichtigt haben, sollte uns diese zuschicken, damit alle Leser davon profitieren können.

Abu Simbel Profanation

44075,195 Unsterblichkeit
45295,195 alles ungefährlich

Alchemist

47340,0 Unsterblichkeit
49745,195 Trap ungefährlich

Alien 8

51736,0 unendlich Leben
42592,255:42593,255 Zeit

Aquaplane

25448,0:25449,195

Automania

64589,4:64590,60

Black Crystal

MAP2-1126690200
MAP3-1126671220
MAP4-3126641220
MAP5-2126671220
MAP6-1126290200

Blue Thunder

Adr. 48011 = Proficompiler (RAND USR 48011)

Bombberman

33248,0 Unsterblichkeit
32851,X X=Anzahl Leben

Bored of the Rings I

START, OUT OF TENT, W, W, W, E, E, S, S., HIDE, N, S, DRINK ALE, N, N, N, E, E, E, S, D, E, N, N, SAY HELP, E, E, IN, OUT, S, S, W, W, N, E, IN, IN, OUT, OUT, E, E, E, E, N, U, TAKE PAPER, D, S, E, E, E, E, IN, E, S, E, E, N, E, DROP PEPPER!

Bored of the Rings II

E, E, GET BATTERY, W, INSERT BATTERY, GET COIN, S, LIFT MAT, GET KEY, UNLOOK DOOR, GET RIFLE, D, GET TREASURE, U, E, GIVE COIN, GET MAP, E, E, N, E, E, S, W, S, E, GET POSTER, W, W, S, W, DROP POSTER, N, W, PRESS RED, S, S, W, GET ROPE, E, E, PLANT BEAN, E, S, CLIMB TREE, EXAMINE MIRROR, D, S, W, EXAMINE MIRROR, E, E, PUMP RAFT, WAIT, WAIT, WAIT, WAIT, SHOOT C, PUMP DRAFT, 6*WAIT, GET COIN, W, UP, UP, INSERT COIN, D, D, E, PUMP RAFT!

Braxx Bluff

23730,255:23731,255

Im Basic-Teil vor dem RAND-Befehl. Programm stürzt in der 3. Phase nicht ab.

Bruce Lee

51795,0 Unendlich Leben 1. Spieler
51803,0 Unendlich Leben 2. Spieler

Cauldron

Unendlich Leben:

10 CLEAR 24599:FOR F=23296 TO 23309:READ A:POKE F,A:NEXT F

20 LET L=USR 23296:POKE 40060,0:LEFT L=USR 24600

30 DATA 221, 33, 24, 96, 17, 232, 159, 62, 255, 55, 205, 86, 5, 201

RUN, Hauptteil laden

Cavelon

24019,146
24080,184

Im Labyrinth "JS Willy" eingeben, danach eine Zahl von 1-6 (=entsprechender Schirm).

Cosmic Cruiser

25373,0

Danger Mouse II

Code-Nr.=7842981

Death Chase

26463,0

Death Star Interceptor

36714,0:38348,0 unendlich Schiffe
36679,0:38170,0 perfektes Schutzschild

Dynamite Dan

52678,0

Entr. to Heades

31394,28:31395,18 Sprung ins Basic, dann RAND USR 29848

Everest Ascent

Mit "ljh590x" kommt man ins Listing.

Everyone's a Wally

58200,183
58217,0:58218,0:58219,0 unendlich Leben (gilt nur für die vollständig gecrackte Version).

Fairlight

61931,201

Fighter Pilot

Landung: Airspeed auf ca. 130-140 Knots, VSJ auf -5 oder -7, Altitude = 100 bis 120

FP/IS Compiler

RAND USR 59300

Frank Bruno's Boxing

Name=Bob, EVBA-membership-code=NG8INI394, Code-Nr. für alle Boxgegner:

Canadian Crusher	STC	IOCHIOIO
Fling-Long-Chop	STC	IS6HIMIO
Andra Puncheredov	STC	IMAHKIO
Tribal Trouble	STC	IQ4HISIO
Frenchie	STC	IK8HIOIO

Ravioli Mafiosi STC INBIIIIOIO
Antipodeon Andy STC IR5IIIMIO
Peter Perfect STC IL9IIIKIO

Frank N Stein

33723,0 Unsterblichkeit
28265,X X=Level (1-15)
27277,X X=Anzahl Leben
23613,((PEEK 23730)-5) BREAK aus
23613,((PEEK 23730)-3) BREAK ein
34124,0

10 BORDER 0:INK 0:CLEAR 24750:LOAD""SCREEN\$: PA-
PER 0:PRINT AT 0,6:LOAD"" CODE:POKE 28287, 255
:RANDOMIZE USR 27890 = unendlich Leben

Fred

unendlich Leben mit

10 CLEAR 24999:REM FREDLOADER
20 DATA 17, 0, 64, 33, 168, 97, 1, 46, 120, 237, 176, 19, 33, 46,
184, 1, 180, 71, 54, 0, 237, 176, 33, 195, 121, 54, 0, 195, 77, 118
30 LOAD "FRED" CODE 25000, 30766
40 FOR F = 65506 TO 65536:READ A
50 POKE F,A:NEXT F
60 RANDOMIZE USR 65506

Gift from the Gods

54459,62:57460,100 unendliche Stammina

Ghostbusters

40191,0:40192,167 Unsterblichkeit
40625,0:40845,0 Unsterblichkeit
42173,0 Unsterblichkeit

Geldcodes:

Ghostbusters = 00166605 = 850.000\$
Robbi = 00741001 = 40.000\$
Schuster,F. = 01010304 = 104.100\$
HL = 70204700 = 23.800\$

Return drücken; auf die Frage "Have You an Account" mit "Y"
antworten und 22444404 = 125.000\$.

Gremlins

SWORD tötet Gremlin am Start.
KNIFE tötet pfeilwerfenden Gremlin,
 schneidet die Düse.
FLASHLIGHT Damit bekommt man die Strippe aus
 dem Briefkasten.
HACKSAW schneidet die Düse.
JOINTING TAPE verbindet die zwei Düsen.
ELECTRIC DRILL bohrt ein Loch in die Metallplatte.
LADDER Damit kommt man in die Dachkammer
 des Departmentstore.
CAMERA jagt Gremlins aus der Bar und auch aus
 der Lüftung.
GIZMO öffnet eine verschlossene Tür.
SPARK IGNITER zündet das Schweißgerät an
TORCH-BOTTLE & das Schweißgerät.
PIPE

H.E.R.O.

10 INK 0:PAPER 0:BORDER 0:CLS
20 LOAD "" CODE 16384
30 POKE 23323,201
40 RANDOMIZE USR 23290
50 POKE XXXXX,XXX
60 RANDOMIZE USR 44106

An die Stelle "XXXXX,XXX" können folgende Pokes gesetzt
werden:

44322,182:44521,182 Unsterblichkeit

44017,X X=Anzahl der Leben
54918,183 unendlich viele Bomben
44088,X X=Anzahl der Bomben

Hulk

1. Bild = BITE LIP
2. Bild = LOOK SIGN
3. Bild = PUSH BUTTON = ON, dann BITE LIP, GO OUT,
Gems sammeln mit TAKE GEM. REMEMBER
NIGHTMARE für 3 BL wegnemen ohne Verwandlung.

Jetpac

25020,0 Unsterblichkeit
25373,X X=Anzahl der Leben (max. 255)
26075,0 nur einen Kanister Benzin für den Start
 der Rakete

25018,0

Jumbly

Namen der Bilder:
Nursery, Flight, Plans, Window, Verse, Vertigo

Jet Set Willy

34483,195:34499,201 ohne Schutzcode
34493,195 Spielbeginn ohne Code-Eingabe (2*XYE)
34785,X-1 X=Leben (max. 32)
34795,X X=Startraum (normal 32)
35123,0 Monster verschwinden
35899,0 Unsterblichkeit
36358,0 Sprung, bis Willy ein Hindernis trifft
36477,0 kein Tod durch tiefes Fallen
36545,0 "THE BANYAN TREE" wird einfacher
36635,239 Interface-2-kompatibel
37874,0 Objekte beim Betreten gesammelt
38240,0 Martha verschwindet = Spielende
39998,0 bis 40191,0 Fass, Martha + Fuß verschw.
41983,256-X X=Anzahl der Flaschen (normal 83)
41983,X X=Objekte, die gesammelt werden
 müssen.
46896,0 bis 49171,0 keine Viecher mehr
59900,255 kein Fehler in "THE ATTIC" mehr

Nach dem Programmstart geht man die Treppe hinunter in die
"FIRST LANDING". Dort stellt man sich in etwa zwischen die
beiden Wörter und gibt "WRITETYPY" ein. Ähnlich wie bei
"Manic Miner" erscheint jetzt ein Schuh. Durch die Eingabe be-
stimmter Zahlenfolgen gelangt man nun in die 60 Räume des Pro-
gramms. Man sollte jedoch vorsichtig sein, wo man sich hinstellt,
da dieser Trick in jedem Raum angewandt werden kann. Sollte in
so einem Raum kein Boden sein, stürzt man sich zu Tode.

Jet Set Willy II

31254,195 Unsterblichkeit
31215,201 Durchlaufen
30027,X X=Startraum -1 = 33

Damit ein weiteres Durchlaufen möglich ist, wird mit folgenden
Datas zurückgesetzt:

4096 FOR Y=31215 TO 31227
4097 READ Z
4098 POKE Y,Z
4099 NEXT Y
4100 STOP
4101 DATA 62, 239, 219, 254, 203, 103, 192, 193, 225
4102 DATA 195, 164, 122, 0

Wenn man ohne Code-Tabelle spielen will:

10 FOR A=64256 TO 64277
20 READ X
30 POKE X,A
40 NEXT A

50 DATA 221, 33, 064, 17, 56, 185, 55, 62, 255, 205, 86, 5, 33, 0,
0, 34, 132, 97, 195, 0, 95
60 RANDOMIZE USR 64256

Anschließend RUN, Hauptteil laden, Codeabfrage=4*YXZ,
ENTERN

Knight Lore

44803,X X=Anzahl der Leben
49759,X X=Anzahl der Objekte sammeln
50084,201 unbegrenzte Zeit und nicht Werwolf
50206,0 unendliche Tage
53567,0 Unsterblichkeit

Lazy Jones

56693,0

Lunar Jetman

36966,244:36945,3 Unsterblichkeit
37999,201 keine Feinde
36965,3 Unsterblichkeit
36965,0 unendlich Leben
36964,244:36965,3 unendlich Leben
43092,X-1 (X max. 99) Stufe

Manic Miner

35136,0 Unsterblichkeit (BREAK im Basic-La-
der) (PAPER 7; 35 POKE 35136,0;
RUN)
33799,X-1 Screenwahl (X=gewünschter Screen)

Man kann auch nach dem Programmstart die Zahlen 6031769 ein-
geben. Es gibt ein System für die Wahl des Screens. Umwandlung
der Nummer des gewünschten Screens minus 1 ins binäre Zahlen-
system und die Tasten drücken, bei denen binär eine "1" steht. Da-
bei entsprechen die Zifferntasten 1, 2, 3, 4, 5 den Binärzahlen 1, 2,
4, 8, 16. Zusätzlich muß immer die Ziffer 6 gedrückt werden.

Beispiele:	Screen	Tasten
	1	6
	2	6,1
	3	6,2
	4	6,1,2
	5	6,3 usw.

Monty Mole

37512,0:47639,255:
47640,0 Unsterblichkeit
36301,201 keine Feinde
38004,0 unendlich Leben
35874,255

Moon Alert

42249,24 unbegrenzte Zeit
42585,2:52596,2 Screen rollt schnell
37035,201 keine Feinde
39754,0 Unsterblichkeit
39756,0 Unsterblichkeit
42404,0 Unsterblichkeit
42404,X X=153-99 Autos am Start
42654,195

Mr. Wimpy

33693,0 Unsterblichkeit
33509,X X=Anzahl der Leben
33721,X X=Anzahl der Schüsse
43105,0 unendlich viele Schüsse
33501,0 1. Bild wird weggelassen

Mugsy

Auf die Frage: "How much You wish to spend on artillery and am-
mo? Sofort 999999999 eintippen = mehrere "Millions of Dollars".

Mutant Monty

54800,0:54867,0:54893,0:55321,0
54933,0

Nightshade

53442,0:53443,0 Unsterblichkeit
Viele Häuser betreten = viele Punkte

Orion

37319,201

PI-Balled

46441,0 Unsterblichkeit
46457,0 Unsterblichkeit
44416,X X=Anzahl der Leben

Pogo

44259,0:44260,33

Push off

26862,167

Raid over Moscow

10 CLEAR 65535:POKE 65535,195
20 PRINT AT9,6; "ORIGINAL-CASSETTE VOM ANFANG
AN STARTEN"
30 FOR F=45043 TO 45081:READ A:POKE F:A:NEXT F
40 DATA 245, 229, 58, 239, 95, 254, 63, 32, 15, 237, 86, 237, 71,
33, 107, 107, 34, 230, 96, 62, 62, 50, 217, 96, 62, 147, 50, 212, 96,
195, 58, 0, 62, 59, 237, 71, 237, 94, 201
50 RANDOMIZE USR 45075
60 MERGE """:LOAD""

Nach dem Laden ist das Programm im Basic. Jetzt für unendliche
Leben POKE 29252, 183, dann RAND USR 32768 eingeben.

Rattler

44092,0:44093,0

River Rescue

33426,0:33452,0

Robot Rumpus

50939,195

Sabre Wulf

43575,255 1. Spieler wird unsterblich
45520,255 2. Spieler wird unsterblich
39393,0 keine Tiere und Krieger mehr
39702,30 keine Monster erscheinen
43575,0 unendlich Leben

Sam Spade

25215,0 Unsterblichkeit
26381,0 unendlicher Sauerstoff

Scuba Dive

55711,X X=Anzahl der Leben
55711,255:55779,255

Sir Lancelot

Folgende Datas für Unsterblichkeit:

10 FOR Y=23300 TO 23329
11 READ Z
12 POKE Y,Z
13 NEXT Y
14 STOP
15 DATA 221, 33, 128, 91, 17, 128, 36, 175, 61, 55
16 DATA 205, 86, 5, 175, 50, 85, 93, 50, 84, 93
17 DATA 50, 86, 93, 50, 253, 127, 243, 195, 8, 92

Anschließend RUN, RAND USR 23300, Hauptteil laden ohne
SCR - headerless!

Spectral Panic

28522,0

Spy Hunter

54494,195:54495,139:54496,208

55711,255:55779,255 unendlich Leben

Starquake

Die Teleport-Codes heißen:

VREOX, EXIAL, ALGOL, DELTA, TULSA, ASOIN, ULTRA, SONIQ, AMAHA, KYZIA, AMIGA, RAMIX. Hat man alle 9 Teile gesammelt, muß man den Code "QUAKE" benutzen.

Stop the Express

34464,183:34926,183:35257,0

Strip Poker

Ohne Pokerspiel, nur Strip:

```

1 CLEAR 42744
2 LOAD "poker" CODE 42800,22567
3 LET M=150:PAUSE M:LET L=USR 43306:PAUSE M:LET
  L=USR 43152:PAUSE M:LET L=USR 43098:PAUSE M:
  LET L=USR 42859
  RUN

```

Super Spy

Mit "ljh590x" kommt man ins Listing.

Tasword II

64516,1:PAPER 1:INK 7

Ergibt im Basic helle Buchstaben auf dunklem Paper und dunklem Border. Setzt man BRIGHT 1 in die erste Basic-Zeile, wird die Schreibfläche heller und der Rand wird gut erkannt.

Terror Daktil

37629,0

Terrormolinos

Bei diesen Gelegenheiten müssen die zehn Bilder gemacht werden: 3x am Strand, 1x wenn Doreen die Bombe findet, 1x auf der Plaza, 1x im Nightclub, 1x vor dem Tower in den Monestry, 1x im Weinkeller, wenn die Spinne erscheint, 1x beim Stierkampf, 1x beim Bootsverleih.

The House that Jack build

26822,0 unendlich Leben

Spielen ohne Code-Karte mit CLEAR 24249:LOAD""CODE (HAUPTTEIL):POKE 24610,0 POKE 24606,80:POKE 23607,248:RAND USR 24520

Tornado Low Level

35006,0

unendlich Leben und Zeit:

```

20 DATA 55, 62, 255, 221, 33, 0, 64, 17, 156, 191, 205, 86, 5, 62,
  0, 50, 190, 136, 50, 15, 132, 201
3000 RANDOMIZE USR 65423

```

Trashman II

38656,142

Transam

25446,0

Unsterblichkeit

Travel W. Trashman

33967,0:33968,128

mehr Startkapital

Tir Na Nog

Nach dem Tod Nr. 6 "RESTART" drücken und man beginnt an der gleichen Stelle wieder.

34202,200

befreit von den Side Guards

Tutankamun

27783,0

Unsterblichkeit

Underwulde

38041,0:38042,0

45019,201

unverletzbar nach Kristall

RID OF MORNING

(alle Feinde verschwinden)

Unsterblichkeit

X=Anzahl der Leben

Wheelie

Codes für die Bilder 1-8:

1-ENTER, 2-WITTY, 3-SHARK, 4-BEBOP, 5-XENON, 6-ZX83B, 7-ZMQL3, 8-HRMEZ

Worse Things happen

35443,0

Dieter Thesen

A View to a Kill30 POKE 28032,255: POKE 28087,8: POKE 29243,8:
POKE 30543,8**Booty**

10 BORDER 0: PAPER 0: INK 0

20 CLEAR 26870

30 LOAD""SCREEN\$

40 LOAD""CODE 26880

50 RANDOMIZE USR 26880

60 POKE 58294,0

70 RANDOMIZE USR 52500

Fantastic Voyage

10 CLEAR 30791: LOAD "VOYAGE" CODE:

POKE 54492,167: BORDER 0: PRINT USR 53248

Project Future

POKE 27662,2

Rocky Horror Show

Für unendliches Zeitlimit eingeben, danach MC laden:

10 CLEAR 65000

20 LOAD""CODE

30 POKE 65310,205: POKE 65311,211: POKE 65312,255

40 FOR A=65491 TO 65498: READ B: POKE A,B: NEXT A

50 RANDOMIZE USR 65281

100 DATA 62, 0, 50, 181, 190, 195, 190, 131

Sweevo's World

POKE 37008,255

für 255 Leben

Daniel Schmidt

Jupitersoft GdBRZX Spectrum Software 16/48K **ab 10 DM**Adventures, Action,
Anwender, Grafik
und Dateiprogramme.**Viel Software stark
im Preis reduziert.**

Kostenloser Katalog bei

**Bernd Denk
Jupitersoft GdBR**

Frühlingsstr. 12

8831 Weiboldshausen

(09141) 2249

ZX-Spectrum - ZX-Spectrum

DISCOVERY 180 DM 388.-

SANYO-MONITOR DM 249.-

(z.B. f. Discovery)

PANASONIC-Drucker DM 698.-

KX-P 1080

KEMPSTON-E-Drucker- DM 149.-

interface

Testatur (dk'ronics) DM 118.-

dk'ronics-Drucker- DM 98.-

interface

48k-Erweiterung DM 49.-

Joystickinterface DM 29.-

GAMES PLAYER

Joystickinterface mit Regelung der Spiel-

geschwindigkeit DM 39.-

SPRACHSYNTHESIZER

(dk'ronics) DM 79.-

3-KANAL-SOUND-

SYNTHESIZER DM 79.-

Liste GKB mit weiteren Angeboten anfordern oder

direkt bestellen bei

U. KUNZ

Junge Hilden 3,
D-7500 Karlsruhe 41
Tel. Bestellannahme
07 21 / 40 26 78 (bis 20 Uhr)

Jahrmarkt (Spectrum 48K)

Falls Sie es leid sind, in immer neuen Versionen als Pacman durch ein Labyrinth düsen zu müssen und es ebenso satt haben, neue Wellen außerirdischer Angreifer abzuwehren, dann könnte vielleicht dieses Programm für etwas Abwechslung sorgen.

Geben Sie das 4,8K lange Basicprogramm ein und Sie werden mit einem Augenschmaus belohnt. Ihr Spectrum bringt Leben in die Szenerie eines Jahrmarkts und braucht dafür

nur eine 12 Bytes lange MC-Routine, da eine simple Blockverschiebung die gesamte Bewegung steuert. Nachdem Sie RUN eingegeben haben, liest der Computer die MC-Daten und die 8 User defined Graphics ein. Dann zeichnet er 5 Einzelbilder für die Animation. Nachdem diese Bilder über RAMTOP abgespeichert wurden, beginnt die Animation. Lehnen Sie sich zurück und lassen Sie sich überraschen. Mit der MC-Routine können Sie übrigens auch selbst eigene Animationen gestalten.

Dietmar Volckmann

Jahrmarkt

```
10 CLEAR 30463: GO SUB 8000
40 BORDER 1: LET g=16: LET r=7
1: LET p=119
50 LET mx=127: LET my=104: POK
E 23302,64
60 FOR n=0 TO 4: BRIGHT 0: CLS

70 REM
71 REM RIESENRAD
80 CIRCLE mx,my,r
130 FOR m=n*2*PI/g/5 TO 2*PI ST
EP 2*PI/g: REM GONDELN
140 LET x=mx+r*COS m: LET y=my+
r*SIN m
141 PLOT x,y: DRAW 0,-7: PLOT x
-7,y-8: DRAW 14,0
142 PLOT x-7,y-9: DRAW 14,0
143 PLOT x-7,y-10: DRAW 14,0
144 PLOT x-7,y-11: DRAW 14,0
145 PLOT x-6,y-12: DRAW 12,0
146 PLOT x-6,y-13: DRAW 12,0
147 PLOT x-5,y-14: DRAW 10,0
148 NEXT m
150 FOR m=n*2*PI/g/5+PI/g TO 2*
PI+2*PI/g STEP 2*PI/g
160 PLOT mx,my: DRAW r*COS m-1,
r*SIN m-1: NEXT m
177 FOR m=0 TO 12 STEP .6: CIRC
LE 127,104,m: NEXT m
180 PLOT 112,20: DRAW 12,0: DRA
W 0,-4: PLOT 143,20: DRAW -12,0:
DRAW 0,-4
187 PLOT 72,15: DRAW 112,0: DRA
W 0,-2: DRAW -112,0: DRAW 0,2
188 PLOT 69,12: DRAW 118,0: DRA
W 0,-2: DRAW -118,0: DRAW 0,2: P
LOT 66,9: DRAW 124,0: DRAW 0,-2:
DRAW -124,0: DRAW 0,2
```

```
198 REM
199 REM ** SPRINGBRUNNEN **
200 LET dx=27: LET dy=8: LET sx
=25: LET sy=80: LET ap=74
210 LET w=ATN (6*sy/sx)
214 LET ag=2*W*SIN w/sx
215 LET v=.8165
220 FOR t=(2*v*SIN w/ag)*n/ap T
O 2*v*SIN w/ag STEP (2*v*SIN (w)
/ag)/ap*5
225 PLOT dx+v*COS w*t,dy+v*SIN
w*t-ag/2*t*t
226 PLOT dx-v*COS w*t,dy+v*SIN
w*t-ag/2*t*t
227 NEXT t
231 FOR m=0 TO 7 STEP .2: PLOT
88+m,32: DRAW 32,72: PLOT 159+m,
32: DRAW -32,72: NEXT m
247 BRIGHT 1
250 FOR m=0 TO 21: PRINT AT m,0
: OVER 1: INK 0: PAPER 5: "
"
251 IF m>16 THEN PRINT AT m,0:
PAPER 4: INK 0: OVER 1: "
"
252 NEXT m
260 FOR z=15 TO 21: PRINT AT z,
25: PAPER 6: " : IF z>18 T
HEN PRINT AT z,25: PAPER 2: "
"
262 NEXT z
300 INK 6: PAPER 0: BRIGHT 1:
305 FOR m=14 TO 21: PRINT AT m,
24: " : AT 14,24: " : NEXT
m: REM ** LICHTERKETTE **
307 LET x1=195: LET x2=254: LET
y1=0: LET y2=60
310 FOR m=y1+n TO y2 STEP 5
315 PLOT x1,m: PLOT x1+1,m: PLO
T x1,m+1: PLOT x1+1,m+1
320 NEXT m: IF n=4 THEN PLOT x
1,y1: PLOT x1+1,y1
350 FOR m=x1+n TO x2-1 STEP 5
355 PLOT m,y2: PLOT m+1,y2: PLO
T m,y2+1: PLOT m+1,y2+1
360 NEXT m: IF n=4 THEN PLOT x
2,y2: PLOT x2,y2+1
399 PAPER 6: INK 0: BRIGHT 1
400 FOR m=1 TO 4: CIRCLE 218,39
,m*3+n-2: NEXT m: PLOT 203,40: D
RAW 32,0: PLOT 219,53: DRAW 0,-2
6
423 FOR z=50 TO 34 STEP -8: FOR
m=241 TO 253 STEP 3: PLOT m,z:
DRAW 0,-5: NEXT m: NEXT z
442 BRIGHT 1: INK 0: PAPER 7
443 PRINT AT 12,13: INK 3+n: PA
PER 0: "SATURN"
```

```
532 PRINT AT 18,11; INK 6; PAPER 2;"CDE";AT 19,11;"EGH";AT 18,18;"DEC";AT 19,18;"FGH"
```

```
536 PRINT AT 21,0; INK 4; PAPER 2;"B B"
```

```
538 PRINT #1;AT 1,0; BRIGHT 1;"JAHRMARKT DIETMAR VOLCKMANN 85"
```

```
609 PRINT AT 13,27; BRIGHT 0; PAPER 6; INK 0;"SCHIE";AT 13,27+N; BRIGHT 1; OVER 1;" "
```

```
620 POKE 23305,p; LET p=p+27; PRINT USR 23300: NEXT n
```

```
5999 REM ** ANIMATION **
```

```
6000 LET z=23300: BORDER 1
```

```
6010 POKE z+5,64: LET p=119
```

```
6020 GO SUB 6600: RANDOMIZE USR Z
```

```
6030 BEEP 2,40: PRINT "NUN EIN BISCHEN BEWEGUNG": BEEP 2,20
```

```
6040 FOR i=0 TO 33: GO SUB 6500: BEEP .3,1: NEXT i
```

```
6050 FOR i=1 TO 130: GO SUB 6500: NEXT i
```

```
6060 PAUSE 60: FOR i=1 TO 60: GO SUB 6600: NEXT i
```

```
6070 FOR m=21 TO 1 STEP -4: FOR i=1 TO m: GO SUB 6500: NEXT i: FOR i=1 TO m: GO SUB 6600: NEXT i: NEXT m
```

```
6080 FOR i=1 TO 30: GO SUB 6600: GO SUB 6500: NEXT i
```

```
6090 GO TO 6000
```

```
6500 POKE z+2,p: RANDOMIZE USR z: LET p=p+27: IF p=254 THEN LET p=119
```

```
6510 RETURN
```

```
6600 POKE z+2,p: RANDOMIZE USR z: LET p=p-27: IF p=92 THEN LET p=227
```

```
6610 RETURN
```

```
8000 PRINT AT 8,10;"JAHRMARKT""  
"BITTE 2 MINUTEN WARTEN BIS ALLE 5 SEQUENZEN GEZEICHNET SIND!" : PAUSE 250
```

```
8005 DATA 33,0,119,17,0,64,1,0,27,237,176,201,0,1,3,7,15,31,63,127,128,192,224,240,248,252,254,255,0,0,60,0,60,0,60,0,0,127,113,113,113,123,96,0,0,254,254,254,254,198,238,0,0,37,41,49,49,41,37,0,0,221,81,221,221,68,93,0,0,220,16,220,220,80,220,0
```

```
8010 FOR i=23300 TO 23311: READ a: POKE i,a: NEXT i: FOR i=USR "a" TO USR "h"+7: READ a: POKE i,a: NEXT i: RETURN
```

```
9999 SAVE "JAHRMARKT" LINE 1
```

Schreibschrift für den 48K Spectrum

ZX Schreibschrift ist ein Programm, das alle Buchstaben des kleinen Alphabets mittels des PLOT-Befehls in Schreibschrift umwandelt. Durch den Befehl COPY und einem Drucker kann die Schrift auf Papier übertragen werden. Die Länge des Textes darf nicht mehr als 31 Zeichen betragen. Es kön-

nen jedoch mehrere Texte hintereinander eingegeben werden.

Nachdem der Computer einen Text geschrieben hat, wird der Benutzer gefragt, ob er noch mehr schreiben möchte. Ansonsten wird das Programm beendet. Befindet sich ein Buchstaben nicht im kleinen Alphabet, so wird er ausgelassen. Die Zeilennummern der DATA-Zeilen entsprechen den ASCII-Werten der einzelnen Buchstaben.

Daniel Dumont

Schreibschrift

```
4 CLEAR
5 LET b=165
6 LET a=-8
7 CLS
10 INPUT a$: IF LEN a$>31 THEN GO TO 10
17 RESTORE
20 FOR i=1 TO LEN a$
22 LET a=a+8
25 LET l=CODE a$(i TO i): IF l<97 THEN GO TO 75
27 RESTORE 1
30 FOR o=1 TO 100
40 READ e: IF e<-2 THEN GO TO 75
50 READ f
60 PLOT a+e,b+f
70 NEXT o
75 NEXT i
80 INPUT "noch etwas(j,n)" : a$: IF a$="j" THEN LET b=b-13: LET a=-8: GO TO 8
85 INPUT "COPY (j/n)?" : a$: IF a$="j" OR a$="J" THEN COPY
86 STOP
97 DATA 0,0,0,1,1,2,1,3,1,4,2,5,2,6,3,7,4,7,5,7,5,7,2,1,3,0,4,0,5,0,6,0,6,1,6,2,6,3,6,4,6,5,6,6,7,1,-3
98 DATA 0,0,1,1,2,2,3,3,4,4,4,5,5,6,5,7,4,8,4,9,3,10,2,9,1,8,1,7,1,6,1,5,2,4,3,2,4,1,5,0,6,1,6,2,6,3,6,4,7,5,-3
99 DATA 0,0,1,1,1,2,1,3,2,4,2,5,3,6,4,7,5,7,6,7,7,7,2,2,3,1,4,0,5,0,6,0,7,0,-3
100 DATA 0,0,0,1,1,2,1,3,1,4,2,5,3,5,4,5,5,5,6,5,2,1,3,0,4,0,5,0,6,10,6,9,6,8,6,7,6,6,6,5,6,4,6,3,6,2,6,1,6,0,7,1,-3
```


Hisoft C für Heimcomputer

Die Programmiersprache C ist als eine moderne Sprache für Computer in aller Munde. Wer C beherrscht, der hat angeblich große Zukunftsaussichten. Wer die "Computerei" zu seinem Beruf machen will und als Einstieg dazu den Heimcomputer benutzen will, für den führt wohl kein Weg an C vorbei. Zudem ist die hier getestete Sprache C aus dem Hause HISOFT gut und preiswert; man bekommt den C-Compiler überall für ca. 90.-DM in der Version für den ZX Spectrum (Cassette), und die Version für die Schneider-Computer kostet um 160.-DM (Diskette). Es gibt auch noch eine Version für CP/M- und CP/M-Plus-Rechner. Sie kostet etwa so viel wie die Schneider-Version.

Der Lieferumfang besteht aus dem 78seitigen Handbuch und einer Cassette beim Spectrum und einem 120seitigen Handbuch mit Diskette beim Schneider. Der Datenträger enthält den C-Compiler und die wichtigen Bibliotheksprogramme STDIO.H und STDIO.LIB. Ab hier unterscheiden sich die Versionen dann aber doch erheblich.

Beim Spectrum hat man nach dem Laden von Cassette über ein kleines Menü die Möglichkeit, den Compiler auf Microdrive zu kopieren. Das ist einfach und unproblematisch. Es empfiehlt sich aber, den Basic-Loader für den Microdrive-Betrieb ein wenig zu ändern. Die Originalversion behält die Hintergrund- (PAPER) und Vordergrundfarbe immer bei. Man kann aber auch INK und PAPER fest vorgeben und den Kopierteil entfernen. Außerdem sollte im Loader ein RS232-Kanal geöffnet werden (z. B. Kanal 4). Die serielle Schnittstelle läßt sich dann mit dem Filepointer 4 ansprechen. So kann man sehr einfach Text auf einem Drucker ausgeben. Die Baudrate muß bei Bedarf mit einem FORMAT-Befehl eingestellt werden, wenn eine andere Geschwindigkeit als 9600 Baud benötigt wird.

Der Editor ist der schon bekannte Zeileneditor aus den Hisoft-Programmen Pascal und Devpac. Da hätte man Besseres bieten können. Zu allem Überfluß weicht er auch noch in Feinheiten von seinen Vorgängern ab. Das ist ärgerlich!

Da hat es der Schneider-Besitzer schon besser. Die einfache AMSDOS-Version ist im wesentlichen identisch mit der Spectrum-Version mit dem alten Zeileneditor. Zum Lieferumfang der Schneider-Version gehört aber zusätzlich das C für CP/M und CP/M Plus sowie die beiden Fullscreen Editoren ED80.COM (CP/M) und ED82.COM (CP/M Plus). Damit läßt es sich arbeiten.

Die Befehle sind ähnlich wie bei Turbo-Pascal eine Untergruppe der WordStar-Befehle. Es gibt sogar ein Auto Indent. Beim Drücken der ENTER-Taste geht der Cursor nicht an den Zeilenanfang, sondern zum Textbeginn der vorhergehenden Zeile. Das macht das Einrücken der verschiedenen Programmblöcke sehr einfach. Außerdem gibt es für die CP/M-Version die spezielle Programmbibliothek CP/M.LIB, die den Zugang zum Maschinensprache-Level von CP/M erlaubt. So können z. B. BIOS- und BDOS-Befehle direkt aufgerufen werden. Für die AMSDOS-Version gibt es drei zusätzliche Bibliotheken (BASCLIB, BASIC2.LIB und EXTCMD.LIB). Sie ermöglichen den Zugang zu den Eigenheiten der Schneider-Computer (Sound, Interrupts, Cassettenrecorder, Joystick, Grafik und RSX-Befehle).

Wer mit dem Compiler programmieren und C von Grund auf lernen will, der braucht unbedingt das Buch "Programmieren in C" von Kernighan/Ritchie. Im Hisoft-Handbuch wird nämlich nur der Unterschied zu der C-Definition in diesem Buch angegeben. Wer aber C lernen will, der benötigt sowieso ein Lehrbuch. Warum also nicht dieses, da es sowieso als die C-Bibel schlechthin gilt?

Die wesentlichen Unterschiede zwischen Hisoft C und der Definition von Kernighan/Ritchie bestehen darin, daß es keine Floating-Point-Zahlen gibt und die Integers INT, LONG und SHORT alle gleich sind (zwei Bytes). Darüber hinaus gibt es für die TYPE-Umwandlung den neuen Operator CAST. Aber das ist keine echte Einschränkung, und es beeinträchtigt auch nicht die Portabilität der Programme: Der CAST-Operator kann für einen anderen Compiler mit #DEFINE CAST ersetzt werden. Eine echte Erweiterung ist das Statement INLINE(...), das es erlaubt, Z80 Maschinencode direkt einzufügen.

Im praktischen Betrieb habe ich die Version 1.1 für den Spectrum getestet. Trotz einiger kleinerer Macken finde ich sie sehr gut. In STDIO.LIB ist die Zeile 1410 (sie enthält das Wort LENGTH) überflüssig. Sie muß entfernt werden. Auch der Editor hat einige kleine Fehler. Das Kommando P(ut) zum Sichern einer Programmquelle sichert unabhängig von den angegebenen Zeilennummern immer den gesamten Quelltext. Das Kommando B (Rückkehr ins Basic) funktioniert auch nur im C-Direktmodus, der normalerweise mit dem Kommando C erreicht wird. Hier kann man C-Befehle wie in Basic direkt ausführen. Ich meine, daß man mit diesen kleinen Fehlern leben kann.

Der C-Compiler kann sogar Quellfiles vom Cassettenrecorder compilieren. Dazu wird der Quelltext in Blöcken auf Band gespeichert. Die Pause zwischen den Blöcken ist lang genug, um den Text zu compilieren. Auch das Einfügen einer Datei (#INCLUDE) von Cassette ist möglich. Die CP/M-Version compiliert von Diskette auf Diskette und erzeugt dort ein .COM File. Dieses ist ohne Compiler lauffähig (enthält aber ein RUNTIMES-Modul) und darf beliebig an Freunde weitergegeben werden, ohne in rechtliche Schwierigkeiten zu

geraten – es sei denn, der Quelltext war abgekupfert. Auch die Spectrum-Version kann mit #TRANSLATE Filename Maschinensprache-Files erzeugen, die allein lauffähig sind.

Seine Stärke auf dem Spectrum spielt C im Zusammenhang mit ROM-Routinen und den Microdrives aus. Die Funktion FOPEN öffnet jedes Microdrive-File zum Lesen. Um die ersten 9 Bytes eines PRINT-FILES zu lesen, muß man jedoch den Data-Pointer manipulieren. In diesen ersten 9 Bytes steht bei den anderen Filetypen der Fileheader. Diesen überspringt der C-Compiler. Eine weitere herausragende Möglichkeit bieten die Variablen der Klasse STATIC. Hier kann die Stelle im Speicher angegeben werden, an der die Variable abgelegt wird. Damit ist es sehr leicht möglich, eine Variable über einer Systemvariablen anzulegen, womit sich ein sehr einfacher Zugriff auf die Systemvariable ergibt.

Beim Einbau von MC-Routinen sollte man vor dem Aufruf der eigentlichen Routine das IX-Register des Z80-Prozessors sichern, da es vom Compiler benutzt wird. Wird das IX-Register von der MC-Routine verändert, ist ein Systemabsturz unausweichlich.

Zusammenfassend kann man den C-Compiler von Hisoft nur weiterempfehlen. Er ist schnell, einfach zu handhaben und recht leistungsfähig. Auch der Preis ist in Ordnung. Floating-Point-Zahlen sollen in einer späteren Version folgen. Das ist nicht weiter schlimm, denn Hisoft ist dafür bekannt, daß man eine alte Version gegen eine geringe Gebühr in eine neue umtauschen kann.

Rainer W. Gerling

uni-Key Tastatur f. QL ab 249.-
QL dt. 128K 444.- / 512K 889.-
LV Imb NEC mit CST-Contr. 777.-
Sandy, Erweiterung, Zubehör a.A.
unicorn soft
3500 Kassel
u. Königstr. 46A
T. 0561/770367

Discovery-Disketten-verzeichnis

Der normale Discovery-CAT-Befehl liefert nur recht magere Informationen über den Disketteninhalt. Man erfährt lediglich die Namen der einzelnen Files und die Größe des auf der Diskette noch unbelegten Speicherplatzes. Die hier vorgestellte CAT-Routine produziert Verzeichnisse von der Art, wie es der Beispielausdruck zeigt. Das Basic-Programm erzeugt das Maschinen-code-Utility. Die Zeilen 30 und 40 sind entsprechend der gewünschten Anfangsadresse an die 485 Bytes lange Routine anzupassen. Das generierte Maschinenspracheprogramm ist an die gewählte Anfangsadresse gebunden, darf also später nicht in einen anderen Adreßbereich geladen werden.

PRINT USR Anfangsadresse ruft die CAT-Routine für den Bildschirmausdruck auf. Ersetzt man PRINT durch LPRINT, wird die Ausgabe zum Drucker geschickt.

Zu jedem File erhält man detaillierte Informationen. Anhand des abgebildeten Beispielausdrucks dürfte die Bedeutung der einzelnen Spalten schnell klar werden. Unter Adr erfährt man, wohin das File normalerweise geladen wird. Je nach Dateityp ist hier eine Variable oder eine Speicheradresse aufgeführt. Bei Autostart-Programmen wird die Nummer der Startzeile angegeben. Prgln gibt die reine Programmlänge wieder, d.h. die Länge ohne Variablen. Hinter "Freie Sektoren"

ren" stehen zwei Zahlen. Die erste gibt die Anzahl der freien Sektoren insgesamt, die zweite die Anzahl der unbelegten Sektoren am Ende der Diskette in einem Block an. Falls sich die Zahlen stark unterscheiden, könnte es sich lohnen, die Diskette mal wieder zu verdichten. Zur Erinnerung: Ein Sektor ist 256 Bytes lang.

Die Routine durchsucht das Inhaltsverzeichnis am Anfang der Diskette. Dort ist jedes File durch einen 16 Bytes langen Eintrag repräsentiert, der drei 16-Bit-Zahlen und den zehn Zeichen langen Namen enthält. Die erste Zahl gibt die Anzahl der noch zum File gehörenden Bytes im letzten Dateisektor an. Die zweite und dritte Zahl beschreiben die Nummer des ersten bzw. letzten zur Dateiaufzeichnung gehörenden Sektors. Die CAT-Routine greift auf den ersten Sektor einer jeden Aufzeichnung zu, denn er beginnt mit einem sieben Bytes langen Fileheader, der mit dem identisch ist, den wir von Cassettenaufzeichnungen her kennen, nur daß hier nicht nochmals der Dateiname erwähnt ist. Der Header enthält also die Informationen, die im Verzeichnis in den Spalten Typ bis Prgln erscheinen.

Es gibt eine Ausnahme von der Regel: Files vom Typ DAT (Dateien, die mit Hilfe von OPEN und CLOSE angelegt worden sind) besitzen keinen Header. Die Daten werden ab dem ersten Byte des ersten Da-

teisektors abgelegt. Diese Methode hat interessante Konsequenzen, wenn man ein Dateifile mit CHR\$ 0, 1, 2 oder 3 beginnen läßt. Das Discovery-System glaubt dann, es liege ein Header vor, und erlaubt die Interpretation des Files als Programm, Zahlen-Array usw.

Umgekehrt kann man mit OPEN, INPUT, INKEY\$ etc. auf jede beliebige Datei zugreifen, also beispielsweise eine Programmaufzeichnung Stück für Stück in eine Stringvariable laden.

Michael Schramm

Basic-Lader

```

10 DEF FN A(X$)=CODE X*-48-39*
(X$)*9*
20 DEF FN B(Y$)=16*FN A(Y$(1))
+FN A(Y$(2))
30 CLEAR 64199: REM Anfangsadr
esse-1
40 LET A=64200: REM Anfangsadr
esse
50 PRINT "Bitte einen Moment w
arten!"
60 LET A$="cd08170600f71222*d1
01cd481711*e5010e00616922ae5c22a
c5c22aa5ccd*c901cd*93014469736b6
57474653a20ff21*e9015e235623ed53
a85ccd*b101cd*bb01cd*93014e616d6
52020202020202054797020204c6e676
5202020416472205072676c6effcd*bb
013aae5cc61032ae5c200f11*e5012aa
a5c2322aa5c"
70 LET A$=A$+"4fcd*c901af21*e5
0106004f097e23234e2346235e235634
ca*390123c5f5cd*b101f1c1626bed42
656f2b2b2b2b2b2be52aa85ceb22a85c
606937ed52ed5bac5c1922ac5c606911
*e501d50e07cd*c901dde13e20d721*d
601dd7e00fe0538023e0457828206004
f090603cd*b301e1cd*9d01dd6e03dd6
6047aa72"
80 LET A$=A$+"01fcb7c280906063
e20d710fb1803cd*9d01dd6e05dd6606
cd*9d013e0dd7c3*66007afe0428f5fe
03200218ec3e20d73e20d77c15200bd6
40d7cd*93012829ff18dad680d73e241
8f06069cd*be01cd*930146726569652
053656b746f72656e3affed5ba85c37e
d52eb2aac5c19cd*9d01ebcd*9d013e0
dd7ed7b3"
90 LET A$=A$+"d5cc3761bc54ffe8
03808cbb93e14d73e01d779fe203006c
bf13e1318f179d7cd*930114001300ff
c1c9e37e23e3feffc8d718f6c501*910
1c5d5e53e205fd701f0d8cd2a19c3301
a060a7e23cd*6e0110f9c93e0dd70620
3e2dd710fb3e0dd7c9cd08173e010602
cd0000c348175052474e4d4243485242
5954444154"

```

Musterausdruck

Diskette: M. Schramm

Name	Typ	Lnge	Adr	Prgln
cat.s	PRG	2236		2236
array1	CHR	803	X\$()	
array2	NMB	505	A()	
druck.c	BYT	620	32768	
Druck	PRG	1482	10	1399
Daten	DAT	138		

Freie Sektoren: 687 676

```

100 LET B=A: LET S=31877
110 IF A$(1)="*" THEN GO TO 170
120 LET X=FN B(A$(1 TO 2))
130 POKE B,X: LET S=S-X
140 LET A*=A$(3 TO ): LET B=B+1
150 IF A*>"" THEN GO TO 110
160 GO TO 220
170 LET C=A+FN B(A$(2 TO 3))+25
6*FN B(A$(4 TO 5))
180 LET M=INT (C/256)
190 POKE B,C-256*M: POKE B+1,M
200 LET A*=A$(6 TO ): LET B=B+2
210 LET S=S+C-A: GO TO 150
220 PRINT
230 IF S THEN PRINT "Fehler in
A*!": STOP
240 PRINT "Die CAT-Routine ist
bereit.", "Aufruf durch PRINT USR
";A,"bzw. LPRINT USR ";A
250 STOP
9000 SAVE #1,"Cat" LINE 10
9010 VERIFY #1,"Cat"
9020 STOP
9500 SAVE #1,"Cat.c"CODE A,485
9510 VERIFY #1,"Cat.c"CODE

```

Nachtrag zum Multiface One (CK 6-7/86)

Wie Horst Müller in seinem Bericht über das Multiface One bereits angekündigt hat, ist schon wieder eine neue Version erschienen, die wir von der Firma Romantic Robots direkt aus England erhalten haben. Leider lagen dem Interface keine Unterlagen bei, so daß hier noch nicht alle Neuigkeiten genannt werden können. Tatsache ist jedoch, daß es einige Änderungen gegeben hat. Geblieben sind folgende Features:

- automatische Kopiermöglichkeit für Wafadrive, Diskette, Microdrive und Cassette,
- eingebautes Joystick-Interface,
- 8-kByte-RAM-Erweiterung,
- eingebautes Toolkit.

Das Toolkit wurde erweitert. Jetzt sind auch Hex-Eingaben möglich. Weggefallen ist das Monitor-Interface. Dafür müssen Beta-Disk-Anwender nicht mehr das Gerät umbauen, da es schon mit abschaltbarem Joystick-Interface geliefert wird. Sobald wir das Interface voll ausgetestet haben, folgt ein Bericht.



Mit neuem
Inhalt

SPECTRUM * Hard- und Software zu günstigen Preisen * SPECTRUM

Das verbesserte ISO-ROM! Jedes ISO-ROM verfügt nun über:

- * Reset ohne Programmverlust! • Deutsche Umlaute!
- * Neuer Zeichensatz! • Verbessertes CAT für MDI
- * Verkürzte Syntax für Microdrive oder Beta-Disk!
- * Erweiterter Editor mit Cursor/MW in Programmzeilen!
- * Variable Zeichenbreite: 32, 36, 42, 51 oder 64 Zeichen pro Zeile (nur bei Monitor-ROM!)

Wählen Sie dazu eine von 2 versch. NMI-Routinen:

Entweder den HEX-Monitor: Unterbricht jedes Programm, ermöglicht Schreiben und Starten eigener MC-Routinen!

Oder die neue Backup-Routine: Überträgt sämtliche Programme auf Ihr Speichermedium! Dabei wird die größte "Lücke" ausgespart, so daß nicht jedes Programm 48 K lang ist!!!

Dieses Programm läuft dann auch ohne ISO-ROM!

Außerdem: POKE-Option! Sie können jederzeit Ihr Game unterbrechen, einen POKE eingeben und weiterspielen. Unentbehrlich für alle Spieler!!!

Backup-Routinen gibt's für Microdrive, Opus Disk, Beta-Disk und Cassette (für alle anderen Speichermedien)!

Fordern Sie zum ISO-ROM unser Info an!

Zu jedem ROM gehört eine deutsche Anleitung mit Einbauplan (auch für Parallelbetrieb mit Sinclair-ROM)!

Monitor-ROM: nur 80.- DM

Backup-ROM: nur 95.- DM

Auf Wunsch bauen wir Ihr ROM für nur 20.- DM auch ein!

Amadeus - Das Spectrum-Musiksystem für alle 3-Kanal-Soundmodule. Synthesizer mit 7 vorgegebenen Instrumenten. Notenteil für bequeme Noteneingabe und Korrektur. Jetzt 45.- DM.

Zusammen mit 3-Kanal-Soundmodul nur 125.- DM.

Jetzt brandneu: Das ISO-FACE! Ermöglicht den Betrieb von bis zu 3 externen ROMs durch einfaches Anstecken an den Spectrum.

- * 3 Steckplätze für 27128-er EPROMs!
- * Umschalter Sinclair/ext. ROM, absturzfür, 3-fach Schalter für die ext. ROMs und prellfreie Reset/NMI-Taster!

* Durchgeführter Bus und stabiles Gehäuse!

Das alles für sage und schreibe nur 100.- DM!!!

(Bei gleichzeitigem Kauf eines ISO-ROMs: 10.- DM Rabatt.)

BETA-PACK - Das Utility-Paket für Ihre Beta-Disk:

1. BETA-TRANS - Bringt Files von Cassette auf Diskette!
 2. BETA-COPY - Kopiert beliebige Files von Disk zu Disk!
 3. BACKUP 48 - Schnelles Backup mit fast 48 K auf einmal!
 4. BETADRESS - Findet autom. Anfangsadressen von Codeblöcken!
- 4 Topprogramme auf 1 Diskette: nur 60.- DM
Bitte Laufwerk und Controller angeben (z.B. 3,5"/2x80/4.12)

ISO-DATEI - Endlich ein Dateiprogramm, das den Speicherplatz der Beta-Disk voll ausnutzt! Bis zu 5000 Einträge auf einer Disk, komfortabler Screeneditor, deutsche Umlaute, Programm und Anleitung in deutsch, sehr benutzerfreundlich! Jetzt 45.- DM

BETA-BOOT - "Boot"-File für jede Diskette! Autom. CAT, zum Laden eines Files nur File per Cursor anfahren. Nur 20.- DM

POKE-STRIPPER - Pokern Sie mit! Ziehen Sie sich warm an, denn Sie spielen gegen 3 Damen. Super-Grafik! Nur 30.- DM

ISO-COPY, QUICKSAVE, QUICKSAVE II je 20.- DM/ ASSI: 15.- DM

LOCK-SAVE, SOFTWARE-DRUCKERPUFFER je 30.- DM

Broschüre ROM-ROUTINEN: 10.- DM/POKE-LISTE: 3.- DM

Gratis-Info und Bestellungen (V-scheck oder Nachnahme) bei:
Olaf Marohn
 Schürferstraße 41
 4600 Dortmund 30

**INDIVIDUAL
SOFTWARE**

Volker Marohn
 Am Beilstück 30
 4600 Dortmund 50

Schnellere Apfelmännchen-Grafik

Das Apfelmännchen-Programm aus CK 6/86 ist hinsichtlich der Bildberechnungsgeschwindigkeit nicht optimal; es läßt sich noch "tunen". Ladet hierzu die alte Version und gebt die Zeilen aus dem Basic-Listing ein. Das Programm wird um etwa 13 % schneller. Außerdem zeigt es nun die zur Bilderstellung verbrauchte Zeit an.

Die nicht unerhebliche Geschwindigkeitszunahme wird dadurch erreicht, daß an zwei Stellen der Aufruf von Gleitkommaoperationen durch wenige Maschinenbefehle ersetzt wird:

1. Die an einem Punkt notwendige Multiplikation mit 2 wird in der alten Version durch eine Addition der Zahl mit sich selbst erzielt (eine Addition geht schneller voran als eine Multiplikation). In der neuen Version wird einfach der Exponent der Zahl in Gleitkommadarstellung um eins erhöht, denn diese ist ja auf die Basis 2 bezogen, d.h. jede Zahl wird in der Form 2^m gespeichert. Man muß nur sicherstellen, daß die Zahl tatsächlich in Gleitkommadarstellung vorliegt und nicht etwa in der speziellen Ganzzahldarstellung. Das ist natürlich sehr unwahrscheinlich, denn dann hätte durch Addition und Multiplikation mehrerer gebrochener Zahlen ausgerechnet eine ganze Zahl entstehen müssen. Sicherheitshalber sollte man jedoch durch das Kommando \$3D dafür sorgen,

daß das oberste Stapелеlement gegebenenfalls in Gleitkommadarstellung überführt wird. Nach jedem Kalkulatoraufruf zeigt HL auf das Exponentbyte der letzten Zahl auf dem Stapel. Man kommt also sehr leicht an die Zahl heran.

2. Der Vergleich mit der variablen Grenze zur Prüfung, ob der Betrag der komplexen Zahl zu groß geworden ist, wurde durch einen festen Vergleich mit 4 ersetzt, weil das ein sehr sinnvoller Wert ist. Glücklicherweise ist 4 eine Zweierpotenz, so daß zum Test wiederum nur der Exponent der Gleitkommazahl herangezogen werden muß. Hierbei ist zu beachten, daß das Exponentbyte den Exponenten plus 128 enthält. Man muß also mit $128+3=131$ vergleichen, denn 4 wird als 2^{3+128} dargestellt. Das oberste Stapелеlement darf getrost gelöscht und danach kann trotzdem noch auf die Zahl zugegriffen werden. Diese Zahl wird nicht wirklich aus dem Speicher entfernt, sondern nur die Systemvariable STKEND, die das Ende des Gleitkommazahlen-Stapels angibt, verringert sich um 5. Nach dem Kalkulatoraufruf gilt $DE = (STKEND)$, d.h. DE gibt die Adresse des Exponentenbytes der "gelöschten" Zahl wieder.

Pro Bild werden leicht einige hunderttausend Gleitkommaadditionen eingespart, und das bringt etliche Minuten an Laufzeitverringerung.

Michael Schramm

Basic-Lader

```
25 DEF FN Z*(X)=": "+(X AND X
(<=9)+STR$ X
245 PRINT "Bildberechnungszeit
: ";PEEK 30997;FN Z*(PEEK 30998
);FN Z*(PEEK 30999)";
370 SAVE N$ CODE 31000,6144
8020 LET A$="2118791100400100180
db0c976af676f22785c327a5ccd811c0
604c5cd791cc110f921bd7622685cefc
802c702c602c102c502c402c20238cda
22d32bc76cda22d32ba7621000022b87
6efe6c00238cd541fd20d763abc761f3
```

```
80b21bd7611d176010a00edb021ba767
e2377efe0e138efe0310431e13104c30
f3d02381afe833805ef02381819efe30
3e40fe0e1043d3834efe50fc102c0023
821bb763520d0efc102c0023821bb767
e230fcb4e2802c6ffed4bb876d444760
c280ded43b876efe7e00fc0"
8030 LET A$=A$+"023818860478fec0
300eed43b876efe0e10fc10238c37a75
210040111879cd367521785c11b8760e
0376edb0113200cd73761e3ccd737632
17791e3ccd73763216793ab876321579
ed7b3d5cc37d1be5f578e6c00f0f0fc6
406778e607846778e63807076f79e6f8
0f0f0f856f792fe607070707f6c6326f
76cbc6f1e1c92ab8763aba760600dd21
b876dd7002dd7001dd70004804cb7920
08cb23cb12cb1118f3ddcb0026ddcb01
16ddcb0216ed52993805dd3400180219
89cb39cb1acblb10e07dc9"
8040 LET A=3E4: LET S=42110
```

Beispiel 1

```
RST $28 ;Diese Routine verdoppelt das
DEFB $3D ;Top-Element auf dem Stapel
DEFB $38 ;durch inkrementieren des
INC (HL) ;Exponenten.
RET
```

Beispiel 2

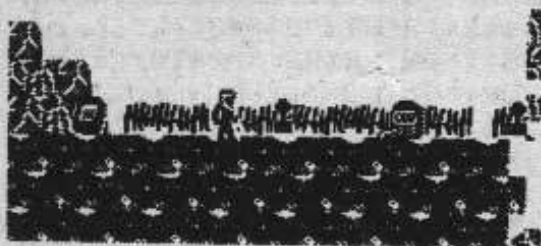
```
RST $28 ;Diese Routine prüft, ob der
DEFB $3D ;Betrag des Top-Elements
DEFB $02 ;kleiner als 4 ist.
DEFB $38 ;Falls der Vergleich positiv
LD A,(DE) ;ausfällt, wird das Carry-
CP 131 ;Flag gesetzt. Die Zahl auf
RET ;dem Stapel wird gelöscht.
```

Ghosts 'n Goblins

Brandneu aus England eingetroffen ist Ghosts 'n Goblins, der neueste Hammer von Elite. Wenn man der Werbung glauben darf, wird dieses Programm der Herbsthit 1986, zumal es gleichzeitig für alle aktuellen Computertypen herausgebracht wurde. Dabei handelt es sich wieder einmal um die Umsetzung eines erfolgreichen Spielhallenhits. Wie üblich müssen einige Abstriche bei der Grafik gemacht werden. Trotzdem bleibt noch genug übrig, um auch Heimcomputerfreunde zufriedenzustellen.

Hinter Ghosts 'n Goblins verbirgt sich ein klassisches Abenteuerspiel. Die Handlung ist schnell beschrieben. Der Spieler schlüpft in die Rolle eines heldenhaften Ritters, der eine schöne Jungfrau aus den Klauen des bösen Herrschers befreien muß. Die Entführung findet in Kurzform vor jedem Spielbeginn statt. Zombies und andere finstere Gestalten wollen die Rettung natürlich verhindern.

Auf dem Bildschirm stellt sich Ghosts 'n Goblins recht ansehnlich dar. Vor dem Spiel

SCORE
3600LIVES
7TIME
2:52

Am Ausgang des Friedhofes

start kann der Anwender entscheiden, ob er über Tastatur (Belegung kann frei definiert werden) oder mit einem Joystick steuern will. Danach beginnt der Kampf. Die ersten Bilder zeigen eine Art Friedhof. Schon hier tauchen die ersten Untoten auf, die zwar nicht besonders schnell sind, aber durch bloße Berührung ein Leben des Helden rauben können. Das gilt auch für alle anderen Gestalten, von denen es schon bald nur so wimmelt.

Schon in der ersten Spielminute wird klar, daß man nur mit viel Geduld und Geschick weiterkommt. Der Spieler kann seine Figur nach links und rechts bewegen, sich ducken, springen oder klettern lassen. Die Zombies lassen sich durch gezielte Pfeilwürfe ausschalten, was Punkte einbringt. Wie schon gesagt, ist es notwendig,

bereits am Anfang höllisch aufzupassen, um kein Leben zu verlieren. Zwar stehen davon neun zur Verfügung, doch muß nach jedem Verlust von vorne begonnen werden, was auf Dauer ziemlich nervt.

Leider bin ich noch nicht so weit gekommen, einen Blick auf die schöne Jungfrau werfen zu können; dafür fehlt mir einfach ein Unsterblichkeitspoke. Ghosts 'n Goblins ist ein Spiel, mit dem man sich manche Stunde interessant vertreiben kann. Ich halte das Programm tatsächlich für einen potentiellen Renner, da Grafik und Animation sowie die Spielidee stimmen.

System: Spectrum 48 K
Hersteller: Elite
Bezugsquelle: Peter West Records
Rolf Knorre

Beliebig viele Stellen nach dem Komma mit dem ZX-Spectrum

Manchmal ist es aus irgendwelchen Gründen wichtig, einmal mehr als die üblichen Spectrum-Stellen bei numerischen Berechnungen zu erhalten. Die beiden vorgestellten Algorithmen gestatten die Berechnung und Ausgabe beliebig vieler Stellen bei der Division bzw. beim Logarithmieren. Der Divisionsalgorithmus (Zeile 40-100) ist denkbar einfach. Der Dividend (Zähler) wird durch fortgesetztes Subtrahieren der Vielfachen des Divisors (Nenners) und nachfolgender Multiplikation mit 10 immer wieder neu gebildet (Zeile 90).

Der Logarithmus-Algorithmus gelingt nach einem ähnlichen Muster. Hier wird nicht gefragt, wie oft eine natürliche Zahl x in einer y als Summand enthalten ist, sondern wie oft x in y als Faktor enthalten ist. Das

fortgesetzte Subtrahieren wird durchs Dividieren ersetzt und das Multiplizieren durchs Potenzieren (vgl. Zeile 1090 und die Abfrage in Zeile 1060). Zur mathematischen Begründung des Verfahrens könnte noch einiges ergänzt werden, doch möchte ich die Schulerinnerungen nicht überstrapazieren. Die Beispielausdrücke mögen genügen. Jeder kann sich eigene erstellen.

Die Division 1:97 zeigt, wenn man genau hinschaut, eine Periodenlänge von 96. Keine Tabellensammlung geschweige denn ein Computer kann das liefern. Die Zahl 2.7182818,... (EXP 1) ist die sog. Eulersche Zahl e . Übrigens, was macht der Spectrum, wenn für z und n jeweils 0 eingegeben wird?

Harald Bensom

Maxi-Fließkomma

```

1 REM Divisions-Algorithmus
2 REM *****
10 INPUT "Zähler: ";z,"Nenner
:n
20 INPUT "Wieviel gewünschte
Stellen? ";s
30 PRINT z;" / ";n;" = "
40 FOR i=1 TO s
50 LET c=0
60 IF z>n*c+n THEN LET c=c+1:
GO TO 60
70 IF i=1 THEN PRINT c;" . ";: G
O TO 90
80 PRINT c;
90 LET z=10*(z-n*c)
100 NEXT i
110 PRINT " . . . ."
120 PRINT "s";" Stellen"
499 REM *****
500 STOP
501 REM *****
1001 REM Logarithmus-Algorithmus
1002 REM *****
1010 INPUT "Basis: ";b,"Numerus:
:n
1020 INPUT "Wieviel gewünschte
Stellen? ";s
1030 PRINT "log( ";b;" )";"n" = "
1040 FOR i=1 TO s
1050 LET c=0
1060 IF n>b*(c+1) THEN LET c=c+
1: GO TO 1060
1070 IF i=1 THEN PRINT c;" . ";: G
O TO 1090
1080 PRINT c;
1090 LET n=(n/(b+c))*10
1100 NEXT i
1110 PRINT " . . . ."
1120 PRINT "s";" Stellen"

```

Spectrum User Club Wuppertal

Club-Zeitschrift
Tips, Tricks, Bauanleitungen
Vorstellung neuester Programme
Erfahrungsberichte
Mitglieder-Börse
Club-Software

Weitere Informationen erhalten Sie (gegen
Einsendung von DM 0.50 Rückporto) von:
Rolf Knorre,
Postfach 20 01 02, 5600 Wuppertal 2

Hardcopy mit dem LPRINT-II-Interface

Nachdem wir bereits zwei Hardcopy-Routinen für das Interface I abgedruckt haben, diesmal ein entsprechendes Utility für das Interface LPRINT II. Auch dieses 305 Bytes lange Maschinencodeprogramm läßt sich in einen beliebigen Speicherbereich legen. Es wird genauso aufgerufen und benutzt wie die in CK 11/85 und 6/86 beschriebenen Hardcopy-Routinen.

Die Interface-I-Routinen sind für LPRINT II nicht geeig-

net, weil über die gewohnte Zeichenausgabetechnik durch RST \$10 nicht beliebige Bytes zur Centronics-Schnittstelle durchgeschleift werden können (Basic-Tokens werden expandiert). Vielmehr läßt man jeden Character in den Akku und führt CALL \$OF12 aus. Ohne angeschlossenes LPRINT-II-Interface wäre das ein sinnloser Sprung mitten in eine der Sinclair-Drucker-Ansteuerungsroutinen. Die Interface-Hardware blendet das Spectrum-ROM jedoch aus und schaltet dafür das eigene ROM ein. Dort steht ab Adresse \$OF12 die richtige Ausgaberroutine.

Michael Schramm

10 REM Spectrum-Hardcopy-Routine fuer das LPRINT-II-Interface

Michael Schramm, Kiel
Computer Kontakt 1986

```
20 DEF FN A(X*)=CODE X*-48-39#
(X*)"9")
```

```
30 CLEAR 59999: REM Anfangsadresse - 1
```

```
40 LET A=6E4: REM Anfangsadr.
```

```
50 PRINT "LPRINT-II-Hardcopy-Routine wird", "erzeugt" "Einen Augenblick bitte..."
```

```
60 LET A*="dffe2c3e002008cd811
ccd941ee68332925c21001822935c262
022955cdffe2c2044cd791ccd941e329
45ca72820f5cd941e32935cc180382bf
e193027dffe2c2024cd791ccd941e329
65ca72007ed7b3d5cc3761bf5cd941e3
2955cc1803804fe213802cf0a3e1bcd1
20f3e41cd120f3e08cd120f3e1bcd120
f3e4bcd120f3a965c0707260017cb14f
dcb3846280317cb14cd120f7ccd12"
```

```
70 LET A*=A*+"0ffd6659fd6e5bfd
465ce5c57ce6070f0f0fb56f7ce618c6
4067119e5c0608fdcb5856280a242424
241804188918ae7e121cfdcb584e2802
121c2410f21608219e5c06085ecb1317
732c10f8fdcb587e28012ff5cd120ff1
fdcb5846c4120f1520ddc1e12c10a33e
0ddcd120f3e0acd120f21925ccb4e2808
cb56cbd628afcb96fd3459fd355a20a5
3e1bcd120f3e41cd120f3e0ccd120f18
92"
```

```
80 LET S=29459
```

```
90 LET X=16*FN A(A*(1))+FN A(A*(2))
```

```
100 POKE A,X: LET S=S-X
```

```
110 LET A*=A*(3 TO 1): LET A=A+1
120 IF A*="" THEN GO TO 90
130 IF S THEN PRINT "Fehler in
A*!": STOP
140 PRINT
150 PRINT "Die Maschinencoderou
tine ist", "bereit." "Aufrufen du
rch PRINTUSR 6E4,... (genaue Bes
chreib. in CK 11/85 und CK 6/86
)" "Das Basic-Programm kann dur
ch NEW geloescht werden."
160 STOP
170 SAVE "LPRINT-Cop"CODE 6E4,3
05
```

N.O.M.A.D.

Bei diesem neuen Programm handelt es sich um ein Actionspiel, das im Science-Fiction-Bereich angesiedelt ist. Wieder einmal stehen sich das Gute und das Böse gegenüber, um im Showdown über das weitere Schicksal des Universums zu entscheiden.

Diesmal geht es um die intergalaktische Verbrechervereinigung TALOS, die von dem skrupellosen Cyrus T. Gross geführt wird. Dessen Name traut man sich, wenn überhaupt, überall nur flüsternd auszusprechen. Auf der anderen Seite steht N.O.M.A.D., eine Vereinigung freiheitsliebender Menschen und Roboter, die es sich zum Ziel gesetzt hat, Gross und seine Bande zu bekämpfen. Dazu wird Nomad 471 eingesetzt, ein hochintelligenter Kampfdroid. Bewaffnet mit zwei 57er Magnum-Strahlern, technisch auf dem neuesten Stand, soll dieser Droid das Innere der gegnerischen Station durchdringen, um an Gross heranzukommen. Da auch der beste Roboter nicht alleine klar kommt, tritt auch der Spieler in Aktion. Über Joystick oder Tastatur muß er die Kampfmaschine steuern und sicher ans Ziel bringen. Insgesamt stehen dazu 4 Droidenleben zur Verfügung, die unter Umständen schnell verbraucht sein können.

Die feindliche Station stellt sich dem Spieler als Labyrinth dar, das von Fallen und Feinden nur so wimmelt. Erschwerend wirkt sich dabei die Steuerung des etwas schwerfälligen Droi-

den aus. Gibt man zuviel Gas auf die Antriebsraketen, sitzt man, eh man sich versieht, vor der Abschlußöffnung einer automatischen Abwehrrakete. Neben diesen Raketen warten auf Nomad 471 z.B. Magnetwände, Killer-Droiden und an-



dere unangenehme Dinge. Alles in allem eine große Schar Feinde, die keine Langeweile aufkommen lassen.

Für ein Schießspiel bietet N.O.M.A.D. recht viel an Aktionen und Grafik. Da man auch die Möglichkeit hat, zusätzliche Leben zu bekommen, muß man bei einer Niederlage nicht gleich resignieren. Leicht zu bewältigen ist das Programm aber keinesfalls.

System: Spectrum 48K
Hersteller: Ocean
Bezugsquelle: P. West Records
Rolf Knorre

it's sinclair time



Die zwei Elite-Renner
zum Wahnsinnspreis
von je **24.--**



2 Bücher - unent-
behrlich - ehrlich!
ZX 81 DM **49.--**
Sinclair QL DM **69.--**



Das schärfste
für den Spectrum -
Peep-show. Bitte
Alter angeben.
DM 29,90



Eine Reise
von New York
nach Ägypten.
Deutsches Grafik-
adventure für
DM 16.--



CK Spezialitäten-Bestellschein

Ich wünsche folgende
Bezahlung:

• Nachnahme
(m.z. 5,70 DM Versandkosten)

• Vorauskassa

(keine Versandkosten)
Bei Vorauskassa bitte Scheck belegen
oder auf Postcheckkonto Körber
45423-756 überweisen.

Anzahl	Best.-Nr.	Artikel
	N1	Bomb Jack
	N2	Gost'n Goblins
	N3	Alles über den ZX81
	N4	QL unter Kontrolle
	N5	Peep-Show
	N6	Super-Story

Name des Bestellers

PLZ/Ort

Anschrift

Datum/Unterschrift

Coupon ausschneiden, auf Postkarte kleben und einsenden an:

Verlag Rätz-Eberle / CK-Software, Postfach 16 40, 7518 Bretten.

Der Colour Pen Graph EB50

Eine Art Schreibmaschinen-Plotter

Rein zufällig entdeckte ich in der Filiale der Warenhauskette divi in Raisdorf bei Kiel den Colour Pen Graph EB50 (Hersteller: Silver Reed). Dieses Gerät, das wie eine Reise-schreibmaschine aussieht, erweckte mein Interesse, weil es nicht mit Typen druckt, sondern mit farbigen, kleinen Kugeltinten schreibt. Im Schreibkopf sind vier dieser Stifte untergebracht (schwarz, blau, rot und grün); auf Tastendruck wird die Farbe gewechselt. Tatsächlich kann man den Pen Graph, der inklusive Batterien (fünf Monozellen) und seiner Hardbox, in deren Unterteil er fest eingesetzt ist, nur rund 2,5 kg wiegt, als reine Reise-schreibmaschine nutzen – allerdings als eine äußerst komfortable!

Der EB50 bietet praktisch alle Funktionen, die man von einer ordentlichen elektronischen Schreibmaschine erwartet, wie beispielsweise setzen von Rändern und Tabulatoren (bis zu 16), optional automatischer Zeilenumbruch nach einem Wortende in Randnähe

und Korrekturmöglichkeit (mit speziellem Korrekturstift). Da der Pen Graph nicht mit starren Typen arbeitet, sondern jedes Zeichen wirklich zeichnet, ergeben sich ganz neue Möglichkeiten. Man kann nicht nur "normal", sondern auch abwärts oder aufwärts schreiben, kursiv oder unterstreichen, in halber oder doppelter Zeichen-größe – selbstverständlich alles in vier Farben.

Diese Druckparameter lassen sich beliebig kombinieren. Die jeweils eingestellte Kombination zeigt der obere Teil eines über der Tastatur angeordneten Displays an. Der Hauptteil des Displays gibt die 15 zuletzt getippten Zeichen wieder. Wählt man den Modus "Vorschreiben", so wird zunächst eine komplette Schreibzeile im Gerät zwischengespeichert, je nach gewählter Schriftgröße maximal 40, 80 oder 160 Zeichen. Nun kann man mit Hilfe von Pfeiltasten das Display-Fenster über der gespeicherten Zeile hin- und herbewegen, um gegebenenfalls noch Korrekturen vorzunehmen. Erst das Betätigen der Wagenrücklauftaste bringt die Zeile aufs Papier. Druckgeschwindigkeit: ca. 7 Zeichen pro Sekunde in Normalgröße, 14 bei halber Größe.

Falls während des Textschreibens mal eine Berechnung anfällt, schaltet man den EB50 einfach in den Taschenrechner-Modus und führt die

Rechnung im Display durch. Das Ergebnis läßt sich natürlich auch ausdrucken. Allerdings bietet der EB50 nur die vier Grundrechenarten.

Drückt man die GRAPH-Taste, so wird am augenfälligsten, daß der Pen Graph eher ein Plotter als ein Drucker ist. Denn jetzt genügen wenige Tastendrücke und die Eingabe von Zahldaten gemäß den Anweisungen im Display, um farbige Kreis- und Blockdiagramme für Statistiken und Vergleiche zu erstellen. Es besteht die Möglichkeit, zunächst den Schreibkopf nur die Umrisse der Grafik umfahren zu lassen, um sicherzustellen, daß das Diagramm auch in ein möglicherweise schon bestehendes Bild paßt. Der EB50 bietet zehn verschiedene Typen solcher Diagramme. Die Abbildung zeigt einige Beispiele.

Für uns Computerbesitzer wird der EB50 jedoch erst dadurch richtig interessant, daß er eine eingebaute Centronics-Schnittstelle aufweist und somit an jeden kommunikationsfreudigen Computer angeschlossen werden kann. Ein kleiner Schiebeschalter wird bewegt, und schon gehorcht der Pen Graph nur noch dem Rechner – die Tastatur ist abgeschaltet. Der EB50 verhält sich jetzt genauso wie ein Typenraddrucker, wobei die im manuellen Betrieb gewählten Einstellungen für Farbe, Schriftgröße und

stil übernommen werden (Randeinstellungen gehen jedoch verloren).

Der Pen Graph als Plotter

Ein spezieller Steuercode (DC2) aktiviert den Plotter-Modus. Nun zeigt der Pen Graph, was wirklich in ihm steckt! Das Blatt Papier wird als Koordinatensystem aufgefaßt, dessen Ursprung beliebig festgelegt werden kann. Durch ASCII-Strings übermittelt man dem Plotter den Befehl, jeden gewünschten Punkt in gerader Linie anzufahren, wahlweise mit gesenktem oder erhobenem Kugeltift. Beispielsweise zieht "D100,50" eine Linie vom augenblicklichen Standort zum Punkt mit den Koordinaten (100,50); "J-10,20" zeichnet eine Linie zu dem Punkt, der relativ zur momentanen Position 10 Einheiten weiter links und 20 Einheiten weiter oben liegt. Spezielle Strings verursachen Textausgabe, Farb-, Größen- oder Schreibrichtungswechsel. Auf diese Weise läßt sich jede gewünschte Grafik erzeugen. Die Auflösungsgenauigkeit beträgt etwa 0,25 mm. Als Anhaltspunkt für die Plot-Geschwindigkeit sei gesagt, daß für die programmgesteuerte Erzeugung der abgebildeten Zeichensatztafel etwa drei Minuten benötigt wurden.

Noch ein paar Worte zur mechanischen Seite: Die Tastatur

**Ihr direkter Draht
zur Redaktion:
Telefon 0 72 52 / 429 48
Für Briefe:
Computer Kontakt
Postfach 16 40
7518 Bretten**

SPECTRUM • QL • ATARI • SPECTRUM • QL • ATARI • SPECTRUM •

Dreaser, Soft- und Hardware

Wir bekommen laufend die aktuellsten Produkte für den Spectrum QL, Atari 600/800/130 XL/XE sowie Atari ST. Nutzen Sie unseren Telefon- und Auftragservice zu den angegebenen Zeiten, damit auch Sie über die Neuheiten informiert sind.

Spectrum		QL	
Alien Highway	31,90 DM	Assembler	98,90 DM
Batman	28,90 DM	Karate	88,90 DM
Bomb Jack	30,90 DM	Pascal	358,00 DM
Planets	38,90 DM	C	398,00 DM
Spindizzy	38,90 DM	Cartridge Doctor	64,90 DM
Pentagramm	38,90 DM	Spook	47,90 DM
Equinox	38,90 DM	Flight Simulation	77,90 DM
Atari		Hardware	
Assembler	(ST) 178,90 DM	AMX Mouse	(Spec) 259,00 DM
C	(ST) 347,90 DM	QL (engl.)	399,00 DM
Pascal	(ST) 327,90 DM	Spectrum 128 (dt)	548,00 DM
Knock Out	(D) 48,90 DM	Discovery 1	689,00 DM
Knock Out	(C) 34,90 DM	SpecDrum	145,00 DM
Mr. Robot	(D) 88,90 DM	Multiface One	199,00 DM
Mr. Robot	(C) 38,90 DM	Spectrum Plus	348,00 DM

Fordern Sie unsere Gratisliste an!

Dreaser Soft- u. Hardware, Im Rosenhag 6, D-5300 Bonn 1, ☎ 02 25 / 25 40 84
Mo. bis Fr. von 17.00-19.00 Uhr, Sa. von 14.00-18.00 Uhr oder Auftragsannahme rund um die Uhr.

SPECTRUM • QL • ATARI • SPECTRUM • QL • ATARI • SPECTRUM •

S P E C T R U M REPARATURDIENST

Rep.- Zeit ca. 5 Tage; 6 Mon. Garantie. Auf Wunsch mit Kostenvoranschlag (kostenlos!)

ROM-MODUL Für 8/16k Eeproms. Für BASIC o. MC-Programme. Auch Zweit-ROM-Betrieb möglich! Autostart! Ideal für Spiele, Utilities etc. Einfachste Anwendung! Incl. Konvertierungsprogramm **DM 99,00**

Belkenheid Computertechnik
Wellinger Weg 5a, 4513 Belm Tel.: 05406/5547

EAR- und MIC-Buchse bei diesem Rekorder entfallen kann. Nützlich ist auch der Monitor-Schalter, mit dem man in der Position "ON" aufgenommene Signale zusätzlich hören kann, was das Auffinden von Anfang und Ende der programmierten Daten leichter macht. Bei der Aufnahme kann man so auch mithören. Die Batterie/Aufnahme-Anzeige flimmert während des Signaleingangs bei der Aufnahme. Bei Wiedergabe und Schnellvorlauf leuchtet die Anzeige auf. Wenn die Batteriespannung nachläßt, leuchtet die Anzeige nur noch schwach. Wenn man an die EAR-Buchse einen kleinen Kopfhörer (doppelt Mono) oder eine alte Box (ca. 4-8 Ohm und zwischen 3 und 6 Watt Nennbelastung) anschließt, kann man den Datenrekorder sogar als Walkman verwenden.

Dieser Datenrekorder eignet sich also für diejenigen, die ein gutes Bandspeichergerät suchen, das über alle Funktionen verfügt und zusätzlich zur übli-

chen Ausstattung von Kassettentekordern einige Funktionen und Annehmlichkeiten bietet, die den Umgang mit dem Computer erleichtern.

Preis: 129.- DM
(ohne Netzteil)
Bezugsquelle: COMPUTER
ACCESSOIRES,
Jägerweg 10, 8012 Ottobrunn

Markus Pisters

Das ZX-Spectrum Spiele Buch

Von Richard Hurley
Moderne
Verlagsgesellschaft
148 Seiten, 29.80 DM
Cass. zum Buch 38.- DM
ISBN 3-478-09080-6

Das Buch enthält fünfzehn Programm-Listings, die alle groß und deutlich abgedruckt sind und bei der Eingabe in den Computer keinerlei Schwierigkeiten bereiten. Da bei den Programmen auch UDGs verwendet werden, wurde darauf verzichtet, diese bereits in den Text einzubauen. Man nahm lieber unterstrichene Großbuchstaben, wodurch das Suchen des richtigen Buchstaben im Grafik-Modus entfällt.



Dreizehn der Programme laufen auch auf dem kleinen ZX-Spectrum mit 16 KByte. Bei den abgedruckten Programmen handelt es sich um Grafik-Spiele, die nahezu Profiqualität erreichen. Ist in den Programmen Maschinen-code enthalten, so wird nicht weiter darauf eingegangen.

Das Buch will vor allem die jüngeren User ansprechen, die noch nicht genug Geld besitzen, um sich jedes Spiel zu kaufen.

Horst Müller

Back up Kopie

von Knight Lore + Underwulde

Die Besitzer dieser beiden Programme werden sich zweifelsohne über die recht eigenartige Weise des Ladens wundern, die trotz Originalkassette Schwierigkeiten verursachen kann. Das bringt aber Vorteile, denn eine BACK UP KOPIE kann man auch von schlechteren Rekordern laden. Und was noch viel wichtiger ist: Jetzt kann man endlich auch die Unsterblichkeitspokes eingeben.

Underwulde

1. Anfertigen eines HEADERS: SAVE "under" CODE 16384,46100
2. Erstellen eines Kopierers:
 - a) Eingabe des mc-Programms
 - b) Abspeichern des mc-Programms: SAVE "copy" CODE 65487,32
3. Breaken ins Vorprogramm (während des Beepens) und Eingabe folgender POKES: POKE 24766,161
POKE 62420,195
POKE 62421,207
POKE 62422,255
Dann Zuladen des Kopierers
4. Starten des Programms: RANDOMIZE USR 24740
5. Einladen des Originalprogramms; Einladen des erstellten HEADERS
6. Einlegen einer Leerkassette für die BACK UP KOPIE, Aufnahmetasten des Rekorders drücken, 0-Taste des Spectrums drücken
7. Erstellte Kopie in den Computer laden und durch folgende Befehle zerlegen: SAVE "uwsc" SCREEN, SAVE "under" CODE 26600,35000
8. Erstellen eines Vorprogramms:


```
10 LOAD ""SCREEN
20 LOAD ""CODE
30 RANDOMIZE USR
26610
```

9. Um mit Unsterblichkeit spielen zu können, muß Vorprogramm folgende Zeile ergänzt werden: PAUSE 0: POKE 59376,0

Knight Lore

1. Anfertigen von zwei HEADERN:
 - a) SAVE "klsc" CODE 16384,6912
 - b) SAVE "lore" CODE 24744,37760
2. Wiederholen UNDERWULDE Punkt 2.
3. Breaken ins Vorprogramm (während des Beepens), Eingabe folgender POKES: POKE 24770,168
POKE 62425,195
POKE 62426,207
POKE 62427,255
4. Zuladen des Kopierers. Starten des Programms durch: RANDOMIZE USR 24744, Originalprogramm einladen, HEADER a) zuladen, wiederholen UNDERWULDE Punkt 6., Kopie laden und den SCREEN save.
5. Wiederholen von Punkt 3. Kopierer zuladen, Eingabe folgender POKES: POKE 65500,168
POKE 65501,96
6. Starten des Programms durch: RANDOMIZE USR 24744, Originalcodes einladen, HEADER b) zuladen, wiederholen Underwulde Punkt 6.
7. Erstellen eines Vorprogramms:


```
10 LOAD ""SCREEN
20 LOAD ""CODE
30 RANDOMIZE USR
24832
```
8. Um mit Unsterblichkeit zu spielen, muß folgende Zeile ergänzt werden: 25 PAUSE 0: POKE 53567,0

Lars Kölbr und Mathias Schröder

Spectrum-Eprommer



248.-

- ★ 2716/2732/2732A/2764/27128
- ★ Intelligente Programmierung
- ★ Menü-gesteuert
- ★ zus. Centronics-Schnittstelle
- ★ Keine zus. Hardware erf.
- ★ Preis 248.- DM inkl. Software für EPROMMER und Drucker
- ★ Spez. Sockel zum Austausch Spectrum-ROM - 27128 24.80 DM
- ★ EPROM-Löschgerät 117.00 DM
- ★ Druckerkabel Centronics 37.00 DM

Alle Preise zuzügl. Versand. Info anfordern.

ZX-Spectrum
C&M Meyer
Rahserstraße 52
4060 Viersen 1
☎ 021 62/229 64

Reparatur-Schnell-
dienst/Ersatzteile
Rufen Sie uns an!

MC-Programm

```
10 FOR I=65487 TO 65518
20 READ A
30 POKE I,A
50 DATA 175,55,221,33,239,255,17,17,0,205,2
55 DATA 8,33,0,64,221,33,239,255,1,254,239,2
60 DATA 237,120,203,71,32,250,229,195,132,9
```

ZX81-Kopierschutzprogramm

Dieses Hilfsprogramm erlaubt, ZX81-Programme so aufzuzeichnen, daß sie ganz normal zu laden, aber mit den üblichen Kopierprogrammen nicht zu vervielfältigen sind. Zur Eingabe müssen hinter dem REM in Zeile 1 192 beliebige Buchstaben eingetippt werden. Die Kontrolle erfolgt durch PRINT PEEK 16511 + 256 * PEEK 16512, was 194 ergeben muß (= Anzahl der Buchstaben plus 2). Bevor das Programm zum ersten Mal durch RUN gestartet wird, ist durch RUN 1000 dafür zu sorgen, daß in der REM-Zeile 1 der Maschinencode erzeugt wird.

Anschließend löscht man alle Zeilen ab 1000 und saved das Programm. Es ist nun wirklich

fertig und kann durch RUN aktiviert werden. Sobald es durchgelaufen ist, darf man den Basic-Teil löschen, denn der entscheidende Maschinencode befindet sich geschützt am oberen Ende des RAM-Bereichs. Man lädt jetzt das Programm, das kopiersicher aufgezeichnet werden soll. Durch RANDUSR 32658 wird gesaved. Hierbei wird so aufgezeichnet, daß das Programm nach erneutem Laden automatisch mit der Ausführung der ersten Zeile startet. Der ganze Schutz macht natürlich nur dann Sinn, wenn das schützenswerte Programm durch geeignete Maßnahmen dafür Sorge trägt, daß der nicht autorisierte Benutzer kein Listing produzieren kann (z.B. als BREAK-Schutz die Systemvariable DF-SZ auf Null poken oder Eingabe eines Kennwortes über INKEY\$).

Michael Schramm

```
1 REM HIER MUESSEN 192 BELIEB
IGE BUCHSTABEN EINGEGEBEN WERDEN
```

```
10 RANDOMIZE USR 16514
```

```
20 PRINT "DAS KOPIERSCHUTZPROG
RAMM VON", "MICHAEL SCHRAMM IST B
EREIT.",,,, "SAVE : RANDOMIZE USR
32658"
```

```
30 STOP
```

```
1000 REM ++++++
1010 LET A$="215D7F220440F901003
EC5017606C5ED730240EB219F4001A30
0EDB0C9B500000000F9D41C7E8F00940
00076217D40222940210000221440213
C40010021FD363B0051C5CD4C037123C
110F6217B40C37B03CDE7023A3B40328
17F2A14402273"
```

```
1020 LET A$=A$+"7FFDCB3BB6213C40
222940217040221440215D7F11ED1210
FE1B7AB320F9CD1E032109401633CDFB
7F215E7F1634CDF87F2A737F22144021
7D40222940427E7E10FC213C401621CD
F87F217B40CD16033A817F323B40CF08
CD1E03231520F9C9"
```

```
1030 LET S=16878
```

```
1040 FOR A=16514 TO 16705
```

```
1050 LET X=16*CODE A$+CODE A$(2)
-476
```

```
1060 POKE A,X
```

```
1070 LET S=S-X
```

```
1080 LET A$=A$(3 TO )
```

```
1090 NEXT A
```

```
1100 IF S THEN PRINT "FEHLER IN
A$"
```

```
1110 IF NOT S THEN RUN
```

DNS

Dieses Programm für den ZX 81 mit 16 KRAM erläutert allen Biologiefreaks und solchen, die es werden wollen, anhand einer Grafik die Feinstruktur

der Desoxiribonucleinsäure, die die Erbinformationen einer Zelle enthält. Das Programm startet sich nach dem Laden von Cassette selbständig.

Volker Herren

```
10 REM (C) VOLKER HERREN 1986
15 CLS
20 PRINT "DIE FEINSTUKTUR D
ER DNS:"
30 PRINT
40 PRINT "DIE DNS (DESOXIRIBONU
CLEINSÄURE)"
50 PRINT "ENTHALTEN DIE ERBINF
ORMATIONEN"
60 PRINT "EINER ZELLE."
70 PRINT "SIE BESTEHEN AUS EIN
ZELLENBAU-"
80 PRINT "TEILEN, DIE IN EINER
GENAU FEST-"
90 PRINT "GELEGTEN ANORDNUNG U
ORHANDEN"
100 PRINT "SEIN MUESSEN."
110 PRINT AT 20,0;"FÜR ZEICHNU
NG SIEHE:"
120 IF INKEY$("<")="1" THEN GOTO 12
0
130 IF INKEY$="1" THEN GOTO 900
900 CLS
1000 PRINT "GRAFISCHE DARSTELLU
NG DER DNS:"
1010 FOR A=2 TO 20
1020 PRINT AT A,2;"0"
1030 PRINT AT A,9;"0"
1040 NEXT A
1050 FOR B=2 TO 20 STEP 5
1060 PRINT AT B,3;" "
1070 NEXT B
1110 FOR D=5 TO 20 STEP 5
1120 PRINT AT D,3;"|||||"
1130 NEXT D
1140 FOR E=3 TO 20 STEP 2
1150 PRINT AT E,1;"0"
1160 PRINT AT E,10;"0"
1170 NEXT E
1175 FOR F=3 TO 20 STEP 2
1180 PRINT AT F,2;"<"
1185 PRINT AT F,9;">"
1190 NEXT F
1200 PRINT AT 2,13;"ZEICHEN:"
1210 PRINT AT 3,13;"=====
"
1220 PRINT AT 5,13;"1)=ADENIN"
1230 PRINT AT 7,13;"2)=THYMIN"
1240 PRINT AT 9,13;"3)=GUANIN"
1250 PRINT AT 11,13;"4)=CYTOSIN"
1260 PRINT AT 13,13;"5)=DESOXIR
IBOSE"
1270 PRINT AT 14,13;"(=ZUCK
ER)"
1280 PRINT AT 16,13;"6)=SÄURER
EST DER"
1290 PRINT AT 17,13;"PHOSPHO
RSÄURE"
1300 PRINT AT 19,13;"<TASTE ""1"
">"
1310 IF INKEY$("<")="1" THEN GOTO 13
10
1320 IF INKEY$="1" THEN CLS
1330 IF INKEY$="1" THEN GOTO 150
0
1500 PRINT "INFORMATI
ONEN:"
1510 PRINT
1520 PRINT "ES IST KLAR, DASS DI
E ZEICHNUNG"
1530 PRINT "NATUERLICH SEHR UNGE
NAU IST."
1540 PRINT "ES WURDE ABER DARAUF
GEACHTET,"
1550 PRINT "DASS DIE EINZELNEN T
EILE IN"
1560 PRINT "IHRER ANORDNUNG GENA
U SO IN DER"
```



```

1570 PRINT "ZEICHNUNG VORKOMMEN
WIE IN DER"
1580 PRINT "NATUR."
1590 PRINT "ICH HOFFE, DASS MIR
DIESES AN-"
1600 PRINT "LIEGEN GEGLUECKT IST
"
1610 PRINT
1620 PRINT "=====
=====
1630 PRINT "      (C) BY VOLKER HERR
EN 1986"
1635 PRINT "      ENGELHARDT
STR.17"
1637 PRINT "      8501 SCHW
ANSTETTEN"
1640 PRINT "=====
=====
1650 PRINT
1660 PRINT "  ""1""=NEUSTART"
1665 PRINT
1670 PRINT "  ""2""=ZEICHNUNG"
1680 PRINT
1700 IF INKEY$<>"1" AND INKEY$<>
"2" THEN GOTO 1700
1710 IF INKEY$="1" THEN RUN
1720 IF INKEY$="2" THEN GOTO 900
1800 STOP
9000 SAVE "DN"
9999 RUN

```

Grafik

Bei diesem Programm für den ZX81 mit 16K RAM wird Grafik mit einer Laufschrift gekoppelt, die dann unter der Grafik am Bildschirmrand eingeblendet wird. Folgende Punkte sind zu beachten:

1. Die Steuerung des Grafikpunktes erfolgt mit den Cursor-Tasten.
2. Die Eingabe von "1" bewirkt eine Fixierung des Punktes. Das Malen kann beginnen.
3. Durch die Eingabe von "0" kann man ungewollte "Pater" wieder auslöschen. Die Steuerung erfolgt auch im Löschmodus über die Cursor-Tasten.
4. Mit "N" wird das Programm neu gestartet, wenn man sich einmal hoffnungslos "verfahren" hat.
5. Durch "Z" wird in den Laufschrift-Modus geschaltet. Hierbei ist darauf zu achten, daß die vorher eingegebene Grafik nicht den ganzen Bildschirm beansprucht, da sonst durch lange Laufschrifttexte die unteren Teile der Grafik einfach überschrieben werden. Beendet wird die Laufschrifteneingabe einfach durch Newline.
6. Mit "S" wird das Programm samt Grafik und Laufschrift auf Band gesaved. Vorher ist jedoch das Band zu starten!
7. "A" bewirkt ein Anhalten des Laufschrifttextes zum näheren Betrachten.

```

10 REM (C) VOLKER HERREN 1986
12 REM ENGELHARDTSTR.17
14 REM 8501 SCHWANSTETTEN
20 LET X=0
30 LET M$=""
40 LET Y=X
50 PLOT X,Y
60 LET A$=INKEY$
90 IF A$="0" OR A$="1" THEN LE
T M$=A$
100 IF A$="Z" THEN GOTO 170
110 IF M$="0" THEN UNPLOT X,Y
120 IF A$="N" THEN RUN
130 LET X=X+(A$="0" AND X<63)-(
A$="S" AND X)

```

```

140 LET Y=Y+(A$="7" AND Y<43)-(
A$="5" AND Y)
150 GOTO 50
170 FAST
180 LET N=1
190 INPUT B$
200 SLOW
210 LET B$=B$+"**"
220 FOR X=1 TO 32-LEN A$
230 LET B$=B$+" "
240 NEXT X
250 LET B$=B$+B$( TO 32)
260 PRINT AT 21,0;B$(N TO 31+N)
270 LET N=N+1
280 IF INKEY$="A" THEN GOTO 280
290 IF INKEY$="S" THEN SAVE "GR
AFIK"
295 IF INKEY$="X" THEN GOTO 10
300 IF N=LEN B$-32 THEN LET N=1
310 GOTO 250
320 POKE 16389,68
330 GOTO 250

```

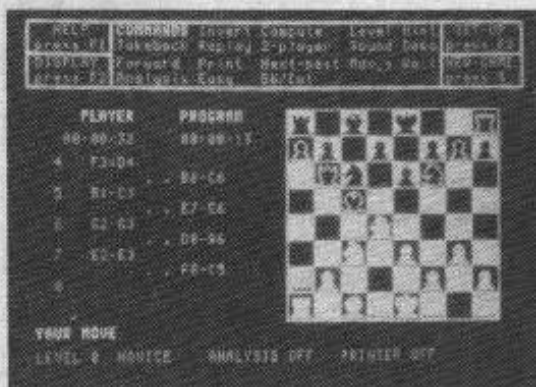
Psion Chess für den QL

Das bislang einzige bei uns erhältliche Schachprogramm für den QL kommt aus dem Hause Psion. Da diese Firma für gute Software bekannt ist, war ich sehr gespannt auf das Programm. Ich kann vorwegnehmen: Es ist so gut, wie ich es mir gewünscht habe. Dem Vorwort der Anleitung kann man entnehmen, daß Psion Chess die Weltmeisterschaft der Microcomputer gewonnen hat. Aber auch ohne diesen Hinweis stellt man nach einigen Partien fest, daß es sich bei diesem Programm um einen ernstzunehmenden Gegner handelt.

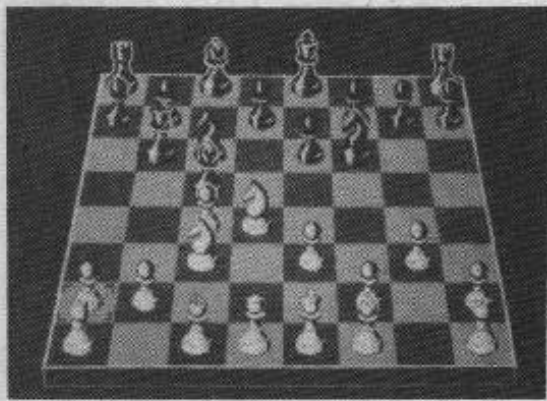
Insgesamt stehen 14 verschiedene Schwierigkeitsstufen zur Verfügung, die jederzeit geändert werden können. Die meisten Stufen regeln die Zeit, die der QL zum Rechnen, also zum Denken bekommt. Das fängt bei 2 Sekunden an und geht bis zu 4 Minuten pro Zug. Darüber hinaus kann man das Programm so einstellen, daß der Computer so lange wie sein Gegner über-

legt, sich also auf dessen Spielstärke einstellt. Die letzte Möglichkeit nennt sich INFINITE TIME. Hier rechnet der QL so lange, bis man über eine Taste den Zug fordert. Wer Psion Chess schlagen will, muß wirklich gut sein.

Neben der reinen Spielstärke bietet das Programm aber noch wesentlich mehr. Da wäre erst einmal die optische Darstellung des Schachbretts und der Figuren. Der Spieler kann zwischen 2- und 3-dimensionaler Darstellung wählen. Beide Varianten haben Vor- und Nachteile. Bei der 3D-Darstellung nimmt das Schachbrett fast die ganze Bildschirmfläche in Anspruch, sieht dabei aber sehr gut aus. Das Brett, die Figuren und auch die Bewegungen vermitteln einen realistischen Eindruck. Gespielt wird übrigens nicht mit den sonst üblichen Eingaben wie z.B. e2-e4, sondern über die Cursortasten und ENTER. Nach kurzer Übung klappt die Bedienung einwandfrei.



Übersichtlich in 2D



Ansprechend in 3D

Das 3D-Brett nimmt viel Platz in Anspruch. Weitere Informationen sind auf dem Bildschirm deshalb nicht möglich. Außerdem ist es nicht immer ganz einfach, die Position einzelner Figuren genau zu erkennen. Aus diesen Gründen haben die Programmierer auch eine 2D-Darstellung eingebaut. Wählt man diese Option, wird das Schachbrett etwa um die Hälfte verkleinert. Der Spieler sieht dann praktisch von oben auf das Brett. Dieser Anblick ist zwar nicht so schön, dafür aber zweckmäßiger. Alle Positionen sind jetzt klar erkennbar. Auch das restliche Bildschirmfeld wird nun voll genutzt. Im oberen Teil stehen die möglichen Kommandos sowie die Belegung einiger Funktionstasten. Links neben dem Spielfeld werden die letzten 5 Züge dokumentiert. Hier ist auch Platz für die beiden Schachuhren.

Drei Funktionstasten sind belegt. F1 ruft die Help-Seiten auf. Hier wurde die komplette Anleitung untergebracht, die man nun jederzeit auf den Bildschirm bringen kann. F2 schaltet zwischen den beiden Darstellungsarten um, und F3 ermöglicht schließlich den Aufbau besonderer Figurenstellungen, um z.B. eine Partie nachzuspielen. Außer den Funktionstasten bietet Psion Chess eine Reihe weiterer Kommandos, die in der Regel über den Anfangsbuchstaben aufgerufen werden.

Level – ändert die Spielstärke;

Easy – reduziert den eingestellten Level;

Hint – macht einen Zugvorschlag;

Takeback – dient der Zugrücknahme;

Forward – hebt eine Rücknahme wieder auf;

Replay – wiederholt eine Partie vom Anfang bis zum Ende automatisch;

Compute – zwingt das Programm, einen Zug zu machen;

Next-Best – zwingt das Programm, seinen Zug zurückzunehmen, um den nächsten besten zu machen;

Analysis – ermöglicht die Analyse einer Partie;

Invert – dreht das Schachbrett;

Wait – hält die Uhren an;

2-Player – erlaubt das Spielen zweier Personen. Das Programm stellt nur noch das Brett;

Sound – schaltet den Sound ein oder aus;

BW/COL – schaltet zwischen farbiger und schwarzweißer Darstellung um;

Print – schaltet die Druckerausgabe ein. Nach jedem Zug wird das Protokoll der Partie mitgeschrieben;

Mdvs – speichert oder lädt den Spielstand.

Neben diesen Befehlen, die immer zur Verfügung stehen, gibt es noch einige spezielle Dinge, auf die ich hier nicht näher eingehen will. Man kann aber schon jetzt sehen, daß es sich bei Psion Chess um ein sehr komfortables Programm handelt. Da auch die Spielstärke und die optische Präsentation gut ist, kann das Programm nur empfohlen werden. Zumindest reine Hobby-Spieler werden vollauf befriedigt sein.

System: QL
Hersteller: Psion
Bezugsquelle: ABC Budde
Rolf Knorre

QL: Was nicht im Handbuch steht!

Wie schon beim ZX-Spectrum gibt es auch beim Sinclair QL etliche Befehle, die nicht im Handbuch stehen. Es handelt sich dabei um folgende ganz schön mächtige und wichtige Befehle: **WHEN**, **WHEN ERROR**, **END WHEN**, **REPORT**, **MISTake**, und die Funktionen **ERNUM**, **ERLIN** und **ERR_NN**. Hierbei steht das **NN** für die 21 verschiedenen Buchstabenkombinationen laut nachstehender Tabelle:

Nr.	Kennung	Bedeutung
-1	NC	Abgebrochen
-2	NJ	Fehlerhafter Job
-3	OM	Speicherüberlauf
-4	OR	Bereichsüberlauf
-5	BO	Puffer voll
-6	NO	Kanal nicht eröffnet
-7	NF	Nicht gefunden
-8	EX	Existiert bereits
-9	IU	In Bearbeitung
-10	EF	Dateiende
-11	DF	Datenträger voll
-12	BN	Ungültige Bezeichnung
-13	TE	Übertragungsfehler
-14	FF	Formatfehler
-15	BP	Ungültiger Parameter
-16	FE	Fehlerhafter Datenträger
-17	XP	Fehler im Ausdruck
-18	OV	Überlauf
-19	NI	Nicht implementiert
-20	RO	Nur Lesen
-21	BL	Syntaxfehler

Die Kürzel ergeben sich aus den englischen Fehlermeldungen, z.B. steht **BL** für **Bad Line**.

Die Frage ist also jetzt nur noch, wie diese Befehle arbeiten. Falls man über **MERGE** ein Basic-Programm lädt, werden alle Zeilen, die von der Syntax her nicht stimmen, durch **MISTake** am Zeilenanfang gekennzeichnet. Eine Überarbeitung des Programms wird dadurch sehr erleichtert.

Das Statement **REPORT** gibt nach einem Fehler die Zeile mit der Fehlermeldung und der Zeilennummer aus, d.h. die normale **QDOS**-Fehlermeldung. Die Voreinstellung ist Kanal 0. (Das sind die Zeilen unten, in die die Befehle einge-

geben werden.) **REPORT #n** gibt auf Kanal **n** aus.

Die Funktionen **ERNUM** und **ERLIN** liefern die letzte Fehlernummer bzw. die Zeilennummer, in der der Fehler aufgetreten ist. Man benötigt diese Funktionen, um den Fehlertyp und den Ort des Fehlers festzustellen. Die Funktionen **ERR_NN** sind Null, wenn alles O.K. ist, und Eins, wenn der entsprechende Fehler auftritt. Welchen Vorteil diese Funktionen gegenüber **ERNUM** bringen, bleibt unklar.

Wenn fast alles zur Fehlerbehandlung vorhanden ist, warum fehlt dann das **ON ERROR GOTO**? Weil es etwas Besseres gibt! Mit **WHEN ERROR** kann eine Prozedur definiert werden, die im Falle eines Fehlers angesprungen wird. Fehler ist alles, was in der Tabelle steht. Ein **STOP**-Statement und das Abbrechen des Programms gilt nicht als Fehler (wie z.B. beim Spectrum). Die Fehler-Prozedur wird mit **END WHEN** abgeschlossen. In der Prozedur kann nun alles stehen, was man gerne hätte. Kommt die Prozedur zum **END WHEN**, wird die Programmausführung mit dem Statement, das auf das Fehlerstatement folgt, fortgesetzt. Man kann also nicht an eine andere Stelle im Programm springen. Zumindest habe ich bisher nicht herausgefunden wie. Nach **WHEN ERROR** darf kein weiteres Statement in der Zeile mehr folgen.

Die Error-Prozedur kann auch mit **CONTINUE** und **RETRY** verlassen werden. **CONTINUE** springt zu dem Statement, das dem fehlerhaften folgt, und **RETRY** wiederholt das fehlerhafte Statement, bei dem natürlich die Fehlerursache abgestellt sein muß, da man sich sonst in einer endlosen Schleife befindet. Leider kann die Fehlerprozedur nicht mit **GO TO** oder **GO SUB** verlassen werden, denn diese Kommandos löschen den Status "Fehlerbehandlung" nicht. Tritt ein weiterer Fehler auf, erhält man die Fehlermeldung "In

Verarbeitung". CONTINUE, RETRY und END WHEN lösen den Status "Fehlerbehandlung". Listing 1 zeigt ein kleines Programm, mit dem die Möglichkeiten der Fehlerbehandlung demonstriert werden.

Fast noch mächtiger ist der Befehl WHEN, der in der Form WHEN <Bedingung> benutzt wird. In der Bedingung muß (so daß Ergebnis meiner Tests) auf der linken Seite eine Variable stehen, die der Super-Basic-Interpreter ständig beobachtet. Falls die Bedingung erfüllt ist, wird die Prozedur abgearbeitet. Eine Überprüfung der Bedingung findet jedesmal dann statt, wenn den zu beobachtenden Variablen ein neuer Wert zugewiesen wird. Ist mit dem neuen Wert die Bedingung erfüllt, erfolgt die Abarbeitung der entsprechenden WHEN-Prozedur. Danach geht es im Basic-Programm an der alten Stelle weiter. Auch diese Prozedur wird mit END WHEN abgeschlossen. Ist eine Variable jedoch Laufvariable einer FOR-

Schleife, kann sie nicht mit WHEN beobachtet werden. Dazu ist dann ein IF in der entsprechenden Schleife notwendig. Ein WHEN ist sinnvoll, wenn man nicht im voraus weiß, wo im Programm die Bedingung erfüllt wird. Listing 2 zeigt ein Demo-Programm, in dem mehrere Variablen mit WHEN beobachtet werden.

Ich frage mich, warum diese Befehle im Handbuch nicht beschrieben sind. Wurden sie schlicht vergessen? Oder enthalten sie Fehler? Letzteres kann nur ein intensives Ausprobieren zeigen. Vielleicht habe ich auch noch nicht alle Möglichkeiten in diesen Befehlen gefunden. Jeder, der noch etwas zu diesem Thema beitragen kann, der sollte an die Redaktion schreiben. Zum Schluß sei noch bemerkt, daß ich alle meine Tests auf einem deutschen QL gemacht habe. Die englischen QLs müßten aber die gleichen Möglichkeiten bieten.

R. W. Gerling

Beispiel 1

```
100 REMark *****
110 REMark Demo für WHEN ERROR
120 REMark R.W. Gerling
130 REMark Mai 1986
140 REMark *****
150 WHEN ERROR
160 nummer=ERNUM
170 SELECT ON nummer
180 ON nummer=-17
190 IF ERLIN=330 THEN
200 PRINT#0;"Es muß eine Zahl
  eingegeben werden"
210 RETRY
220 END IF
230 ON nummer=-7
240 PRINT#0;"Laufwerk nicht gefunden"
250 PRINT#0;"Catridge vergessen?"
260 ON nummer=REMAINDER
270 PRINT#0;"Ich weiß nicht was falsch ist"
280 END SELECT
290 END WHEN
300 REMark *****
310 CLS
320 REPEAT schleife
330 INPUT#0;"Bitte die Nummer des
  Laufwerks eingeben: ";drive
340 DIR "mdv"&drive&"_"
350 END REPEAT schleife
```

Beispiel 2

```
100 REMark *****
110 REMark Demo für Befehl WHEN
120 REMark R.W. Gerling
130 REMark Mai 1986
140 REMark *****
150 WHEN a=5
160 PRINT "'A' hat jetzt den Wert Fünf"
```

```
170 END WHEN
180 WHEN b=8
190 PRINT "'B' hat jetzt den Wert Acht"
200 END WHEN
210 WHEN c>40
220 PRINT "'A*B' ist größer als vierzig"
230 END WHEN
240 REMark *****
250 FOR i=1 TO 7
260 a=i
270 FOR j=5 TO 9
280 b=j
290 c=a*b
300 PRINT a,b,c
310 END FOR j
320 END FOR i
330 REMark *****
```

Pengi

Pengi ist ein kleiner Pinguin, der auf seiner heimatlichen Scholle eine Menge Probleme bekommen hat. Die Scholle, ein kleines Labyrinth aus Eisblöcken, wird von den boshafte Snobees heimgesucht, die sich sofort auf den kleinen Pinguin stürzen, wenn dieser sich nicht dagegen wehrt. Pengi kann aber die Eisblöcke verschieben. Trifft ein solcher Block einen Snobee, wird dieser vernichtet, was Punkte einbringt. Außerdem kann der Pinguin die Blöcke so stellen, daß ein Snobee eingeschlossen und damit ungefährlich wird. Vier Pinguin-Leben stehen zur Verfügung; bei entsprechend hoher Punktzahl kann man aber ein Zusatzleben erhalten.

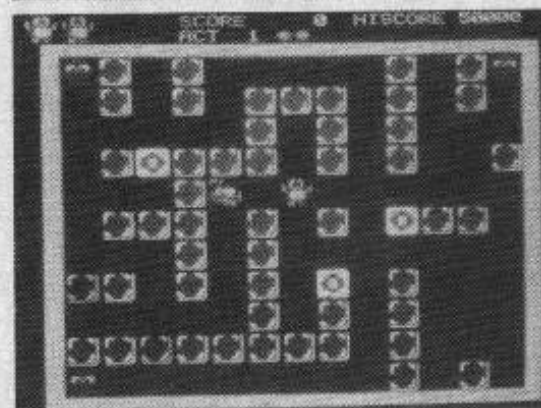
Auf dem Monitor sieht das Spiel recht gut aus. Nach einem Titelbild, das eine schöne Eislandschaft zeigt, erhält der Spieler eine kurze Erklärung. Auf Tastendruck wird das Spielfeld aufgebaut; die erste Runde kann beginnen. Die Anordnung der Eisblöcke ändert sich von Runde zu Runde, so daß das Spiel nicht so schnell langweilig wird. Wer die Spielidee noch nicht kennt oder einfach ein unterhaltsames Spiel für seinen QL sucht, ist mit Pengi gut bedient.

System: QL
Hersteller: Gigasoft
Bezugsquelle: ABC Budde

Rolf Knorre



Zur Einstimmung



Hier wird geklotzt!

Das BEEP-Kommando des QL

Der Soundbefehl des QL ist eine deutliche Verbesserung gegenüber dem simplen BEEP des Spectrum. Daß der Soundbefehl beim QL auch BEEP heißt, täuscht ein bißchen über seine Möglichkeiten hinweg; er kann mehr, als man glaubt. Leider ist die Vielzahl der Parameter verwirrend. Man weiß eigentlich gar nicht so recht, was man angeben soll. Dabei soll das vorliegende Programm helfen. Die Einstellung der Parameter wird digital mit Ziffern und analog durch Schieberegler angezeigt. Drückt man die Cursor-Tasten zusammen mit SHIFT, werden die Parameter grob eingestellt. Ohne SHIFT erfolgt dies in Einerschritten. Die Schrittweite für die Grobeinstellung ist jeweils die letzte Zahl in den DATA-Statements. Sie kann nach Belieben geändert werden.

Bei kurzen Tönen kann der Ton durch Drücken einer beliebigen Taste neu gestartet werden. Zum Verständnis der Tonparameter sei auf die Darstellung im Teil "Begriffe" des QL-Handbuches verwiesen. Es fällt immer wieder auf, daß kleine Veränderungen eines Parameters den Ton völlig ändern.

Hier einige Beispiele für eigene Tonexperimente:

0, 1, 1, 4500, 0, 5, 0, 9 (aus QL-Handbuch)
 0, 1, 11, 4500, 0, 5, 0, 11
 2000, 10 (Bccp als Ersatz für PRINT CHR\$(7))
 0, 10, 100, 2500, 4
 0, 40, 1500, 1, 0, 3, 12
 0, 30, 110, 500, 1
 0, 20, 30, 200, -7
 0, 60, 160, 2000, 1

Das abgedruckte Listing zeigt ein SuperBasic-Programm, das ein einfaches Spielen mit den Tonparametern ermöglicht. Viel Spaß damit!

Rainer W. Gerling

```

1000 REMark *****
1010 REMark *      BEEP Demo      *
1020 REMark *      R.W. Gerling    *
1030 REMark *      (c) Juni 1986    *
1040 REMark *****
1050 :
1060 DEFine PROCedure TITEL
1070 OPEN#1,con_512x256a0x0:CLS:CSIZE 3,1
1080 INK 7
1090 AT 3,9:PRINT"QL Sound-Demo"
1100 INK 4:CSIZE 2,1
1110 AT 6,7:PRINT"Ein Ausprobierprogramm von"
1120 AT 7,14:PRINT"R.W. Gerling"
1130 AT 8,17:PRINT CHR$(127);" 1986"
1140 CSIZE 1,0:AT 24,17:PRINT"Drücke eine beliebige Taste"
1150 PAUSE
1160 END DEFine
1170 :
1180 REMark *****
1190 :
1200 DEFine PROCedure CHANGE
1210 neu=ton%(waagrecht)+inc
1220 IF neu>max(waagrecht) THEN neu=max(waagrecht)
1230 IF neu<min(waagrecht) THEN neu=min(waagrecht)
1240 SYMBOL waagrecht,neu
1250 ton%(waagrecht)=neu
1260 CURSOR 64*(waagrecht-1)+16,210
1270 PRINT ton%(waagrecht);" "
1280 END DEFine
1290 :
1300 REMark *****
1310 :
1320 DEFine PROCedure SYMBOL(xp,yp)
1330 LOCAL yalt,x,y
1340 x=64*(xp-1)+16
1350 yalt=180-(ton%(xp)-min(xp))/(max(xp)-min(xp))*135
1360 y=180-(yp-min(yp))/(max(yp)-min(yp))*135
1370 CURSOR x,yalt:PRINT" "
1380 CURSOR x,y:INK 4:PRINT"<==>":INK 7
1390 yalt=y
1400 END DEFine
1410 :
1420 REMark *****
1430 :
1440 DEFine PROCedure INIT
1450 DIM ton%(8),par$(8,10),min(8),max(8),start(8),inc0(8)
1460 RESTORE 1500
1470 FOR i=1 TO 8
1480 READ par$(i),min(i),max(i),start(i),inc0(i)
1490 END FOR i
1500 DATA " Dauer",-32768,32767,0,500
1510 DATA " Höhe 1",0,255,0,10
1520 DATA " Höhe 2",0,255,0,10
1530 DATA " Grad 1",-32768,32767,0,500
1540 DATA " Grad 2",-8,7,0,1
1550 DATA " Anzahl",0,15,0,1
1560 DATA " Verfremd.",0,15,0,1
1570 DATA " Zufall",0,15,0,1
1580 FOR i=1 TO 8:ton%(i)=start(i)
1590 FOR i=1 TO 8:inc=0
1600 waagrecht=1:sprung=0
1610 END DEFine
1620 :
1630 REMark *****
1640 :
1650 DEFine PROCedure SCREEN
1660 CLS:CSIZE 0,0:
1670 INK 7:PRINT TO(23);"QL Sound-Demo von R.W. Gerling ";
CHR$(127);" 1986"

```

Computer-Kontakt
das Heft mit den
preisgünstigen
Kleinanzeigen


```

1680 BLOCK 3,180,508,20,4
1690 BLOCK 511,15,0,20,4
1700 BLOCK 511,3,0,200,4
1710 FOR i=1 TO 8
1720 CURSOR 64*(i-1),24:INK 0:PAPER 4:PRINT par$(i);
1730 BLOCK 3,180,64*(i-1),20,4
1740 END FOR i
1750 INK 4: PAPER 0
1760 AT 24,0:PRINT"Auswählen mit ";CHR$(188);CHR$(189);" ";:
    INK 7:PRINT CHR$(124);:INK 4
1770 PRINT" Ändern mit ";CHR$(190);CHR$(191);" bzw. mit SHIFT ";
    CHR$(190);CHR$(191);" ";:INK 7:PRINT CHR$(124);:INK 4
1780 PRINT" Grundstellung mit G ";:INK 7:PRINT CHR$(124);:INK 4
1790 PRINT" Ende mit E"
1800 END DEFine
1810 :
1820 REMark *****
1830 :
1840 DEFine PROCedure PFEIL
1850 neu=waagrecht+sprung
1860 IF neu=9 THEN neu=1
1870 IF neu=0 THEN neu=8
1880 CURSOR 64*(waagrecht-1)+34,225:PRINT " "
1890 CURSOR 64*(neu-1)+34,225:PRINT CHR$(190)
1900 waagrecht=neu
1910 END DEFine
1920 :
1930 REMark *****
1940 :
1950 DEFine PROCedure ANZEIGE
1960 FOR i=1 TO 8
1970 CURSOR 64*(i-1)+16,210:INK 7:PAPER 0:PRINT ton%(i);
1980 CURSOR 64*(i-1)+16,36:INK 7:PAPER 0:PRINT max(i);
1990 CURSOR 64*(i-1)+16,190:INK 7:PAPER 0:PRINT min(i);
2000 END FOR i
2010 END DEFine
2020 :
2030 REMark *****
2040 REMark *      Hauptprogramm      *
2050 REMark *****
2060 :
2070 TITEL
2080 INIT
2090 SCREEN
2100 FOR i=1 TO 8:SYMBOL i,start(i)
2110 ANZEIGE
2120 PFEIL
2130 REPeat main
2140 a=CODE(INKEY$(-1))
2150 SElect ON a
2160 ON a=101,69
2170     EXIT main
2180 ON a=103,71
2190     FOR i=1 TO 8:SYMBOL i,start(i):ton%(i)=start(i)
2200     BEEP
2210 ON a=192
2220     sprung=-1
2230     PFEIL
2240 ON a=200
2250     sprung=1
2260     PFEIL
2270 ON a=208
2280     inc=1
2290     CHANGE
2300 ON a=216
2310     inc=-1
2320     CHANGE
2330 ON a=212
2340     inc=inc0(waagrecht)
2350     CHANGE
2360 ON a=220
2370     inc=-inc0(waagrecht)
2380     CHANGE
2390 ON a=REMAINDER
2400 END SElect
2410 BEEP ton%(1),ton%(2),ton%(3),ton%(4),ton%(5),ton%(6),
    ton%(7),ton%(8)
2420 END REPeat main
2430 CLS
2440 REMark *****
2450 REMark *      Ende      *
2460 REMark *****

```

Sinclair QL - QUILL und ABACUS auf einer Cartridge

Wer mit dem QL und speziell mit Quill und Abacus arbeitet, kann sich durch ein neues Boot-Programm Zeit sparen. Er muß dazu nur die Cassette einlegen, mit F1 starten und kann anschließend zwischen Quill und Abacus wählen. Beim Verlassen des jeweiligen Programms kommt man nach Wahl ohne Cassettenwechsel und Neustart in das nächste Programm. Zugleich wird damit der Speicher bei Abacus von 21 K auf 22 K vergrößert.

Ein Nachteil besteht darin, daß Hilfstexte, die mit F1 aufzurufen sind, gelöscht werden. Um das gewünschte Ergebnis zu erhalten, müssen Sie das folgende Programm eingeben und mit SAVE MDV1_BOOT auf einer Cartridge sichern:

```

1 CLEAR: TRA 1
2 AT 8, 10: INPUT "QUILL =
  Q ABACUS = A ":A$
3 IF A$ = "A" OR A$ = "a"
  THEN GO TO 5
4 EXEC_W MDV1_QUILL:
  RUN
5 EXEC_W MDV1_ABACUS:
  RUN

```

Danach müssen Sie nur noch die Programme Quill, Abacus und PRINTER_DAT von der Originalcartridge auf Ihre neue Cartridge bringen. Dazu legen Sie das Original in MDV2 und geben folgendes ein:

```

COPY MDV2_QUILL TO
MDV1_QUILL
COPY MDV2_ABACUS TO
MDV1_ABACUS
COPY MDV2_PRINTER_DAT
TO MDV1_PRINTER_DAT

```

Manfred Reiner

Tom Rowley

Sprühende Ideen mit Atari Grafik**SPRÜHENDE IDEEN**
108 Atari Grafik
1987-Sony

250 Seiten.
Sprühende Ideen ist ein Lehrbuch, das mit den Grafikmöglichkeiten des ATARI in die Gestaltung von Objekten, in Farbgebung und die Entwicklung von Bildschirmtexten einführt. Für den Leser gerufen Kenntnisse der Programmiersprache Basic – auch wenn das Buch gelegentlich die Vorteile der Maschinensprache zeigt.

Bestellnummer TW 104

DM 49,-

C. Lorenz

Das große Spielebuch für Atari

200 Seiten.
Dieses Buch enthält eine Reihe aktueller Programme für den Atari 800 XL und 800 XL und ist eine Weiterführung von Band 1, dem großen Spiele-Buch für Atari. Es bringt eine Reihe neuer Spiele, Programme zur Sounderzeugung und ein Kapitel über Grafik-Spielerien mit dem Atari. Außerdem enthält es einige Tips und Programme zum Zeichensatz des Atari.

Bestellnummer H 820

DM 29,80

C. Lorenz

Das große Spielebuch für Atari

151 Seiten.
Aufregende Computerspiele in Atari-Basic. Neben Spielen finden Sie hier eine Reihe hochinteressanter Anregungen für eigene Programme. 3D-Grafik, Bewegung und Scrollen, Grafik und Ton in FORTH, Tonprogrammierung usw.

Bestellnummer H 821

DM 29,80

Handbuch für Hacker**HANDBUCH FÜR HACKER**
1987-Taschenrechner

In diesem Buch finden Sie endlich alle Antworten auf die Fragen, die sich jeder stellt und bislang keiner beantworten konnte. Mit vielen Nummern und Datenbanken. Endlich wird hier erklärt, wie man mit Datex-P arbeitet. Viele Tips und Tricks zum Mailboxbetrieb. Steigen Sie mit ein in die Unterwelt der DPU.

Bestellnummer 6

DM 38,-

Raeto West

C 64/SX 64 – Computer Handbuch

688 Seiten.
Insgesamt ein Jahr verarbeitete der Autor Raeto West mit der Analyse und Dokumentation auf den C64. Das Ergebnis seiner Arbeit: Das einzige enzyklopädische 64er Buch, das neben Ihrem Computer liegen sollte. Das Buch ist geprägt von solidem Fachwissen und umfangreicher Kompetenz. Es enthält über 300 Programmierungen aller 64er Funktionen – auch die schwierigen, seltenen und meist genötigten.

Bestellnummer TW 102

DM 66,-

Don Inman / Kurt Inman

Der Atari Assembler

276 Seiten, 82 Abb., ca. 100 Programme.
Mit diesem Buch können Sie das Programmieren in Assembler lernen und sich gleichzeitig mit der Anwendung des Atari Assembler Moduls auf Ihrem Atari 400- oder 800-Modell vertraut machen. Das Buch ist eine ausgezeichnete Einführung für Leser mit einigem Grundwissen in Basic, setzt aber keinerlei Assembler-Kenntnisse voraus.

Bestellnummer ID 202

DM 38,-

Karl-Heinz Koch

ATARI Spiele programmieren

240 Seiten.
Das Buch führt Schritt für Schritt in das Programmieren in BASIC ein. Dabei werden schon mit den ersten einfachen Befehlen faszinierende Grafikeffekte erzielt. So werden die Befehle und ihre Wirkung optisch erfahrbar gemacht. Auf Verständlichkeit wird besonders Wert gelegt, was für Bücher dieser Materie leider keine Selbstverständlichkeit ist.

Bestellnummer BI 907

DM 32,-

Julian Reschke / Andreas Wlethoff

Das ATARI Profibuch

296 Seiten. Das richtige Buch für Sie, wenn Sie einen ATARI 400, 800, 800 XL, 800 XL oder 130XE besitzen. Das Buch enthält eine vollständige Liste aller Systemadressen und -routinen mit Erläuterungen; detaillierte Informationen über Speicherbelegung und Ausnutzung des Betriebssystems; einen Tabellenteil mit Zeichencodes, Adressen, Kommandos, Tonhöhen, Farben, Fehlermeldungen sowie einen alphabetischen Nachschlageteil.

Bestellnummer SY 105

DM 42,-

Hal Hicksman

Der ATARI als Musikbox

194 Seiten. Dieses Buch führt Sie in die Grundbegriffe der Musik ein. Erst danach werden die Möglichkeiten der Computermusik dargestellt. Das Buch ist für Anfänger geschrieben und zeigt, wie man mit einfachen Basicprogrammen Melodien, Akkorde und Kanons erstellen kann. Sie müssen weder Noten lesen noch ein Instrument spielen können, um mit diesem Buch Ihren ATARI in ein mehrstimmiges Orchester zu verwandeln.

Bestellnummer MT 123

DM 29,80

Chaos Computer Club

Die Hackerbibel

259 Seiten.
Überall in den Medien ist seit geraumer Zeit Rede von den Hackern. Und wenn man "Hacker" sagt, meint man in diesem unserm Lande zu meist Mitglieder des Chaos Computer Club Hamburg und seine Ableger. Sei es der 130.000 DM Coup bei der Hamburger Sparkasse oder die diversen Auftritte in Tagesschau, bei Frank Elstner und anderswo: über sie ist viel berichtet worden. Hier erstmals ein Buch von ihnen. Nicht nur über das "Wie" des Hackens, sondern auch über das "Warum".

Bestellnummer H 500

DM 33,33

BUCH-BESTELLKARTE

Bitte liefern Sie mir folgende Bücher:

Anzahl	Bestell-Nr.	Titel	Einzel-Preis inkl. MwSt.

Name des Bestellers

Anschrift

PLZ/Ort

Telefon

Ich wünsche folgende Bezahlung:

- ☐ Nachnahme (+ 5,70 DM Porto + Versandkosten)
☐ Vorauskasse (keine Versandkosten)

bei Vorauskasse bitte Scheck beilegen oder auf Postscheckkonto Karlsruhe 43423-756 überweisen.

Datum/Unterschrift

Coupon ausschneiden, auf Postkarte kleben und einsenden: Verlag Rätz-Eberle, Postfach 16 40, 7518 Bretten.



COMMODORE - NEWS

Liebe C 64-User!

Auf den hart umkämpften europäischen Märkten ist Commodore einer der wenigen Hersteller, die eine lückenlose Produktlinie vom Homecomputer für Einsteiger (C64) über den professionellen Personal Computer bis zum innovativen "Mediencomputer" (Amiga) anbieten. Dies ist sicher einer der Gründe dafür, daß Commodore seine führende Marktstellung nicht nur hält, sondern – allen Unkenrufen zum Trotz – weiter ausbauen kann.

Gerade das Gegenteil widerfuhr einer anderen, ebenfalls sehr bekannten Firma. Die in München ansässige SM Software AG hat es nicht geschafft, das Raubkopieren ihrer Programme in den Griff zu bekommen, und mußte Konkurs anmelden. Allerdings ist die ehemalige Tochtergesellschaft SM Soft-Training GmbH mit ihren Lernprogrammen, die dem-

nächst auch auf PC verfügbar sein sollen, weiterhin auf dem Markt vertreten. Wir werden einige dieser Programme in den nächsten Ausgaben vorstellen.

Für die jetzige Ausgabe haben wir einige Leckerbissen für Euch, um den nun herannahenden Herbst ein bißchen aufzuheilen. Neben dem zweiten Teil des Grafikurses zeigen wir am Beispiel des C 64, wie man eine Mailbox aufbaut. Und für Leute, die ständig Ärger mit dem Schreib/Lesekopf ihrer Floppy haben, bringen wir eine komplette Justageanleitung.

Soviel für diesmal und viel Spaß beim Studium dieser Ausgabe.

Frank Wolk

P.S. Schreibt uns doch mal, was Euch besonders interessiert. Vielleicht können wir das dann berücksichtigen.

The Golf Construction Set

Golf gehört in letzter Zeit zu den beliebtesten Sportarten, die auf dem Computer simuliert werden. Eigentlich sollte man annehmen, daß das Nobelspiel sehr schwer und mit wenig Erfolg den Weg in heimische Wohnzimmer finden würde. Doch der Schein trügt. Mit großem Enthusiasmus golft man in Deutschland auf den großen

Plätzen der Welt um die Wette, und nur wenige haben es bisher geschafft, sich dem Reiz dieses Spiels auf dem Computer zu entziehen.

Der Grund für diese Faszination liegt aller Wahrscheinlichkeit nach in der hohen Präzision, die mit den schier unendlichen Einzeleinstellungen in je-

den Schlag gelegt werden kann. Das vorliegende Programm bietet eine solche Schritt-für-Schritt-Simulation. Nachdem man den gewünschten Kurs ausgewählt hat, müssen vor dem ersten Schlag die aktuellen Wind- und Geländedaten beachtet und einige Schläger aus dem Korb zurückgelassen werden, weil der Träger sonst über-



lastet wäre. Nun erscheint auf der rechten Bildschirmseite eine Übersicht über das Gelände mit einer speziellen Notation, die Berge, Hügel und sonstige Hindernisse oder Besonderheiten des Geländes anzeigt. Vor jedem Schlag wählt man den richtigen Schläger, den Schlagwinkel, die Fußstellung, das geplante Schlagziel und sogar den richtigen Schwung beim eigentlichen Schlag.

Der große Vorteil des Golf Construction Set liegt aber nicht in diesen hervorragend dargestellten Simulationsschritten. Über das Hauptmenü wird ein Konstruktionsset ange-

wählt, mit dem man eigene Golfplätze erstellen kann. Wer also die mitgelieferten Plätze gemeistert hat, kann sich selbst neue, immer schwerere Aufgaben stellen.

Das Golf Construction Set ist wohl eine der besten Golfsimulationen für den C64. In Grafik und Animation kann sie mit der Konkurrenz auf jeden Fall mithalten. Der Konstruktionsteil hebt das Spiel über das Niveau der anderen Programme hinaus und macht es auch auf längere Sicht interessant.

System: C64
Hersteller: Ariolasoft
Bezugsquelle: Profisoft GmbH
Osnabrück
Thomas Tai

Denksport auf dem Commodore 64

Von John Stephenson
Carl Hanser Verlag
200 Seiten, 29.80 DM
ISBN 3-446-14333-5

Professor John Stephenson's Buch fehlt das sonst übliche "Zisch", "Bumm" und "Boing" der Computerspiele. Dafür enthält es 47 Spiele, die Konzentration, Mathematikkenntnisse sowie logisches und strategisches Denken verlangen. Sie reichen von Konzentrationsspielen, wie z.B. Memory oder



die Bewegung eines Springers über das Schachbrett, über Wortspiele, wie z.B. das bekannte Galgenmännchen, bis zu Logik- und Strategiespielen, wie Schiffe versenken, Nimm und die Türme von Hanoi. Im mathematischen Teil geht es um Kopfrechnen, komplexe Zahlen, analytische Geometrie, Algebra sowie um zwei Programme zur Nullstellenbestimmung. Im Anhang finden sich ein Loader und einige Programme zur

Der Profi-C-Compiler

Die Programmiersprache C, zur Zeit in aller Munde, wurde bereits 1972 von Dennis Ritchie und Brian Kernighan für das Betriebssystem UNIX entwickelt. Der Profi-C-Compiler ist eine Weiterentwicklung der ursprünglichen Sprache. Data Becker hat dieses Programm für den C128 herausgebracht, vorausgegangen war C für den C64. Daß selbst IBM seine PCs auf C umrüstet, legt Zeugnis für den erfolgreichen Weg von C ab.

Der C-Compiler besteht aus mehreren Programmteilen.



musikalischen Untermalung
des Ganzen.

Jedem Programm hat Stephenson einige Seiten zur Erklärung gewidmet, die über Sinn des Spiels, Spielablauf, den Programmaufbau und, falls nötig, über die Mathematik des Programms informieren. So bekommt man nicht nur interessante und unterhaltsame Spiele, sondern dazu auch noch Nachhilfe in Mathematik sowie Tips zur Basic-Programmierung.

Einziger Mangel: Abgesehen von den wenigen im Buch enthaltenen Fehlern, geschehen beim Abtippen besonders der längeren Programme (die längsten beanspruchen über zwei Seiten) häufig Fehler, so daß eine Prüfsummenroutine oder eine Diskette/Cassette zum Buch, die einem zusätzlich noch das recht langwierige Abtippen ersparen würden, wünschenswert wären.

Axel Klenk

Der Kommandoprozessor beinhaltet einige resistente Befehle, d.h., die Befehle müssen nicht mehr nachgeladen werden. Transiente Befehle können in C vom Anwender selbst entwickelt werden.

Eine komfortable Eingabe von C-Programmen ermöglicht der Editor. Die Profi-C-Version für den 128 beinhaltet eine RAM-Floppy, die deutliche Geschwindigkeitsvorteile bringt. Der gesamte Inhalt einer RAM-Floppy kann auch auf Diskette abgespeichert werden, womit eine Nachladung möglich wird. So können bis zu acht Floppy-Laufwerke verwaltet werden. Sie erhalten Laufwerksnamen wie A, B, C, usw.

Auch die vielen vom C 64 auf den C 128 umgestiegenen User brauchen sich nicht zu grämen: Ihre Programme laufen auch unter Profi-C und sind auf der Quellfilebene 100 % kompatibel. Sie müssen nur neu kompiliert und gelinkt werden. Alte Bibliotheken sind jedoch auf den neuen Stand umzuschreiben.

Im Gegensatz zum C64 ist der PC128 ja weitaus komplizierter aufgebaut. Man kann ihn zum Beispiel im 80-Zeichen-Modus oder im 40-Zeichen-Modus verwenden. Gleiches gilt für das schnelle Diskettenlaufwerk 1570/71. Profi-C nutzt diese Vorteile optimal aus. Als RAM-Floppy ist von den Autoren die zweite Bank, also die anderen 64 K des 128, gewählt worden. Sie ist auch weitgehend kompatibel zu den 1541/70/71-Laufwerken.

Der Kommandoprozessor des Compilers ist wie der von MS-DOS und CP/M aufgebaut. Im Zusammenhang mit C steht meist UNIX mit seinen Bibliotheken. Auf der Systemdiskette befinden sich einige solcher Dateien, z.B. eine Grafikdatei, welche speziell auf den C128 zugeschnitten ist.

Der C-Compiler legt beim Compilieren die Fehler in einem Error-File ab. Im Basic dagegen steigt der Compiler aus und zeigt den Fehler bzw. die Fehlerzeile. Dann muß neu gestartet werden. Also ein Riesenvorteil von C! Der C-Compiler ermöglicht das Editieren aller Errors, indem der Anwender ein Error-File lädt und überarbeitet. Der Befehlsvorrat dazu besteht aus wenigen Befehlen. Ein lauffähiges Programm entsteht dann durch Verwendung der erwähnten Programmbibliotheken.

Der Profi-C-Compiler bringt eine Effizienz, die den professionellen Programmierer begeistert. Aber auch Einsteiger können sich aufgrund des guten Handbuchs relativ leicht einarbeiten. Wer up to date sein will, kommt um diese lohnende Investition nicht herum.

System: Commodore 128

Hersteller: Data Becker

Preis: Disk 298.- DM

Thorsten Traub

Shogun

Genau passend zur Wiederholung der gleichnamigen Fernsehserie erschien das Computerspiel zu dem bekannten Fernost-Spektakel. Damit wäre ein weiteres Trio perfekt: Buch, Film und Computerspiel. Die Qualität von Buch und Film ist

unbestritten. Jedermann konnte sich davon überzeugen. Gleichen Ansprüchen wird das Computerspiel jedoch nicht gerecht. Die Handlung baut nämlich nicht wirklich auf der des Originals auf. Es wurde im Gegenteil ein ganz normales Action-Adventure lediglich mit einem guten Titel versehen.

Im Programm kommen natürlich die Hauptpersonen des Buches vor. Zu Beginn kann man sich sogar aussuchen, in welche Rolle man schlüpfen möchte: Fürst Toranaga, Kapitän Blackthorne oder vielleicht ein mutiger Samurai? Die Wahl bleibt vollkommen dem Spieler überlassen. Für die Handlung selbst ist dies von untergeordneter Bedeutung, denn theoretisch kann jeder das Ziel des Spiels erreichen, den Thron des Shogun. Die derzeitige Regierung soll gestürzt und die Macht übernommen werden. Zu diesem Zweck muß der Spieler, egal in welcher Person er verkörpert ist, 20 Verbündete finden, die ihm diesen Putsch ermöglichen.

Der Weg dorthin ist weit und reichlich mit Stolpersteinen versehen. Nicht jeder kann um ein Bündnis gebeten werden. Zum richtigen Zeitpunkt müssen Feinde erkannt und ausgeschaltet oder ein Freund richtig behandelt werden, um sich mit ihm verbünden zu können. Die jeweilige Handlung wird durch Symbole unter dem Handlungsfenster dargestellt. Auf Knopfdruck stoppt das Spiel und man kann ein Icon anwählen.

"Shogun" besteht aus etwa 128 Bildschirmen, auf denen sicherlich halb Japan abgebildet ist. Die Grafiken sind nicht vom Allerfeinsten, doch in Farbe sicherlich annehmbar. Beim Sound geizte man ein wenig mit den Möglichkeiten des C 64, so daß eigentlich nur noch die Handlung bleibt, die sicherlich nicht so interessant und komplex ist, wie man es von einem Spiel, das sich mit solchen Federn schmückt, erwarten dürfte.

System: C 64
Hersteller: Virgin Games
Preis: ca. 40,- DM
Bezugsquelle:
Profisoft GmbH, Osnabrück
Thomas Tai

Grafikkurs für den C 64

2. Teil: der VIC-II

Im ersten Teil dieses Kurses wurde besprochen, wie der Bildschirmspeicher und der Farbspeicher des C 64 aufgebaut sind. In diesem Heft werden wir uns ausführlich mit dem Baustein beschäftigen, der für die Video-Aufbereitung zuständig ist, mit dem VIC-II. Die ersten einführenden Bemerkungen dazu wurden bereits im ersten Teil des Kurses gemacht.

Die Register des VIC-II

Eine Liste der Register des VIC-II sei zunächst vorgestellt. Die Basisadresse des VIC-II (im folgenden nur noch als VIC bezeichnet) liegt bei \$D000 (#53248). Einige Informationen zum VIC finden sich schon im C-64-Handbuch. Dort ist die Dokumentation der Register aber nicht sehr übersichtlich gestaltet.

Der Rechner selbst kann bis zu 8 Sprites gleichzeitig verwalten. Sprites sind frei definierbare Grafikobjekte, die innerhalb des Bildschirms mittels der entsprechenden Register bewegt werden können.

Für jedes dieser 8 Sprites gibt es ein Register für die Verwaltung der x-Position und ein Register für die Verwaltung der y-Position:

Register #00:
x-Position von Sprite 0.

Register #01:
y-Position von Sprite 0.

Register #02:
x-Position von Sprite 1.

Register #03:
y-Position von Sprite 1.

Register #04:
x-Position von Sprite 2.

Register #05:
y-Position von Sprite 2.

Register #06:
x-Position von Sprite 3.

Register #07:
y-Position von Sprite 3.

Register #08:
x-Position von Sprite 4.

Register #09:
y-Position von Sprite 4.

Register #10:
x-Position von Sprite 5.

Register #11:
y-Position von Sprite 5.

Register #12:
x-Position von Sprite 6.

Register #13:
y-Position von Sprite 6.

Register #14:
x-Position von Sprite 7.

Register #15:
y-Position von Sprite 7.

Der C 64 kann in x-Richtung maximal 320 Punkte und in y-Richtung maximal 200 Punkte auflösen. Will man ein Sprite über den Bildschirm bewegen und die dafür benötigte Information in nur einem Register unterbringen, so sind dabei nur die Werte 0 bis 255 zulässig. Damit läßt sich in der y-Richtung jede Position einnehmen, in x-Richtung kann man aber auf diese Art und Weise den rechten Bildschirmrand nicht erreichen. Abhilfe schafft hier ein weiteres Register, in dem für das jeweilige Sprite ein Flag gesetzt wird. Anhand dieses Flag wird entschieden, ob das entsprechende Sprite in x-Richtung um 256 Punkte nach rechts verschoben wird oder nicht. Weil es somit pro Sprite nur zwei logische Zustände gibt, reicht schon ein Bit aus, um den Zustand eines Sprites zu beschreiben. Folgerichtig werden für 8 Sprites 8 Bits benötigt, die zu einem Byte zusammengefaßt werden können. Auf diese Weise ist dann auch das folgende Register organisiert:

Register #16:
8 Flags für die Verschiebung der Sprites in x-Richtung um 256 Punkte (1 = Verschiebung aktiviert, 0 = keine Verschiebung).

Register #17:

Bit 0-2: Für die vertikale Verschiebung des Bildschirms in Rasterzeilen. Unter Verwendung dieses Registers kann ein "Softscrolling" programmiert werden.

Bit 3: Ist dieses Bit gelöscht, werden vertikal nur 24 Zeilen dargestellt. Normalerweise wird dann der Bildschirmrand oben und unten um jeweils 4 Punktzeilen zur Bildschirmmitte hin gerückt. Mit Hilfe von Bit 0-2 läßt sich der verbleibende Bildausschnitt so zentrieren, daß ganze 24 Zeilen dargestellt werden. Eine ganze Zeile ist dann unter dem Bildschirmrahmen versteckt.

Bit 4: Wird dieses Bit gelöscht, ist der Bildschirm ausgeschaltet. Die gesamte Sichtfläche des Monitors ist mit der Farbe des Rahmens gefüllt.

Bit 5: Ein Flag, das für unsere weiteren Betrachtungen eine wesentliche Bedeutung hat, denn mit ihm wird der Grafikmodus des Rechners gesteuert. Bei gesetztem Bit ist der Grafikmodus aktiv.

Bit 6: Ist dieses Bit gesetzt, arbeitet der Rechner im sogenannten "Extended-Background-Color-Mode". Damit ist es möglich, nicht nur individuell die Zeichenfarbe zu wählen, sondern zusätzlich noch eine Hintergrundfarbe für jedes einzelne Zeichen auszusuchen. Die Anzahl der darstellbaren Zeichen ist auf die ersten 64 Bildschirpcodes reduziert (siehe Tabelle der Bildschirpcodes im C 64-Handbuch).

Bit 7: Enthält Bit 9 des Rasterregisters.

Register #18:

Enthält bei einem lesenden Zugriff 8 von 9 Bits der Punktzeile, die gerade auf dem Bildschirm aufbereitet wird. In das Register kann ein Wert geschrieben werden, dem dann ein mögliches Interruptereignis zugeordnet wird.

Register #19:

Dieses Register enthält die x-Koordinate des Lightpens.

Register #20:

Enthält die y-Koordinate des Lightpens.

Register #21:

Ähnlich wie in Register 16 ist hier jedem Sprite ein Bit zugeordnet. Ist das jeweilige Bit gesetzt, so ist das der Bitnummer entsprechende Sprite eingeschaltet und somit sichtbar.

Register #22:

Bit 0-2: Zuständig für die horizontale Verschiebung des Bildschirms mit einer Auflösung von einer Punktweite.

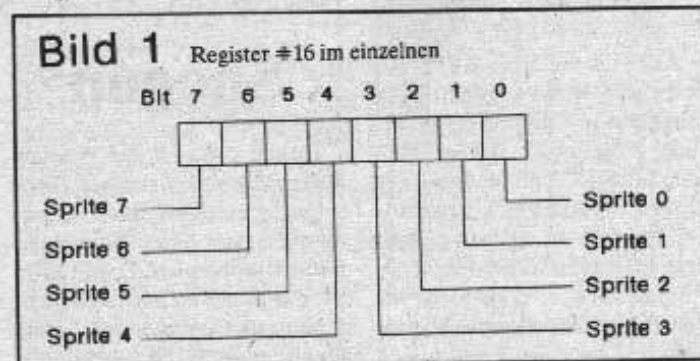
Bit 3: Kontrolliert, wie viele Zeichen je Zeile dargestellt werden. Ist dieses Bit gelöscht, wird der Bildausschnitt verschoben. Mit Hilfe der ersten 3 Bits kann der Bildschirm so zentriert werden, daß nur noch 38 Zeichen je Zeile dargestellt werden. Ist dieses Bit gesetzt, werden 40 Zeichen pro Zeile dargestellt (normaler Einschaltzustand).

Bit 4: Mit diesem Bit wird der "Multicolor-Mode" eingeschaltet. In diesem Modus wird keine hohe Punktauflösung erzielt, dafür können aber verschiedene Farben für jeden einzelnen Punkt des Bildschirms gesetzt werden.

Bit 5: Stellt das Reset-Bit des VIC dar und muß im normalen Betrieb gelöscht werden.

Register #23:

Dieses Register ist ebenfalls flagartig organisiert. Ist für das jeweilige Sprite das entsprechende Bit gesetzt, wird das Sprite in y-Richtung doppelt so breit dargestellt.



Register #24:

Dieses Register hat für unsere weiteren Betrachtungen ebenfalls besondere Bedeutung, da hier unter anderem festgelegt wird, wo der Beginn des Zeichengenerators liegt. Der Zeichengenerator selbst setzt sich aus den Daten zusammen, die benötigt werden, um in der Standardbetriebsart Zeichen auf dem Schirm darzustellen.

Bit 1-3: Legt den Beginn des Zeichengenerators fest, wenn der Grafikmodus ausgeschaltet ist. Ansonsten wird Bit 3 herangezogen, um einen von zwei möglichen Grafikschrmen zu selektieren.

Bit 4-7: Mit diesen Bits wird die Basis des Video-RAMs festgelegt. Dadurch ist es möglich, den Bildschirm anfang auf eine in Grenzen frei wählbare Adresse zu legen.

Register #25:

In diesem Register werden bestimmte Interruptzustände verwaltet. Die folgenden Beschreibungen beziehen sich auf einen lesenden Zugriff.

Bit 0: Dieses Bit wird gesetzt, wenn das Ereignis, das durch einen schreibenden Zugriff auf Register 18 definiert wurde, eingetroffen ist.

Bit 1: Berührt ein Sprite ein Zeichen, dann wird dieses Bit gesetzt.

Bit 2: Falls 2 beliebige Sprites sich berühren, so wird dieses Bit auf den logischen Zustand 1 gebracht.

Bit 3: Eine Aktion des Lightpens beeinflusst dieses Bit.

Bit 7: Wird eines der vorhergehend beschriebenen Bits gesetzt, wird automatisch auch dieses Flag auf 1 gesetzt.

Nur mit einem schreibenden Zugriff können die Bits dieses Registers gelöscht werden. Dazu muß für das zu löschende Bit eine logische 1 geschrieben werden.

Register #26:

Mit diesem Register wird definiert, welches der Ereignisse aus Register 25 einen Interrupt auslösen darf. Die Bitbelegung entspricht der des Registers 25. Das 7. Bit des Registers ist nicht belegt.

Die Register 25 und 26 lassen sich besonders in Maschinensprache effektiv verwenden, da auf ein entsprechendes Ereignis per Interrupt sofort reagiert werden kann.

Register #27:

Auf dem Bildschirm ist es damit möglich, Zeichen und Sprites gleichzeitig darzustellen. Mit Hilfe dieses Registers kann nun definiert werden, ob ein Zeichen über oder unter einem Sprite steht. Für den letzteren Fall muß das korrespondierende Bit gelöscht werden.

Register #28:

Es gibt für Sprites zwei Darstellungsarten. Eine davon wird "Multicolor-Mode" genannt. Soll diese Darstellungsart eingeschaltet werden, muß das dem Sprite entsprechende Bit gesetzt werden. Im "Multicolor-Mode" können Sprites in mehreren Farben wiedergegeben werden.

Register #29:

Mit diesem Register wird festgelegt, ob Sprites in x-Richtung doppelt so breit dargestellt werden. Dazu ist für jedes einzelne Sprite das korrespondierende Bit zu setzen.

Register #30:

Kollidieren zwei Sprites miteinander, so werden in diesem Register die den Sprites zugeordneten Flags gesetzt.

Register #31:

Mit Hilfe dieses Registers läßt sich feststellen, ob ein Sprite ein Zeichen auf dem Bildschirm berührt. Ist das der Fall, so steht in dem dem Sprite zugeordneten Flag der Bitwert 1.

Register #32:

Rahmenfarbe.

Register #33:

Hintergrundfarbe (#0).

Register #34:

Hintergrundfarbe (#1).

Register #35:

Hintergrundfarbe (#2).

Register #36:

Hintergrundfarbe (#3).

Register #37:

In diesem und im folgenden Register werden die Farbinformationen für Multicolor-Sprites abgelegt.

Register #38:

siehe Register 37.

Register #39:

Farbe für Sprite #0.

Register #40:

Farbe für Sprite #1.

Register #41:

Farbe für Sprite #2.

Register #42:

Farbe für Sprite #3.

Register #43:

Farbe für Sprite #4.

Register #44:

Farbe für Sprite #5.

Register #45:

Farbe für Sprite #6.

Register #46:

Farbe für Sprite #7.

Damit ist die Belegung des VIC vollständig. Es ist jedoch klar, daß eine bloße Beschreibung der Registerinhalte des VIC nicht ausreicht, um "den Durchblick" zu bekommen. Deshalb werden wir uns noch ausführlich mit dem Video-RAM und dem Zeichengenerator beschäftigen und in den weiteren Kursteilen auf die Programmierung von hochauflösender Grafik und Spriteanimation eingehen.

Der Zeichengenerator

Der Zeichengenerator des C64 ist in der Standard-Betriebsart des VIC aktiviert. Bereits im ersten Teil unseres Kurses wurde gezeigt, daß sich jedes Zeichen auf dem Schirm aus 8 Bytes zusammensetzt. Der Zeichengenerator enthält nun für alle Zeichen des Commodore-Zeichensatzes die Daten, die benötigt werden, um ein Zeichen zu bilden. Auf den genauen Aufbau des Zeichengenerators werden wir eingehen, wenn beschrieben wird, wie der Zeichensatz des Rechners verändert werden kann.

Der VIC kann immer nur auf einen 16 Kilobyte langen Block des Rechners zugreifen. Aufgrund dieser Besonderheit muß auch in dem aktuellen 16-K-Block der Zeichengenerator abgelegt sein, denn schließlich könnten Sie sonst keine Zeichen auf dem Bildschirm erkennen. Auch darf der Block nicht irgendwo im Rechner beginnen, sondern nur an den folgenden festen Adressen:

Block	Beginn	Ende	Bit 10 dez
3	\$0000	\$3FFF	113
2	\$4000	\$7FFF	102

1	\$8000	\$BFFF	011
0	\$C000	\$FFFF	000

Um dem Rechner mitzuteilen, welcher der 16-K-Blöcke aktiviert werden soll, wird der aus obiger Tabelle ersichtliche Dezimalwert in die Speicherzelle \$DD00 (#56576) geschrieben. Bei dieser Speicherstelle handelt es sich um ein CIA-Register (ein Steuerbaustein des C64). Bei einem schreibenden Zugriff ist darauf zu achten, daß die anderen Bits der Speicherstelle unverändert bleiben.

Nach dem Einschalten des Rechners ist offenbar Block 3 aktiv, denn wie Sie bereits wissen, beginnt das Video-RAM dann bei \$0400 (#1024). Auch dieses muß in dem Block liegen, mit dem der VIC gerade arbeitet.

Es sei hier noch einmal ausdrücklich darauf hingewiesen, daß im Gegensatz dazu die Startadresse des Farb-RAMs nicht variierbar ist. Hier liegt der Beginn immer bei \$D800 (#55296).

Einen Haken scheint die ganze Sache trotzdem zu haben. Wenn Sie ein Programm schreiben, wird dieses im Rechner beginnend mit der Adresse \$0800 (#2048) abgelegt. Irgendwann ist dann vielleicht ihr Programm so groß geworden, daß es einen Speicherplatz von z.B. 20 Kilobyte benötigt. Dann ist das gesamte Programm länger als der Block 3, und demzufolge müssen die Daten des Zeichengenerators doch überschrieben sein, oder?

Die Antwort darauf müßte lauten: "Richtig!", wenn da nicht eine Eigenart des VIC dafür sorgen würde, daß der Zeichengenerator, wenn er bei \$1000 (normale Einstellung) oder bei \$9000 beginnt, seine Zeicheninformation nicht mehr aus dem jeweils aktuellen 16-K-Block holt, sondern dort nachschaut, wo der Zeichengenerator auch physikalisch vorhanden ist. Die Basisadresse des Zeichengenerators lautet \$D000 (#53248), und durch seine Länge von 4 Kilobyte reicht er bis \$DFFF.

Und schon hat man das nächste Problem. Gerade eben noch wurde festgestellt, daß die Basisadresse des VIC bei \$D000 liegt. Demzufolge würden also

die physikalische Startadresse und die Startadresse des VIC zusammenfallen. Wo liegt da des Rätsels Lösung? Ganz einfach: Der Zeichengenerator enthält als ROM Informationen, die nur gebraucht werden, um die Zeichenkonturen aufzubauen. Ansonsten belegt er lediglich Platz innerhalb des Rechners, der besser genutzt werden könnte. Es liegt deshalb nahe, den Zeichengenerator nur dann anzusprechen, wenn dieser benötigt wird, um ihn anschließend wieder auszublenden.

Innerhalb eines 16-K-Blocks kann der Zeichengenerator in Abständen von 2 Kilobyte abgelegt werden. Die Startadresse des Generators wird dazu codiert in den Bits 1-3 des Registers 24 geschrieben. Dabei sind folgende Codierungen möglich:

Bit 3 2 1	dez	zu addierender Offset
0 0 0	00	\$0000, #00000
0 0 1	02	\$0800, #02048
0 1 0	04	\$1000, #04096
0 1 1	06	\$1800, #06144
1 0 0	08	\$2000, #08192
1 0 1	10	\$2800, #10240
1 1 0	12	\$3000, #12288
1 1 1	14	\$3800, #14336

Soll der Zeichengenerator also mit der Adresse \$6000 beginnen, muß beachtet werden, daß diese Adresse im 16-K-Block 1 liegt. Die Differenz zwischen dem Blockbeginn und dem erwünschten Beginn des Zeichensatzes ist also \$2000. Demzufolge muß in das Register 24 der Wert 8 geschrieben werden, um dem Rechner mitzuteilen, daß dort der Zeichengenerator zu finden ist. Zusätzlich muß der Rechner wissen, wo der 16-K-Block liegt, in dem der neue Zeichensatz zu finden ist. In diesem Fall müßte also in das Register \$DD00 der dezimale Wert 2 eingetragen werden.

Das Video-RAM

Auch das Video-RAM ist im Speicher des Rechners frei verschiebbar, wobei hier ebenfalls darauf zu achten ist, daß jeweils nur ein 16-K-Block angesprochen werden kann. Im Gegensatz zum Zeichengenerator ist das Video-RAM sogar in Schritten von einem Kilobyte ablegbar, was in der Tatsache begründet ist, daß in der Stan-

dard-Betriebsart des C64 nur 1000 Zeichen = 1000 Byte auf einmal darstellbar sind. Der Beginn des Video-RAMs ist ebenso wie der Beginn des Zeichengenerators in codierter Form abzulegen. Die möglichen Anfangsadressen sind in der folgenden Tabelle dargestellt:

Bit 7 6 5 4	dez	zu addierender Offset
0 0 0 0	000	\$0000, #00000
0 0 0 1	016	\$0400, #01024
0 0 1 0	032	\$0800, #02048
0 0 1 1	048	\$0C00, #03072
0 1 0 0	064	\$1000, #04096
0 1 0 1	080	\$1400, #05120
0 1 1 0	096	\$1800, #06144
0 1 1 1	112	\$1C00, #07168
1 0 0 0	128	\$2000, #08192
1 0 0 1	144	\$2400, #09216
1 0 1 0	160	\$2800, #10240
1 0 1 1	176	\$2C00, #11264
1 1 0 0	192	\$3000, #12288
1 1 0 1	208	\$3400, #13312
1 1 1 0	224	\$3800, #14336
1 1 1 1	240	\$3C00, #15360

Wollen Sie zum Beispiel das Video-RAM auf die Adresse \$8400 (#33792) legen, so müssen Sie wie üblich zunächst feststellen, in welchem 16-K-Block diese Adresse liegt (in unserem Fall in Block 1). Zwischen dem Start des 16-K-Blocks und dem Video-RAM ergibt sich dann eine Differenz von \$0400. Deshalb muß der Wert 16 in das Register 24 geschrieben werden. Außerdem will der Rechner wissen, welcher der vier 16-K-Blöcke aktuell ist, was wie weiter oben beschrieben durchgeführt wird. Somit können Sie also die folgenden POKE-Befehle eingeben, um das Video-RAM auf die Adresse \$8400 zu bringen. Beachten Sie auch, daß der Zeichengenerator immer vorhanden ist, wenn er im Bereich von \$1000 bis \$1FFF oder von \$9000 bis \$9FFF lokalisiert ist. Das wird hier ausgenutzt.

10 POKE 56576, (PEEK (56576) AND 252) OR 1:REM schaltet Block 1 ein.

20 POKE 53248+24, (PEEK (53248+24) AND 15) OR 16:REM schaltet Video-RAM um.

30 POKE 53248+24, (PEEK (53248+24) AND 240) OR 4:REM Beginn des Zeichengenerators definieren.

40 POKE 648,132:REM Zeiger auf Video-RAM aktualisieren.

Durch die AND- und OR-Verknüpfungen werden wirklich nur die Bits geändert, die auch geändert werden sollen; alle anderen bleiben unverändert. Der letzte POKE-Befehl ist notwendig, damit Betriebssystemroutinen folgerichtig verarbeitet werden können. Der Wert des Zeigers, der mit dem POKE-Befehl in Zeile 40 übergeben wird (in diesem Fall also 132), berechnet sich aus: Zeiger = (dezimale Startadresse aus Video-RAMs, absolut)/256

Starten Sie das Programm, wenn Sie es abgetippt haben, und schauen Sie sich an, was dann passiert. Der Bildschirm füllt sich mit "wirren" Zeichen, die aber verschwinden, wenn Sie den Bildschirm löschen. Der Bildschirm beginnt aber tatsächlich an einer anderen Adresse. Das wird deutlich, wenn Sie versuchen (wie im ersten Teil unseres Kurses beschrieben), z.B. in der linken oberen Ecke einen weißen "*" durch POKE-Befehle zu erzeugen.

Damals hätten Sie POKE 1024,42:POKE 55296,1 eingegeben. Jetzt müssen Sie folgendes schreiben:

POKE 33792,42:POKE 55296,1

Wollen Sie wieder zur Standardkonfiguration zurückkehren, müssen Sie das Listing wie folgt abändern: 10 POKE 56576, (PEEK (56576) AND 252) OR 3:REM schaltet Block 3 ein

40 POKE 648,4

Danach beginnt das Video-RAM wieder bei \$0400, und der Zeichengenerator stellt alle Zeichen in Großschrift oder in Grafikzeichen dar.

Beim nächsten Mal werden wir die hier dargestellten Grundlagen für die Entwicklung einer Routine verwenden, mit der es möglich wird, komfortabel zwischen zwei unabhängigen Bildschirmen hin- und herschalten. Bis dahin: Spielen Sie ruhig ein wenig mit den Registern, denn "Probieren geht über Studieren!"

F. Lenczewski

LOGO für Einsteiger

Von Dr. Dietrich Senftleben
Vogel-Verlag
210 Seiten, 30.- DM
ISBN 3-8023-0802-6

Dieses Buch ist für Leute gedacht, die als erste Programmiersprache nicht Basic, sondern LOGO gewählt haben.

Der LOGO-Kurs beginnt mit dem "kleinen LOGO-Einmaleins", das anhand der recht einfach zu bedienenden Turtlegrafik dieser Sprache erläutert wird. In diesen Lektionen lernt man den LOGO-Zeileneditor und das Prinzip der Programmspeicherung sowie wesentliche Grundzüge der Sprache (Primitives, selbstdefinierte Befehle, Variablen und einfache Rekursionen) kennen. Am Ende jeder Lektion findet man kleine Aufgaben, die zum Experimentieren anregen sollen, und Kurzbeschreibungen, die den Inhalt der Lektionen für Fortgeschrittene kurz zusammenfassen. Dieser Abschnitt

des Buches ist eindeutig für Anfänger konzipiert. Ein Großteil der Seiten wird von Bildschirmfotos und Abbildungen der Tasten, die man zu drücken hat, in Anspruch genommen.

Die wesentlichen Bestandteile von LOGO (Worte und Listen, Schleifen, Drucken und die zentrale Vokabel "Output") werden nur anhand von Beispielen fast ohne zusätzliche Erläuterungen eingeführt, so daß Einsteiger auf jeden Fall Probleme bekommen dürften. Typische Phrase: "Wir sehen uns - ohne viel Erklärungen - das nachfolgende Programm an." Positiv zu bewerten ist dagegen wiederum der Anhang des Buches. Er liefert eine für Fortgeschrittene verwendbare Kurzgrammatik, Übersichten über Vokabeln, typische Anfängerfehler sowie nützliche Programmbausteine, die das Zeichnen von Kreisen und Bögen und die Musikprogrammierung in LOGO erlauben.

Axel Klenk

Das RP-System von Brillant Software

Ein neuer Trend in der Softwarebranche scheint der Vertrieb von durchaus brauchbaren Programmen mit einer unverständlichen Anleitung zu sein. Ein Beispiel dafür ist das RP-System, der "schnelle Weg zum selbstgeschriebenen Superprogramm", das im Brillant Software-Verlag erschienen ist.

Zum Lieferumfang gehören eine beidseitig gut gefüllte Diskette und ein fast beängstigend dickes Begleitbuch, dessen Umfang gut 250-300 Seiten beträgt (genauere Zahlen sind unmöglich, da man sich die Nummerierung der Seiten größtenteils gespart hat). Hat man sich durch zwei Vorworte, eine Einführung und mehrere Loblieder auf das System und die Sprites gelesen, stellt man fest, daß über drei Viertel des Buches nur Hardcopies der mitgelieferten Screens und Sprites enthalten. Das ist zwar recht nett, da man sich das Ausprobieren von 19 Screens und 1024 (!) Sprites spart und gleichzeitig noch eine Erläuterung bekommt. Aber leider geht das Ganze sehr auf Kosten des Inhalts.

Der erste Teil des RP-Systems ist der Laufschrift-Editor. Mit ihm können Texte erstellt werden, die später mit wählbarer Geschwindigkeit über den Bildschirm wandern. Das Ergebnis des Editiervorgangs ist ein Textfile, weshalb nicht ganz verständlich ist, wieso man einen Editor benötigt und dem Programm nicht einfach einen oder mehrere Strings übergibt.

Beim Screen-Editor handelt es sich zweifellos um den besten Teil des RP-Systems. Dies ist ein Grafik-Programm mit einigen Schwächen, aber auch mit zahlreichen Stärken, das zum Entwurf von Spielfeldern dienen soll. Der Screen-Editor arbeitet ausschließlich im Multicolormodus, wobei die Farben einfach über die Funktionstasten gewählt werden können. Er verschafft die Möglichkeit, Punkte zu setzen, Linien zu ziehen, Flächen zu füllen, Bildausschnitte zu vergrößern oder zu kopieren und Sprites in die Bitmap miteinzubeziehen. Da sich

jedoch ohne weitere Funktionen das Selbstzeichnen als langwierig herausstellen dürfte, hat man 19 Screens auf die Diskette gebannt, die meist durch Herauskopieren von Ausschnitten oder durch ein Auf-den-Kopf-Stellen mehrfach verwendet werden können. Beim Laden und Speichern von Screens kennt das RP-System ein eigenes und zusätzlich noch das Format des Multi-Color-Grafik-Tools (zufällig ebenfalls von Brillant Software) sowie das des Koala-Painters.



Hat man einige Screens auf die immerhin drei vorhandenen Bitmaps (deren Handhabung anfangs verwirrend erscheint, sich später jedoch noch als halbwegs praktisch erweist) verteilt, so kann man fast beliebig manipulieren. Die Darstellung zweier Bitmaps gleichzeitig, ein Splitscreen oder das Scrollen der Bilder sind kein Problem mehr. Zweifelloso positiv am Screen-Editor ist auch, daß nicht nur die Farben, sondern auch die für die Commodore-Grafik wichtigen Bitmuster bei allen Befehlen bedacht wurden. Fragt sich nur, wer bei der knappen Erläuterung mit diesem Feature etwas anfangen kann.

Zwiespältiger präsentiert sich hingegen der Sprite-Editor des RP-Systems. Er enthält einiges an Standards, wie Directory, Editieren in Normal- oder Multicolormodus, Spiegeln und Kopfstellen der Sprites, aber auch ein punktweises Verschieben und die Möglichkeit, zwei Sprites gleichzeitig darzustel-

len. Der Hit des Sprite-Editors ist zweifellos der Animations-test, mit dem zeichentrickähnliche Sequenzen von bis zu 8 Sprites ausgetestet werden können. Ein Nachteil ist, daß keine einzelnen Sprites, sondern nur Spritesets geladen und gespeichert werden können. Außerdem bauen Programm und Verfasser der Anleitung mit den Begriffen "Spritemuster", "Spriteblock", "Sprite-nummer" und "Nummer des Sprite-musters = Speicherzelle" eine dermaßen gekonnte Verwirrung auf, daß die Orientierung in den ohnehin recht umfangreichen Spritesets größtenteils Glücksache ist.

Recht deutlich mißlungen ist mir der Versuch, mit dem Sound- und Melodie-Editor des RP-Systems etwas anzufangen. Noch recht erträglich ist der Sound-Editor, mit dem auf einfache Art und unter Verzicht auf die Mischung von Wellenform, Ringmodulation und Synchronisation, dafür aber mit vollem Gebrauch der Filter, eine Menge aus dem SID herausgeholt werden kann. Enttäuschend dagegen der Melodie-Editor. Abgesehen davon, daß das Notensystem vertikal verläuft (woran man sich gewöhnen kann), läßt sich nur eine Stimme auf einmal programmieren. Zusätzlich teilt die Anleitung stolz mit, daß das Programm selbstverständlich in der Lage ist, punktierte Noten/Pausen zu verarbeiten, Tonarten und Taktstriche zu setzen und zwischen Baß- und Violinschlüssel zu unterscheiden. Sie gibt aber nicht an, wie man diese wichtigen und in der Software zweifellos auch vorhandenen Funktionen anwenden kann. Um es kurz zusammenzufassen: Die Programmierung von "Alle meine Entchen" gestaltete sich sehr kompliziert bis fast unmöglich. An schwierigere Werke, wie z.B. in der Anleitung vorgeschlagene aktuelle Schlager, möchte ich mit diesem Editor, oder besser gesagt, mit dieser Anleitung nicht herangehen.

Als letztes bleiben noch die RP-Systembefehle. Dabei handelt es sich um Maschinenrouti-

nen, die über SYS und entsprechende Mengen von Parametern aufgerufen werden. Mit dem freundlicherweise beigelegten Programmkopf läßt sich das Ganze auf einige Zuweisungen und einen Unterprogrammaufruf verkürzen. Diese Befehle sind im großen und ganzen in drei Gruppen zu unterteilen: 1. Befehle zum Aus- bzw. Einschalten der RP-Funktionen (Verbiegen des Interrupt-Vektors), 2. Befehle, die man eigentlich von Monitorprogrammen her kennt, wie z.B. Speichern und Laden, Verschieben und Füllen von Speicherbereichen und ein Peek unter das Basic-ROM. Die dritte Gruppe bilden die für das eigentliche Spielprogramm wichtigen Befehle. Ein einziger organisiert z.B. die gesamte Speicherverwaltung des VIC in Block 3. Andere schreiben Texte auf den HiRes-Screen, setzen oder steuern Sprites oder Laufschrift, lesen Joystickwerte oder synchronisieren gar Spritebewegung und Sound, so daß z.B. ein "Tapp" nur dann ertönt, wenn das Pferd tatsächlich einen "Huf" auf den "Boden" setzt. Soweit das mit der mangelhaften Kenntnis von Sprites und Sound/Melodien zu überprüfen war, arbeiten diese Befehle alle einwandfrei. Überhaupt hat man den Eindruck, daß das RP-System selbst hervorragend funktioniert, während einem die Anleitung einiges von seinem Können verheimlichen will. Wenn sie nicht bald überarbeitet wird, verfehlt das RP-System seinen Zweck, denn wer diese Anleitung verstehen und tagelang basteln, testen und rätseln will, kauft sich keinen Speiegenerator, sondern schreibt seine eigenen Programme.

System: C 64
Hersteller: Brillant Software
Axel Klenk

Centr. GLP II centr/ser. 598.-
STAR NL 10 auch f. Comm. 848.-
CITIZEN 120D centr/ser. 749.-
SEIKOSHA SP1000 A/AS/VC 749.-
unicorn soft
3500 Kassel
u. Königstr. 46A
T. 0561/770367

Mailbox Marke Eigenbau

In letzter Zeit explodiert die Datenfernübertragung (DFÜ) förmlich. Täglich kommen neue Mailboxen ("elektronische Briefkästen") und ganze Scharen von neuen Benutzern dieser Art der Kommunikation dazu. Dieser Bericht soll den neuen Benutzern die Angst vor den vielfältigen Problemen mit der Datenfernübertragung nehmen. Als Sysop (System-Operator = Betreiber der Mailbox) einer Mailbox im Frankfurter Raum habe ich in der Zwischenzeit sowohl die Probleme der Benutzer als auch die Schwierigkeiten der Betreiber kennengelernt. Und schließlich habe ich auch noch meine Probleme in Erinnerung, die ich am Beginn der DFÜ-Tätigkeiten hatte.

Als Neuankömmling in der DFÜ sollte man sich an einige wenige Regeln halten, damit das neue Hobby richtigen Spaß macht. Auch dem Sysop der angerufenen Mailbox wird der Benutzer einen Gefallen tun, wenn er sich richtig verhält (z.B. richtiges Verlassen der Mailbox etc.). Nachdem der Rechner eingeschaltet und das Terminalprogramm geladen ist, muß man die Parameter einstellen. Sie sagen dem Rechner, wie er die ankommenden Pfeiftöne der Mailbox in lesbare Texte umwandelt und auf den Schirm bringt. Die Parameter einer Mailbox werden zumeist in einer dreistelligen Buchstaben-Zahlen-Kombination angegeben. Hinter der Telefonnummer kann man z.B. "7/N/1" oder "8/N/2" finden. Die erste Ziffer gibt die Anzahl der Datenbits an. Sie ist im Normalfall immer 7, bei manchen Mailboxen allerdings 8. Wenn's mal mit der Übertragung nicht klappt, so sollte man immer zuerst diese Ziffer ändern (von 7 auf 8 oder umgekehrt). Der nun folgende Buchstabe gibt die Art der Paritätsprüfung an. In deutschen Mailboxen gibt es diese Paritätsprüfung nicht. Deshalb wird hier auf "N" wie "None" (also keine Prüfung) eingestellt. Die letzte Ziffer gibt die Anzahl der Stoppbits an, die das Ende eines jeden übertrage-

nen Zeichens angeben. Sie ist in den meisten Fällen 1.

Sind diese Parameter eingestellt (wenn nichts bekannt ist, immer die normale Kombination "7/N/1" nehmen), wird nochmals überprüft, ob alle Geräte (Akustikkoppler, Rechner etc.) richtig angeschlossen sind und funktionieren. Wenn das der Fall ist, kann endlich die Nummer einer Mailbox gewählt werden. Ist die Verbindung hergestellt (aus dem Telefon pfeift es!), legt man schnell den Telefonhörer in den Akustikkoppler oder schaltet das Modem ein. Viele Mailboxen warten jetzt auf ein Erkennungszeichen, damit sie "wissen", daß die andere Stelle zur Datenübertragung bereit ist. Man drückt deshalb ein paar Mal auf die RETURN-Taste des eigenen Rechners, bis die Mailbox die Begrüßungs-Meldung ausgibt. Mit ihr hat man schon einmal die größte Hürde übersprungen.

Die normale Übertragung läuft übrigens mit 300 Baud ab. Das bedeutet eine Geschwindigkeit von 300 Bits in der Sekunde. Dem DFÜ-Neuling wird diese Geschwindigkeit allerdings am Anfang noch zu schnell sein. Die Ausgabe der Texte kann deshalb durch Druck auf die Tasten CTRL/S angehalten und mit CTRL/Q wieder gestartet werden. Mit der Zeit gewöhnt man sich aber an die 300 Baud und wird nach kürzester Zeit auch sämtliche Bücher nur noch mit 300 Baud lesen!

Nach der Begrüßungsmeldung, die übrigens einige wichtige Hinweise enthalten kann, wird man in den meisten Fällen nach dem Namen gefragt. Da wir aber als "Erst-Anrufer" noch nicht in die Mailbox eingetragen sind, geben wir einfach das Wort "gast" ein. Es teilt dem Rechner mit, daß man das erste Mal in der betreffenden Mailbox ist und sich nur mal umsehen möchte. Über einige andere Punkte gelangt man dann schließlich in das Hauptmenü der Mailbox. Dies enthält z.B. folgende Punkte:

1. Infobox
2. Pinboard
3. Userbox
4. 64'er-Box
5. Andere Rechner
6. Newsbox
7. Softbox
8. Nummernbox
9. Unterhaltung (Witze)

Jetzt kann zwischen diesen Punkten gewählt werden, bis man schließlich an den gewünschten Text 'rankommt. Eine so aufgebaute Mailbox funktioniert also ähnlich wie BTX, indem man sich von einem Menü zum anderen immer weiter durcharbeitet, um an den gewünschten Text heranzukommen. Sollten Sie sich nicht mehr zurechtfinden, so tippen Sie "Help" oder "Hilfe" ein. Fast sämtliche Mailboxen unterstützen den Benutzer dann mit Tips zum weiteren Gebrauch. Außerdem gibt es die Möglichkeit, mit dem Sysop direkt zu "sprechen". Dazu tippt man "com" oder "kom" ein, worauf fast alle Systeme den Sysop an die Tasten rufen. Der Sysop wird dann gerne weiterhelfen, wenn er gerade anwesend sein sollte.

Eine Mailbox lebt von Eingaben, also von Texten und Anregungen von der Seite der Anrufer. Das Interessanteste und Aktuelle in einer Mailbox steht im sog. "Pinboard". Es ist eine Art Schwarzes Brett mit Verkaufs- und Such-Anzeigen, die der Benutzer schreiben und lesen kann. Es gibt aber auch die Möglichkeit, anderen Benutzern Briefe zu schreiben (Brief = Mail). Damit die aber nicht jeder lesen kann, sondern nur der eigentliche Empfänger, muß diese Information irgendwie geschützt werden. Die meisten Systeme lassen deshalb eine "Eintragung als Benutzer" zu. Um eingetragener Benutzer zu werden, muß man dem Sysop der Mailbox lediglich eine entsprechende Nachricht zukommen lassen, in der neben dem Namen auch ein Paßwort übermittelt wird, das natürlich nicht in allen Mailboxen gleich sein sollte. Die Mailbox fragt dann ganz zu Beginn nach dem Namen und dem ausgewählten Paßwort. Als eingetragener Be-

nutzer sollte man also statt dem Namen "gast" den richtigen Namen eingeben, worauf das System nach dem Paßwort fragen wird. Nach dem erfolgreichen "Login" (das bedeutet, daß Sie mit Namen und Paßwort identifiziert wurden) sucht die Mailbox nach Briefen anderer Benutzer an Sie.

Jetzt noch einige kleine Tips, die dem Sysop und dem Benutzer das Leben erleichtern: Eine Mailbox lebt von Eingaben. Jeder Sysop freut sich über Verbesserungsvorschläge von Seiten der Benutzer. Schließlich kann er durch sie die Box attraktiver machen, und er weiß, daß seine Benutzer auch an der Gestaltung teilnehmen und nicht nur rein passiv konsumieren. Oftmals kommen auch Mails an, die einen beleidigenden Inhalt haben. Als Sysop muß man sich das ja alles durchlesen. Derartige Texte werden aber bei keiner Mailbox übernommen! Außerdem erzeugen solche Mails auch noch Frust bei den Sysops. Vor allen Dingen, wenn eine Mail kommt wie: "Alles Mist", "Blöde Mailbox" etc. Da ist's doch viel besser, den Grund der Wut zu schreiben, als einfach nur zu meckern.

Wichtig für die Mailbox ist auch, daß der Benutzer sie ordnungsgemäß verläßt. Dies geschieht bei den meisten Mailboxen durch Eingabe von "logoff" oder "ende", ist aber von Mailbox zu Mailbox verschieden. Wenn nämlich die Box durch einfaches Auflegen des Benutzers verlassen wird, dann ist sie für nachfolgende Benutzer ca. 1-2 Minuten lang gesperrt.

Übrigens, wer meine Mailbox kennenlernen möchte, der kann gerne den ganzen Tag über anrufen: Mailbox CVCM, Tel. 06192/7960, 24h online (online = geöffnet), Parameter: 7/N/1.

Lars Schmidt

**Computer-Kontakt
hat preisgünstige
Kleinanzeigen**

Das Versatile-Interface-System für den C 64

"Nicht schon wieder ein Ein-/Ausgabe-Interface im Test" wird der eine oder der andere Leser stöhnen. Doch es gibt kleine, aber feine Unterschiede zwischen den einzelnen Produkten. Deshalb aufgepaßt!

Das Testgerät besteht aus dem Interface, das in den Userport des C 64 eingesteckt wird, sowie den Ein-/Ausgabe-Platinen. Das sind zwei Digital-Ausgangsmodule mit je 4 Relaisausgängen, ein Digital-Eingangsmodule mit acht Eingängen und ein Analog-Eingangsmodule mit vier regelbaren Analog-Eingängen. Der erste Unterschied zu Konkurrenzprodukten fällt sofort nach dem Auspacken auf: Das Gerät ist vorbildlich verarbeitet, sicher verschraubt, und die einzelnen Module sind durch Plexiglasscheiben geschützt, was auch einen positiven optischen Effekt hat. Die Flachbandkabel vom Interface zu den Modulen sind steckbar und auch ausreichend lang, so daß der Modul-Turm auch weiter entfernt vom Computer stehen kann. Die Platinen selbst sind sehr sauber verarbeitet und gut aufgebaut. Im Test gab es keine Probleme damit.

Die Anschlüsse der Platinen, an die man ja später die "Fühler" bzw. "Verbraucher" anschließt, wurden nicht einfach in Form von Lötflächen konzipiert, sondern sind durch Schraubklemmen herausgeführt. So lassen sich nicht nur die für den Laien schwierige Lötarbeiten einsparen, sondern auch schon einmal fixierte Drähte leicht wieder umklemmen.

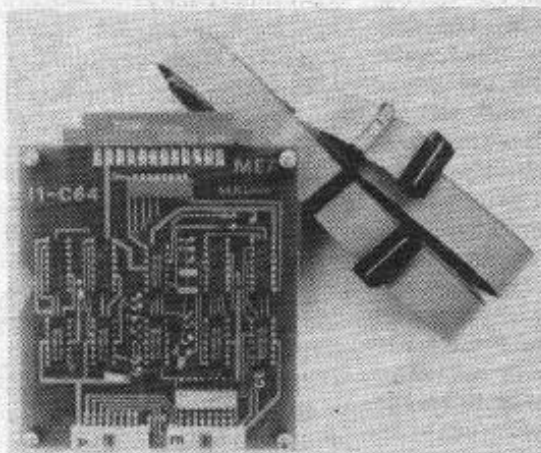
Dem Interface beigelegt ist eine knappe Erläuterung der Funktionsweise der einzelnen Module sowie die Erklärung der Anschlüsse. Für den Laien ist diese Anleitung leider erst nach mehrmaligem Durchlesen verständlich. Aber mit ein klein wenig technischem Verständnis nimmt man die Hürde leicht!

Neben den oben aufgeführten Modulen gibt es noch ein Digital-Ausgangsmodule mit

acht Ausgängen, einen Analog-Ausgang sowie eine Erweiterungs-Box, Geräte, die allerdings dem Testgerät nicht beigelegt waren. Zum Betrieb des VIS ist jetzt lediglich noch das 24V-Netzteil erforderlich. Je nach Bedarf kann man sich auf diese Weise sein eigenes Interface zusammenstellen.

Mit dem Digital-Ausgangsmodule kann man die vier bereits auf dem Modul vorhandenen Relais vom Computer aus ansteuern. Sie können bis zu 250 Volt schalten (50 bis 250 Watt), womit normale 220-Volt-Geräte kein Problem darstellen. Der Schaltzustand der Relais wird über LEDs angezeigt – ein Vorteil, den nur wenige Geräte bieten. Durch einen einfachen POKE-Befehl können die Relais einzeln angesteuert werden, wobei die Adresse durch einen sog. "Jumper" (steckbare Lötbrücke) gewählt werden kann. Auch hier zeigt sich wieder die Durchdachtheit des Systems: Es müssen lediglich die Jumper umgesteckt und nicht etwa umständlicherweise Lötbrücken geöffnet bzw. geschlossen werden. Zusätzlich kann durch Umstecken eines Jumpers eine "Zyklus-Überwachung" aktiviert werden. Falls der Computer mal abstürzt oder sich das Programm aufhängt, wird das Relais nach einer gewissen Zeit abgeschaltet.

Über das Digital-Eingangsmodule können maximal acht digitale Eingänge durch einen einfachen PEEK-Befehl abgefragt und ausgewertet werden. An das Interface wird dazu eine Spannung angelegt (z.B. 24 Volt). Sie dient als Festlegung der Schaltschwelle, die bei etwa 50 % der angelegten Spannung liegt. Ist die an einen Eingang angelegte Spannung also kleiner als 12 Volt, so steht in der Speicherstelle eine "0" und bei über 12 Volt eine "1". So werden die analogen Spannungen in digitale Impulse umgewandelt. Eine Anwendungsmöglichkeit wäre z.B. der Anschluß eines Fototransistors. Erreicht



Lötseite der Interface-Platine

die Spannung einen gewissen Punkt (in diesem Falle abhängig vom Lichteinfall), so wird dieser Zustand dem Computer gemeldet. Über ein Digital-Ausgabemodule kann dann z.B. eine Lampe betätigt werden. Auch bei diesem Modul wird der Schaltzustand wieder durch LEDs angezeigt.

Die an das Analog-Eingangsmodule angelegte Spannung wird in ein 8-Bit-Wort, also eine Zahl zwischen 0 und 255, umgewandelt. Es gibt vier solcher Eingänge auf dem Modul, die zudem noch einzeln regelbar sind, so daß der Meßbereich zwischen einem Volt (ein Schritt = 4 mV) und zehn Volt (ein Schritt = 40 mV) festgelegt werden kann. Eine praktische Anwendung wäre beispielsweise der Bau eines Thermometers. Über den Fühler würde die variable Spannung dem Interfa-

ce zugeführt und so dem Computer weitergegeben. Das Auslesen der Werte erfolgt wieder mittels eines PEEK-Befehls.

Das VIS bietet eine Unmenge an Anwendungs-Möglichkeiten und Komfort. Bei heutigen Interfaces ist die solide Bauweise noch immer keinesfalls die Regel. Trotz seines relativ hohen Preises ist das VIS sein Geld wert. Das gut durchdachte Konzept des VIS, das fast keine Anwendungsmöglichkeit offen läßt, sollte einen Interessenten von den Vorteilen dieses Qualitäts-Gerätes überzeugen.

System: C 64
Preis: 648,- DM

Bezugsquelle:
Ing.-Büro Manfred Kühn
Friedrich-Ebert-Allee 61
2000 Schenefeld

Lars Schmidt

C 64 und PC 128

S/S Modul Hardcopy + Turbo Tape + Turbo Disk + Renew/Old	30,00 DM	Super-Hardcopy-Modul Druckt fast jedes Computer-Bild	45,00 DM
Supersoftwarepaket I 10 Programme auf Disk, oder Kassette. Anwendungs- und Spielprogramme	18,95 DM	S/6 Modul Turbo Disk + Hardcopy + Formattieren + ID Changer + Renew/Old	39,00 DM
10 Disketten S/5/DD Superpreise	14,50 DM	S/1 Modul Turbo Tape	19,90 DM
S/4 Modul Turbo Tape + Turbo Disk + DOS S.1 + Renew/Old	23,00 DM	S/2 Modul Turbo Disk + S/1 und S/2 ohne Gehäuse	19,90 DM
Modul 2000 Turbo Tape + Turbo Disk + Monitor 8000 + ID Changer siehe Test OK Nr. 11/86	45,00 DM	Profi Betriebssystem 7x schnelleres Laden, Old und DOS Funktionen, Funktionstabellen, im Gehäuse	49,00 DM
Modul 3000 Monitor + Turbo Disk + Turbo Tape + Hardcopy + Filecopy + ID Changer + Backup + Renew/Old	65,00 DM	Betriebssystem-umschaltplatte 128 Damit ist es möglich, im 64er Modus auf dem 128 weitere Betriebssysteme zu betreiben. Platine inkl. Schalter, weitere Platinen auf Anfrage	39,00 DM
Modus S-128 Beschleunigt den Ladevorgang der 1570-71 um ca. 5-7 mal im 64er Modus	49,00 DM	Leerplatte	9,95 DM

Alle Module im Gehäuse, mit Roset (Aufpreis von 5,00 DM). Die Programme belegen keinen Speicherplatz. 24 Stunden Service, Info gegen 1,00 DM. Weiteres Zubehör und Commodore ICs auf Lager, z.B. 6526

Klemmer & Schulte Electronic

Berrenratherstr. 496 • 5030 Hürth-Effern
Ladenlokal Mo.-Fr. 13.00-18.00 Uhr
Sa. 10.00-13.00 Uhr, ☎ 022 33/9 37 36

Der Star Painter – Neu von Sybex

Der Star Painter ist kein gewöhnliches Grafikprogramm wie z.B. Koala Painter oder Blazing Paddles, die den Multicolor-Modus mit einer Auflösung von 320*200 Punkten benutzen. Der bedeutsamste Unterschied zu diesen Programmen ist die Größe der zusammenhängenden Grafik, die 640*344 Punkte (= 220160 Pixels) in einfarbiger Darstellung umfaßt – in der Hardcopy, versteht sich! Dazu kommt noch ein Hintergrundspeicher, der die Aufgabe hat, beim Erstellen einer Zeichnung mehrmals benötigte Grafikelemente beizubehalten. Er umfaßt zusätzlich noch einmal eine Speicherkapazität von 256*168 Punkten. Hinzu kommen verschiedene Editiermöglichkeiten, eine praxiserleichternde Texteingabe und umfangreiche Möglichkeiten zur Ein/Ausgabe.

Zunächst einmal kann Star Painter damit all das, was gewöhnliche Zeichenprogramme auch können, also Rechtecke, Linien, Strahlen und Kreise zeichnen. Es besteht zusätzlich die Möglichkeit, Kreise so zu "verziehen" bzw. zu "quetschen", daß eine Ellipse entsteht. Das ist besonders praktisch, wenn man zum Beispiel einen Baum zeichnen will. Besonders interessant finde ich die Möglichkeit, Text in die HiRes-Grafik einzufügen. Hier stehen zwei Zeichensätze zur Auswahl, von denen der zweite auch selbst definiert werden kann. Das heißt, man kann auch auf einen bereits definierten Zeichensatz zurückgreifen (zum Beispiel Griechisch, Russisch oder auch einfach nur eine

andere Schriftart), der bereits auf Diskette gespeichert ist. Man kann die Zeichen auch auf den Kopf stellen und in alle möglichen Richtungen drehen. Und das alles mit allen möglichen Sonderzeichen, deutschen Umlauten und in Proportional-schrift. Der Rechner simuliert übrigens eine DIN-Tastatur. Y und Z sind also im Gegensatz zur ASCII-Tastatur vertauscht. Ebenso befinden sich die Umlaute auf den von der Schreibmaschine her gewohnten Stellen.

Star Painter ist das einzige Programm, bei dem am oberen und linken Rand Lineale mitlaufen, die das Formatieren der Zeichnung erleichtern. Meines Wissens nach ebenso einzigartig ist die Möglichkeit, die Zeichengeschwindigkeit zu variieren. Da mit einem C 64 ja nicht so ohne weiteres 220160 Bildschirmpunkte auf einen Schlag dargestellt werden können, ist man zwangsläufig darauf angewiesen, mehrere HiRes-Seiten zu benutzen. Das geht beim Star Painter recht problemlos vor sich. Mit den Tasten 1-9 wird das Fenster angewählt, das man bearbeiten will, und 0 bezeichnet den Hintergrundspeicher. Es ist natürlich auch möglich, zwischen mehreren Seiten hin und her zu scrollen. Im Hintergrundspeicher kann man kleinere Grafiken zur späteren Verwendung ablegen. Einzelne Bilder größer als 24*21 Punkte sind jedoch nicht sinnvoll, da sie nicht in die Grafik zurückkopiert werden können.

Mit dem Get-Befehl, der nicht nur im Hintergrundspeicher Anwendung findet, kann man ein solches 24*21 Punkte großes Grafik-Element ausschneiden und mit Put bzw. Append an anderer Stelle in die eigentliche Grafik einfügen. Mit Append wird das Grafik-Element in die Grafik kopiert, ohne die darunter liegenden Punkte zu löschen. Mit Put werden die nicht gesetzten Grafik-Punkte mitkopiert, d.h. die darunter liegende Grafik wird gelöscht. Im normalen Grafikspeicher können größere Ausschnitte auch mit Move kopiert

werden. Man zieht wie in der Box-Funktion ein Rechteck auf, kann dieses Rechteck nun in der gesamten Grafik hin- und herbewegen und an gewünschter Stelle wieder mit Put bzw. Append in die vorhandene Grafik einfügen. Im Append-Modus kann ein kopierter Ausschnitt auch als Pinsel zum Zeichnen benutzt werden, indem er bei gedrücktem Joystick-Button über die Grafik bewegt wird. So lassen sich interessante Effekte erzielen. Da der Star Painter auch die Möglichkeit bietet, den im Puffer befindlichen Ausschnitt zu bearbeiten, kann man sich auf einfache Weise einen Zeichenstift für besondere Linien erstellen. Erwähnenswert ist auch die Fill-Funktion. Mit ihr lassen sich Flächen sogar mit selbst definierten Füllmustern ausfüllen.

Der Star Painter kann auch zoomen. Dazu begibt man sich in einen speziellen Zoom-Editor, der in seinen Funktionen dem Ausschnitt-Editor entspricht. Bei der Bearbeitung kann man die Grafik sogar rotieren, invertieren und spiegeln. Wer sich dann das komplette Bild betrachten will, kann mit "Whole" die ganze Grafik von 640*344 Punkten stark verkleinert auf den Bildschirm bringen.

Einzige Voraussetzung für die Übernahme von Grafiken aus anderen Programmen ist eine Auflösung von 320*200 Punkten. Natürlich lassen sich auch die mit dem Star Painter erstellten Grafiken laden und wie schon angesprochen, kann man auch selbst definierte Zeichensätze, Maßstäbe und Hintergrundspeicher abspeichern.

Die vielleicht wichtigste Funktion von Star Painter ist die Ausgabe der Grafik auf einem Drucker. Den meisten Programmen fehlt diese überaus wichtige Option völlig. Beim Star Painter hingegen wurde sie besonders vielseitig gestaltet; man kann damit jeden Drucker ansteuern. Schon fertig angepaßt sind der MPS 801, der MPS 802, der Melchers CP-BOX, Epson-Drucker mit diversen Interfaces sowie kompa-

tible Geräte. Sollte Ihr Drucker in dieser Auswahl nicht enthalten sein, so können Sie Star Painter ohne viel Aufwand auf Ihren Drucker umstellen. Das macht man am besten beim "Installieren" des Programms. Diese Einstellung ist im Prinzip nur ein einziges Mal notwendig, denn Star Painter merkt sich die Installation und lädt sich dann selbst mit der entsprechend eingestellten Modifikation.

Wie mächtig dieses Zeichenprogramm wirklich ist, begreift man erst, wenn man sich über einen längeren Zeitraum hinweg wirklich ernsthaft mit ihm auseinandergesetzt hat. Mich fesselte dieses Programm, bei dem wirklich alles stimmt, tagelang an den Computer. Ein besonderes Lob gilt hier auch dem 166seitigen Lehrbuch, das im Lieferumfang enthalten ist. Nach dessen Lektüre kann auch jemand, der zuvor noch nie vor einem Computer saß, problemlos mit dem Star Painter arbeiten. Aber auch dem Profi in Sachen Computer bietet das Trainingsbuch mit Beispielprogrammen eine Hilfestellung.

Erwähnenswert ist noch, daß im Lieferumfang außer vielen Hilfsprogrammen, Hintergrundgrafiken, Maßstäben und ähnlichem auch ein Kopierprogramm enthalten ist, um von Star Painter eine Sicherheitskopie für den Eigenbedarf zu ziehen. Diese Tendenz bei den Anbietern von Software finde ich dem Käufer gegenüber nur reell. Diverse Firmen, wie zum Beispiel Data Becker, klügeln dagegen immer neue Kopierschutzvarianten aus, die dann bei einem Kopierversuch entweder das Programm unbrauchbar machen oder den Schreib/Lese-Kopf des Laufwerks in einer Endlosschleife einen "Bump" (Anschlagen des Schreib/Lese-Kopfes) ausführen lassen, was zur Dejustation des Laufwerks führt.

System: C64
Hersteller: Sybex Verlag
Preis: 79,- DM (Diskette)

Stefan Selbach

Commodore 64



Time Loader

Das Programm "Time Loader" läuft auf einem Commodore 64 mit Floppy. Mit ihm kann man sich andere Programme zu einer beliebigen Zeit nachladen lassen. Der "Time Loader" wird über zwei Menüpunkte gesteuert. Wählt man im Menü die "1", so kann man die momentane Uhrzeit einstellen. Sie wird wie folgt eingegeben: z.B. für 17.31 Uhr und 45 Sekunden die Ziffernfolge 173145.

Im zweiten Menüpunkt stellt man die Uhrzeit ein, bei der ein Programm nachgeladen werden soll. Nach Drücken der RETURN-Taste muß man

noch den Filenamen des zu ladenden Programms eingeben. Dabei können auch Wildcards benutzt werden. Nach Drücken der Space-Taste wird das Hauptprogramm gestartet, das beim Erreichen der im Menüpunkt 2 eingestellten Uhrzeit das Programm lädt und startet.

Soll eine Datasette statt einer Floppy benutzt werden, so muß man in Zeile 530 die "8" in eine "1" umschreiben. Bei Programmen, die mit einem SYS gestartet werden, muß in Zeile 540 das RUN durch den SYS-Befehl ersetzt werden. Dabei sollte man nicht vergessen, die CLR-Taste hinter dem SYS-Befehl zu drücken.

Slawek Gödickemeier

```

10 POKE53281,0:POKE53280,0:PRINT"
11 PRINT"
20 PRINT"
30 PRINT"
40 PRINT"
50 PRINT"
52 GOSUB10000
55 PRINT"
60 PRINT"
65 PRINT"
70 PRINT"
80 PRINT"
90 PRINT"
91 PRINT"
100 PRINT"
105 IFV=1THENPRINT"PROGRAMM
ARBEITET 1":GOTO500
110 GETA$:IFA$=""THEN11
120 IFA$="1"THEN150
130 IFA$="2"THEN200
140 GOTO110
150 PRINT"
160 PRINT"GEBEN SIE BITTE DIE UHRZEIT E
IN"
170 PRINT"UND DRUECKEN SIE RETURN."
180 OPEN1,0:PRINT"UHRZEIT (Z.B. 182056
)= ";;INPUT#1,TI$:CLOSE1:GOTO11
200 PRINT"GEBEN SIE BITTE DIE UHRZEIT
EIN, BEI DER"
210 PRINT"DAS PROGRAMM GELADEN WERDEN SO
LL."

```

Frame-Flash

Dieses Programm erzeugt auf dem Rahmen eine schnelle Folge von verschiedenen Farbtönen. Es ist in Maschinensprache geschrieben und wird mit

SYS 828 gestartet. Beendet wird es durch den Druck auf die Space-Taste oder durch RUN/STOP/RESTORE. Der Farbeffekt kann durch POKE 845,X verändert werden.

Norbert Ivens

Basic-Lader

```

100 FORI=828TO851:READA:POKEI,A:S=S+A
110 NEXTI:IFS<>3126THENPRINT"DATA-ERROR
!!!":STOP
140 DATA162,0,160,0,232,142,32,208,173,1
,220,201,239,240,8,200,192,30,208,241
150 DATA76,62,3,96
160 PRINT"DRUECKEN SIE A SPACE "
170 SYS828

```

Assembler-Listing

```

,033C A2 00 LDX #00
,033E A0 00 LDY #00
,0340 E8 INX
,0341 8E 20 D0 STX D020
,0344 AD 01 DC LDA DC01
,0347 C9 EF CMP #EF
,0349 F0 08 BEQ 0353
,034B C8 INY
,034C C0 1E CPY #1E
,034E D0 F1 BNE 0341
,0350 4C 3E 03 JMP 033E

```

```

220 OPEN1,0:PRINT"UHRZEIT (Z.B. 182056)
= ";;INPUT#1,LA$:CLOSE1:PRINT"
230 OPEN1,0:PRINT"FILENAME (MAX. 16 Z.) =
";;INPUT#1,FL$:CLOSE1:PRINT"
240 L=LEN(FL$)
245 P=VAL(LA$)
246 T=LEN(LA$)
250 IFL>16THEN200
255 IFT<6THEN200
256 IFP<00000000RP>235959THEN200
260 PRINT"DISKETTE MIT DEM PROGRAMM
"FL$" EINLEGEN."
270 PRINT"
SPACE"
280 GETR$:IFR$(">") THEN280
290 V=1:PRINT":GOTO11
500 IFLA$=TI$THEN520
510 GOTO11
520 PRINT":POKE198,0
530 PRINT"+CHR$(34)+FL$+CHR$(34)+
",8"
540 PRINT"RUN";
560 POKE631,13:POKE632,13
580 POKE198,2
9999 END
10000 S$=LEFT$(TI$,2)
10001 M$=MID$(TI$,3,2)
10002 SE$=RIGHT$(TI$,2)
10003 PRINT" UHRZEIT = "S$;
: "M$; " "SE$; " "
10004 RETURN

```

Eastern Fight

Nach dem Start werden zunächst die Daten für die Maschinenprogramme und die Grafik eingelesen, die nach ca. 3 Min. erscheint. Nun können Sie durch Druck auf die Space-Taste mit dem Spiel beginnen.

Bei "Eastern Fight" müssen Sie möglichst erfolgreich gegen einen computergesteuerten Gegner antreten, den Sie mit der linken Figur (Joystick in Port 2) bekämpfen. Außer nach rechts und links können auch Tritte in verschiedenen Höhen

ausgeteilt werden, indem Sie den Feuerknopf drücken und den Joystick gleichzeitig nach oben, unten oder rechts halten.

Um nun einen Punkt zu bekommen, müssen Sie Ihren Gegner natürlich treffen. Dieser darf dabei aber nicht angreifen. Wenn die Zeit abgelaufen ist, können Sie Ihre Punkte zählen und neu beginnen. Verzweifeln Sie nicht, wenn Sie am Anfang immer gegen den Computer verlieren. Er wurde nämlich so programmiert, daß er seine Spielfigur um ca. 10 % schneller bewegen kann.

Christian Rduch

```
100 POKE53280,0:POKE53281,0:POKE646,1:PR
INT"*****EASTERN FIGHT"
101 PRINT"***** (C) BY RADDISH-SO
FT"
102 PRINT"*****BITTE WARTEN SI
E 4 MIN." :FORT=1TO4000:NEXT
103 FORX=832TO865:READQ:POKEX,Q:NEXT
104 DATA120,169,51,133,1,169,0,133,95,13
3,90,133,88,169,208,133,96,169,240,133
105 DATA89,169,224,133,91,32,191,163,169
,55,133,1,88,96
106 SYS832:POKE850,160:POKE56576,PEEK(56
576)AND252:POKE53272,8:POKE648,192
107 PRINT"J"
201 AD=30000:FORI=1000TO1075:READA#,PS:C
=0:FORN=1TOLEN(A#):STEP2
202 A=ASC(MID$(A#,N,1))-48:B=ASC(MID$(A#
,N+1,1))-48:A=A+7*(A>9):B=B+7*(B>9)
203 POKEAD,16*A+B:C=C+PEEK(AD):AD=AD+1
204 NEXT:PRINTI;" " :IFC<>PSTHENPRINT"ER
ROR":STOP
205 PRINT"OK":NEXT
206 GOTO2000
1000 DATA A0648C13C0C88C10C0C88C16C0C88C
38C0C88C3EC0C88C39C0C88C3AC0C88C3CC0, 4266
1001 DATA 8C3DC0C88C3BC020BC7BA900850285
FD85FB85FC85FE8DBC0220EE79A9018D, 4206
1002 DATA C0028DBF0278A9808D1403A9758D15
035860ADB02F0034CEB7CA502F0034C, 3265
1003 DATA DD75A90085FCAD00DCC977F023C97B
F0262910F0034CDD75AD00DC2908F01F, 3909
1004 DATA AD00DC2901F01FAD00DC2902F01F4C
DD75A90185024CDD75A90285024CDD75, 3266
1005 DATA A90385024CDD75A90485024CDD75A9
0585024A502F013C901F01BC902F07DC9, 3416
1006 DATA 03F00AC904F0094CFD754C89774CCC
764C0B774C4A77A5FBF005C6FB4C8977, 3800
1007 DATA A4FCC010F050AD00D0C9C8B03BAD00
D069078D00D0AD02D069078D02D0AD04, 3826
1008 DATA D069078D04D0AD06D069078D06D0B9
207D8DF8C3C8B9287D8DF9C3C8B9287D, 4146
1009 DATA 8DFAC3C8B9287D8DFBC318A5FC6904
85FCA90A85FB4C8977A90085024C8977, 4189
1010 DATA A5FBF005C6FB4C8977A4FCC010F050
AD00D0C932903BAD00D0E9088D00D0AD, 4365
1011 DATA 02D0E9088D02D0AD04D0E9088D04D0
AD06D0E9088D06D0B9387D8DF8C3C8B9, 4099
```

```
1012 DATA 387D8DF9C3C8B9387D8DFAC3C8B938
7D8DFBC318A5FC690485FCA90A85FB4C, 4645
1013 DATA 8977A90085024C8977A5FBF005C6FB
4C8977A4FCC00CF029B9487D8DF8C3C8, 4401
1014 DATA B9487D8DF9C3C8B9487D8DFAC3C8B9
487D8DFBC318A5FC690485FCA91485FB, 4812
1015 DATA 4C8977A90085024C8977A5FBF005C6
FB4C8977A4FCC00CF029B9547D8DF8C3, 4289
1016 DATA C8B9547D8DF9C3C8B9547D8DFAC3C8
B9547D8DFBC318A5FC690485FCA91485, 4797
1017 DATA FB4C8977A90085024C8977A5FBF005
C6FB4C8977A4FCC00CF029B9607D8DF8, 4357
1018 DATA C3C8B9607D8DF9C3C8B9607D8DFAC3
C8B9607D8DFBC318A5FC690485FCA914, 4895
1019 DATA 85FB4C8977A90085024C8977EAA5FD
F0034CE977A90085FEA502F013C901F0, 4099
1020 DATA 22C902F02FC903F040C904F02E4CDE
77EAD04DC2903F015C901F018C902F0, 3795
1021 DATA 294CE977AD04DC2903F0184CC977A9
0185FD4CE977A90285FD4CE977A90385, 3844
1022 DATA FD4CE977A90485FD4CE977A90585FD
A5FDF013C901F01EC902F017C903F00A, 4297
1023 DATA C904F0094C09784CA4794CE1784C23
794C65794C7878ADB0C02F006CEBC024C, 3368
1024 DATA A479A4FEC010F051AD08D0C9C8B03B
AD08D069088D08D0AD0AD069088D0AD0, 3979
1025 DATA AD0CD069088D0CD0AD0ED069088D0E
D0B97C7D8DFCC3C8B97C7D8DFDC3C8B9, 4368
1026 DATA 7C7D8DFEC3C8B97C7D8DFFC318A5FE
690485FEA9098DBC024CA479A90085FD, 4428
1027 DATA 4CA479ADBC02F006CEBC024CA479A4
FEC010F051AD08D0C932903BAD08D0E9, 4139
1028 DATA 088D08D0AD0AD0E9088D0AD0AD0CD0
E9088D0CD0AD0ED0E9088D0ED0B96C7D, 3768
1029 DATA 8DFCC3C8B96C7D8DFDC3C8B96C7D8D
FEC3C8B96C7D8DFFC318A5FE690485FE, 5151
1030 DATA A9098DBC024CA479A90085FD4CA479
ADBC02F006CEBC024CA479A4FEC00CF0, 4014
1031 DATA 2AB98C7D8DFCC3C8B98C7D8DFDC3C8
B98C7D8DFEC3C8B98C7D8DFFC318A5FE, 5243
1032 DATA 690485FEA9118DBC024CA479A90085
FD4CA479ADBC02F006CEBC024CA479A4, 3820
1033 DATA FEC00CF02AB9987D8DFCC3C8B9987D
8DFDC3C8B9987D8DFEC3C8B9987D8DFF, 5351
1034 DATA C318A5FE690485FEA9118DBC024CA4
79A90085FD4CA479ADBC02F006CEBC02, 3933
1035 DATA 4CA479A4FEC00CF02AB9A47D8DFCC3
C8B9A47D8DFDC3C8B9A47D8DFEC3C8B9, 5239
1036 DATA A47D8DFFC318A5FE690485FEA9118D
BC024CA479A90085FDAD1ED08D0D02F0, 4235
1037 DATA 0CA502C903901BA5FDC90390034CA5
7AA5FCC90C90F7EEBF0220837A20EE79, 3814
1038 DATA 4CA57AA5FDC90390E5A502C9039003
4CA57AA5FEC90C90D6EEC00220837A20, 3978
1039 DATA EE794CA57AA9648D00D08D04D0A97C
8D02D08D06D0A9C88D08D08D0CD0A9E0, 4172
1040 DATA 8D0AD08D0ED0A9918D01D08D03D08D
09D08D08D0A9A68D05D08D07D08D0DD0, 3863
1041 DATA 8D0FD0A9018D20D08D22D08D25D0A9
008D26D0A004A9089326D088D0F8A909, 3750
1042 DATA 8D23D0A9038D21D0A004A907992AD0
88D0F8A9FF8D15D08D1CD0A0108CF8C3, 4203
1043 DATA C88CF9C3C88CFAC3C88CFBC3A0288C
FDC3C88CFCC3C88CFCC3C88CFEC360A9, 5774
1044 DATA 008D04D4A90F8D18D4A9088D05D4A9
008D06D4A9788D00D48D01D4A9818D04, 3420
```



```

1045 DATA D460AD08D0ED00D0C90B9008A9018D
BE024CBF7AA9008502A90185FDCE15C0, 3677
1046 DATA CE15C0AD15C0C95A90034C077B18AD
15C0690A8D15C0CE14C0AD14C0C95A90, 3566
1047 DATA 034C077B18AD14C0690A8D14C0CE12
C0AD12C0C95A90034C077B18AD12C069, 3041
1048 DATA 0A8D12C0CE11C0AD11C0C95AD011AD
12C0C95AD00AAD14C0C95AD0034C227B, 3686
1049 DATA 4C9D7BAD15C0C95CB0F6A9018D0B02
A010B9B07D99F8C288D0F7A01BB9C17D, 4495
1050 DATA 9920C388D0F7A013B9DD7D9948C388
D0F718ADB0F0269598D2CC318ADC00269, 4152
1051 DATA 598D54C3ADB0F02C0C0029010F01CA0
18B9F17D9970C388D0F74C927BA01FB9, 4209
1052 DATA 0A7E9970C388D0F74C927BA00DB947
7E9978C388D0F7A01CB92A7E99C0C388, 4363
1053 DATA D0F7ACBF02A95188F0069953C04CA2
7BACC002A95188F0069967C04CB07B4C, 4138
1054 DATA 31EA90085FBA9C485FCA206A90085
FDA9CA85FEA000B1FB91FDC00F9E6FC, 5192
1055 DATA E6FECAD0F020277CA90085FBA9CA85
FCA91685FDA215A000B1FB8502A002B1, 4460
1056 DATA FBA00091FBA502A00291FB18A5FB69
0385FBA5FC690085FCCAD0DC18A5FB69, 4546
1057 DATA 0185FBA5FC690085FCC6FDA5FDD0C7
60A90085FBA9CA85FCA206A000A90085, 4603
1058 DATA 02B1FB29C0F01BC9C0F009C980F00F
E6024C587C18A502690385024C587CE6, 3628
1059 DATA 02E602B1FB2930F023C930F00EC920
F01418A502690485024C817C18A50269, 3082
1060 DATA 0C85024C817C18A50269088502B1FB
290CF023C90C00EC908F01418A50269, 2903
1061 DATA 1085024CAA7C18A502693085024CAA
7C18A50269208502B1FB2903F023C903, 2784
1062 DATA F00EC902F01418A502694085024CD3
7C18A50269C085024CD37C18A5026900, 3081
1063 DATA 8502A50291FBC8D006E6FCCAD00460
4C337C4C317CAD01DC2910F0034C257D, 3536
1064 DATA EAA9008DBB02A9608D11C0A95A8D12
C08D14C08D15C020EE79A9018DBF028D, 3701
1065 DATA C002A029A920994FC088D0FAA0EFA9
3599F8C288D0FA4C31EA101114151011, 3890
1066 DATA 161710111819101112131011181910
1116171011141510111213101118191A, 609
1067 DATA 1C1B1D10111213101118191E201F21
10111213101118192224232510111213, 726
1068 DATA 29282D2C29282F2E2928313029282B
2A2928313029282F2E29282D2C29282B, 1334
1069 DATA 2A292831303432353329282B2A2928
31303836393729282B2A292831303C3A, 1454
1070 DATA 3D3B29282B2A35040905351A050914
3509131435150D353513090535080102, 861
1071 DATA 050E3535353510150E0B140535150E
043504051235030F0D10151405123535, 760
1072 DATA 35353510150E0B14053504010D0814
35080102050E35130905350705170F0E, 632
1073 DATA 0E050E3504010D0914350801143504
051235030F0D1015140512350705170F, 562
1074 DATA 0E0E050E3513100103053514011314
053506150512350E0515051335131009, 579
1075 DATA 050C35150E050E1413030809050405
0E20, 243
2000 FORT=784*64T0810*64:POKET,0:NEXT
2010 FORT=0T062:READQ:POKET+784*64,Q:NEX
T
2020 FORT=36T052:READQ:POKET+785*64,Q:NE

```

```

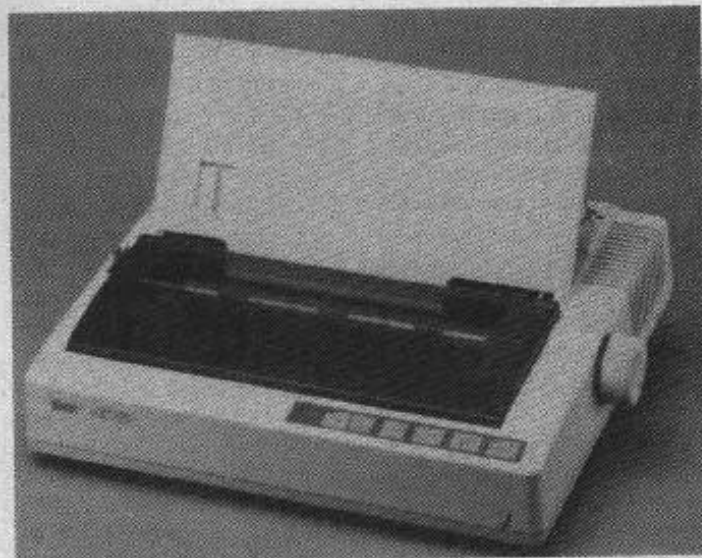
XT
2030 FORT=0T062:READQ:POKET+786*64,Q:NEX
T
2040 FORT=0T062:READQ:POKET+787*64,Q:NEX
T
2050 FORT=0T059:READQ:POKET+788*64,Q:NEX
T
2060 FORT=0T059:READQ:POKET+789*64,Q:NEX
T
2070 FORT=0T059:READQ:POKET+790*64,Q:NEX
T
2080 FORT=48T057:READQ:POKET+791*64,Q:NE
XT
2090 FORT=0T059:READQ:POKET+792*64,Q:NEX
T
2100 FORT=12T057:READQ:POKET+793*64,Q:NE
XT
2110 FORT=0T062:READQ:POKET+794*64,Q:NEX
T
2120 FORT=0T062:READQ:POKET+795*64,Q:NEX
T
2130 FORT=33T062:READQ:POKET+796*64,Q:NE
XT
2140 FORT=0T053:READQ:POKET+797*64,Q:NEX
T
2150 FORT=9T062:READQ:POKET+798*64,Q:NEX
T
2160 FORT=0T062:READQ:POKET+799*64,Q:NEX
T
2170 FORT=33T062:READQ:POKET+800*64,Q:NE
XT
2180 FORT=0T056:READQ:POKET+801*64,Q:NEX
T
2190 FORT=39T062:READQ:POKET+802*64,Q:NE
XT
2200 FORT=0T062:READQ:POKET+803*64,Q:NEX
T
2210 FORT=51T062:READQ:POKET+804*64,Q:NE
XT
2220 FORT=0T062:READQ:POKET+805*64,Q:NEX
T
2240 DATA,,60,,63,,255,,250,,251,,2
50,,234,,235,,170,,1,104,,1,104,,5,85
2250 DATA,5,85,,21,213,,21,221,,21,118,,
5,90,,1,90,,3,90,,1,221,,1,117
2260 DATA64,,80,,84,40,,149,168,,149,1
60,,20,128
2270 DATA,1,85,,3,87,,1,253,,1,117,,1,22
1,,5,93,,21,85,,21,85,,85,77,,85,195
2280 DATA1,87,,1,92,,1,92,,1,80,,5,112,1
2,5,64,255,5,67,255,5,127,243,9,252,,42
2290 DATA240,,40,,
2300 DATA,,,,,64,,80,,80,,84,,84
,,85,,85,,21,64,,53,64,,5,64,,5,64
2310 DATA,1,80,,193,80,,241,92,,252,92,,
15,170,,3,234,,
2320 DATA,1,85,,3,87,,1,253,,1,117,,5,22
1,,5,85,,21,85,,21,117,,21,197,,85,197
2330 DATA,87,13,,87,1,,21,1,,21,193,,5,1
13,,5,109,,61,109,,252,163,,48,163,,160
2340 DATA,,,,,64,,64,,64,,64,,
,64,,64,,80,,80,,80,,80,,80,,80
2350 DATA,,80,,92,,168,,168,,
2360 DATA,1,85,,3,87,,1,253,,5,117,,5,22
1,,5,85,,5,117,,5,93,,1,92,,1,84,,3,84
2370 DATA,,84,,87,,1,223,,1,85,,1,117,,

```

```

1,117,,14,191,,2,190,,10,130
2380 DATA192,,128,,176,,160
2390 DATA,1,85,,3,87,,1,253,,1,117,,1,22
1,,1,85,,1,85,,1,93,,1,85,,1,93,,1,81
2400 DATA,1,81,,5,113,,5,117,,5,69,,5,12
1,,5,122,,50,250,,62,142,,14,131
2410 DATA64,,64,,64,,80,,80,,80,,8
0,,112,,64,,64,,64,,192,,,,,192
2420 DATA,,192
2430 DATA,,,,,60,,63,,255,,250,,251,
,,250,,234,,235,,170,,1,104,,1,104
2440 DATA,5,85,,5,85,,21,213,,23,93,,21,
117,,5,85,,1,165,,3,169,,1,173,,1,119
2450 DATA,1,93,,3,117,,1,245,,1,125,,1,2
13,,1,85,,1,92,,1,80,,1,80,,1,80,,5,112
2460 DATA,5,112,,5,64,,5,112,,5,67,,5,79
,,5,127,,5,112,,14,112,,10,192
2470 DATA64,,80,,86,128,,90,128,,218,,
,20,,,,,64,,84,2,,85,82,
2480 DATA85,94,,85,90,,85,92,,16,,,,,
,,192,,252,,63,252,,243,
2490 DATA,15,,15,192,,63,192,,62,128,,6
2,192,,58,128,,58,192,,58,128,,58,5
2500 DATA,10,85,,21,85,,85,85,,1,85,150,5
,85,80,5,87,80,5,197,84,21,1,84,23,1,92
2510 DATA20,1,93,24,1,117,40,3,117,40,1,
213,8,1,125,1,213,1,80,,1,80,,1,80
2520 DATA,1,80,,1,80,,1,80,,1,112,,1,64,
,1,64,,5,67,,5,79,,5,124,,6,240,,10,176
2530 DATA,2,128,64,,84,,86,,90,128,,5
0,130,,2,,26,,90,,5,90,,21,88,,85,92
2540 DATA,85,192,,84,,80,,112,,,,,
,,,,,240,,63,12
2550 DATA,,243,
2560 DATA,,240,,252,,3,252,,3,232,,3,23
6,,3,168,,3,172,,3,168,,1,160,,5,161
2570 DATA,21,85,,85,85,,93,221,,87,85,,2
1,87,,6,148,,2,148,,1,148,,7,92,,21,245
2580 DATA,85,213,1,87,117,1,85,5,1,92,,1
,87,252,,85,255,,213,131,3,198,128,,2
2590 DATA128,,32,,160,,1,160,,5,128,,21
,,84,,92,,80,,112,,,,,
2600 DATA,,,,,64,,84,,85,,85,64,,53,8
0,,1,84,,255,86,,12,218,128,,10,128
3040 B=57344
3050 READZE:IFZE=-1THEN3300
3060 FORT=0T07:READQ:POKEB+ZE*8+T,Q:NEXT
3070 GOTO3050
3080 DATA27,0,0,4,6,21,21,102,89,28,0,0,
0,64,64,64,80,80,30,0,0,0,252,253,253
3090 DATA253,253,31,0,0,0,0,0,0,85,33,
0,1,1,5,5,22,86,85,35,85,101,85,149,85
3100 DATA86,102,85,36,148,84,101,85,153,
166,105,85,37,0,0,0,64,80,80,85,38,255
3110 DATA255,255,255,255,255,253,253,39,
255,255,255,255,255,2,2,40,255,255
3120 DATA255,255,255,255,0,0,41,255,255,
255,255,255,255,170,186,42,255,255
3130 DATA255,255,255,255,170,171,43,255,
255,255,255,255,255,0,128,44,253,253,253
3140 DATA255,255,255,255,255,45,10,42,85
,255,255,255,255,255,46,170,234,85,255
3150 DATA255,255,255,255,47,170,170,85,2
55,255,255,255,255,48,186,170,85,255
3160 DATA255,255,255,255,49,170,170,85,2
55,255,255,255,255,50,171,170,85,255
3170 DATA255,255,255,255,51,128,160,85,2
55,255,255,255,255,52,170,170,170,170
3180 DATA170,170,170,170,53,85,85,85,85,
85,85,85,85,54,0,0,85,255,255,255,255
3190 DATA255,55,0,0,1,85,85,21,0,0,56,0,
5,85,85,85,85,5,0,57,0,80,84,85,85,85,85
3200 DATA5,58,0,0,0,0,85,85,85,85,59,0,0
,0,0,80,84,80,80
3210 DATA91,85,89,105,89,89,89,89,89,90,
85,106,102,102,102,102,102,106,92,85
3220 DATA106,86,86,106,101,101,106,93,85
,106,86,86,106,86,86,106,94,85,101,102
3230 DATA102,106,86,86,86,95,85,106,101,
101,106,86,86,106,96,85,106,101,101,106
3240 DATA102,102,106,97,85,106,86,86,86,
89,89,89,98,85,106,102,102,106,102,102
3250 DATA106,99,85,106,102,102,106,86,86
,106,100,85,85,89,89,85,89,89,85
3260 DATA101,37,37,37,37,37,37,37,37,102
,88,88,88,88,88,88,88,88
3270 DATA103,37,37,37,9,2,0,0,0,104,98,8
8,98,96,128,0,0,0,105,85,85
3280 DATA85,85,170,0,0,0,106,85,85,85,85
,85,150,40,0
3281 DATA81,168,168,168,168,168,168,168,
0,1,85,106,102,102,106,102,102,102,2
3282 DATA85,105,102,102,105,102,102,105,
3,85,106,101,101,101,101,101,106,4,85
3283 DATA105,102,102,102,102,102,105,5,8
5,106,101,101,106,101,101,106,6,85
3284 DATA106,101,101,106,101,101,101,7,8
5,106,101,101,101,102,102,106,8,85
3285 DATA102,102,102,106,102,102,102,9,8
5,89,89,89,89,89,89,89,10,85,106,86
3286 DATA86,86,102,102,106,11,85,102,102
,102,105,102,102,102,12,85,101,101,101
3287 DATA101,101,101,106,13,85,102,106,1
06,102,102,102,102,14,85,102,102,106
3288 DATA106,106,102,102,15,85,106,102,1
02,102,102,102,106,16,85,106,102,102,106
3289 DATA101,101,101,17,85,106,102,102,1
02,102,106,106,18,85,106,102,102,106
3290 DATA105,102,102,19,85,106,101,101,1
06,86,86,106,20,85,106,89,89,89,89
3291 DATA89,89,21,85,102,102,102,102,102
,102,106,22,85,102,102,102,102,89,89,89
3292 DATA23,85,102,102,102,102,106,106,1
02,24,85,102,102,102,89,102,102,102
3293 DATA25,85,102,102,102,89,89,89,89,2
6,85,106,86,89,89,101,101,106
3299 DATA-1
3300 PRINT"*****"
3310 PRINT"      IF
      ";
3320 PRINT"+++++!#%$+++++!+++++
++++!#%$++++"
3330 PRINT"(((&')*)*)+&(((((&((((
(&')**))*)+&")
3340 PRINT"666,-./01223,66666666,6666666
6,-./01223,"
3350 FORT=49632T049751:POKET,160:NEXT
3360 FORT=49752T049911:POKET,52:NEXT
3370 FORT=49912T050151:POKET,53:NEXT
3380 PRINT"*****789:;*****
789:;      789:;"
3500 POKE53270,216:SYS30000:FORT=1T0500:
NEXT:POKE49169,96
9999 GOTO9999

```

Der Star NL10, sauber verpackt

den. Gegen Aufpreis ist außerdem ein automatischer Einzelblatteinzug erhältlich, der einfach anstelle dieser Abdeckung aufgesteckt wird.

Mit dem Commodore-Interface bietet der Star NL 10 dem Anwender den gesamten Befehlssatz des MPS 803 und damit die volle Kompatibilität zu diesem Drucker einschließlich des Commodore-Zeichensatzes. Die Besonderheit ist jedoch, daß auch die anderen Fähigkeiten des NL 10 mit dem Interface nutzbar bleiben. Sie werden unter Programmkontrolle fast alle über die Epson-üblichen Escape-Charactersequenzen erreicht.

Neben den vier Schriftarten Pica, Elite (96 Zeichen pro Zeile), Schmalschrift (136 Zeichen pro Zeile) und der Near-Letter-Quality-Schrift (18x23 Punktmatrix, Standard: 9x11) gibt es einen Proportional-schriftmodus sowie Breitschrift, Fettschrift, Schrägschrift und die Möglichkeit, Schriften in doppelter und vierfacher Größe zu drucken. Ebenso ist Hoch- und Tiefstellen und Unterstreichen vorgesehen. Es kann zwischen den acht internationalen Zeichensätzen sowie zwischen ASCII- und Commodore-Modus umgeschaltet werden, wobei allerdings der Commodore-Zeichensatz nicht vollständig von der NLO-Schrift unterstützt wird (Grafikzeichen erscheinen in Normalschrift). Der Master-Print-Modus erlaubt das Einstellen und Mischen von

sieben Druckmodi mit einem Befehl.

Des weiteren existieren leistungsfähige Kommandos, die der formatierten Ausgabe dienen. So können die vier Blatt-ränder, horizontale und vertikale Tabulatoren und der Papiervorschub in beide Richtungen (z. B. in 72tel oder 216tel Zoll Schritten) beliebig eingestellt werden. Ferner ist der links- oder rechtsbündige und der zentrierte Druck von Zeilen möglich. Natürlich wird auch die Steuerung des automatischen Einzelblatteinzugs durch Charactersequenzen übernommen. Die Anwendung von Macros gestattet das Zusammenfassen von bis zu dreißig Bytes langen Befehlsstrings, die dann mit einer einzigen Escape-sequenz aktiviert werden können. Doch es fehlt wie auch beim Master-Print-Mode die Sequenz, die diesen Zustand wieder deaktiviert, ohne jeden Befehl einzeln abschalten zu müssen. 96 Zeichen können in Normalform als Proportional-schriftzeichen oder in NLO-Form frei definiert und ins RAM des NL 10 geladen werden.

Neben der 7-Nadel Commodore-Grafik, die in einfacher (auch invers) und doppelter Dichte einsetzbar ist, verfügt der NL 10 im ASCII-Modus über vier verschiedene Grafikmodi. Die 8-Nadel Grafik kann in einfacher (60 Punkte pro Zoll), doppelter (120) und vierfacher (240) Dichte auf das Pa-

pier gebracht werden. Außerdem ist die Grafik bei doppelter Dichte wahlweise auch mit doppelter Geschwindigkeit ausführbar.

Wenn das deutsche Handbuch (es lag beim Test noch nicht vor) genauso gut wird wie das englische, dann läßt es kaum Wünsche offen. Viele Basic-Beispielprogramme erleichtern auch dem Einsteiger die Einarbeitung in den umfangreichen Befehlssatz.

Mit der Arbeitsgeschwindigkeit von 120 Zeichen pro Sekunde (30 Zeichen pro Sekunde im NLO-Modus) liegt der STAR NL 10 mit an der Spitze in seiner Preisklasse. Die vielfältigen

Anwendungs- und Anpassungsmöglichkeiten, das hervorragende Preis-/Leistungsverhältnis und das Steckmodulkonzept erschließen ihm ein großes Software-reservoir (für den C 64 z. B.: Print Shop, Textomat+) und machen ihn für einen weiten Käuferkreis interessant.

System: Commodore 64
(IBM PC, Centronicsdrucker)
Hersteller: Star Micronics
Deutschland GmbH
Frankfurter Allee 1-3
6236 Eschborn/Ts.
Preis: 1145.- DM
(inkl. Interface)

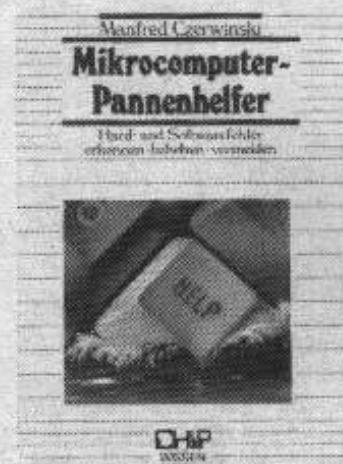
job

Microcomputer Pannenhelfer

Von M. Czerwinski
Vogel-Buchverlag
168 Seiten, 28.- DM
ISBN 3-8023-0652-X

Die Freude am Computer währt nur so lange, wie er reibungslos funktioniert. Treten jedoch die fast unvermeidlichen hard- und softwarebedingten Pannen auf, ist guter Rat teuer. Mit dem Mikrocomputer-Pannenhelfer, einem neuen Buch aus der erfolgreichen Reihe "Chip Wissen" kann man die Fehler am Computer, Bildschirm und Drucker sowie in Programmen aufspüren, Abhilfe schaffen und weiteren Fehlfunktionen vorbeugen.

Die Bereiche Hardware und Software werden getrennt behandelt. Eine ausführliche Fehlersuchtafel schließt den Hardwareteil ab. Im Softwareteil werden die Standardfehler-meldungen verschiedener Basic-Versionen (MBasic 2.0, C-64-Basic, Applesoft-Basic, Schneider-Basic) erläutert, logische Fehler behandelt und diverse Testmethoden vorgestellt. Der Hardware-Teil enttäuscht jedoch. Da für verschiedene Computer geschrieben, konnte der Autor nicht besonders tief auf die jedem Gerät eigenen, speziellen Macken und Fehlerquellen eingehen, so daß das Ganze etwas oberflächlich



wirkt und nur globale Aussagen zu den einzelnen Pannen und deren Vermeidung gemacht werden.

Ganz anders jedoch der Software-Teil. Hier glänzt das Buch mit einer Fülle von informativen Erläuterungen, Tips, Tricks und Hilfsprogrammen, die dem Anfänger wie dem Profi gleichermaßen nützlich sind. Abgerundet wird dieser Teil durch die Vorstellung verschiedener Testhilfen, die das Aufstöbern und Korrigieren von Fehlern erheblich beschleunigen. Der Softwareteil entschädigt also für vieles, was man im Hardwareteil vermißt, so daß der Preis von 28.- DM - sicher nicht wenig - trotzdem angemessen erscheint.

Frank Wölk

Green Beret

Daß wir auch bei indizierten Spielen (also Spielen, die die Gewalt verherrlichen) keinen Index librorum prohibitorum aussprechen und dem Leser den Bericht vorenthalten, sehen Sie an dem folgenden Spiele-Test.

Bei "Green Beret" handelt es sich um ein Kriegsspiel, dessen zweifelhafter Sinn darin besteht, "feindliche Soldaten" abzuschlachten. Aus erziehungstechnischen Maßnahmen und verantwortungsvollem Denken heraus sollte man solche Spiele Kindern auf gar keinen Fall zugänglich machen. Da Spiele aber auch und gerade von Jugendlichen im späten Kindesalter (ca. 9 Jahre) gespielt werden, sollte sich meiner Meinung nach jeder Programmierer für die Auswahl eines Themas schämen, mit dem Kinder in diesem prägenden Alter noch nichts anzufangen wissen.

Jetzt aber zum Spiel selbst. Sie sind eine hochtrainierte Kampfmaschine mit dem Auftrag, alle vier feindlichen strategischen Abwehranlagen zu infiltrieren, um Gefangene zu befreien. Sie sind allein gegen eine immense Übermacht. Alles, was Sie bei sich tragen, ist ein Messer. Sie benötigen nichts als Ihr Geschick und "strategisches Können"(!)? Zunächst müssen Sie über Plattformen und Leitern durch die vier feindlichen Abwehrstationen vordringen: die Brücke, den Hafen, die Raketenstation und last not least das Gefangenennager. Unterwegs wird versucht, Ihnen mit Minen, Granaten, Düsenjägern, die Bomben werfen, schießenden Soldaten und hageren Gestalten, die Ihnen mit Karate-Tritten und Schlägen die Zukunft vermiesen wollen, den Weg so steinig wie möglich zu gestalten.

Aber für was hat man denn sein blutiges Messerchen, wenn nicht dafür, einen Soldaten nach dem anderen rücksichtslos

abzustecken? Erwischt man einmal einen Kommandanten (das sind die Typen im weißen Jackett, die ständig versuchen, sich dünne zu machen, wenn's brenzlich wird), so ist man um ein Waffensystem reicher. Das sind dann Flammenwerfer, Granaten und Raketenab-



schußgeräte. Mit diesen geht das Abmurksen wesentlich rationeller vor sich, da man gleich drei oder vier Feinde auf einmal ins Jenseits befördern kann.

Auch die Grafik ist nicht gerade das Gelbe vom Ei, aber dennoch akzeptabel. Die Farbwahl ist so fies gestaltet, daß sich die Gegner zum Beispiel auf der Brücke kaum vom Hintergrund abheben und so der Schwierigkeitsgrad unnötig in die Höhe getrieben wird. Dafür erkennt man den feigen Kommandanten um so schneller.

Der hitparadenreife Sound ist spitzenmäßig gestaltet und erinnert leicht an einen Destruction-Mix. Es sieht so aus, als würden die Möglichkeiten des Sound Interface Device (SID 6581) im C64 voll ausgenutzt.

Negativ aufgefallen ist mir bei diesem Spiel auch ein kleiner Programmierfehler. Eine Restore-Funktion, die das Spiel an den Anfang zurücksetzen sollte, wurde schlicht und einfach vergessen. Diese Funktion liegt bei anderen Spielen normalerweise auf der Taste "Restore" bzw. "Run Stop/Restore". Da aber diese beiden Tasten hier nicht gesperrt wurden, stürzt das Programm beim Drücken der Restore-Taste auf Nimmerwiedersehen ab. Auch

für eine Pause läßt sich das Programm nicht unterbrechen.

Eine freiwillige Selbstkontrolle (FSK) für Software, wie sie auch schon bei Video-Filmen seit geraumer Zeit praktiziert wird, wäre keine schlechte Lösung.

System: C64
Hersteller: Ocean
Preis: Disk 45.-, Cassette 35.-
Bezugsquelle: Rushware
Stefan Selbach

Uns würde einmal Ihre Meinung zu derartigen indizierten Spielen interessieren. Vor allem unsere jüngeren Leser sollten sich einmal melden und schreiben, was sie dazu meinen.

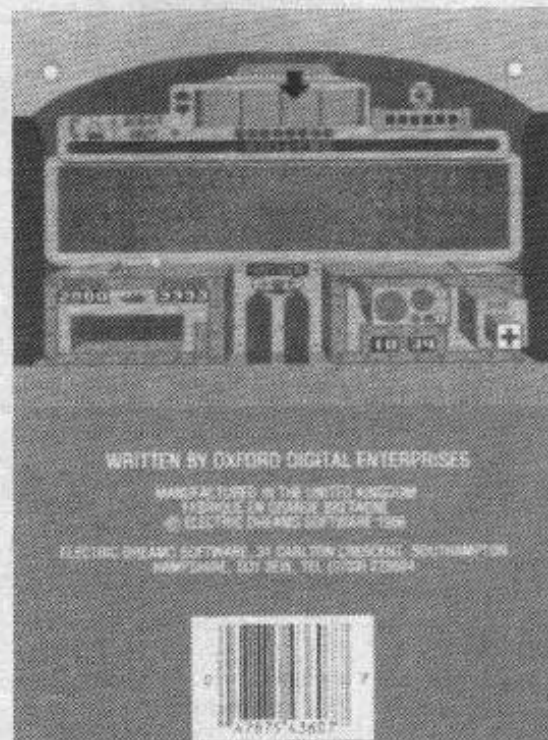
*Verlag Rätz-Eberle,
c/o Stefan Selbach,
Postf. 1640, 7518 Bretten 1.*

R.M.S. Titanic

Am 15.04.1912 versank mit der Titanic das größte Linienschiff aller Zeiten, 46.329 BRT schwer, 883 Fuß lang und 29 Fuß breit. Der Zusammenstoß mit einem Eisberg, nach dem das Schiff in kurzer Zeit sank, riß 1200 Menschen und eine unvorstellbare Zahl von Wertgegenständen mit in die Tiefe.

Soweit zur Hintergrundstory des vor einiger Zeit bei Activision erschienenen Spiels "R.M.S. Titanic", bei dem Sie die Rolle des Besitzers und Geschäftsführers der "Titanic Salvage Company" übernehmen. Es handelt sich dabei um eine Bergungsfirma mit dem Ziel, die Titanic zu heben. Zu diesem Zweck sind Sie mit dem T.S. 1, einem Unterwasserfahrzeug mit allerlei technischen Finessen ausgerüstet.

Ziel des Spieles ist es, mit T.S. 1 zur Titanic hinabzutauchen, sich im Labyrinth der Räume der Titanic zurechtzufinden und den Auslöser der acht, vor kurzer Zeit angebrachten Luftsäcke, deren genaue Lage jedoch nicht bekannt ist, aufzustöbern, um so die Titanic zu bergen. Doch wie im Leben auch geht dies nicht so einfach, wie man sich dies zunächst vorstellt. Da rauben neugierige und sensationshungrige Presseleute einem die kostbare Zeit mit Interviews, bei denen die Antworten jedoch gut überlegt sein wollen, da eine schlechte Publicity sich äußerst ungünstig auf die Zahlungsmoral der Kapitalgeber auswirkt. Ohne deren Zuwendungen wäre das Unterfangen binnen kurzer Zeit zum Schei-



Das Cockpit
des T.S.1

**Computer-Kontakt
das Heft mit den
preisgünstigen
Kleinanzeigen**

tern verurteilt. Doch gemessen an den Kosten, die das Projekt verschlingt, sind diese Finanzspritzen nur ein Tropfen auf den heißen Stein. Deshalb müssen neben den Auslösern für die Luftsäcke auch die Bordtresore gefunden werden, um Geldvorräte aufzufrischen.

Die Umsetzung dieser Handlung liegt im Bereich der grafischen Gestaltung über dem Durchschnitt der meisten C-64-Spiele. Die Bedienung und der Ablauf der Handlung sind relativ einfach, allerdings gibt die Anleitung bei einigen Details nur unzureichend Hilfestellung. Sie ist der große Schwachpunkt dieses ansonsten vorzüg-

lichen Spieles. Die uns vorliegende Version beschränkt sich auf die englische Sprache und ist noch dazu kompliziert geschrieben. Ohne ausreichende Sprachkenntnisse kann man die Vielfalt der Bedienungsmöglichkeiten nur erahnen. Trotzdem ist "R.M.S. Titanic" alles in allem ein äußerst gelungenes Strategiespiel, das auch bei uns so manchen begeisterten Anhänger finden wird, der Wert auf originelle Spielideen, eine gute grafische Gestaltung und hohe Spielmotivation legt.

System: C64
Hersteller: Activision
Preis: 59,- Disk, 39,- Cas.
Frank Wölk

Flight Deck

Terroristen halten die Welt in Atem. Ein vierköpfiges Kommando hat eine Atombombe gestohlen und droht nun, diese auf die Millionenstadt New York abzuschießen, wenn nicht gewisse Forderungen erfüllt werden. Doch Amerika wäre nicht Amerika, wenn die Nation nicht versuchen würde, weder diesen Forderungen nachzukommen noch eine der schönsten Städte der Welt zu verlieren. In einer Nacht-und-Nebel-Aktion wird ein Flugzeugträger unter dem Kommando eines Topagenten zu der Insel geschickt, auf der die Zentrale der Organisation liegt.

Das Ziel, die Zerstörung dieser Zentrale, liegt nun in den Händen des Spielers. Er befiehlt einen Flugzeugträger mit 10 Düsenjägern und hat alle nötigen Vollmachten. Das Spiel selbst vollzieht sich in drei Stufen. Es beginnt mit den Vorbereitungen zum Start, denn bis man sein erstes Flugzeug in der Luft hat, vergeht einige Zeit. Beim Transport des Flugzeugs zur Startrampe darf keine Beschädigung auftreten. Der Start selbst wird in einem kleinen Fenster angezeigt, wobei sich der Pilot haargenau an die Vorgaben eines kleinen Zielquadrates halten muß. Um das Flugzeug auf den richtigen Kurs zu bringen, schaltet man in den Kartenmodus, in dem auch alle wichtigen Daten für den Piloten sichtbar werden.

Erreichen die ersten Düsenjäger das Eiland, muß der genaue Sitz der Terroristen erkundet werden. Hierzu wird das Gelände Millimeter für Millimeter abfotografiert, und erst wenn die ganze Insel bekannt ist, wird das Quartier der Terroristen sichtbar. Dabei ist man vor Angriffen der Erpresser nicht sicher und muß sich mit den Maschinengewehren an Bord zur Wehr setzen. Hat man schließlich ein vollständiges Bild von der Insel, kommt die nächste Stufe, der Inselmodus. Die Bomber starten vom Flugzeugträger und versuchen, das Quartier zu zerstören. Gelingt das im vorgegebenen Zeitlimit, ist die Mission beendet und der Kommandant des Flugzeugträgers kann sich seine Lorbeeren beim Präsidenten abholen.

Die Grafik von Flight Deck ist nicht gerade berauschend, vor allem weil man von der Konkurrenz Besseres gewohnt ist. Trotzdem kann ich das Spiel guten Gewissens empfehlen, denn Strategie und Action finden hier eine optimale Vereinigung. Außerdem treten im Programm immer wieder Sequenzen mit gesprochenen Worten auf, die das Spiel zwar nicht reizvoller, aber ein wenig interessanter machen.

System: C64
Hersteller: Eaglesoft
Bezugsquelle: Profisoft GmbH
Thomas Tai

The Way of the Tiger

Vor langer, langer Zeit legte eine alte Dienerin ein Baby auf die Stufen des Kwon-Tempels. Die Mönche nahmen sich des Kleinen an und bildeten den Heranwachsenden in den Techniken des Ninja-Kampfes aus. Bevor er sich aber selbst mit dem hohen Titel "Ninja" schmücken konnte, waren einige sehr harte Prüfungen durchzustehen, die er entweder meisterte oder daran starb.

In die Rolle dieses Prüflings schlüpft nun der Spieler bei "The Way of the Tiger", wenn der Meister seinen Lehrling in die Zauberwelt von Orb schickt, in der viele gefährliche Wesen aus der Unterwelt Gewehr bei Fuß stehen, dem angehenden Ninja den Garaus zu machen. Zunächst kommt es in der Wüste zum unbewaffneten Zweikampf. Neben blitzschneller Reaktion ist auch große Aufmerksamkeit gefordert, denn die Gegner können hinter jedem Felsblock hervorkommen und unseren Kämpfer von hinten überfallen oder aus dem Boden wachsen, ohne dem überraschten Prüfling eine Abwehrchance zu lassen. Besonders angenehm fällt bei diesem Programmteil die liebevoll gestaltete Hintergrundgrafik auf. Hoch über den Köpfen der Kämpfenden ziehen weiße Wolken, und im Vordergrund wiegen sich Blumen und Gräser im Wind.

Jäh wird man aus diesem Naturschauspiel auf den schlüpfrigen Steg über einen morastigen See versetzt. Nun muß sich der Spieler nicht mehr mit Händen und Füßen wehren, sondern geht mit einer langen Stange auf die ebenso bewaffneten Gegner los. Skelette steigen aus dem Wasser, Monster greifen von hinten an, und ehe man sich versieht, rutscht man aus und ertrinkt in den eiskalten Fluten. Und das wäre nun wirklich kein würdiger Tod für einen Ninja.

Nach diesem Balanceakt geht es dem schwersten Teil der Prüfung entgegen. Der Schwertkampf ist nicht nur wegen der Endgültigkeit eines Hiebes so gefährlich, sondern hier trifft der Spieler auch auf seinen Lehrmeister, dessen

Überwindung das Bestehen der Prüfung bedeuten würde.

Die ganze Zeit über schaut der Kwon-Gott wohlwollend auf seinen Schützling herab und gibt ihm innere Kraft und Ausdauer, sobald er einen Kampf gewonnen hat. Stellt sich der Spieler allerdings dilettantisch



an, kann er auf keine Unterstützung hoffen, und wenn sich die Anzeige der Energiepunkte langsam dem Ende nähert, muß er sogar auf sein baldiges Ableben gefaßt sein.

Die Hintergrundgrafik ist, wie erwähnt, sehr gut gelungen und fasziniert wirklich jedesmal von neuem. Auch die Bewegungsabläufe sind flüssig, wenn auch etwas langsam, was wohl weniger mit dem mangelnden Können der Programmierer, als mit der Gemächlichkeit des C64 zu erklären ist. Im positiven Gesamtbild von "The Way of the Tiger" fällt allerdings der magere Sound negativ auf. Im Titelbild wird man zwar von einer mitreißenden Melodie empfangen, doch im Spiel selbst herrscht außer den mageren Kampfgeräuschen eine gespenstische Stille. Insgesamt gesehen also ein empfehlenswertes Spiel, bei dem nur der Sound unter den Möglichkeiten des C64 bleibt.

System: C64, MSX, CPC
Hersteller: Gremlin Graphics
Preis: ca. 40,- DM
Bezugsquelle: Profisoft GmbH, Osnabrück
Thomas Tai

C 64 – ständig im Griff (Band 1)

Von
Manfred Walter Thoma
Dr. Alfred Hüthig Verlag
96 Seiten, 21.80 DM
ISBN 3-7785-1302-8

Dieses kleine Nachschlagewerk soll als Unterstützung der Arbeit am Commodore 64 dienen. Es kann und will auch nicht dicke Handbücher ersetzen, sondern dokumentiert die einzelnen Themenbereiche in möglichst knapper, präziser Form. Von der Speicheraufteilung über die Variablentypen, die Register des VIC- und SID-Chips, die CIAs, die Sprites und die Basic-Befehle bis hin zum DOS und der Diskettenorganisation reicht die Themenpalette, die in den neun Kapiteln behandelt wird.

Die Beschreibung der Register beschränkt sich auf die Angabe der Speicheradressen und der Funktionen der entsprechenden Bits. Ein wenig ausführlicher wird nur auf den Spriteaufbau eingegangen. Die Basic-Befehle sind alphabetisch aufgelistet und mit kurzen Beispielen und Hinweisen versehen (inklusive den Systemvariablen ST und TI bzw. TI\$). Sie werden durch eine Liste der Basic-Fehlermeldungen komplettiert. Auch die Fehlermeldungen, die bei der Arbeit mit den DOS-Befehlen auftreten können, werden in Deutsch erklärt und aufgelistet. Ebenso sind die Basic-Tokens aufgeführt.

Darüber hinaus findet der Anwender einige nützliche und oft vermißte, kleine Basic-Routinen, die z.B. dem Auslesen des Fehlerkanals, dem Aneinanderhängen von Programmen (MERGE), dem Laden und

Speichern von Maschinenprogrammen und dem Retten von nicht korrekt geschlossenen Dateien dienen.

Dieser kleine Band wird seinem Anspruch, den C-64-User bei seiner Arbeit zu unterstützen, vollauf gerecht. Durch seine ungewöhnliche Vollständigkeit (mir fiel nur das Fehlen des OPEN-Befehls in Verbindung mit Programmdateien auf) bei gleichzeitiger, erfreulicher Kürze und Übersichtlichkeit ist er als Nachschlagewerk fast unentbehrlich. Gerade ein solches sollte jedoch über ein Stichwortverzeichnis verfügen, das sich aber bei zukünftigen Auflagen sicher noch einfügen läßt. Der Preis von 21.80 DM ist für einen 96seitigen, spiralgebundenen Band im Taschenbuchformat, der vorwiegend dem schnellen Auffinden von Befehlen und Speicheradressen dient und kein Systemhandbuch ersetzen kann, doch relativ hoch. Somit dürfte er nur für Anwender interessant sein, die schon die entsprechende "Primärliteratur" besitzen.

job



Handbuch mit zahlreichen Erklärungen zum System selbst und seinem psychologischen und pädagogischen Hintergrund, eine praktische Übersichtskarte der Tastaturbelegungen in verschiedenen Kursen (normal ist bei Soft Learning die deutsche QWERTZ-Belegung) und zusätzlich ein Büchlein mit Cassette für eine kostenlose Probelektion in Esperanto.

Der zweite notwendige Teil der Ausrüstung ist ein Sprachkurs mit Grund- und Aufbaukursen in Französisch, Italienisch, Spanisch, Englisch (mit Management-Englisch) und Intensivkursen in den gleichen Sprachen sowie in Russisch und Schwedisch. Angefügt ist wiederum eine Kursdiskette und 4 Audio-Cassetten mit 7-8 Lektionen nebst einem Handbuch, in dem der Inhalt der Diskette/Cassetten mitprotokolliert und übersetzt wurde. Letzter und wichtigster Ausrüstungsgegenstand ist ein gemütlicher Sessel vor dem Computer, der die tiefenpsychologische Wirkung von Soft Learning zu verstärken hat.

Hat man nun alle diese Dinge der Anleitung oder dem Instinkt gemäß zusammengesteckt, geladen, zurückgespult, gestartet, gelesen und geglaubt, so wird einem vom Computer eine Fremdentität (in meinem Fall eine englische) zugewiesen, und man hat Gelegenheit, sich in der Phantasie in eine ruhig gelegene Allee des englischen Badeortes Bournemouth zurückzuziehen, denn vor dem eigentlichen Lernvorgang muß man die richtige Einstellung zum Lernen gefunden haben. Zu diesem Zwecke zeigt der Bildschirm eine Spannungsgrafik, während einem aus dem Recorder eine angenehme Frauenstimme entgegenschallt: "Erinnere Dich an eine Zeit, in der Dir das Lernen besonderen Spaß gemacht hat."

Auch wenn letzteres schwerfallen sollte, so sind die nachfolgenden Lektionen wirklich nicht schlecht aufgezogen. Untermalt von klassischer Musik, die manchmal ablenkend wirkt, kann man auf Bildschirm, Cassette und im Handbuch eine zusammenhängende Geschichte mitverfolgen, die einen, in ein-

zelne Lektionen aufgeteilt, in diverse Bereiche der englischen Sprache einführt. Zwischen oder in den Lektionen findet man im Handbuch kurze Erklärungen und Tabellen zu Zahlen, Zeiten, Konjugation und ähnlich Unerläßlichem.

Nach Abschluß jeder Lektion folgt eine Entspannungs- und Atmungsübung, die mit einem Keyword-Vokabeltraining kombiniert wird. Dabei werden den englischen Vokabeln neben ihrer deutschen Bedeutung sogenannte Keywords zugewiesen, wobei es sich um deutsche Worte handelt, die ähnlich geschrieben oder ausgesprochen werden. So findet man als Keyword zu "holidays" das schöne deutsche Wort "Holundertee".

Um auf diese Art und Weise das im Gedächtnis Gespeicherte etwas zu beleben, bietet das Soft-Learning-System sogenannte Aktivierungsspiele an, bei denen es sich um so aufregende Dinge wie "Texthandeln" (ein pantomimisches Spiel), "Multiple Choice" (Bedeutungszuordnungsspiel oder einfach: Vokabeln abhören), "Silbenrätsel", "Zettelkasten" (Lernmethode aus der Vor-Soft-Learning-Zeit), ein Lückentext-Diktat und einige kleine Grammatikübungen handelt. Verblüfft einen da noch der wunderliche Satz im Handbuch: "Sollten Sie während einer Lektion einschlafen, so ist das nicht weiter schlimm, allerdings sollten Sie sie dann wiederholen?"

Soft Learning bietet also einfaches und komfortables Lernen; dem üblichen Lernstress stehen Ruhe und Entspannung gegenüber. Allerdings wird im Begleitmaterial an manchen Stellen das Eigenlob übertrieben. Ein wesentlicher Nachteil ist das Baukastensystem, das durch seine große Anzahl an Einzelteilen jede Übersicht erfolgreich verhindert.

Bezugsquelle:
SM Softtraining
Fasangartenstraße 4
8000 München 83

Grund- und Aufbaukurse kosten je 198.-, Intensivkurse 98.-, Zusatzwortschätze oder ein Wortschatzeditor je 49.- und die Systembasis 89.-.

Axel Klenk

Soft Learning

"Soft Learning ist die denkbar optimale Lernmethode unserer Zeit", so heißt es auf den Verpackungen der zahlreichen Teile, die notwendig sind, um das Soft-Learning-System zum Laufen zu bringen. Es sind dies

die Systembasis mit der Systemdiskette, ein Userport-Stecker mit erstaunlich kurzem Kabel zur Kopplung des Computers mit dem eigenen, handelsüblichen Audio-Cassettenrecorder mit DIN-Buchse, ein Einfüh-

Paint Boutique

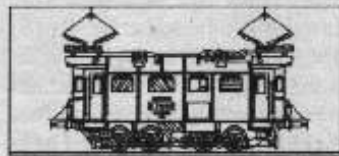
Mal- und Zeichenprogramme für Homecomputer erfreuen sich ständig steigender Beliebtheit. Dies ist nicht zuletzt dem Oldie unter diesen Programmen, dem "Koala-Painter", zu verdanken. Programme neueren Datums lehnen sich immer noch in Aufbau und Gestaltung an diese Vorlage an.

Auch das vor kurzem erschienene Programm "Paint Boutique" vermittelt auf den ersten Blick diesen Eindruck. Nimmt man jedoch die wirklich gut gelungene Dokumentation zur Hand, wird einem schnell bewußt, daß dieses Programm sehr viel mehr bietet, als man bei flüchtiger Beachtung annehmen könnte. Der Hauptunterschied zu den bisherigen Mal- und Druckprogrammen wie "Koala-Painter" oder "Printshop" besteht in der Möglichkeit der Zusammenstellung und Bearbeitung von Text und Grafik. Als besonderes Bonbon sei hier die Möglichkeit des Einlesens und Verarbeitens von Textomat-Files genannt.

"Paint Boutique" ermöglicht dem Anwender die Erstellung von beliebigen Druckerzeugnissen, wie eigener Briefkopf, Zeitung, Gruß- und Glückwunschkarten, Etiketten, Flugblätter... Die Anwendungsmöglichkeiten sind fast unbegrenzt.

Als Peripheriegeräte benötigt man ein oder zwei Diskettenlaufwerke sowie einen Matrixdrucker. Durch ein gesondertes, sehr einfach zu bedienendes Install-Programm ist es dem Anwender möglich, fast jeden Matrixdrucker anzupassen. Das Programm bietet Editoren für Text, Grafik, Zeichen und Randeinstellung des Aus-

drucks. In diesen Editoren kann der User bestehende Grafiken verändern und eigene entwickeln. Weiterhin gibt es die Möglichkeit, bereits geschriebene Texte und beliebige, maximal vierfarbige Grafiken anderer Programme einzubinden.



Das gesamte Layout wird auf acht HiRes-Seiten erzeugt, in einem gesonderten Unterprogramm zusammengestellt und kann anschließend ausgedruckt werden. Nachfolgend eine Auflistung der speziellen Features von "Paint Boutique":

- eingebauter Grafikeditor mit umfangreichem Befehlssatz,
- gesonderte Erstellung von Großsymbolen (Initialen) mit Lupenbefehl,
- drei Zeichensatzgrößen (24, 16, 8 Punkte), Zeichen beliebig erstellbar,
- beliebiger Rand, komfortabel zu erstellen (Spiegelsymmetrie),
- Texteingabe in eigenem Editor oder Übersetzung von Textomat-Files,
- Anwahl eines Formats mit bis zu 640 x 800 Punkten,
- Zusammenbau von Text und Grafik in separatem Editor,
- Verwendung beliebig vieler Zeichensätze auf einer Seite.

Zum Umfang des mit 69.- DM recht preiswerten Programms gehört weiterhin eine Grafikdiskette mit vorgefertigten Grafiken und Zeichensätzen. Eine weitere Zusatzdisket-

te kann für DM 29.- bezogen werden. Laut Data Becker werden weitere Disks in Kürze folgen.

"Paint Boutique" vereinigt die Vorteile der meisten anderen Programme in sich, ohne gleichzeitig deren Fehler zu übernehmen. Besonders her-

ausragend ist die 1a-Druckeranpassung. Auch mit Exotendruckern muß man nicht auf die Fähigkeiten dieses Programms verzichten.

System: C 64
Hersteller: Data Becker
Preis: 69.- DM (Disk)
Frank Wölk

Commodore 128 Intern

Von Gerits, Schieb, Thrun
Verlag Data Becker
512 Seiten, 69.- DM
ISBN 3-89011-098-3

Vom C 64 bereits altbekannt und bewährt, gibt es nun aus der "Intern-Serie" von Data Becker die Fortsetzung des Bestsellers "64 Intern". Geeignet ist dieses Buch für all jene, die tiefer in den Commodore 128 Einblick nehmen möchten. Gewisse Grundkenntnisse in Maschinensprache sind aber für die volle Nutzung unabdingbar, jedoch erhält auch der Neuling und Unbedarfte einige grundlegende Erklärungen zum Speicheraufbau oder den verschiedenen Betriebsarten. Wie auch bei "64 Intern" besteht das Buch nicht nur aus dem ROM-Listing, im Gegenteil, Hardwarebausteine wie der VIC-SID-Chip werden ebenso ausführlich erklärt wie der neue 8563 VDC-Chip. Daß dem Memory Management (kurz: MMU) ein eigenes Kapitel gewidmet wurde, ist mindestens ebenso positiv hervorzuheben wie die Erklärung der verschiedensten wichtigen Kernel-Routinen nebst Beispielen, um diese am besten in eigene MC-Programme einzubinden.

Etwas dürftig fiel das erste Kapitel aus, in dem die Verbindungen zur Außenwelt mittels User Port, RS-232 und Datasetten-Anschluß beschrieben werden. Die Autoren übergangen eine ausführliche Erläuterung der Schnittstellen zugunsten einer dafür um so besser ausgefallenen Beschreibung der einzelnen Chips. Begonnen wird dabei mit dem bereits aus dem C 64 bekannten VIC-Chip, der wirklich bis in die letzte Kleinigkeit erschöpfend erklärt wird. Die Registerbelegung und die verschiedenen Betriebsarten werden ebenso beschrieben wie der

Umgang mit Sprites, die Zeichendarstellung und die Programmierung von Farbe und Grafik.

Im dritten von insgesamt zehn Kapiteln wird die Ein- und Ausgabesteuerung durch den CIA 6526 erklärt. In den darauffolgenden Abschnitten werden der SID-Chip sowie der neue



8563 VDC-Chip, der den RGB-Ausgang versorgt, durchleuchtet. Der Hauptbestandteil (das ROM-Listing) ist sehr ausführlich dokumentiert, was für den durchweg hohen Informationsgehalt der einzelnen Kapitel von "128 Intern" spricht.

Auf den Hardwareaufbau und die Beiträge der einzelnen Bausteine in den verschiedenen Modi wird im letzten Kapitel kurz eingegangen. Abgeschlossen wird dieses hervorragende Buch von einigen nützlichen Tabellen, gefolgt von einem Stichwortverzeichnis. Bei einer derartigen Informationsfülle ist es leicht, zu erkennen, daß dieses praktische Arbeitsbuch und Nachschlagewerk ein absolutes Muß für jeden darstellt, der sich eingehender mit seinem 128'er beschäftigen will. Der einzige schwache Punkt ist der hohe Preis von 69.- DM, doch ist diese Investition absolut empfehlenswert.

Frank Wölk



Die beiden Deutschlands - eng umschlungen

Der Amiga: Pro und Kontra

Selten haben sich in der letzten Zeit die Geister derart an einem Computer geschieden wie beim Amiga. Von den einen als aufgetakelter Super-Spiele-Computer bezeichnet, sehen andere in ihm einen Schritt zu einer neuen Computergeneration der Ultra-Leistungsklasse. Auch unter unseren freien Mitarbeitern führte der Amiga zu sehr hitzigen und angeregten Diskussionen. Zwei freie Mitarbeiter der CK-Redaktion, die bereits über mehrere Monate intensiv mit dem Amiga gearbeitet haben, wollen hier ihre gänzlich konträre Meinung zum Ausdruck bringen und zur Diskussion stellen.

Zur Pro-Seite nimmt Thomas Tai aus Bramsche Stellung, der in der Vergangenheit für viele Softwaretests und Programmbeschreibungen in der CK und im CPC-Magazin verantwortlich zeichnete. Sein Schwerpunkt lag in der Arbeit mit Computern der Heimcomputerklasse (wenn man das heute noch so bezeichnen darf); er arbeitete aber in der letzten Zeit auch mit PCs und speziell mit dem Atari ST und dem Amiga.

Die Kontra-Position wird von Manfred Walter Thoma vertreten, der neben diversen Kursen in der CK und im CPC-Magazin speziell zu allgemeinen Themen der Computerei Stellung nahm. Als Programmierer und Systemanalytiker liegt sein Arbeitsschwerpunkt vor allem bei der Anwendung und Nutzung von Personal Computern.

Beide Autoren trafen sich zu einem kurzen Gespräch in Hamburg, das hier pauschal in Pro- und Kontra-Meinungen von Manfred Walter Thoma wiedergegeben wird. Die Autoren und der Verlag würden sich freuen, wenn Sie, die Leser, Ihre Meinung über den Amiga in Form eines Leserbriefes äußern würden.

Pro: Der Amiga ist uneingeschränkt ein Supercomputer. Seine technische Qualität sowie das äußere Erscheinungsbild sprechen für den professionellen Charakter dieses Compu-

ters. Seine hervorragenden grafischen Eigenschaften werden von dem ausgezeichneten RGB-Monitor gut wiedergegeben. Gerade die mit anderen Computern nicht vergleichbare Grafik, die in Punkto Auflösung und Geschwindigkeit wohl so ziemlich ungeschlagen ist, ordnen den Amiga in die Kategorie der Super-Grafik-Computer ein. Das eingebaute Laufwerk mit 880 KByte Speicherkapazität sowie die hohe Verarbeitungsgeschwindigkeit lassen jede denkbare Art von Anwendungsprogrammen zu. Der hervorragende Stereo-Sound rundet das Bild eines Spitzen-Computers ab.

Kontra: Der Amiga ist ein Computer der mittleren Preisklasse (3000-4000 DM), der für dieses Preisniveau einen selbstverständlichen, aber guten Eindruck hinterläßt. Von seiner grundlegenden technischen Konzeption her (Verteilung einzelner Aufgaben auf spezielle unabhängige Einheiten) kann der Amiga durchaus überzeugen, obwohl z.B. die sehr geringe Übertragungsrate zwischen Laufwerk und Rechner dieses Konzept wieder in Frage stellt. Eine Auflösung der Grafik mit 640 mal 400 Pixel muß nach dem heutigen Standard als "normal" bezeichnet werden und ist im Gegensatz zur Farbvielfalt des Amiga nicht erwähnenswert. Hervorragend ist allerdings die Verarbeitungsgeschwindigkeit der Grafik, speziell beim Füllen von Flächen. Die Bedeutung der Farbvielfalt spielt allerdings bei den meisten Anwendungsprogrammen nur eine untergeordnete Rolle. Bei Textverarbeitungs-, Kalkulations- oder Anwendungsprogrammen im allgemeinen ist der Farbmonitor eher hinderlich als nützlich. Die musikalischen Fähigkeiten sind beeindruckend, aber für die professionelle Anwendung bedeutungslos.

Pro: Durch die Benutzeroberfläche kann auch ein absoluter Computerlaie innerhalb von ca. zwei Wochen problemlos mit dem Amiga arbeiten. Durch seine grafischen Fähigkeiten lassen sich die Symbole

(Icons) zur Auswahl von Funktionen und Programmen so eindeutig darstellen, daß die Auswahl zum Kinderspiel wird. Gerade heute, wo der Benutzer mehr anwenden als selbst programmieren will, ist diese Komfortabilität besonders wichtig. Die Maus spielt dabei eine funktionell bedeutende Rolle: Mit Hilfe eines einzigen Bedienungselements läßt sich mit immer wiederkehrenden Handbewegungen jede beliebige Funktion auswählen. Der Benutzer muß nicht erst ein dickes Handbuch lesen und viele, teilweise sehr unverständliche Anweisungen lernen, sondern kann gleich den Computer benutzen.

Kontra: Die Benutzeroberfläche und die Maus bilden für den Anwender über kurz oder lang einen erheblichen Hemmschuh. Obwohl in der ersten Zeit die Benutzung der Oberfläche dem Computerlaien eine wertvolle Hilfe ist, stellen sich nach kurzer Zeit Probleme, die mit der Benutzeroberfläche nicht mehr zu lösen sind (verketten, sortieren etc.). An dieser Stelle ist der Benutzer gezwungen, sich mit dem Handbuch auseinanderzusetzen. Das liegt darin begründet, daß die Anzahl der Funktionen, die über die Benutzeroberfläche möglich sind, sehr eingeschränkt ist. Diese begrenzte Anzahl von Anweisungen könnte ein Anwender auch in zwei Wochen erlernen, ohne daß eine Benutzeroberfläche notwendig wird. Zur Maus ist zu sagen, daß sie immer auf dem Schreibtisch stört, vor allem dann, wenn wegen jedem "Pipifax" irgendwas mit der Maus bestätigt werden muß. Es gibt sehr wenige sinnvolle Anwendungen mit der Maus.

Pro: Nach der problemlosen Einführung des Benutzers in das System mit Hilfe der Benutzeroberfläche steht dem erfahrenen Anwender ein Kommandointerpreter (CLI) zur Verfügung, der keine Wünsche mehr offen läßt. Es sind Möglichkeiten ähnlich wie unter MS-DOS vorhanden. Der Amiga bietet also für jeden Anwender das Richtige.

Kontra: Der Kommandointerpreter ist ein sehr mächtiges Werkzeug, auf das der Anwender über kurz oder lang zurückgreifen muß, um all seine Probleme zu lösen. Benutzt er allerdings den CLI, so liegt die Benutzeroberfläche ballastartig brach.

Pro: Mit Hilfe der Maus und der Benutzeroberfläche lassen sich Anwendungsprogramme sehr einfach bedienen. Der Anwender kann, schon bedingt durch das Konzept, dem ein derartiges Programm unterliegen muß, alle relevanten Funktionen über die Maus einleiten. Die Einarbeitung in komplexe Programme ist daher sehr schnell möglich. Auch diejenigen Benutzer, die ein Programm sehr unregelmäßig benutzen, finden immer schnell wieder den Einstieg. Hier ist ein unverkennbarer Vorteil zu finden.

Kontra: Das Angebot an kommerziell nutzbaren Anwendungsprogrammen ist derzeit noch nicht so groß, daß man hier ein abschließendes Urteil fällen kann und darf. Allerdings bezweifle ich, daß z.B. ein Textverarbeitungsprogramm durch die Benutzung einer Maus bzw. der Benutzeroberfläche einfacher erlernbar oder sinnvoller nutzbar wäre. Die Maus wird wieder zu einem relativ nutzlosen und platzraubenden Objekt. Die traditionelle Benutzung (z.B. der Cursor-taste) ist m. E. vorzuziehen.

Pro: Die Maus kann beim Amiga jederzeit durch die Tastatur simuliert bzw. ersetzt werden!

Kontra: Wozu dann die Maus?

Pro: Wenn das Angebot an spezieller Software für den Amiga nicht reicht, kann man mit dem "SideCar" auch Standard-Software unter MS-DOS auf dem Amiga benutzen. Das bedeutet, daß neben den hervorragenden Programmen, die speziell auf den Amiga zugeschnitten sind und dessen Eigenschaften nutzen, auch Standardprogramme aus der MS-DOS-Palette verfügbar sind.

Kontra: Hier liegt doch der Hase im Pfeffer! Zum einen wurde von Commodore ein Rechner entwickelt, der bewußt nicht MS-DOS-kompatibel ist, um größere Chancen für den Absatz neu entwickelter Software und Hardware zu erzielen. Zum anderen ist der MS-DOS-Markt z.Z. so dominant, daß man nicht umhinkommt, irgendwo MS-DOS-kompatibel zu sein. Das wurde inzwischen bemerkt, so daß der "SideCar" nach der Markteinführung den Amiga kompatibel macht. Und das für einen Aufpreis von etwa 2000.- DM!! Dafür bekommt man schon bald einen IBM-kompatiblen Computer! Für weniger Geld könnte man z.B. - wenn man will - einen IBM-Kompatiblen mit einer Benutzeroberfläche ausstatten (z.B. MS-WINDOW). Was soll's also?

Pro: Durch die schnelle grafische Darstellung mit einer Auflösung von 640 mal 400 Punkten stellt der Amiga ein sehr preiswertes Werkzeug zur Erstellung von Konstruktionszeichnungen bzw. Zeichnungen im allgemeinen dar. Im besonderen können sich jetzt auch kleinere Ingenieurbüros oder Architekten ein CAD-System leisten, das ihren Ansprüchen genügt. Zum anderen zeigen Werbefirmen sowie Firmen, die sich mit der grafischen Animation beschäftigen (z.B. das Fernsehen) ein großes Interesse am Amiga. Für wenig Geld ein grafisches System mit hervorragenden Eigenschaften!

Kontra: Ohne Frage liegt hier die Stärke des Amiga. Fraglich bleibt nur, wie er in diesem Bereich angenommen wird und inwieweit das Softwareangebot den Kunden zufriedenstellt. Hier bleibt abzuwarten, wie leistungsfähig und vor allem wie billig die CAD-Programme werden, um einen guten Durchsatz gegenüber den verbreiteten CAD-Programmen (z.B. AutoCAD, MEGA-CAD) zu erzielen.

Pro: Nicht zuletzt seien die besonderen musikalischen Fähigkeiten des Amiga erwähnt. Durch den Einsatz einer MIDI-Schnittstelle lassen sich beliebige Synthesizer steuern, die einen professionellen Einsatz auf

Bühne und Studio ermöglichen. Was der Amiga schon "von Haus aus" an Soundmöglichkeiten und Kompositionsvielfalt bietet, ist beeindruckend. Durch geeignete Software ist eine ideale Kompositionshilfe für Musiker vorhanden. Die durch Software erzeugte und durchaus verständliche Sprachausgabe läßt auch Anwendungen für Behinderte (z.B. Blinde) zu.

Kontra: Die genannten musikalischen Aspekte des Amiga sind Randerscheinungen und nur für einen sehr kleinen Prozentsatz der Anwender sinnvoll. In der Regel reicht ein einfaches "Piep" bei Anwendungsprogrammen zur Kennzeichnung von Fehleingaben aus. Inwieweit die Möglichkeit der Sprachausgabe einen Computer blindengerecht macht, sei dahingestellt (neben Aspekten der Ergonomie des gesamten Systems). Es bleibt aber zu hoffen und abzuwarten, ob Softwarefirmen sich in breiterem Maße mit derartigen Projekten auseinandersetzen, um auch für den blinden Computerfan ein System zu entwickeln, das zu erschwinglichen Preisen im privaten Rahmen erhältlich ist. Das wäre ein Schritt nach vorne, aber nicht nur eine Forderung an den Amiga.

Pro: Resümee. Der Amiga ist einer der leistungsfähigsten Computer der neuen Generation. Durch seine moderne Architektur, seinen leistungsfähigen Prozessor und durch sein ausgewogenes Gesamtkonzept ist er richtungsweisend für die Zukunft. Dieses Konzept wird sich kurz- bzw. mittelfristig als Computer für den Anwender durchsetzen. Die Vielfalt der Möglichkeiten wurden bis jetzt noch nicht erschöpfend bearbeitet, so daß hier in der nächsten Zeit neue Perspektiven deutlich werden. Der Wunsch vieler Anwender an ein einfaches und vor allem bedienungsfreundliches System wird das Konzept des Amiga zu großer Beliebtheit und Popularität führen. Daneben werden speziell die grafischen Fähigkeiten neue Maßstäbe setzen, an denen sich zukünftige Produkte zu messen haben. Der Amiga ist eindeutig ein Produkt der neuen PC-Klasse, die alle Elemen-

te eines "Homecomputers" (Musik, Sprites, Joystick etc.) zusätzlich beinhaltet und von daher auch zur Entspannung und nicht nur zur Arbeit dienen kann.

Kontra: Resümee. Der Amiga ist ein Zwitter zwischen Home- und Personal-Computer. Für einen Homecomputer ist er allerdings mit ca. 3500.- DM bei weitem zu teuer und daher nur für den zahlungskräftigen Hobbyisten finanzierbar. Im großen und ganzen bietet der Amiga einige interessante Aspekte, die allerdings z.Z. noch recht unausgewogen erscheinen. Er ist weder Fisch noch Fleisch. Rüstet man den Amiga nach kommerziellen Ansprüchen auf (Zweitlaufwerk, größerer Arbeitsspeicher, SideCar) gelangt man in eine Preiskategorie, die zwischen der PC- und AT-Klasse liegt. Gerade heute, wo IBM-

Kompatible zwischen 3000.- und 4000.- DM erhältlich sind, die mit einer Grafikkarte, zwei Laufwerken oder einem Laufwerk plus Festplatte und einem freien Arbeitsspeicher von 640 KByte ausgestattet sind, fällt die Entscheidung zugunsten des Amiga sehr schwer. Der Vorteil der IBM-Kompatiblen liegt eindeutig in der universellen Nutzbarkeit, was nicht zuletzt an der gigantischen Anzahl der verfügbaren Software liegt. Hat man als Anwender allerdings ein ganz spezielles Problem, so kann hier der Amiga Vorteile bieten. Das muß aber von Fall zu Fall entschieden werden. Ich persönlich sehe keine Veranlassung, auf den Amiga umzusteigen, und würde mich auch heute noch für einen IBM-Kompatiblen in der Preisklasse von 3000-4000.- DM entscheiden.

Manfred Walter Thoma

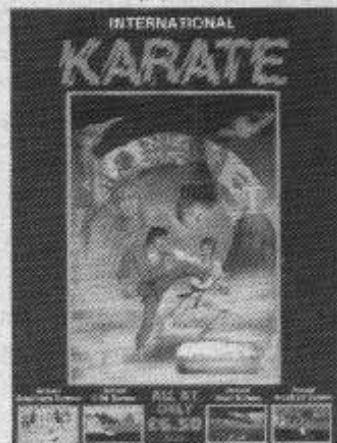
International Karate

"International Karate" ist weitgehend an "The Way of the Exploding Fist" von Melbourne House angelehnt. Die Grafik unterscheidet sich in der Qualität kaum von der des Vorbildes, dagegen zeigt sich die Sounduntermalung wesentlich aufwendiger und einfallsreicher, und auch die Kampfgeräusche klingen ziemlich realitätsnah. Dem Spieler stehen sechzehn Bewegungsmöglichkeiten bzw. Schlagvarianten zur Verfügung, die mit dem Joystick gelenkt werden. Die Bedienung weicht nur leicht von der von "Exploding Fist" ab.

Ziel von "International Karate" ist es, Karatekämpfe auf allen fünf Kontinenten erfolgreich zu bestreiten, um schließlich den schwarzen Gürtel zu erhalten. Nach jeweils drei gewonnenen Kämpfen wird der nächsthöhere der sechs möglichen Gürtel (weiß, gelb, grün, lila, braun, schwarz) verliehen. Auf den Reisen zwischen den Kontinenten absolviert der Spieler zwei zusätzliche Prüfungen und verdient sich damit Bonuspunkte. Während eines Kampfes werden vom Kampfrichter (je nach Ausführung eines erfolgreichen Angriffs) halbe oder ganze Punkte verteilt.

Der Kämpfer, der innerhalb von dreißig Sekunden die meisten Wertungen oder zuerst zwei volle Punkte erzielt, ist Sieger.

Neben dem Spiel gegen den Computer ist natürlich auch der



Kampf zweier Karatekas gegeneinander vorgesehen, wobei der Sieger dann gegen den Computer antreten muß.

"International Karate" ist eine der besten Karatesimulationen für den Commodore 64.

System:

C 64, Schneider CPC, Spectrum, Atari

Hersteller:

Activision

Preis: Cas. 25.-, Disk 39.-
job

CK-PROGRAM

SINCLAIR

Paint (nicht veröffentlicht), Pyramide (7/84), Superhim (9/84), Drawer (9/84), Säulendiagramm (10/84), Große Buchstaben (10/84), Farben beim Spectrum (10/84), Prodomo (11/84), Toolkit (12/84), Libelle (12/84), 3-D Schrift (12/84), Neuer Zeichensatz (12/84), Krümelmonster (1/85), Fast L/S (1/85)
Best.-Nr. S 1

Puzzle (4/85), Sprites mit Demo (3/85), Darts (5/85), Uhr (7/85), Roulette
Best.-Nr. S 10

Catalog (2/85), Solitaire (2/85), Fillroutine (2/85), Computer Figures (2/85), Ku Bernd (2/85), Sterngrafik (2/85), Manic Train (3/85), Senso (4/85)
Best.-Nr. S 11

Schattler (5/85), Spectrum-Infosystem (5/85), Cowboy (6/85), Player's Dream (6/85), Tape (6/85), Soundexaminer (6/85), Hovercraft (7/85)
Best.-Nr. S 20

ATARI

Lunar Lander (12/84), Car Race (7/84), Turbo Worm (1/85), Munsterjagd (3/85), Bewegte Grafik (3/85), Digger (2/85), 15 und 3 (4/85), Bundesligasimulation (3/85), *3-D Laby (10/84), Zeichensatzeditor (2/85), Mini-Trickfilmstudio (9/85), Rolly Dolly (11/84), *Musik-Editor (4/85), HELPI nur bei Kassettenversion, *Programme sind nur mit Erweiterung lauffähig.
Best.-Nr. A 10 Diskette, Best.-Nr. A 10a Kassette

Sound-Demo I (5/85), Sound-Demo II (nicht veröffentlicht), The Run and Jump Construction Set (6/85), Bank Panik (7/85) nur mit Erw., Funktions-Plötter (5/85), Blockade (9/85), Jewel Eater (5/85), Zellen-Assembler (7/85), Joystick-Controller (9/85), Horizontales-Scrolling (5/85), *Converter (DOS III in DOS II) (9/85), *Nur auf Diskette
Best.-Nr. A 11 Diskette, Best.-Nr. A 11a Kassette

TI 99/4A

Burglar Time (12/84), Cowboy (7/84), Desert Flight (9/84), Fassadenkletterer (11/84), Hangman (nicht veröffentlicht), Miner-Pat (1/85), Nova-Madaga (1/85), Parachute Jumper (5/84), Permanente Kleinbuchstaben (10/84), Pokellistengenerator (12/84), Screen Utilities (11/84), Burglar Time (12/84), Cowboy (7/84), Desert Flight (9/84), Fassadenkletterer (11/84), Hangman (nicht veröffentlicht), Miner-Pat (1/85), Nova-Madaga (1/85), Parachute Jumper (5/84), Permanente Kleinbuchstaben (10/84), Pokellistengenerator (12/84), Screen Utilities (11/84)
Best.-Nr. TI 1 Diskette, TI 1a Kassette

Alpha Lock (2/85), Cube (3/85), Eponst (4/85), Jungler (4/85), Macropede (4/85), Merge-Flir (3/85), Motor ON (2/85), Pooyan (2/85), Progload (3/85), Rotation (3/85), Vokabel (2/85), Achtung: Macropede, Merge-Flir und Progload nur auf Diskette.
Best.-Nr. TI 10 Diskette, Best.-Nr. TI 10a Kassette

Ballade pour Adeline (nicht veröffentlicht), Rugalmulator (6/85), *SuperDisk-Katalog (6 und 7/85), Fluch des Pharaos (6/85), Plot (6/85), Säulendiagramm (5/85), Calculator (7/85), Texter (6/85), Würfel-Duell (7/85), *Nur auf Diskette
Best.-Nr. TI 11 Diskette, Best.-Nr. TI 11a Kassette

*Goto/Jump (10/85), Cavern (9/85), Crazy Man (11/85), Devil Quest (10/85), Eklent-Star (11/85), High Res. Grafik mit Demo (11/85), Soundeditor (10/85), *Sprite-Utilities (11/85), *Sektor 0 (9/85), *Nur auf Diskette
Best.-Nr. TI 12 Diskette, Best.-Nr. 12a Kassette

COMMODORE

Duell (7/84), Mäuserennen (9/84), Speicherplatzanzeige (10/84), Basic-Erweiterung (10/84), Through the wall (11/84), Maze-Ball (11/84), Prüfummgengenerator und Indikator (11/84), Grafik-Erweiterung (12/84), Bierkiste (12/84), Phalanx (1/85), Nürburgring (1/85)
Best.-Nr. C 1 Diskette DM 20,-, Best.-Nr. C 2 Kassette DM 16,-

Handballmanager (3/85), Defender (2/85), Ghosts (4/85), Pugno (4/85), Delete (4/85), Merge (2/85), Find (2/85), Screen-Designer (2/85), Scr.Des.Obj C000 (2/85), Data-Gen (2/85), Rahmenfarben (3/85), Auto-Starter 1.0 (4/85), Code 64 (5/85), File-Load (5/85), File-Save (5/85)
Best.-Nr. C 10 Diskette DM 20,-, Best.-Nr. C 10a Kassette DM 16,-

Die Grafik-Erweiterung Teil I und II sowie die Basic-Erweiterung EXB V 1.8 kann auf Diskette / Kassette inklusive Anleitung bezogen werden. (Siehe auch CK-Hefte 10/84, 12/84 und 5/85)
Best.-Nr. D585 Diskette DM 13,-, Best.-Nr. K585 Kassette DM 10,-

FREIE AUSWAHL

3er Pack

3 Kassetten nach Ihrer Wahl nur **40,-**

3 Disketten nach Ihrer Wahl nur **50,-**

5er Pack

5 Kassetten nach Ihrer Wahl nur **60,-**

5 Disketten nach Ihrer Wahl nur **75,-**

MM SERVICE

Paint (9/85), Window (9/85), 3-D Plotter (10/85), Superlist (10/85), Centipede (10/85), Nitro Joe (11/85)

Best.-Nr. 5 30

Höllenzur (1/86), The Servant (1/86), Snake-Smasher (3/86), Assi (3/86), List/LLIST (3/86), Programmzellen löschen (3/86), Spectrum-Quick-copy (3/86), On Error Goto (3/86)

Best.-Nr. 5 40

Ass. Tips Uhr (7/86), GLP-Druckerroutine (7/86), Datenverschlüsselung (7/86), Pro Dos (7/86), Apfelmännchen (7/86), Tele (7/86), Dishmaster (5/86), Spectrace (5/86)

Best.-Nr. \$ 50

Kassette
16,--

Diskette
20,--

Kassette
16,--

DL Designer 64K (10/85), Joypoint 64K (10/85), Musiccreator 64K (11/85), Chefredacteur 64K (1/85), Unprotector V1.0 16K (1/85), Key Maker 16K (1/85)

Best.-Nr. A 12 Diskette, Best.-Nr. A 12a Kasette

Cherry Harry (3/86), Mission X auf dem Altar (5/86), *Basic-Erweiterung (5/86), Mini-Billard (10/85), Zeichen-Zauberei (3/86), Sound-Demo (3/86). *Nur auf Diskette

Best.-Nr. A 13 Diskette, Best.-Nr. A 13a Kasette

*Revolver Kid (1/86), *Pys-Dos (7/86), *Text im Grafikfenster (7/86), Rollerball (7/86), Kung Fu (9/86), *Disk-Menü (9/86), Titan (9/86).
*Nur auf Diskette

Best.-Nr. A 14 Diskette, Best.-Nr. A 14a Kassette

Diskette
20,--

Kassette
16,--

Bierbörse (3/86), Blüchem (5/86), * Copy 5 (1/86), Hardcopy für Seikosha CP 100 (1/86), Lander (3/86), Moondriver (5/86), Polargrafik (3/86),
Ti-Point (3/86), 40-Zeichen-Mode (3/86), Turtle Jumper (1/86). * Nur auf Diskette

Best.-Nr. TI 13 Diskette, Best.-Nr. TI 13a Kassette

TI-Assemblerdiskette 1 Für die TI-Assembliertrecks haben wir auf vielfachen Wunsch eine Diskette mit den Quellcodes der übertragenen Assemblerprogramme Macropede (4/85) und Super Disk Catalog (6 und 7/85) zusammengestellt. Zum Ändern und Lernen.

Best.-Nr. TI 14 Diskette

Ti-Assemblerdiskette 2 Diese Diskette enthält die Quellcode-Dateien für die Assemblerprogramme Adressdatei (4/86) und Disk-Utilities (8/86)

Best-Nr. TI 15 Diskette

* Willy der Wurm (7/86), Seikoshia GP 50 Hardcopy (7/86), Tarzan (7/86), * Adressdatei (5/86), Soundeditor plus (9/86), Screen Editor (7/86), * Assembler-Grafik-Generator (7/86), The Plague (9/86), * Disk Utilities (9/86). * Nur auf Diskette

Best.-Nr. TI 16 Diskette, Best.-Nr. TI 16a Kassette

**Sonder-
für alle
modore
ten und
erten**

Balloon (7/85), Rescue in the Stone-Age (6/85), Solitaire (7/85), Sprite Editor (9/85), Data Loader (6/85), Rahmenroutine (7/85)

Best.-Nr. C 11 Diskette DM 20,-, Best.-Nr. C 11a Kasette DM 16,-

Raddish One (10/85), 3-D-Hubschrauber Simulator (11/85), Head to Head Biathlon (1/86)

Best.-Nr. C 12 Diskette DM 13,-, Best.-Nr. C 12a Kasette DM 10,-

Formel One (3/86), Directory in Farbe (3/86), Raddish In the Jungle (5/86)

Best.-Nr. C 13 Diskette DM 13,-, Best.-Nr. C 13a Kasette DM 10,-

Best.-Nr. C 14 Diskette DM 13,-, Best.-Nr. C 14a Kassette DM 10,-

[illegible]

SOFTWARE-BESTELLSCHEIN

Coupon ausschneiden und einschicken an:

Verlag Rätz/Eberle CK-Software
Postfach 1640-7518 Bretten

Anzahl	Bestell-Nr.	Jer/Ser-Pack	Preis	Name des Bestellers	
				Straße	PLZ/Ort
				Telefon	Datum/Unterschrift
				Ich wünsche folgende Bezahlung:	
				☐ Nachnahme (zuzüglich DM 5,70 Porto- und Versandkosten)	
				☐ Vorauskasse (keine Versandkosten) Bitte Scheck beilegen oder auf Postcheckkonto Karlsruhe 43429-756 überweisen.	



INFOS - NEWS - BERICHTE - REVIEWS - TESTS - PROGRAMME - LISTINGS - ALLES FÜR ATARI-USER

Hallo Atarianer!

Im Mittelpunkt dieses Heftes steht "MicroMon", unser Top-Programm des Monats, das zum ersten Mal nur für Cassette ausgelegt ist! Es handelt sich dabei um einen Maschinensprachemonitor, der die ersten Einblicke in die Assemblerprogrammierung des Atari geben kann. Die geschickte Programmierung (z.B. liegt der Monitor unter dem Betriebssystem und nimmt daher keinen Speicherplatz weg) hat uns bewogen, MicroMon zum Top-Programm dieser Ausgabe zu machen.

Als Ausgleich finden die Disketten-User ebenfalls ein MS-Programm, den Disk-Collector von unserem Stammleser Stefan Funk. Mit diesem sehr hilfreichen Programm lassen sich komfortabel alle Files verwalten, die bei nachlässiger Beschriftung oft stundenlange Suchaktionen erfordern.

Gleich zwei Spiele haben wir für die Freunde der "leichten Unterhaltung": das Basic-Spiel "Goff" und meinen in Assembler geschriebenen "Puzzler".

Als "ernsthafte Anwendung" (was immer das sein mag) finden Sie ein Karteikartenprogramm und das zweite Listing zu unserem MIDI-Interface-Bauplan. Und in Peters Assembler-Ecke dreht sich alles um den Display-List-Interrupt, kurz DLI genannt. Vielleicht werden einige sich gleich mit dem MicroMon versuchen.

Nun zu unserem großen ST-Teil, den wir in Zukunft weiter ausbauen werden. Hier findet der interessierte ST-User wie gewohnt Softwaretests und Buchbesprechungen aller Kategorien. Unseren Bericht über das hervorragende GfA-Basic möchte ich an dieser Stelle besonders erwähnen. Wir beabsichtigen, in Zukunft für den ST vor allem Programme in diesem (wie wir meinen besten) Basic-Dialekt zu veröffentlichen. Wir warten also gespannt auf Listings und Tricks mit/über GfA-Basic.

Bis zur nächsten CK verbleibe ich als Euer
Thomas 1000

Satte 320 KByte für den Atari 800 XL

Der ST hat es gezeigt: Mehrere 100 KBytes bis zu einem Megabyte Speicher für den häuslichen Computer sind heute kein Traum mehr. Moderne Speicherchips erlauben auf geringstem Raum riesige Speichermengen. Nun sind die Preise dieser Chips so weit gefallen, daß sie auch für Home-Computer interessant werden. So kann

man mit der neuen Erweiterung des Compy-Shops ganze 256 KByte in seinen 800 XL einbauen, die sich mit den bisherigen 64 K zu respektablen 320 KByte addieren. Ganz nebenbei wird der 800 XL dann auch kompatibel zum 130 XE.

Die Speichererweiterung ist auf einer doppelseitig beschich-

teten Platine sauber untergebracht. Sie ist kompakt aufgebaut und findet daher im Inneren des 800 XL reichlich Platz. Zum Einbau müssen zwei Chips von der Computerplatine entfernt und in die Speichererweiterung eingesetzt werden. Anschließend wird die Erweiterung in die leeren Sockel eingesteckt. Jetzt wird's schwieriger, denn nun warten 16 Drähtchen auf jemanden, der geschickt mit dem Lötkolben umgehen kann. Wer sich das nicht zutraut, kann die Erweiterung auch gegen Aufpreis beim Compy-Shop einbauen lassen.

Mit Hilfe eines Schalters, der in einem Lüftungsschlitze des Gehäusedeckels montiert wird, kann man den Zusatzspeicher ein- und ausschalten. Man ist somit auch gegen Programme gewappnet, die sich allergisch zeigen sollten und mit der Erweiterung nicht mehr funktionieren wollen. Die zusätzlichen 256 K machen den 800 XL kompatibel mit dem 130 XE. Damit ist sichergestellt, daß der Speicher von zukünftigen und schon vorhandenen Programmen (z.B. Typesetter oder Basic XE) angenommen wird.

Die Software

Das ist dann auch gleich der Kern der Sache. Denn was nützt Ihnen der schönste Speicher, wenn Sie nichts damit anfangen können? Sie dürfen nämlich nicht erwarten, daß sich mit dem Einbau auch der unter Basic nutzbare Speicher erhöht. Nein, hier bleibt es bei den bisherigen ca. 30 bis 38 KByte, je nach DOS und Basic. Auf jeden Fall kann man die 256 KByte aber als RAM-Disk verwenden und somit auch umfangreichere Werke darin auf-

bewahren. Freundlicherweise bekommt man gleich eine Disk mit einer überarbeiteten Version von DOS XL zu der Erweiterung. Dort findet man ein passendes RAM-Disk-Programm, das den Zusatzspeicher in 960 Sektoren zu je 256 Bytes einteilt. Natürlich kann man auch die RAM-Disk von DOS 2.5 nehmen, aber die bietet nur knappe 500 Sektoren zu 128 Bytes.

Die Speichererweiterung kostet komplett 248.- DM. Wer sie einbauen lassen will, muß noch einmal dreißig Mark drauflegen. Diese Investition lohnt sich bestimmt, wenn man viel mit Compilersprachen oder Assembler arbeitet und eine große RAM-Disk das Arbeiten vereinfacht.

Bezugsquelle:
Compy-Shop
Gneisenastraße 29
4330 Mülheim

Peter Finzel

Atari-Club

Wir, das sind 2 Atari-Freaks im Alter von 13 und 15 Jahren, haben einen Atari-Club gegründet und wollen selbstgeschriebene Software untereinander austauschen. Auch Tipps geben wir bekannt. Bitte legt einen frankierten Umschlag bei, wenn ihr Fragen habt und uns schreibt. Einen Clubbeitrag braucht ihr voraussichtlich nicht zu bezahlen. Nur wenn ein Clubmagazin (auf Diskette) erscheint, müßt ihr dieses bezahlen. Bitte schreibt an die Adresse, die euch am nächsten ist.

Volker Wiebe	oder
Data-Light	Tobias Schuster
Heidschnuckenweg 7	Data-Light
2120 Lüneburg	5481 Burg Sülze

Neues für die Atari-Spielebox

Eigentlich konnte man kaum noch Besseres erwarten. Schon die erste Ausgabe von "Spy vs. Spy" überzeugte durch das ausgefeilte Spielprinzip und durch farbenfrohe Bilder. Der Nachfolger dieses Spieles mit dem Untertitel "The Island Caper" ist nun schon einige Zeit auf dem Markt und erfreut sich ebenfalls steigender Beliebtheit. Kein Wunder, denn hier wurde wirklich gute Arbeit geleistet. Es handelt sich nicht um eine schlichte Neuauflage des alten Spielprinzips, vielmehr wurden neue Elemente mit aufgenommen und unglaublich viel Wert auf Details gelegt.

Spy vs. Spy II

Ganz kurz zur Story. Die zwei bekannten Spione sind mit Flugzeugen zu einer einsamen tropischen Insel unterwegs, auf der eine streng geheime Rakete niedergegangen ist. Nachdem beide in bester Agentenmanier mit dem Fallschirm auf der Insel gelandet sind, beginnt das eigentliche Spiel. Schwarzer und weißer Spion machen sich auf die Suche nach den drei Teilen der Rakete, die gefunden und montiert werden müssen. Danach muß das U-Boot gefunden werden, das den erfolgreichen Spion von der Insel transportiert. In bester 007-Manier gibt's beim Einstieg in das U-Boot erst mal ein Küßchen einer holden Schönheit.

Wer auch nur einmal die bekannten Comics des MAD-Magazines gelesen hat, der weiß, daß es bei der Suche nach den Raketeanteilen zu manchem

Hinterhalt kommen wird. Der smarte Spion erledigt so etwas mit seinem Trupplator (neuestes Modell FSS 85), mit dem Gruben ausgehoben und Kosmos-Bomben gelegt werden. Nur darf man das alte Sprichwort nicht vergessen, das da lautet: Wer anderen eine Grube gräbt...

Wie auch in der ersten Ausgabe von "Spion gegen Spion" können zwei Spieler dank geteiltem Bildschirm gleichzeitig gegeneinander antreten. Wenn man keinen Mitspieler greifbar hat, übernimmt der Computer den Part des Gegners. Neu ist hier, daß die Screens nicht mehr statisch sind, sondern unabhängig scrollen. Daneben ergibt sich auch noch ein netter 3D-Effekt, wenn man zwischen den Bäumen hindurchläuft. Grafik und Animation sind durch und durch hervorragend programmiert. Besonders überzeugt die Liebe zum Detail, die z.B. im Titelbild und in den Schlussszenen reichlich zum Ausdruck kommt. Gesamturteil: Sehr empfehlenswert.

Knockout

Videospiele rund um den Boxsport sind in letzter Zeit sehr populär geworden. Leider hat sich bisher noch niemand die Mühe gemacht, ein solches Spiel auf den Atari zu übertragen. Verständlicherweise waren daher die Erwartungen hochgespannt, als "Knockout" von Kingsoft als erster Vertreter dieser Gattung angekündigt wurde. Doch gleich vorweg: Die Erwartungen haben sich nicht so ganz erfüllt. Das beginnt schon mit dem Titelbild, das grafisch eher mittelmäßig einzuordnen ist. Und auch während des ganzen Spieles wird man den Gedanken nicht los, daß hier das grafische Potential des Computers nicht ausgenutzt wurde.

Vor Beginn des Spieles erscheint eine Hi-Score-Liste, die nach jedem Spiel auf der Diskette verewigt wird. Dann geht es richtig los; erster Gegner im Ring ist "Macho Macaroni". Der vom Joystick gesteuerte Boxer wird als Drahtgitter

räumlich vor dem Gegner dargestellt, so daß man die Möglichkeit hat, sowohl seine Schläge als auch die des Gegners zu sehen. Im oberen Teil des Bildschirms werden Punkte gezählt und der Zustand beider Boxer angezeigt. Sinkt eine der beiden Kraftskalen auf Null, so fällt der Boxer zu Boden und wird ausgezählt. Wenn er Glück hat, wird der Kampf fortgesetzt. Auch eine Möglichkeit zum k.o. ist vorgesehen: Jeder gelandete Treffer erhöht den k.o.-Balken, jeder Schlag, den man einstecken muß, verkürzt den Balken. Sobald dieser den rechten Rand erreicht hat, ist der Gegner k.o.

Ist der Gegner ausgezählt, wird der nächststärkere von Disk geladen und der Kampf beginnt von vorne. Ob Sie's jemals bis zu "Smart Stallone" schaffen werden? Gesamturteil: Mittelmaß.

Racing Destruction Set

Hand auf's Herz: Wer ist (oder zumindest war) nicht von einer Carrerabahn mit flitzenden Modellautos fasziniert? Genau in diese Lücke stößt Ariolasoft mit dem "Racing Destruction Set", das vom amerikanischen Partner Electronic Arts übernommen wurde. Hier kann man sich eine eigene Rennstrecke basteln, sogar ein Auto selbst zusammenstellen, und dann geht's ab durch die Mitte, entweder zu zweit oder gegen den Computer. Also wieder eines jener berühmten "Construction Sets", bei denen man alles selbst machen soll? Keine Sorge, auf der beidseitig bespielten Diskette sind ca. 50 fertige Rennstrecken enthalten, die man sofort ohne großartigen Aufwand verwenden kann.

Für all diejenigen, die gerne selbst Hand anlegen wollen, bietet der "Baukasten" eine reiche Auswahl an Elementen, wie z.B. Kurven, Kreuzungen oder Sprünge, die zu einer abwechslungsreichen Strecke verarbeitet werden können. Der Editor funktioniert recht praktisch und prüft die Rennstrecken immer gleich nach der Ein-

gabe, ob auch alle Elemente zueinander passen. Die beste Eigenschaft dieses Baukastens ist der 3D-Effekt, mit dem die Rennstrecke etwas räumliche Tiefe erhält. Dadurch sind auch Berg- und Talstrecken möglich.

Nach der Rennstrecke muß man sich ein passendes Gefährt zusammenstellen. Man hat die Auswahl zwischen neun Typen, die vom Rennwagen über einen Jeep bis zum Mondauto (!) reichen. Wem Motorisierung und Reifen nicht gefallen, kann unter einer Reihe von Alternativen auswählen. Nachdem zwei Autos ausgewählt sind, geht es zum Rennen. Die Piste wird in einem gesplitteten Bildschirm in schräger Aufsicht dargestellt, und, ähnlich wie beim oben vorgestellten "Spy vs. Spy", können sich zwei Spieler gleichzeitig beteiligen.

Hier endet aber auch schon die Gemeinsamkeit mit dem Agentenspiel, denn die Bildschirmgrafik des Rennens läßt viel zu wünschen übrig.

Das Rennen kann man im herkömmlichen Sinne auf Zeit, aber auch auf möglichst viele Crashes fahren (darum Destruction). Dann gewinnt derjenige, dessen Fahrzeug am längsten heil bleibt. Naturgemäß gibt es in diesem Modus auch die Möglichkeit, den anderen zu behindern (z.B. Ölspur) und am eigenen Fahrzeug einen Rammschutz anzubringen. Sieht man einmal von der Grafik des Rennens ab, so überzeugt das Spiel durch seine vielfältigen Möglichkeiten, die auch in der deutschen Anleitung gut beschrieben sind. Durch den Nachteil der Grafik lautet das Gesamturteil nur "gut".

Peter Finzel



1 Mb Laufwerk f. ATARI ST 496.-
Disk 3.5" 2D ab 55.-
Grünmonitor f. ATARI ab 298.-
unicorn soft
3500 Kassel
u. Königstr. 46A
T. 0561/770367

Atari 600XL/800XL/130XE-User!
Lichtgriffel mit Programmen und
deutscher Anleitung nur 49.- DM.
Versand gegen Scheck/Nachnahme.
Info gratis! Firma Schilbauer,
Postfach 1171A, 8458 Sulzbach, 84
09661/6592 bis 21.00 Uhr.

PETER'S ASSEMBLERECKE

Display-List Interrupts

PM-Grafik, Scrolling und Page-Flipping sind alles Techniken, mit denen Atari-Computer vor einigen Jahren Meilensteine gesetzt haben. Doch das I-Punktchen des grafischen Potentials sind die Display-List-Interrupts (kurz DLI). Mit deren Hilfe können noch mehr Farben, noch mehr bewegte Objekte oder auch mehrere Zeichensätze gleichzeitig dargestellt werden.

Diese DLIs haben nur einen Nachteil: Sie sind ungemein kritisch in der Ausführungszeit und daher ausschließlich den Assemblerprogrammierern vorbehalten. Das ist natürlich ein Grund für die Assembler-ecke, sich dieses Themas anzunehmen. Den Anfang machen zwei Beispiele. Später werden wir uns mit dem Timing der DLIs befassen, ein Thema, das besonders Leute mit etwas Assembler-Erfahrung interessieren wird. Zunächst aber ein paar Grundlagen für alle, die sich mit Interrupts und Rasterzeilen noch nicht so auskennen.

Sie wissen ja sicherlich, daß von Ihrem Computer erzeugte Bild besteht aus 50 Einzelbildern pro Sekunde. Aber auch jedes dieser Einzelbilder (Frames) wird nicht im Ganzen ausgegeben, sondern von einem Elektronenstrahl in einzelnen Rasterzeilen auf den Bildschirm gezeichnet. Nur durch die Trägheit des Auges und der

Nachleuchtdauer des Bildschirms sehen wir ein stehendes, meist flimmerfreies Bild.

Für die Grafikprogrammierung braucht man eine Möglichkeit, den Prozessor an die Erzeugung des Bildes zu koppeln. Damit vermeidet man Bildstörungen (Flackern etc.), und Bewegungen werden gleichmäßig und flüssig. Beim Atari wurden zu diesem Zweck zwei Interrupts verwendet. Der eine tritt auf, wenn die Ausgabe eines Frames gerade beendet ist und der Elektronenstrahl zu seinem Ausgangspunkt zurückkehrt. Wir kennen ihn bereits aus früheren Assembler-eken; es ist der VBI. Der zweite Interrupt, der DLI, läßt sich bereits während des Bildaufbaues auslösen.

Es dürfte inzwischen bekannt sein, daß die Grafik des Atari durch einen eigenen Videoprozessor, den Antic, aufgebaut wird. Auch dieser Prozessor hat ein eigenes Programm (die Display-List), in dem festgelegt wird, welcher Speicherbereich in welchem Modus darzustellen ist. So besteht die Display-List eines GRAPHICS-0-Bildschirms u.a. aus 24 Anweisungen zur Darstellung einer Textzeile. Man kann nun bei jeder Antic-Anweisung ein Bit setzen (das höchstwertigste), so daß der Videoprozessor einen Display-List-Interrupt auslöst.

Die Möglichkeiten einer solchen Einrichtung sind enorm. Da der Interrupt auftritt, während der Elektronenstrahl noch unterwegs ist, kann man einige für die Bilderzeugung verantwortliche Hardware-Register ändern. So ist es möglich, oberhalb und unterhalb der Interruptzeile verschiedene Zeichensätze zu benutzen, wenn man CHBASE (\$D409) mit einem DLI ändert. Auch kann man Farbreister neu besetzen und somit mehr Farben darstellen oder die Positionsregister der Player verändern etc.

Eine der eindrucksvollsten Möglichkeiten für DLI habe ich bei meinem Programm "Schreckenstein" verwendet. Mit Hilfe eines DLIs werden die Register für Fein-Scrolling (HSCROL und VSCROL) in der Mitte des Bildschirms geändert. Dadurch sind zwei Fenster möglich, die sich vollkommen unabhängig voneinander scrollen lassen.

Technik

Bleibt eigentlich nur noch zu klären, wie man mit DLIs umgeht. Drei Schritte sind zu tun:

1. Das DLI-Bit in der gewünschten Zeile der Display-List setzen.
2. Den DLI-Vektor VDSLST (\$0200, \$0201) auf die Interrupt-Routine richten.
3. Den Interrupt mit Bit 7 von NMIIEN (\$D40E) freigeben.

Nach diesen Schritten würde der Vektor VDSLST durchsprungen, nachdem Antic die Anweisung mit dem gesetzten DLI-Bit verarbeitet hätte (genauer dazu noch später). Was jetzt noch fehlt, ist die vom DLI angesprungene Routine. Da dieser Programmteil im Interrupt läuft, muß man ein paar Regeln sehr sorgfältig beachten:

- Alle Register müssen vor der Verwendung auf den Stack gerettet werden. Braucht man nur den Akku, so genügt ein PHA-Befehl.
- Vor Veränderungen der Hardware-Register sollte ein Schreibbefehl auf die Adresse WSYNC (\$D40A) ausgeführt werden. Damit lassen sich Störungen vermeiden; der genaue Mechanismus wird uns später noch beschäftigen.

tigen. Wichtig ist auch, daß alle Veränderungen direkt an den Hardware-Registern ausgeführt werden müssen, denn die Schattenregister werden ja erst im VBI übertragen.

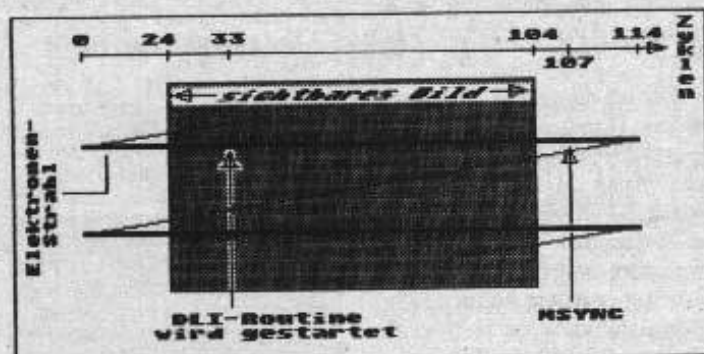
- Vor dem Verlassen der DLI-Routine müssen die ursprünglichen Inhalte der Register wiederhergestellt werden.

Ein ganz einfaches Beispiel für eine DLI-Routine und deren Verwendung in Basic finden Sie in Listing 1. Hier wird nur die Hintergrundfarbe des Textes etwa in der Mitte des Bildschirms geändert.

Mehr DLIs

Natürlich kann der DLI noch viel, viel mehr. In den Listings 2 und 3 bekommt jede Zeile eines GRAPHICS-0-Bildschirms mit Hilfe eines DLIs eine eigene Farbe zugewiesen. Die Farben sind in der Tabelle FARBTAB (ab \$603) zusammengefaßt und können z.B. von Basic geändert werden - eine praktische Methode, wenn man einzelne Zeichen hervorheben will. Im Basic Listing 3 ist ein kleines Demo enthalten, an dem Sie sehen werden, was man alles damit anstellen kann.

Listing 2 zeigt Ihnen die inneren Vorgänge dieses schon relativ komplizierten Interrupt-Beispiels, in dem VBI und DLI zusammenarbeiten. Im ersten Teil werden VBI- und DLI-Routinen wie oben beschrieben eingerichtet. Der VBI, der immer nach der Ausgabe eines kompletten Bildes stattfindet, hat nur die Aufgabe, einen Zeiger in die Farbtabelle auf Null zu setzen. Ist der erste DLI ausgelöst, wird das vom Zeiger markierte Tabellenelement (nach einem STA WSYNC) in das Farbregister 2 übertragen. Danach wird der Zeiger weitergeschaltet, so daß der nächste DLI das zweite Element überträgt. Sind alle 24 DLIs vorbeigelaufen, so tritt wieder ein VBI in Erscheinung, setzt den Zeiger zurück, und das Spiel beginnt von vorne. Sie werden sich vielleicht fragen, warum im vorigen Beispiel kein VBI zum Zurücksetzen des Farbregisters nötig war. Ganz einfach, weil die VBI-Routine im ROM dies



Timing bei Farbwchsel

Aufgabe beim Kopieren der Schattenregister in die Hardware-Register automatisch erledigt.

Besonderheit

Wenn Sie sich die Routine zum Setzen der DLI-Bits in der Display-List genauer ansehen, werden Sie feststellen, daß der erste DLI nicht, wie anzunehmen, in der ersten Zeile des GRAPHICS-0-Bildschirms ausgelöst wird, sondern schon davor. Jede Display-List beginnt mit drei Anweisungen für je 8 Leerzeilen und genau bei der letzten dieser Anweisungen wird das DLI-Bit schon gesetzt. Das hat seinen Grund. Wenn nämlich eine Anweisung in der Display-List mehrere Rasterzeilen erfordert (z.B. der Textmodus Antic 2), dann wird der DLI erst in deren letzter Rasterzeile ausgelöst. Würden wir daher die Anweisung für die erste Textzeile mit einem DLI-Bit versehen, so hätte der Farbwechsel erst Auswirkungen auf die zweite Zeile. Wir lernen daraus, das DLI-Bit immer in der Zeile vorher zu setzen.

Timing

Obwohl der 6502 in Ihrem Atari mit seinen 1.79 MHz schon beachtlich schnell läuft, muß man bei einer DLI-Routine mit der Laufzeit aufpassen. Keine Angst, es ist noch lange nicht kritisch, wenn wie bisher nur eine einziges Farbbregister geändert werden soll. Aber wenn man mehr als 3 Hardware-Register überschreiben will, muß schon mit jedem Maschinenzyklus gerechnet werden.

Wieviel Zeit bei einem DLI zur Verfügung steht, wird an Bild 1 klar. Es zeigt zwei Rasterzeilen eines Bildes, wobei in der ersten Zeile ein DLI stattfinden soll. Man muß noch wissen, daß die Zeit, die der Elektronenstrahl für eine Rasterzeile benötigt, beim 6502 für 114 Zyklen ausreicht. Es vergeht nun schon einige Zeit von der Auslösung des DLIs bis zum Sprung durch den Vektor VDSLST. Das führt dazu, daß die DLI-Routine erst ab Zyklus 33 der ersten Zeile zum Zuge kommt.

Das Register WSYNC

Man könnte jetzt einfach ein Farbbregister mit einer LDA-STA-Kombination neu besetzen. Die Folge davon wäre aber, daß der Farbwechsel mitten in der Rasterzeile stattfinden würde. Durch einige zusätzliche Effekte (DMA!) zittert die Farbgrünze auch noch hin und her und führt zu einer Verunstaltung der Grafik. Aus dieser Klemme hilft uns das Register WSYNC (SD40A, Wait for SYNC). Ein Schreibbefehl auf diese Adresse hält den Prozessor solange an, bis der Elektronenstrahl das Ende der Rasterzeile erreicht hat. Genauer gesagt nimmt der 6502 sieben Zyklen vor Anfang der nächsten Rasterzeile seine Arbeit wieder auf. Zu diesem Zeitpunkt befindet sich der Elektronenstrahl außerhalb des sichtbaren Bereiches und jegliche Änderungen der Grafik bleiben unsichtbar.

Da der Haltezustand des Prozessors im Zyklus 107 aufgehoben wird, muß der Schreibbefehl, der selbst vier Zyklen benötigt, spätestens bei 103 beginnen. Man kann sich leicht ausrechnen, daß für den ersten Teil des DLIs maximal 70 Zyklen verbleiben. Da der Speicher des Ataris aber mit dynamischen Speicher-ICs aufgebaut ist, werden einige Zyklen für den sog. "Refresh" abgezweigt (9 pro Zeile). Findet der DLI in einer Zeile statt, in der der Antic Informationen für die Bildschirmgrafik per DMA aus dem Speicher holen muß, so verringert sich die Zeit für den DLI ganz enorm (20 bis 40 Zyklen beansprucht der Antic je nach Grafikmodus für sich). Man kann also davon ausgehen, daß vom Sprung durch den DLI-Vektor bis zu WSYNC ca. 21 bis 61 Zyklen zur Verfügung stehen. Was ist in dieser Zeit zu tun? Erstens müssen die Register gerettet werden, zweitens bietet es sich an, die neuen Werte für Farben etc. gleich in die CPU-Register zu schreiben. Hier ein kleines Beispiel, in dem die Änderung von drei Farbbregistern vorbereitet wird:

```
PHA
TXA
PHA
TYA
```

```
PHA
LDA #FARBE1
LDX #FARBE2
LDY #FARBE3
STA WSYNC
```

Damit hat man die zweite Phase des DLIs schon gut vorbereitet. Man kann die Werte nach WSYNC nun mit drei Store-Befehlen schnellstmöglich in die Hardware-Register übertragen. Die Fortsetzung des obigen Beispiels wäre z.B.:

```
STA COLPF0
STA COLPF1
STA COLPF2
```

Das ist auch bitter nötig, denn die Zeit ist knapp. Von WSYNC bis zum Zeilenende sind 7 Zyklen, bis zum Auftauchen des Strahles am Anfang der nächsten Rasterzeile sind etwa 17 Zyklen Zeit. Etliche davon beansprucht die DMA des Antic (1 bis 3), mit eingeschalteter PM-Grafik sind es 5 mehr. Das bedeutet, daß in dieser "kritischen" Phase des DLIs ca. 16 bis 23 Zyklen genutzt werden dürfen. Bedenkt man, daß die drei STA-Befehle schon 12 Zyklen dauern, dann wird klar, daß hier keine großen Sprünge möglich sind.

Viel Zeit

Für die letzte Phase des DLIs gibt es dann keine zeitliche Einschränkung mehr. Sie sollte nur zu Ende sein, bevor der DLI beim nächsten Frame ausgelöst wird, also in etwa 24000 Zyklen. Man benutzt diese Phase, um die Register in umgekehrter

Reihenfolge wieder vom Stack zu holen und den Prozessor per RTI vom Interrupt zu erlösen. Zu obigem Beispiel wäre dies:

```
PLA
TYA
PLA
TXA
PLA
RTI
```

Natürlich kann man auch noch andere Aufgaben an dieser Stelle erledigen. Etwa einen Zeiger weiterführen (s. Beispiel) oder z.B. den DLI-Vektor auf eine andere Routine richten. Das würde dazu führen, daß ein weiterer DLI ganz andere Aufgaben ausführen könnte. Denkbar wären auch DLIs, die mit mehreren STA WSYNC arbeiten und sich somit über mehrere Rasterzeilen erstrecken. Auf diese Weise erzeugt man diese Farbbalken, die so ungeheuer plastisch aussehen.

An diesen Ausführungen haben Sie sicherlich bemerkt, daß DLIs ganz schön verzwickelt sein können. Das trifft nicht zu, wenn man nur eine Farbe ändern will, aber es gibt auch Fälle, in denen man mit jedem Maschinenzyklus rechnen muß. Ich erinnere mich noch bestens an einen DLI in meinem letzten Spiel, der vier Players versetzt, zwei Farben ändert und die Bildschirmbreite modifizieren sollte. Im wahrsten Sinne des Wortes - das kostete Schweiß!

Peter Finzel

Listing 1

```
10 REM *****
20 REM * Zweifarbiges GR.0
30 REM *
40 REM * Einfaches DLI-Beispiel
50 REM *****
60 REM
100 REM * DLI-ROUTINE IN PAGE 6
110 FOR I=1536 TO 1546:READ A:POKE I,A
NEXT I
115 REM * PLA
120 DATA 72
125 REM * LDA #C2
130 DATA 169,194
135 REM * STA WSYNC
```



```

140 DATA 141,10,212
145 REM *          STA COLPF2
150 DATA 141,24,208
155 REM *          PLA
160 DATA 104
165 REM *          RTI
170 DATA 64
200 REM * DLI-Bit in D.-List setzen
210 DLIST=PEEK(561)*256+PEEK(560)
220 POKE DLIST+16,128+2
300 REM * DLI freigeben
310 VDSLST=512:NMIEN=54286
320 POKE NMIEN,64:REM * DLI aus
330 POKE VDSLST,0:REM * Vektor
340 POKE VDSLST+1,6:REM * eintragen
350 POKE NMIEN,192:REM * DLI ein

```

Listing 2

```

100 REM *****
110 REM * Vielfarbeffekt in BR.0
120 REM * BASIC-Loader
130 REM * P. FINZEL 1986
140 REM *****
200 MPBM=1536:FARBTAB=1540
220 BDSUB 500
230 X=USR(MPBM)
300 FOR I=0 TO 23
310 X=PEEK(FARBTAB+I)
320 POKE FARBTAB+I,14
330 FOR T=0 TO 50:NEXT T
340 POKE FARBTAB+I,X
350 NEXT I
360 GOTO 300
500 REM * Vielfarbeffekt installieren
510 S=0:RESTORE 600
520 FOR A=1536 TO 1664:READ D:POKE A,D
:S=S+D:NEXT A
530 IF S<>14129 THEN ? "DATEN-FEHLER!"
:STOP
590 RETURN
600 DATA 104,76,29,6,20,34,52,66,84,98
,116,130,148,162,180,194,212
610 DATA 226,244,2,20,34,52,66,84,98,1
16,130,0,173,48,2,172,49,2,133
620 DATA 212,132,213,160,2,177,212,9,1
28,145,212,200,177,212,9,128
630 DATA 145,212,200,200,200,177,212,9
,128,145,212,192,27,208,245
640 DATA 169,0,141,14,212,169,98,141,0
,2,169,6,141,1,2,169,121,141
650 DATA 34,2,169,6,141,35,2,169,192,1
41,14,212,96,72,138,72,174,28
660 DATA 6,189,4,6,141,10,212,141,24,2
08,232,142,28,6,104,170,104
670 DATA 64,169,0,141,28,6,76,95,228

```

Listing 3

```

*****
* DISPLAY-LIST-INTERRUPTS
*
* Vielfarbiges GRAPHICS 0 Display
*****

COLPF2 EQU $D01B Hardware-Reg. fuer Farbe
WSYNC EQU $D40A Warten auf Zeilenende
NMIEN EQU $D40E NMI-Freigabe

VDSLST EQU $0200 Vektor fuer DLIs
VVBLKI EQU $0222 VBI-Vektor (immediate)
BDLSTL EQU $0230 Zeiger auf D-List
BYSVBV EQU $E45F VBI-Routine im ROM

HILFZP EQU $D4 Hilfsregister in Zero-P.
*
* im ATMAS Monitor mit G 0601 starten
*
ORG $0600

PLA fuer BASIC
JMP START Tabelle ueberspringen

*
* Tabelle der Zeilenfarben:
*
FARBTAB DFB $14,$22,$34,$42,$54,$62
DFB $74,$82,$94,$A2,$B4,$C2
DFB $D4,$E2,$F4,$02,$14,$22
DFB $34,$42,$54,$62,$74,$82

ZEIGER DFB 0 Zeiger in Tabelle

START LDA BDLSTL D-List-Zeiger in
LDY BDLSTL+1 Zero-Page kopieren
STA HILFZP f. indirekte
STY HILFZP+1 Adr.-Art
LDY #2 DLI-Bit in
LDA (HILFZP),Y dritter Leerzeile
ORA #$80 setzen
STA (HILFZP),Y
INY DLI-Bit in der
LDA (HILFZP),Y LMS-Anweisung
ORA #$80 der D-List setzen
STA (HILFZP),Y
INY LMS-Adresse
INY ueberspringen
INTSET INY Schleife f.
LDA (HILFZP),Y DLI-Bit in restl.
ORA #$80 Display-List
STA (HILFZP),Y
CPY #27 schon alle Zeilen?
BNE INTSET nein-->

*
* DLI und VBI aktivieren
*
LDA #0 Interrupts aus
STA NMIEN (VBI & DLI)
LDA #DLIPBM:L den Vektor auf

```

```

STA VDSLST      die Int.-Routine
LDA #DLIPGM:H   setzen
STA VDSLST+1
LDA #VBIPGM:L   VBI-Programm
STA VVBLKI      einbinden
LDA #VBIPGM:H   (immediate)
STA VVBLKI+1
LDA #C0         und VBI & DLI
STA NMIIEN      freigeben
RTS

```

*
* DLI-Routine Vielfarben-Effekt in BR.0
*

```

DLIPGM  PHA          Akku retten
        TXA          X-Reg. retten
        PHA
        LDX ZEIGER   Index laden
        LDA FARBTAB,X Farbe aus Tabelle
        STA WSYNC    Zeilenende abwarten
        STA COLPF2    Farbe eintragen
        INX          Zeiger weiter-
        STX ZEIGER   schalten
        PLA          X-Reg. vom Stack
        TAX
        PLA          Akku besorgen
        RTI          Schluss!

```

*
* VBI-Routine Vielfarbeneffekt
*

```

VBIPGM  LDA #0       Zeiger auf Anfang
        STA ZEIGER   der Farbtabelle
        JMP SYSVBV   VBI weiter

```

Der hungrige Goff

Ziel des Spieles ist es, den hungrigen Goff so über das Spielfeld zu lenken, daß sein Hunger nach den saftigen Früchten, die sich auf dem Bildschirm befinden, gestillt wird. Leider ist Goff so gierig, daß er nicht aufhört zu laufen und nach jeder gegessenen Frucht noch schneller wird. Passen Sie jedoch auf, daß Sie ihn nicht in

die zurückgelassenen Würmer lenken. Auch darf er bei seiner Jagd nach Früchten nicht in die Wände laufen.

Sollte es Goff gelingen, alle Früchte im ersten Bild zu essen, gelangt er in ein zweites, in dem es ihm noch schwerer fällt, seinen Hunger zu stillen. Im ersten Bild muß er fünf Früchte pro Bild einsammeln. Bei jedem neuen Bild wird es eine Frucht mehr.

Wolfgang Heemsoth

```

1 PAGE=PEEK(106)-12:CH5=PAGE*256
2 POKE 559,0
3 FOR I=128 TO 471:POKE CH5+I,PEEK(573
44+I):POKE CH5+1024+I,PEEK(57344+I):NE
XT I
4 READ C:IF C=-1 THEN 6
5 FOR I=0 TO 7:READ A:POKE CH5+C*8+I,A
:NEXT I:GOTO 4
6 READ C:IF C=-1 THEN 8
7 FOR I=0 TO 7:READ A:POKE CH5+C*8+I+1
024,A:NEXT I:GOTO 6
8 GOSUB 51
9 BI=1:XR=0:YR=0:CHR=164:TIM=50:H=32:T
I=TIM:SC=0:L=4:MU=1:FRU=MU+4
10 GRAPHICS 17:POKE 756,PAGE:POKE PAGE:
POKE 711,36:GOSUB 22

```

```

11 POSITION 0,22:? #6;"SC:";SC:POSITIO
N 15,22:? #6;"L:";L:POSITION 6,23:? #6
;"LEVEL ";MU
12 POKE 756,POK:POK=POK+4:IF POK>PAGE+
4 THEN POK=PAGE
13 POKE 77,0:5=PEEK(632):FOR M=1 TO TI
:NEXT M
14 IF 5=11 THEN XR=-1:YR=0:CHR=163
15 IF 5=7 THEN XR=1:YR=0:CHR=161
16 IF 5=13 OR 5=9 OR 5=5 THEN XR=0:YR=
1:CHR=164
17 IF 5=14 OR 5=10 OR 5=6 THEN XR=0:YR
=-1:CHR=165
18 SOUND 0,50,0,12:COLOR H:PLOT X,Y:X=
X+XR:Y=Y+YR:LOCATE X,Y,Z:COLOR CHR:PLO
T X,Y:SOUND 0,0,0,H=Z
19 IF Z=F THEN 39
20 IF Z<32 THEN 32
21 GOTO 12
22 IF BI>8 THEN BI=1
23 POKE 559,0:GOSUB 26:ON BI GOSUB 65,
60,73,77,81,85,92,98
24 GOSUB 29:POKE 559,34:CHR=164:FRU=MU
+4
25 RETURN
26 COLOR 6:PLOT 0,0:DRWTO 19,0:DRWTO
19,21:DRWTO 0,21:DRWTO 0,0
27 F=INT(RND(0)*3)+10
28 X=9:Y=10:H=32:COLOR 164:PLOT X,Y:TI
=TIM:RETURN
29 A=INT(RND(0)*18)+1:B=INT(RND(0)*19)
+1:LOCATE A,B,C:IF C<32 THEN 29
30 COLOR F:PLOT A,B:RETURN
31 REM [0]
32 COLOR 162:PLOT X,Y:GOSUB 37
33 FOR M=120 TO 150:SOUND 0,M,10,10:50
UND 0,M+1,10,10:POKE 756,POK:POK=POK+4
:IF POK>PAGE+4 THEN POK=PAGE
34 NEXT M:GOSUB 37
35 L=L-1:IF L<0 THEN 48
36 M=32:XR=0:YR=0:CHR=164:TI=TIM:GOTO
18
37 FOR M=0 TO 3:SOUND M,0,0,0:NEXT M:R
ETURN
38 REM [0]
39 FOR M=150 TO 100 STEP -2.5:SOUND 0,
M,10,10:SOUND 1,M-20,10,10:NEXT M:SOUN
D 0,0,0,0:SOUND 1,0,0,0
40 SC=SC+50:TI=TI-5:H=136
41 FRU=FRU-1:IF FRU<1 THEN MU=MU+1:BI=
BI+1:FRU=MU+4:XR=0:YR=0:SC=SC+200:GOSU
B 43:GOTO 10
42 GOSUB 29:GOTO 11
43 FOR M=15 TO 0 STEP -2:FOR M1=60 TO
40 STEP -1:SOUND 0,M1,10,M:SOUND 1,M1-
10,10,M:NEXT M1:NEXT M
44 SOUND 0,0,0,0:SOUND 1,0,0,0
45 IF BI=4 OR BI=7 OR BI=8 THEN FOR M=
15 TO 0 STEP -2:FOR Q=50 TO 100 STEP 2
:SOUND 0,0,18,M:NEXT Q:NEXT M:L=L+1
46 SOUND 0,0,0,0:POSITION 17,22:? #6;L
47 RETURN
48 FOR M=15 TO 0 STEP -1:FOR M1=20 TO
35 STEP 2:SOUND 0,ABS(M1-M),10,M:NEXT
M1:NEXT M:SOUND 0,0,0,0
49 POSITION 5,10:? #6;"GAME OVER"
50 FOR M=0 TO 1800:NEXT M:GOTO 8
51 GRAPHICS 17:POKE PAGE:POKE 788,198
52 POSITION 1,5:? #6;"[0] PRESENT"
53 POSITION 1,2:? #6;"Wolfgang heemsoth"
54 POSITION 0,9:? #6;" <<< <<< <<< <
<< "
55 POSITION 0,10:? #6;" < < < <
< "
56 POSITION 0,11:? #6;" < < < < <<
<<< "
57 POSITION 0,12:? #6;" < < < < <
< "
58 POSITION 0,13:? #6;" <<<< <<< <
< "
59 POSITION 5,17:? #6;"SCORE:";SC
59 POSITION 4,20:? #6;"press start":PO
SITION 1,22:? #6;"[0] END this game"
60 POKE 756,POK:POK=POK+4:IF POK>PAGE+
4 THEN POK=PAGE
61 FOR M=1 TO 20:NEXT M

```



```

62 IF PEEK(53279) <> 6 AND PEEK(644)=1 T
HEN 60
63 RETURN
64 REM BILD 1
65 POKE 709,84
66 RETURN
67 REM BILD 2
68 POKE 709,196:POKE 708,12
69 COLOR 39:PLOT 5,9:DRWTO 5,5:DRWTO
8,5:PLOT 11,5:DRWTO 14,5:DRWTO 14,9
70 PLOT 6,14:DRWTO 13,14
71 RETURN
72 REM BILD 3
73 POKE 708,164:POKE 709,244
74 COLOR 41:PLOT 1,1:DRWTO 6,6:PLOT 1
8,1:DRWTO 13,6:PLOT 1,20:DRWTO 6,15:
PLOT 18,20:DRWTO 13,15
75 RETURN
76 REM BILD 4
77 POKE 709,102:POKE 708,180
78 COLOR 45:PLOT 4,4:DRWTO 9,9:PLOT 1
0,9:DRWTO 16,3:PLOT 3,17:DRWTO 9,11:
PLOT 10,11:DRWTO 15,16
79 RETURN
80 REM BILD 5
81 POKE 709,180:POKE 708,20
82 COLOR 46:PLOT 7,3:DRWTO 2,8:PLOT 1
2,3:DRWTO 17,8:PLOT 2,12:DRWTO 7,17:
PLOT 17,12:DRWTO 12,17
83 RETURN
84 REM BILD 6
85 POKE 709,70:POKE 708,230
86 COLOR 47:PLOT 3,15:DRWTO 3,10:DRW
TO 16,18:DRWTO 16,15
87 PLOT 3,6:DRWTO 3,3:DRWTO 16,3:DR
WTO 16,6
88 COLOR 62:PLOT 5,10:PLOT 7,10:PLOT 1
2,10:PLOT 14,10:PLOT 5,12:PLOT 7,12
89 PLOT 12,12:PLOT 14,12:PLOT 3,10:PLO
T 16,10:PLOT 9,11:PLOT 10,11:PLOT 3,12
:PLOT 16,12
90 RETURN
91 REM BILD 7
92 POKE 709,212:POKE 708,118
93 COLOR 68:PLOT 6,3:DRWTO 3,3:DRWTO
3,6:PLOT 13,3:DRWTO 16,3:DRWTO 16,6
:PLOT 3,15:DRWTO 3,18:DRWTO 6,18
94 PLOT 13,18:DRWTO 16,18:DRWTO 16,1
5:PLOT 6,7:DRWTO 8,9:PLOT 13,7:DRWTO
11,9
95 PLOT 6,14:DRWTO 8,12:PLOT 13,14:DR
WTO 11,12
96 RETURN
97 REM BILD 8
98 POKE 709,86:POKE 708,164
99 COLOR 61:FOR M=1 TO 17 STEP 2:PLOT
M,8:NEXT M:FOR M=2 TO 18 STEP 2:PLOT M
,13:NEXT M
100 PLOT 5,3:DRWTO 3,3:DRWTO 3,5:PLO
T 3,16:DRWTO 3,18:DRWTO 5,18
101 PLOT 14,18:DRWTO 16,18:DRWTO 16,
16:PLOT 14,3:DRWTO 16,3:DRWTO 16,5
102 RETURN
103 DATA 1,96,215,255,190,192,112,80,1
60
104 DATA 2,17,68,17,68,0,127,107,127
105 DATA 3,6,235,255,125,3,14,10,5
106 DATA 4,28,62,42,62,28,28,28,8
107 DATA 5,28,62,62,62,62,28,16,40
108 DATA 6,255,129,129,153,153,129,129
,255
109 DATA 7,102,153,165,90,90,165,153,1
02
110 DATA 8,56,92,108,12,27,54,62,24
111 DATA 9,60,66,153,165,165,153,66,60
112 DATA 10,4,8,8,28,28,62,62,28
113 DATA 11,0,4,2,2,6,14,28,120
114 DATA 12,0,12,16,56,116,116,124,40
115 DATA 13,0,20,20,127,28,127,20,20
116 DATA 14,54,121,121,54,54,79,79,54
117 DATA 15,0,20,34,73,20,73,34,20
118 DATA 20,60,90,153,255,255,153,90,6
0
119 DATA 29,60,66,129,129,129,129,66,6
0
120 DATA 30,0,60,66,66,66,66,60,0,-1
121 DATA 1,96,208,254,255,247,112,80,4
0

```

```

122 DATA 2,68,17,68,17,0,127,107,127
123 DATA 3,6,11,127,255,239,14,10,20
124 DATA 4,28,62,42,62,28,28,28,8
125 DATA 5,28,62,62,62,62,28,4,6
126 DATA 6,255,195,165,153,153,165,195
,255
127 DATA 7,102,153,129,90,90,129,153,1
02
128 DATA 8,56,92,108,12,27,54,62,24
129 DATA 9,0,60,66,90,90,66,60,0
130 DATA 10,4,8,8,28,28,62,62,28
131 DATA 11,0,4,2,2,6,14,28,120
132 DATA 12,0,12,16,56,116,116,124,40
133 DATA 13,0,65,34,28,28,28,34,65
134 DATA 14,54,79,79,54,54,121,121,54
135 DATA 15,0,20,42,85,42,85,42,20
136 DATA 20,60,102,231,129,129,231,102
,60
137 DATA 29,0,0,24,36,36,24,0,0
138 DATA 30,0,36,36,36,36,36,0,-1
139 REM *****
140 REM *
141 REM * WOLFGANG HEEMSOOTH *
142 REM * (C) 1985 *
143 REM * 2810 VERDEN *
144 REM *
145 REM *****

```

Slow Motion

Tippen Sie das Listing ab, und speichern Sie es auf Disk oder Cassette ab. Nach dem Start legt das Programm auf Page 6 im Computer das Maschinenspracheprogramm ab, ruft es mit X=USR (1536) auf und löscht sich selbst. Die Slow-Motion-Funktion ist nun installiert.

Slow Motion verlangsamt alle Abläufe des Computers. Drücken Sie zum Beispiel während eines Listvorganges die Start-Taste, so wird das Listen

verlangsamt. Auch die Geschwindigkeit, mit der der Computer arbeitet, läßt sich einstellen. Dabei gilt: START und OPTION gedrückt = höhere Geschwindigkeit; START und SELECT gedrückt = niedrigere Geschwindigkeit.

Auch Programme (Spiele) lassen sich bremsen, so daß man vielen in die Karten gucken kann und Programmläufe rascher durchschaut.

Nach dem Drücken der Reset-Taste ist Slow Motion ausgeschaltet, läßt sich aber mit X=USR (1536) wieder starten.

Frank Hendrich

```

29000 REM *****
29010 REM *SLOW-MOTION*
29020 REM *****
29030 DIM HEX$(3):ZELLE=1536
29040 READ HEX$:IF HEX$="END" THEN X=U
SR(1536):GOTO 29180
29050 HI=ASC(HEX$(1,1))-48:IF HI>9 THE
N HI=HI-7
29060 LO=ASC(HEX$(2,2))-48:IF LO>9 THE
N LO=LO-7
29070 WERT=HI*16+LO
29080 POKE ZELLE,WERT
29090 ZELLE=ZELLE+1:GOTO 29040
29100 DATA A9,07,A2,06,A0,08,20,5C,E4,
68
29110 DATA 60,A9,08,8D,1F,D0,AD,1F,D0,
C9
29120 DATA 06,F0,0B,C9,02,F0,14,C9,04,
F0
29130 DATA 10,4C,62,E4,A2,30,A0,48,88,
D0
29140 DATA FD,CA,D0,F8,4C,62,E4,AD,25,
06
29150 DATA C9,02,F0,EC,CE,25,06,4C,22,
06
29160 DATA AD,25,06,C9,60,F0,DF,EE,25,
06
29170 DATA 4C,22,06,END
29180 ? "K":FOR I=29000 TO 29180 STEP
10:? I:NEXT I:? "POKE 842,12:L":POSI
TION 0,0:POKE 842,13:STOP

```

1 Ohrenschmaus: Soundmaschine

Verstärkung, 10 Melodien, Schlagzeug, bis zu 5000 Noten, auch von eigenen Programmen nutzbar, Eingabe über Tastatur oder Joystick. Mit Demos auf 2 Diskettenseiten, ausführliches Handbuch.

ATARI 400 - 130 XE, ab 48 K

Best.-Nr. AT 1

29.80 DM

2 Bargeldlos: Slotmaschine

Simulation eines Spielautomaten, vor Kneipenbesuch dringend zu empfehlen.

ATARI 600 XL (64 K)/800 XL

Best.-Nr. AT 2

Diskette 29.- DM

3 Vielfalt: Atari Power Superbuch

Bauanleitungen, Listings, Tips & Tricks... 75 Seiten DIN-A4, nicht im Buchhandel erhältlich!

Best.-Nr. AT 3

29.- DM

4 Nachschlag: Die Hexenküche

Aufschlußreiß für Ein/Aussteiger und Profis gleichermaßen: Tips & Tricks, Kniffe, Drehs etc. Maschinensprache-Programme als Listings. Turned Ihren Atari ganz schön an (und Sie auch)!

Best.-Nr. AT 4

29.80 DM

5 Zuschlag: Disk zu Hexenküche

Damit kann man viel Zeit sparen.

Best.-Nr. AT 5

19.80 DM

6 Fix: ATMAS-II Macro Assembler

8 K Quelltext in 4 Sekunden assembliert! Erzeugung von Bildschirmcode, Full-Screen-Editor, scrollt in beide Richtungen, integrierter Monitor, 50 seitiges Handbuch und Disk im Ringordner.

ATARI 400 - 130 XE

Best.-Nr. AT 6

Diskette 49.- DM

7 Geistsparend: ATMAS Toolbox

Rechenroutinen, I/O, Makros, Customizer, Fast circle, Scrolling, und noch einiges mehr. Auf Diskette mit Anleitung dabeist.

ATARI 400 - 130 XE, ab 48 K

Best.-Nr. AT 7

19.80 DM

8 Praktisch: Monitor XL

Verknüpft Basic-Programme mit Mode-Routinen, eingeben, korrigieren, listen, Single-Step, Disk laden/speichern, Directory-Anzeige, deutsche Fehlermeldungen auch für Basic und DOS. Der Basic-Speicherplatz bleibt unberührt, Anleitung und Disk.

ATARI 600 XL (64 K)/800 XL/130 XE

Best.-Nr. AT 8

19.80 DM

9 Brandneu: Design Master

Bedienung über Fenster-Technik, Auflösung 320 x 192, Fadenkreuz, Maßstabgitter ein/ausblendbar, 2 Screens gleichzeitig, über 122.000 Punkte im Direktzugriff, über 100 verschiedene Schriften, Hardcopy für fast alle Matrix-Drucker (ab 8 Nadeln), Ausdruck in verschiedenen Größen möglich, ausführliche deutsche Anleitung.

ATARI 600 XL (64 K)/800 XL/130 XE

Best.-Nr. AT 9

Diskette 19.80 DM

!!
9
extre
take
↓

Anzahl	Best. Nr.	Artikel	Preis Incl. MwSt.
	AT 1	Soundmaschine	
	AT 2	Slotmaschine	
	AT 3	Atari Power Superbuch	
	AT 4	Die Hexenküche	
	AT 5	Disk zur Hexenküche	
	AT 6	Macro Assembler	
	AT 7	ATMAS Toolbox	
	AT 8	Monitor XL	
	AT 9	Design Master	

Name des Bestellers

Ich wünsche folgende Bezahlung:

Anschrift

☐ Nachnahme
(Zuzügl. DM 5,70 Versandkosten)

PLZ/Ort

☐ Vorkasse
(keine Versandkosten)

Datum, Unterschrift

Bei Vorkasse bitte Scheck belegen
oder überweisen auf Postcheckkonto
Karlsruhe 43423-755

Coupon ausschneiden, auf Postkarte kleben und einsenden an:

Verlag Rätz-Eberle/CK-Software, Postfach 1640, 7518 Bretten

6502-Maschinensprache

Ein Kurs für Atari-User – 2. Folge

Willkommen zum zweiten Teil unseres Assemblerkurses. In der ersten Folge haben wir uns mit ganz grundsätzlichen Dingen wie Zahlensystemen und dem Aufbau des 6502-Prozessors beschäftigt. Diesmal werden wir zunächst das Handwerkszeug besprechen, das wir zum Programmieren in Assembler benötigen. Anschließend widmen wir uns den ersten Befehlen der 6502-Maschinensprache.

Das Handwerkszeug

Um es gleich vorwegzunehmen: Um Ihren Atari-Computer sinnvoll in Maschinensprache programmieren zu können, brauchen Sie spezielle Software. Die Entwickler haben dem 800XL nur Basic mit auf den Weg gegeben. Wie Sie in späteren Folgen noch sehen werden, kann man zwar auch unter Basic Maschinenprogramme eingeben und aufrufen, allerdings sind diese Möglichkeiten zur Entwicklung von Maschinenprogrammen denkbar ungeeignet.

Um das zu verstehen, müssen wir uns erst einmal ansehen, was Maschinensprache eigentlich ist. Jedem Maschinenbefehl ist ein binärer Code zugeordnet, an dem der Prozessor den Befehl erkennen und ausführen kann. Beim 6502 wurden 8 Bit für die Codierung der Befehle verwendet, eine Größe, die sich durch den Aufbau der Hardware (8-Bit-Rechner) anbot. Mit diesen 8 Bit ließen sich maximal 256 verschiedene Codes für Maschinenbefehle darstellen, von denen beim 6502 aber nur etwas mehr als die Hälfte tatsächlich benutzt werden.

Damit wissen wir, daß sich Maschinenbefehle als 8-Bit breite Binärzahlen darstellen lassen. Wie Ihnen aber sicher aus der letzten Folge noch bekannt ist, gibt es für solche Binärzahlen noch andere Darstellungen, die man sich besser merken kann. Da wäre zunächst die hexadezimale Schreibweise, mit der ein Byte

in zwei Ziffern geschrieben werden kann. Diese Darstellung von Maschinenbefehlen wird sehr häufig bei Programmen zur Fehlersuche verwendet (oft als "Debugger" oder "Monitor" bezeichnet). Natürlich kann man die Binärzahl auch ins dezimale System umrechnen. Das ergibt dann die endlosen DATA-Zeilen, die über und über mit Zahlen vollgestopft sind und die Sie sicher schon oft in Basic-Listings gesehen haben.

Jetzt können Sie sich auch vorstellen, warum Basic nicht gerade zum Programmieren in Maschinensprache optimal ist. Man müßte dazu eine Tabelle haben, in der zu jedem Befehl der passende Code in dezimaler Schreibweise enthalten ist. Dann müßte man jeden Befehl aus dieser Liste herausuchen und den Code in DATA-Zeilen eintragen. Eine FOR-NEXT-Schleife wäre zusätzlich notwendig, um das Maschinenprogramm in den Speicher zu bringen, damit es per USR-Befehl aufgerufen werden kann. Sie können sich lebhaft vorstellen, daß diese Art der Eingabe sehr, sehr mühsam ist und außerdem zu Programmen führt, die nach kurzer Zeit nicht mehr überblickt, geschweige denn verändert werden können.

Der Assembler

Damit man die Befehle der Maschinensprache mit ähnlichem Komfort wie z.B. Basic-Programme eingeben kann, werden spezielle Übersetzungsprogramme als "Assembler" benutzt. Um den Befehlen ein übersichtliches Aussehen zu verleihen, benutzt man Abkürzungen für deren Funktion. Für den Befehl "Lade den Akkumulator" steht das Kürzel "LDA", für "Speichere das X-Register" steht "STX". Da der 6502 in den Vereinigten Staaten entwickelt wurde, gehen diese Kürzel auf die englische Bezeichnung der Befehle zurück. Damit erklärt sich auch, daß das obige Beispiel (Store X-Register) zu STX wird.

Ein Assembler braucht drei Funktionen. Er muß die Eingabe des Programmtextes erlauben, d.h. er benötigt einen Editor. Zweitens muß er den eingegebenen Text in ein lauffähiges Maschinenprogramm übersetzen können, und schließlich sollte es eine Möglichkeit geben, das übersetzte Programm zu starten bzw. darin nach Fehlern zu suchen. Die meisten für den Atari erhältlichen Assembler enthalten diese drei Teile und erlauben somit eine mehr oder weniger komfortable Programmmentwicklung auf Maschinenebene.

Verwendbar für unsere Zwecke ist das Editor/Assembler-Modul von Atari, der MAC/65 von OSS und der AT-MAS-II. Die Beispiele in späteren Folgen dieser Artikelserie werden mit letzterem erstellt, lassen sich aber leicht für die anderen Assembler anpassen. Die Kürzel für die Maschinenbefehle sind in allen Assemblern gleich. Man hält sich hier an die Empfehlung des Herstellers. Unterschiedlich sind lediglich spezielle Anweisungen zur Steuerung der Übersetzung. Diese sog. "Direktiven" werden wir uns später noch genauer ansehen.

Das Format

Im Unterschied zu Basic erwartet der Assembler, daß seine Programme gemäß einem fest vorgegebenen Schema eingegeben werden. So ist es z.B. nicht möglich, zwei Befehle in eine Zeile zu schreiben. Sehen wir uns dazu in Bild 1 eine Zeile eines Assemblerprogrammes näher an.

Die Zeile beginnt mit einem sog. Label, das hier MARKE genannt wurde. Beim Assemblieren dieser Zeile würde sich der Assembler den Namen des Labels zusammen mit der Adresse des nachfolgenden Befehles merken. Damit sind die Voraussetzungen gegeben, daß dieser Befehl das Ziel eines Sprungbefehles sein kann. Labels ersetzen also die Funktion der Zeilennummern in Basic.

Das feste Eingabeformat des Assemblers erfordert auch, daß Zeilen ohne Label eingerückt geschrieben werden. Dazu genügt ein Leerzeichen am Anfang der Zeile. Besser ist es natürlich, wenn man etwas weiter einrückt und so das Listing übersichtlich gestaltet.

Beim nächsten Element in der Zeile handelt es sich um den Befehl selbst. Wie wir bereits gesehen haben, werden hierzu Abkürzungen verwendet, die beim 6502 einheitlich drei Zeichen lang sind.

Wie die meisten der Basic-Anweisungen brauchen auch viele der Maschinenbefehle zusätzliche Informationen. Diese werden in das dritte, sogenannte "Operanden-Feld" geschrieben. Falls keine Operanden nötig sind, beginnt hier sofort das Kommentarfeld, das nur dazu dient, den Ablauf des Programms näher zu erläutern. Manche Assembler brauchen vor Beginn des Kommentarfeldes noch einen Strichpunkt als Trennzeichen, bei den meisten geht's auch ohne. Einen Strichpunkt (oder einen Stern bei ATMAS) braucht man aber in jedem Fall am Anfang einer Zeile, die man ausschließlich für Kommentare verwenden möchte. Diese Kommentare werden beim Assemblieren einfach ignoriert und benötigen somit keinen Speicherplatz im endgültigen Programm. Da Maschinenprogramme dazu tendieren, lang und unübersichtlich zu werden, empfiehlt es sich, von den Kommentaren ausgiebig Gebrauch zu machen.

Direktiven

Neben den Maschinenbefehlen beherrscht jeder Assembler eine Anzahl von Pseudo-Befehlen. Diese Direktiven, wie man sie auch nennt, erzeugen keinen Maschinencode, sondern dienen nur zur Steuerung des Übersetzungsvorganges oder zum Einfügen von Daten und Tabellen. Da diese Direktiven je nach Umfang des Assemblers verschieden sind, sollen hier

nur die grundsätzlichen Dinge besprochen werden.

Anfangsadresse

Während Sie sich bei Basic um solche Dinge gar nicht kümmern müssen, kann man in Assembler bestimmen, wo das übersetzte Programm im Speicher abgelegt wird. Beim MAC/65 lautet die Direktive dazu "*" (oder ".ORG"), beim ATMAS-II schreibt man "ORG" mit der gewünschten Adresse im Operandenfeld. Natürlich ist nicht jede beliebige Adresse möglich, denn sonst würde man eventuell wichtige Speicherbereiche (z.B. im DOS) überschreiben. Hier bietet wieder jeder Assembler andere Features, beim ATMAS ist z.B. ein Bereich ab \$A800 für die übersetzten Programme reserviert und beim MAC/65 kann man mit LOMEM einen Bereich nach dem DOS (ab ca. \$2000) reservieren.

Daten und Tabellen

Auch in Assembler gibt es eine Möglichkeit, Daten direkt ins Programm mit aufzunehmen, wie man es bei Basic mit der DATA-Anweisung tun kann. Während aber bei DATA Zahlen und Strings beliebig vermischt werden können, heißt es in Assembler aufpassen. Hier gibt es Direktiven für die verschiedenen Datentypen, wie Strings, Bytes und Words. Bei ATMAS kann man mit ASC einen String, mit DFB einzelne Bytes und mit DFW Words in den Assemblercode einbinden. Ein "Word" ist übrigens eine Zahl mit 16 Binärstellen (vier Hex-Stellen), ein Format, das z.B. zum Aufbewahren von Speicheradressen verwendet wird. Bei MAC/65 heißen die Direktiven ".BYTE" für Strings und Bytes sowie ".WORD".

Symbolische Namen

Bei der Besprechung des Eingabeformates sind wir bereits auf die Labels gestoßen. Diese Einrichtung bietet dem Programmierer die Möglichkeit, anstelle von Zahlenwerten symbolische Namen zu verwenden. Wenn man statt "Springe zur Adresse \$A937" einen Befehl als "Springe zum Label SCHLEIFE" lesen kann, dann erhöht das die Verständlichkeit eines Programmes ungemein.

Der Assembler bietet aber auch die Möglichkeit, neben Labels auch beliebige Konstanten mit einem Namen versehen zu können. Damit kann man z.B. wichtigen Einsparungen ins Betriebssystem einen Namen geben oder die Hardware-Register benennen. Die Direktive dazu heißt beim MAC/65 einfach "=", beim ATMAS-II "EQU". Beim Assemblieren werden die Namen durch die Konstanten ersetzt. Beispiele dazu werden Sie in den nächsten Teilen dieser Serie noch reichlich finden.

Erste Befehle

Es ist soweit; nun lernen Sie die ersten Befehle der 6502-Maschinsprache kennen. Sie wissen ja sicher noch aus der letzten Folge, daß alle Rechenoperationen beim 6502 über den Akkumulator laufen müssen. Bevor man aber mit dem Akku rechnen kann, muß man erst einmal die Möglichkeit haben, dort Werte einzutragen.

Zum Laden eines Wertes in den Akku dient der Befehl mit dem Kürzel LDA, das für "Load Akkumulator" steht. Wenn Sie sich nun Tabelle 1 genauer ansehen, stellen Sie fest, daß es mehrere Formen dieses Befehles gibt. Diese Formen nennt man "Adressierungsarten". Sie bestimmen, woher der Wert für den Akku stammen soll. Grundsätzlich könnte man sich zwei Fälle vorstellen:

1. Der Akku soll mit einer Konstanten belegt werden. Dazu verwendet man die "Immediate"- (unmittelbare) Adressierung, die durch das Doppelkreuz vor dem Operanden gekennzeichnet wird. Schreibt man also LDA #37, so wird die Zahl 37 in den Akku geschrieben.

2. Eine andere Möglichkeit wäre es, den Inhalt einer Speicherzelle in den Akku zu schaffen (mehr oder weniger ein PEEK). Dazu dient die "absolute" Adressierung. Der Operand wird hier als direkte Adresse aufgefaßt. Wenn Sie daher LDA 1536 schreiben, wird nicht 1536 in den Akku geschrieben (ginge ja gar nicht!), sondern der Inhalt

der Speicherzelle 1536 in den Akku geladen.

Sie sehen jetzt schon, auf was die verschiedenen Adressierungsarten hinauslaufen. Neben den gerade angeschnittenen verfügt der 6502 aber noch über einige weitere Adressierungen, die einen vielfältigen Einsatz der Befehle garantieren:

Zero-Page

Auch hier wird eine Adresse als Operand angegeben. Nur ist der Adressbereich auf die Speicherzellen 0 bis 255 (eben jene Zero-Page) beschränkt. Der Vorteil ist, daß solche Befehle weniger Speicherplatz als absolut adressierte Befehle benötigen (nur 2 statt 3 Bytes) und auch schneller ausgeführt werden.

Indiziert

Was tun Sie in Basic, wenn Sie auf mehrere nacheinander liegende Adressen (z.B. ab 1536) zugreifen wollen? Richtig, Sie schreiben z.B. A = PEEK (1536+I) und verändern die Variable I. Genauso funktioniert's auch in Assembler, nur gibt es dafür eine spezielle Adressierungsart namens "Indiziert". Die Rolle der Variablen I übernimmt ein Index-Register. Die Adresse des Speicherzugriffes wird dann aus der angegebenen Adresse plus dem

Inhalt des Index-Registers gebildet. Dem obigen PEEK-Befehl entspricht in Assembler die Anweisung LDA 1536,X.

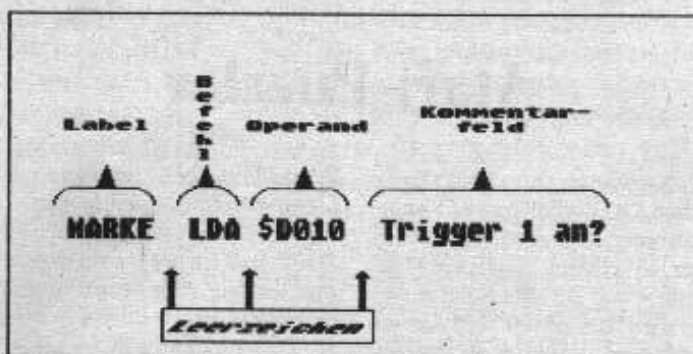
In der Schreibweise unterscheidet sich diese Adressierungsart durch das an den Operanden angehängte Index-Register. Nicht alle Befehle lassen allerdings die Wahl zwischen X und Y frei; bei manchen ist das fest vorgeschrieben. Bei LDA sind beide Möglichkeiten zulässig.

Zero-Page indiziert

Das ist eine Mischform der beiden vorangegangenen Formen. Sie verwendet die kurzen Zero-Page-Adressen zusammen mit der Indizierung. Die Einschränkung ist dabei wieder, daß nur Bytes in der Zero-Page angesprochen werden können. Dafür sind die Befehle nur zwei Bytes lang. Beispiel: LDA \$80,X

Indirekt

Hier geht man davon aus, daß zwei nacheinander liegende Speicherzellen eine neue Speicheradresse enthalten. Man spricht in so einem Fall auch von einem Zeiger, Pointer oder Vektor, der auf das gesuchte Byte deutet. Die indirekte Adressierung brauchen Sie z.B., wenn Sie ein Zeichen des Bildschirms per PEEK-Befehl lesen wollen. Die Anfangs-



Syntax des LDA-Befehls

Befehl:	LDA	Flags:	N, Z
Adr.-Art	Syntax:	Opcode	Länge
Immediate	LDA #Dat	A9	2
Zero-Page	LDA Zadr	A5	2
Zero-Page, X	LDA Zadr, X	B5	2
Absolut	LDA Adr	A9	3
Absolut, X	LDA Adr, X	B9	3
Absolut, Y	LDA Adr, Y	B9	3
(Indir.), Y	LDA (Zadr, Y)	B1	2
(Indir.), X	LDA (Zadr, X)	A1	2

Die Adressierungsarten des LDA-Befehls

adresse des Bildschirmspeichers steht in den Adressen 88 und 89. Klar, man braucht zwei Speicherzellen, denn eine Adresse hat 16 Bit Länge, und in jedes Byte passen nur acht. Daher werden die unteren acht Bit (das sog. LSB) in eine Speicherzelle gelegt, die oberen 8 Bit (das MSB) in die darauffolgende. Der indirekte Zugriff würde in Basic so aussehen:

```
A = PEEK (PEEK(88) +
PEEK(89) * 256)
```

Damit würden wir indirekt, d.h. über den Umweg des Zeigers in 88,89, auf das erste Zeichen des Bildschirms zugreifen. In Assembler geht das viel einfacher. Der Befehl LDA(88),Y erledigt die gleiche Aufgabe wie die verschachtelten PEEKs. Aber was hat das "Y" am Ende zu suchen? Das ist eine Eigenheit des 6502-Prozessors, denn dieser kennt keine einfache indirekte Adressierung. Der 6502 kann nur indirekt zugreifen, wenn er gleichzeitig noch indizieren, also das Index-Register zur Adresse zählen darf. Im Beispiel würde er die Bildschirmadresse aus 88,89 holen, dann erst das Y-Register addieren und anschließend auf diese Adresse zugreifen. Wenn wir eine rein indirekte Adressierung wünschen, muß daher das Y-Register zuvor auf Null gesetzt werden.

Vielleicht wundern Sie sich jetzt, warum man das so kompliziert gemacht hat, aber Sie werden bald merken, daß es in vielen Fällen sehr praktisch ist.

Der 6502 beherrscht noch eine weitere Abart der gerade besprochenen Adressierung. Man vertauscht dazu die Reihenfolge des Ablaufes und addiert zuerst das Index-Register zum Operanden und benutzt diese Adresse zur Indirektion. Auf diese Weise kann mit dem Index-Register einer von mehreren indirekten Zeigern ausgewählt werden. Die Umkehrung schlägt sich auch in der Schreibweise nieder: LDA (88,X)

Als Operanden sind bei den letzten beiden Adressierungsarten nur Adressen in der Zero-Page möglich. Wohl können beliebige Adressen indirekt angesprochen werden, aber der Zeiger muß sich immer in der Zero-Page befinden.

Damit sind wir am Ende der zweiten Folge angelangt. Nachdem nun die Grundlagen vollständig besprochen wurden, können wir im nächsten Teil mit ganzer Kraft in den 6502-Befehlssatz einsteigen.

Peter Finzel

Diese Serie enthält Auszüge aus dem neuen Buch unseres freien Mitarbeiters P. Finzel, das in Kürze erscheinen wird.

Atari-Puzzler

Nachdem das Basic-Spielchen "Rollerball" so gut angekommen ist, habe ich etwas tiefer in meiner Spielekiste gekramt und ein altes Spiel ausgegraben, das zu 100 % in Assembler geschrieben ist. Es läuft auf allen Ataris ab 48 K. Ich hoffe, daß es trotz der vielen Datas ebenfalls seine Fans findet. Um es unseren Lesern so bequem wie möglich zu machen, habe ich den "Puzzler" mit einer Zeilen-Prüfsumme ausgestattet, so daß beim Erzeugen des eigentlichen MS-Programmes auf Disk eventuelle Fehler sofort erkannt werden. Die fehlerhafte Zeile wird dann automatisch gelistet. An dieser Stelle möchte ich besonders auf die vierstimmige Musik von Johannes

Plenio hinweisen, die wir auch schon als Basic-Listing veröffentlicht haben. Herzlichen Dank, daß ich die Daten für die Puzzler-Interrupt-Musik übernehmen durfte.

Tippen Sie das abgedruckte Listing fehlerfrei ab und starten Sie es mit RUN. (Um Abstürzen vorzubeugen, sollte man es auch während der Eingabe öfters abspeichern! 32767 REM nicht vergessen - hilft manchmal.) Jetzt werden Sie aufgefordert, eine Diskette einzulegen, auf der sich ein DOS befindet. Wenn Sie nun RETURN drücken, wird auf der Diskette eine Datei mit dem Namen "Puzzler.COM" erzeugt (wenn keine Fehler enthalten sind).

Speichern Sie spätestens jetzt das Basic-Listing ab und schalten Sie den Computer aus. Laden Sie jetzt das DOS ohne Basic, indem Sie den Computer einschalten, während Sie die Option-Taste drücken. Sobald sich das DOS-Menü meldet, können Sie die Datei "Puzzler.COM" mit der Binärladefunktion laden. Das Programm startet dann automatisch. Bei DOS 2 ist hierfür die Funktion L zuständig.

"Puzzler" ist die Lösung für ein altes Problem: Da hat man nun für den Farbwunder-Computer Atari tolle Zeichenprogramme und erstellt in stundenlangen Sitzungen Kunstwerke aller Art. Im Gegensatz zu den altmodischen Künstlern, die noch mit Öl und Leinwand wirken, fällt es dem Computerkünstler jedoch schwer, seine Werke auszustellen, um sie der breiten Öffentlichkeit zugänglich zu machen. Ist das Bild also gelungen, so wird in der Regel die Familie genötigt, dieses zu bewundern, bevor es wieder auf der nächstbesten Diskette verschwindet. Echte Malgenies lehnen es sogar ab, ihre Kunst in tristem Grau als Hardcopy auf dem Drucker auszugeben.

Nun gibt es jedoch den "Puzzler" und damit eine Möglichkeit, daß sich Ihre Freunde und Bekannten oft Stunden bis ins Detail mit Ihren Werken beschäftigen. Nachdem Sie den "Puzzler" gestartet haben, erscheint das Titelbild mit der Aufforderung, den Namen ei-

nes Bildes einzugeben. Legen Sie also eine Diskette mit Ihren Bildern ein und geben Sie den entsprechenden Namen ein. Sie können auch mit RETURN ein Inhaltsverzeichnis anfordern, durch das Sie mit der SELECT-Taste blättern können. START lädt dann das angezeigte Bild. "Puzzler" kann sowohl Bilder von der Maltafel (bzw. dem KoalaPad) als auch im Micropainter-Format einlesen, so daß die verbreitetsten Malprogramme verwendet werden können. Das Programm stellt sich dabei automatisch auf das verwendete Format ein.

Ist ein Bild glücklich geladen, darf man die START-Taste drücken. Das Bild wird nun in 6*10 Felder aufgeteilt, die der Computer munter durcheinander schiebt. Diese Barbarei kann jederzeit mit dem Feuerknopf des Joysticks 1 abgestellt werden. Nun ist es Aufgabe des Spielers, das ursprüngliche Bild wieder herzustellen. In das freie Feld kann jeweils eines der 4 benachbarten Felder geschoben werden, indem man in die Richtung drückt, in die sich dieses Feld bewegen soll.

Wer den Überblick verliert, der kann mit OPTION alle Felder blinken lassen, die sich noch nicht am richtigen Platz befinden. Ist das Bild wieder restauriert, werden die Punkte ausgegeben, die vom Grad des Durcheinanders abhängen, das der Computer hinterlassen hat.

Thomas Tausend

Puzzler

```
10 REM *****
20 REM * P u z z l e r (c) by TT86 *
30 REM *****
40 REM * ein Maschinensprache-Spiel *
50 REM * von Thomas Tausend *
60 REM * Ried 17 * 8961 Sulzberg *
70 REM * Tel. (08376)/1664 ab 19.00 *
80 REM *****
90 REM * Für ATARI ab 64K und Disk *
99 REM *****
100 DIM A$(22):ZN=1000
101 PRINT CHR$(125); "Puzzler:"
102 PRINT "Bitte DOS II Disk"
103 PRINT "einlegen und RETURN"
104 PRINT "druecken.":INPUT A$
105 PRINT "Bitte etwas Geduld!":?
```

```

109 OPEN #1,8,0,"D:\TEST.COM"
110 READ A$:SS:IF A$="0" THEN END
120 S=0:FOR N=1 TO 21 STEP 2
130 GOSUB 200
140 S=S+A:PUT #1,A
150 NEXT N:?" ":
160 IF SS(>)S THEN ? "FEHLER IN ";ZN:LI
ST ZN:END
170 ZN=ZN+10
180 GOTO 110
200 A=(ASC(A$(N))-48-(ASC(A$(N))>60)*7
)*16+ASC(A$(N+1))-48-(ASC(A$(N+1))>60)
*7
210 RETURN
1000 DATA FFFF004FB40A900000406,1209
1010 DATA 8D0C86A9078D1006A9018D,809
1020 DATA 1206A9038D0606201147A9,638
1030 DATA 000D1A06A900A052204B44,770
1040 DATA A90000F002201F44A9008D,1001
1050 DATA 160620D947201E434C8952,772
1060 DATA A210A90C9D42032056E4A9,1100
1070 DATA 039D4203A9439D4403A945,931
1080 DATA 9D4503A9049D4A032056E4,902
1090 DATA 10034CA94C60A210A5509D,1024
1100 DATA 4403A5599D4503A9009D48,952
1110 DATA 03A91E9D4903A9079D4203,837
1120 DATA 2056E410034CA94C204143,850
1130 DATA 8D0C022041438D0C4022041,943
1140 DATA 438D0C5022041438D0C602A2,1074
1150 DATA 10A90C9D42032056E4A550,1022
1160 DATA 05CDA55905CEA9000D2406,1203
1170 DATA 8D26060D1C0620E04C206D,841
1180 DATA 44209043A900054DAD1FD0,1110
1190 DATA C903D00620F64C20F64C20,1150
1200 DATA EE43AD7602C90FF0E5AE78,1579
1210 DATA 02E00FD0F9C90B0006200E,1170
1220 DATA 414CC440C907D006200D41,1061
1230 DATA 4CC440C90EFC40F7410006,1393
1240 DATA 200D424CC440C90DD0BC20,1009
1250 DATA 90424CC440AD2606C909F0,1213
1260 DATA 77205F43A200A9000D2206,825
1270 DATA 8D2006A3C065CBA5CE05CC,1593
1280 DATA A9000D0006A00710028B1,740
1290 DATA CB2A91CB0000C0FFD0F460,1740
1300 DATA EE0006AD0006C902D0E510,1103
1310 DATA A5CB692005CBA5CC690005,1456
1320 DATA CCEE2206AD2206C920D0C9,1337
1330 DATA E0E010D0B418A5CD690405,1496
1340 DATA CDASCE690005CEEE2606AE,1476
1350 DATA 1C06E00DA44548A9009DA4,1250
1360 DATA 4560CA9DA445E0BE1C0660,1269
1370 DATA AD2606C900F078205F43A9,1141
1380 DATA 000D2206AA3BA5CDE9B405,1147
1390 DATA CDASCEE90005CEA9000D22,1492
1400 DATA 06A5CD05CBA5CE05CCA900,1509
1410 DATA 8D0006A00010028B1C06A,873
1420 DATA 91CB00C8C000D0F460E00B,1550
1430 DATA 06AD0006C902D0E518A5CB,1225
1440 DATA 692005CBA5CC690005CCEE,1530
1450 DATA 2206AD2206C920D0C9E0E0,1351
1460 DATA 10D0B7CE2606AE1CF841F3,1415
1470 DATA 4206CABDA44548A9009DA4,1250
1480 DATA 4560E09DA445CA8E1C0660,1269
1490 DATA AD2406C905F078A9000D22,1120
1500 DATA 06205F4318A5CD692005CB,1075
1510 DATA A5CE690005CCA24038A5CB,1463
1520 DATA E92005CFA5CCE90005D0A0,1716
1530 DATA 00B1CB91CFA90091C0C0C0,1641
1540 DATA 04D0F318A5CB692005CBA5,1493
1550 DATA CC690005CCCAE001D0D2CE,1697
1560 DATA 0006EE2206AD2206C920D0,938
1570 DATA B718A5CE690505CEEE2406,1307
1580 DATA 18AD1C06690AA0BDA44540,1010
1590 DATA A9009DA445AE1C06609DA4,1192
1600 DATA 45AD1C06690A0D1C0660AD,835
1610 DATA 2406C900D0034C1D43A900,795
1620 DATA 8D2206205F4318A5CE6905,800
1630 DATA 05CE38A5CDE92005CBA5CE,1745
1640 DATA E90005CCA24038A5CBE920,1493
1650 DATA 05CFA5CCE90005D0A000B1,1620
1660 DATA CF91CBA90091CFC8C004D0,1600
1670 DATA F338A5CBE92005CBA5CCE9,1070
1680 DATA 0005CCCAE001D0D2CE0006,1394
1690 DATA EE2206AD2206C920D0B638,1170
1700 DATA F442EF43A5CEE90A05CECE,1775
1710 DATA 2406AD1C0638E90AA0BDA4,1071
1720 DATA 4548A9009DA445AE1C0660,1012
1730 DATA 9DA445AD1C0638E90A0D1C,1065
1740 DATA 0660AD300205CFAD310205,1022
1750 DATA D0A003B1CFC90FF007C94F,1490
1760 DATA F0034C3B43E90191CFC0C0,1423
1770 DATA C7D0EA0006006A210A907,1335
1780 DATA 9D4203A9009D40039D4903,860
1790 DATA 2056E410034CA94CAE6006,962
1800 DATA 6020B343A9038D0F02A904,1005
1810 DATA 8D01D2A90005148D0006AD,994
1820 DATA 00060D00D2EE0006200E43,842
1830 DATA C903D0F0A9000D00D2A9A0,1501
1840 DATA 8D01D26048A514AAE414F0,1363
1850 DATA F960A09000D1E060D1F06,973
1860 DATA A9018D020620C643AD1000,1013
1870 DATA D0F8A9000D020606AD0206,1051
1880 DATA F00DAD1E0669018D1E0690,909
1890 DATA 03EE1F0660A90018AD0AD2,971
1900 DATA 693290034C0D4218693290,700
1910 DATA 034C0D4118693290034C90,831
1920 DATA 4218693290034C0E414CC6,821
1930 DATA 43A200F043EB44BDA445D0,1402
1940 DATA 6845D026E0E03CD0F3A90F,1570
1950 DATA 209D44A9018D1206A9000D,902
1960 DATA 0606201147A9028D1A06A9,645
1970 DATA 0BA0492048444CCD4460A2,1026
1980 DATA 00A9059D4203A9459D4403,866
1990 DATA A9459D4503A9149D4003A9,1057
2000 DATA 009D49032056E4201848AD,800
2010 DATA 4003C901D0034CA440600D,1037
2020 DATA 4203AE1A06BD06490D4403,803
2030 DATA E00D06490D4503980D4003,1209
2040 DATA A9000D4903A2002056E460,990
2050 DATA A2000E2206A55005CFA559,1191
2060 DATA 05D0A000B1CF9D3646E0C0,1590
2070 DATA C004D0F518A5CF692005CF,1530
2080 DATA 9002E6D0EE2206AD2206C9,1276
2090 DATA 20D0DE60A2000E2206A550,1155
2100 DATA 05CFA55905D0A0000D3646,1400
2110 DATA 91CFE0C8C004D0F518A5CF,1029
2120 DATA 692005CF9002E6D0EE2206,1347
2130 DATA AD2206C920D0DE60A9018D,1203
2140 DATA F002AD1E0605D4AD1F0605,1139
2150 DATA D520AD9A90385F3A94705,1553
2160 DATA F420E0D02032EC44E74545,1477
2170 DATA A5F38D4403A5F40D4503A9,1411
2180 DATA 090D4203A90C0D4003A900,705
2190 DATA 8D4903A2002056E4AD1FD0,1137
2200 DATA C906D0F9201848A000A0A0,1092
2210 DATA 28C906D011201848A9200D,950
2220 DATA 160620D947201E434CA840,705
2230 DATA 4C0B45A0FFC8B1F310F029,1499
2240 DATA 7F91F3C8A99B91F360443A,1649
2250 DATA 20202020202020202020,352
2260 DATA 20202020202020202020,352
2270 DATA 20202020202020533A9B45,509
2280 DATA 3A90000102030405060700,249
2290 DATA 090A000C000E0F10111213,154
2300 DATA 1415161718191A30313233,359
2310 DATA 3435363738394142434445,662
2320 DATA 464748494A4B4C4D4E4F50,825
2330 DATA 5152535455565700010203,594
2340 DATA 0405060708090A0B0C0D0E,99
2350 DATA 0F10111213141516171819,220
2360 DATA 1A30313233343536373839,551
2370 DATA 4142434445464748494A4B,770
2380 DATA 4C4D4E4F50515253545556,891
2390 DATA 571D2020202020202020,916
2400 DATA E3466520676562656E2053,1050
2410 DATA 69652064656E204E616D65,966
2420 DATA 6E20646573902020202020,800
2430 DATA 6C64657320616E2C2064661,936
2440 DATA 7320696360206C6164656E,1003
2450 DATA 20736F6C6C3A9B1D1D1D20,806
2460 DATA 20202020202020202020,352
2470 DATA 2020203132333435363738,516
2480 DATA 3930313233343536373839,502
2490 DATA 3031323132333435363738,567
2500 DATA 3930313233343536373839,502
2510 DATA 3031323132333435363738,567
2520 DATA 3930313233343536373839,502
2530 DATA 3031323132333435363738,567
2540 DATA 3930313233343536373839,502
2550 DATA 3031323132333435363738,567
2560 DATA 3930313233343536373839,502
2570 DATA 3031323132333435363738,567
2580 DATA 3930313233343536373839,502
2590 DATA 3031321D1D427261766F21,744
2600 DATA 2053696520606162656E20,895
2610 DATA 6573206765736360616666,1071
2620 DATA 742065696E9B50757A7A6C,1160

```



```

2630 DATA 65E446DF47207A75206C6F,1215
2640 DATA 6573656E219B1D1D202020,769
2650 DATA 20496872652050756E6B74,986
2660 DATA 653A202020202020202020,447
2670 DATA 2020202020202020202020,743
2680 DATA A900B0C0B0A0300205CBA0,1060
2690 DATA 310205CCA946A00391CBA0,1290
2700 DATA 06A200B09C49C9FF00791,1434
2710 DATA CBE0C84C2F47A9008557AD,1391
2720 DATA 1206F01FA956C891CBAE06,1270
2730 DATA 06B09049C8E891C08C0E06,1352
2740 DATA BD9049C891CBE8B09049B0,1733
2750 DATA 1006209847A906B01A06A0,793
2760 DATA 05A90B204B44AD1206F00C,937
2770 DATA 207C4DA030A240A907205C,975
2780 DATA E420A047A90E0DC502A900,1319
2790 DATA 0DC002A900B0C60260A941,1311
2800 DATA C091C0C0A5C891C0C0A5CC,2033
2810 DATA 91C0B0A002A9F091C0A90E,1722
2820 DATA 0D0002A9470D0102A9C00D,1029
2830 DATA 0ED46048A0A060B0AD48D,1087
2840 DATA 19D0EE0A06A0A06C900D0,1085
2850 DATA EDA932B00A06040A260A9,1208
2860 DATA 0C9D4203E047DB482056E4,1170
2870 DATA A260A9039D4203A9629D44,1148
2880 DATA 03A9459D4503AD16069D4A,902
2890 DATA 03A9089D4B032056E460A2,1019
2900 DATA 00A9089D4203A9009D40B3,807
2910 DATA 9D4903A97D2056E460207F,1120
2920 DATA 4DAC0E06A97791C0C8A94B,1349
2930 DATA 91C0A062A2E4A9074C5CE4,1568
2940 DATA AD1206C909F01EAD1F00C9,1290
2950 DATA 05D017ADF0C2C92D0010A9,1302
2960 DATA 1C0DF02A909B0B06068D12,913
2970 DATA 064C0F4CAE1006C0A0E04D4,1137
2980 DATA BE1006E000D024EE0C06AD,1061
2990 DATA 0C06CD1006F01CA900B010,855
3000 DATA 060D04D4AC0E0610B1C069,1064
3010 DATA 0191C0C8B1C0690091C04C,1450
3020 DATA 62E4AC0E06A06060B0949,1110
3030 DATA 91C0E0C8B0904991C0A900,1703
3040 DATA 0D0C064C6E40A97F8D0ED4,1080
3050 DATA A21020A04CA903A2109D42,1019
3060 DATA 03A9F09D4403A94C9D4503,1114
3070 DATA A9069D4A032056E410034C,850
3080 DATA A94CA210A9059D4203A945,1061
3090 DATA 9D4403A9459D45DC48D749,1272
3100 DATA 03A9149D4803A9009D49B3,826
3110 DATA 2056E410034CA94CA200BD,1050
3120 DATA 4545C920F01BA9048D1A06,984
3130 DATA A900B55A900B55A014A9,1146
3140 DATA 09204B44A900B5544C2B40,764
3150 DATA 205249A900B01A06A900B5,850
3160 DATA 54A900B55A014A909204B,949
3170 DATA 44A9078D1F00AD1F00C905,1242
3180 DATA F007C90B0F54C2E40AD1F,1297
3190 DATA D0C905F0F9A314690AC514,1420
3200 DATA DBFC4CCC484C1F44A201E8,1382
3210 DATA 0D45459D4345E00AF004C9,1299
3220 DATA 20D0F1A92E9D4345A00AB9,1344
3230 DATA 4545E89D4345C8C00D0F4,1520
3240 DATA A920E89D4345E011D0F8A9,1592
3250 DATA 989D434560E045BA46684B,1272
3260 DATA AE494545334A78B64A9EA4,1208
3270 DATA 405B134C64060706060606,398
3280 DATA 0702020202020202020202,27
3290 DATA FF7D0A0A0A0A0A0A0A0A,1622
3300 DATA 0A0A0A0A0A0A0A0A0A0A,1200
3310 DATA 0A20202020202020202020,450
3320 DATA 2020202020A9BBA20F0D0,1079
3330 DATA 49D34AA0F5A0FAA0FAA0EC,1979
3340 DATA A0E5A0F220BA202020BA20,1227
3350 DATA 20202020202020202020,352
3360 DATA 202020BA9E0A0A0A0A0A,1217
3370 DATA 0A0A0A0A0A0A0A0A0A0A,1510
3380 DATA 98202020202020202020,630
3390 DATA 20202020202020202020,352
3400 DATA F4E0FEDE1F3A0A0F4E1F5,2454
3410 DATA F3E5EE49B1D1C1C1C1C1,1262
3420 DATA 1C1C1C1C1C1C1C1C1C1C,308
3430 DATA 1C1C1C1C1C1C1C00647275,527
3440 DATA 656360656E007369650062,937
3450 DATA 6974746500F3F4E1F2F400,1636
3460 DATA 756D0065696E006E657565,971
3470 DATA 730062696C640065696E7A,964
3480 DATA 756C6164656E000000F0F5,1132
3490 DATA FAFACE05F2000963000062,1677
3500 DATA 7900F4E0FEDE1F300F4E1,2266
3510 DATA F5F3E5EE4001C1C1C1C1,1451
3520 DATA 1C1C1C1C1C1C1C1C1C1C,308
3530 DATA 1C1C1C1C1C1C1C1C1C1C,308
3540 DATA 1C1C1C1C1C1C1C1C1C1C,308
3550 DATA 1C1C1C1C00676562656E00,625
3560 DATA 7369D44ACF4B650064656E,1200
3570 DATA 006E616D656E0064657300,843
3580 DATA 70757A7A6C65730065696E,1113
3590 DATA 500064617300696368006C,819
3600 DATA 6164656E00736F6C6C006F,961
3610 DATA 6465720064727565636865,1054
3620 DATA 6E0073696500F2E5F4F5F2,1633
3630 DATA EE00756D00736963680064,907
3640 DATA 617300696E68616C747376,1095
3650 DATA 65727A656963686E697300,1076
3660 DATA 6465720064697368657474,1075
3670 DATA 6500616E7A75736568656E,1078
3680 DATA 001C1C1C1C1C1C1C1C1C1C,200
3690 DATA 1C1C1C1C1C1C1C1C1C1C,308
3700 DATA 2020202020202020202020,352
3710 DATA 2020209B00000000000000,251
3720 DATA 0000000000000000000000,0
3730 DATA 0000000000000000000000,0
3740 DATA 0000000000000000000000,0
3750 DATA 00000000001010101010,6
3760 DATA 01010101010101010101,11
3770 DATA 010101010061636874756E,647
3780 DATA 6700657300697374006665,850
3790 DATA 686C657200D0040C04C6E75,1216
3800 DATA 6D6D65720000000000006175,647
3810 DATA 66676574726574656E0000,977
3820 DATA 00626974746500F3F4E1F2,1490
3830 DATA F400647275656368656E41,1150
3840 DATA 00010101010101010101,10
3850 DATA 01010101010101010101,23
3860 DATA 0D0D0D0D0D0D0D0D0D0D0D,143
3870 DATA 0D0D0D0D0D0D0D0D0D0D0D,130
3880 DATA 74686F6D61730074617573,1097
3890 DATA 656E640067727565737374,1092
3900 DATA 0064656E00726573740064,857
3910 DATA 65720077656C74000D00E1,897
3920 DATA F4E1F2E900697374006265,1479
3930 DATA 7373657200616C7300E32E,1038
3940 DATA 2E2E2E2E2E2E2E000D0D0D,361
3950 DATA 0D0D0D0D0D0D0D0D0D0D0D,143
3960 DATA 0D0D0D0D0D0D0D0D0D0D0D,143
3970 DATA 0D0D0D0D0D0D0D0D0D0D0D,143
3980 DATA 0D0D0D0D0D0D0D0D0A9C9D,442
3990 DATA 42032056E460980D4A900,1177
4000 DATA 05D5A21020A04C20A0A9A9,1300
4010 DATA D705F3A940B5F420E6D0A2,1052
4020 DATA 00A000B1F3297F30CC4CC7,1203
4030 DATA 4DE9209D0740E0C8E003D0,1656
4040 DATA F0A906B0D06060D12062011,782
4050 DATA 4720E04C40040AD1F00C9,1164
4060 DATA 06D0F960443A2A2E2A9BA9,1139
4070 DATA 000D2006A55885CBA55985,1155
4080 DATA CCA200BD6845DDA445F007,1429
4090 DATA B6CF20464DA6CFE0E03CD0,1617
4100 DATA 016018A5CB690485CBA5CC,1303
4110 DATA 690085CCE2006AD2006C9,1130
4120 DATA 0AD0D4A9000D200618A5CB,1170
4130 DATA 69D885CBA5CC690485CC4C,1540
4140 DATA 054DA5CB0D5A06A5CC0D50,1200
4150 DATA 06A200A000A900F1CB91CB,1209
4160 DATA C8C004D0F518A5CB692005,1519
4170 DATA CBA5CC690085CCE0E020D0,1710
4180 DATA E1AD5A0685CBAD500685CC,1437
4190 DATA 604C9F4D4C824DA9F00D1A,1267
4200 DATA 02A99E8D2002A94D0D2002,942
4210 DATA A900A2039D00D29D04D2CA,1274
4220 DATA 10F760A9030D0FD2A9000D,1207
4230 DATA 00D20D40060D42060D5206,871
4240 DATA 0D5406A9078D1F00A9C00D,1299
4250 DATA 2002A94D0D2902A9010D1A,009
4260 DATA C84DC34E0260A9000D07D2,1175
4270 DATA EE2006AD2006C905F0034C,1028
4280 DATA B34EA9000D2006AD5206C9,1075
4290 DATA 00D044AE40060D7F50C9FF,1372
4300 DATA F03020B94E0D4A06E00D7F,1352
4310 DATA 5020B94E0D4C06E00D7F50,1226
4320 DATA 20B94E0D4E06E00D7F5020,1100
4330 DATA B94E0D5006E00E4006A910,1119
4340 DATA 0D52064C2C4EA2300E4006,849
4350 DATA A90F4CE04DAE4A060D0F4E,1201
4360 DATA 0D04D2F002A9C60D05D2EE,1550

```

```

4370 DATA 4AB6AE4E06BD5F4FBD06D2,105B
4380 DATA F089A984187D5F4C554E,1112
4390 DATA A980BD07D2EE4E6AD5686,1242
4400 DATA F083CE5606AE5086BD7F4F,1196
4410 DATA F08D8D5086A908BD5606A9,1067
4420 DATA 008D8D1D2AD5086BD08D2F0,1210
4430 DATA 06A9A0186D5606BD01D2EE,1150
4440 DATA 5086AD5406D085A908BD54,964
4450 DATA 06AE4C06BD3F4FBD02D2F0,1186
4460 DATA 03AD5406BD03D2CE5406CE,1122
4470 DATA 5406EE4C06CE5206A9018D,1015
4480 DATA 1A0260182A2A2A2A600000,412
4490 DATA 000000C44EBF4F00000000,544
4500 DATA 0000000000000042425757,306
4510 DATA 4949424242425757494942,796
4520 DATA 0049496161575749494949,806
4530 DATA 6161575749004C4C676757,886
4540 DATA 574C4C4C4C676757574C00,847
4550 DATA 4200420042004200420042,396
4560 DATA 0042004200490049004900,351
4570 DATA 4900490049004900490051,446
4580 DATA 0051005100510051005100,405
4590 DATA 5100510057000000000000,249
4600 DATA 0000000000000000000000,0
4610 DATA 0000000000000000000000,0
4620 DATA 000000202A2C000F000000,141
4630 DATA 202A2D000F000F00000000,157
4640 DATA 0000000000000000000000,0
4650 DATA 00000020402040200040004,22
4660 DATA 0002000400020000000000,8
4670 DATA 0000000000000000000000,0
4680 DATA 003C000000000000003C00,120
4690 DATA 3C003C003C003200000032,280
4700 DATA 00350000003C0000000000,113
4710 DATA 00000000000000440044,136
4720 DATA 00440044003C04F8B5000,727
4730 DATA 000035003C000000000044,181
4740 DATA 0040000001000000000000,65
4750 DATA 320032003200320032002D,295
4760 DATA 3235003C00000000000000,163
4770 DATA 00002D00000020002D0000,130
4780 DATA 0000003200350000000000,103
4790 DATA 0000000044004000000000,132
4800 DATA 00000000020020002000,120
4810 DATA 2000200020002000000020,200
4820 DATA 002D000000000032000000,95
4830 DATA 000000002D002D002D002D,180
4840 DATA 002D002D002D0000002D00,180
4850 DATA 3200000000003500000000,103
4860 DATA 0000003200320032003200,200
4870 DATA 3200320032000000320035,253
4880 DATA 00000000003C0000003501,114
4890 DATA 00000000350020002D0032,180
4900 DATA 0035000000010000000100,55
4910 DATA 0001010000010100010101,7
4920 DATA 0001010100020101000201,10
4930 DATA 0100030101000301010001,12
4940 DATA 0101000201010001010101,10
4950 DATA 0101010202010103020C50,202
4960 DATA 0751010104010101050101,280
4970 DATA 0106020101070201010001,31
4980 DATA 0101010101010202010103,15
4990 DATA 0201010401010105010101,19
5000 DATA 0602010107020101000401,34
5010 DATA 00090401000A0501000005,46
5020 DATA 01000C0601000D0601000E,54
5030 DATA 0701000F00010000000101,26
5040 DATA 00010101010101020101,11
5050 DATA 01000101000101010501,13
5060 DATA 0101060101010001010100,22
5070 DATA 00010100010102000101,9
5080 DATA 0000010100000101050001,10
5090 DATA 0106000101000001010000,19
5100 DATA 0001090000010A0000010B,33
5110 DATA 0000010C0000010D000001,20
5120 DATA 0E0700010FFF00000000A9,461
5130 DATA 000D22060D2006A2072041,626
5140 DATA 43DD9852F0034C9252CAD0,1479
5150 DATA F2A206204143CAD0FAA200,1396
5160 DATA 2041439DC402E8E006D0F5,1434
5170 DATA A200204143CAE000D0F0A5,1381
5180 DATA 5085CBA55985CC200552AD,1307
5190 DATA 5E08519F5206C900F02220,1113
5200 DATA 4143AC200691CB203052CE,1050
5210 DATA 0006AD0006C900D0E0CE01,1036
5220 DATA 06AD0106C9FFD0E14CB351,1411
5230 DATA 20414305CDAC2006A5CD91,1227
5240 DATA CB203052CE0006AD0006C9,957
5250 DATA 00D0ECCE0106AD0106C9FF,1293
5260 DATA D0E24CB351D02041430D00,1291
5270 DATA 0629000D5E06AD0006297F,763
5280 DATA 0D0006C900F006A9000D01,905
5290 DATA 06602041430D0106204143,578
5300 DATA 0D00066010A5CB695005CB,1156
5310 DATA A5CC690085CCEE2206EE22,1361
5320 DATA 06AD2206C9C0F0034C5F52,1100
5330 DATA 30A5C0E9D005CBA5CCE91D,1040
5340 DATA 05CCA9010D2206AD2206C9,1102
5350 DATA C1F0034C7E5230A5CBE920,1417
5360 DATA 05CBA5CCE91E05CCA9000D,1615
5370 DATA 2206EE2006AD2006C920F0,1000
5380 DATA 0160606060203C40207451,706
5390 DATA 4CA040203C404C63400001,704
5400 DATA 001AC7C900FFE002E10200,1262
5410 DATA 4000000000000000000000,64
5420 DATA 0,0

```

32767 REM

Atari-Karteiverwaltung

Als Speichermedium benötigt man hier einen Cassettene recorder oder noch besser ein Diskettenlaufwerk. Besitzt ein Anwender auch einen Drucker, so kann er sich die Kartei seitenweise ausdrucken lassen. Die Bedienung des Programmes erklärt sich fast von selbst. Abgespeichert wird nach dem Abtippen mit CSAVE oder bei Diskettenbesitzern mit SAVE"D: KARTEI.BAS.

Nach dem Laden und Starten des Programms erscheint das Hauptmenü. Hier besteht die Möglichkeit, die einzelnen Funktionen bequem anzusprechen. Mit der "1" gelangt man in das Untermenü für "Laden und Speichern". Ist die entspre-

chende Funktion gewählt, wird man aufgefordert, den Filenamen einzugeben. Nach nochmaligen "Return" wird die Funktion ausgeführt.

Punkt 2 des Hauptmenüs dient der Festlegung der gewünschten Titelparameter, z.B. Name, Mitglieds-Nr., Wohnort, PLZ., Artikel-Nr. usw. Hier können 10 Parameter eingegeben werden.

Punkt 3 legt mit den unter Punkt 2 eingegebenen Parametern eine Kartei an. Nach Eingabe des Karteiblattes wird man gefragt, ob eine neue Eingabe erfolgt.

Drückt man im Hauptmenü die "4", so gelangt man in eine

Suchroutine. Gesucht werden kann eine bestimmte Seite oder ein bestimmtes Wort. Leider ist das Atari-Basic nicht gerade sehr schnell. So kann es bei größeren Karteien zu längeren Wartezeiten führen, wenn man ein bestimmtes Wort suchen läßt.

Mit Punkt 5 kann man in der Kartei blättern und nach Drücken des Sternchens im jeweiligen Karteiblatt Änderungen vornehmen.

Mit Punkt 6 hat man die Möglichkeit, an eine Kartei neue Karteiseiten anzufügen. Diese Seiten können danach mit Funktion 7 nach dem besten Parameter einer Seite sortiert werden.

Funktion 8 ist der Lesemodus. Mit ihm kann man in der

Kartei blättern und ebenfalls Änderungen vornehmen.

Punkt 9 des Hauptmenüs ist nur für Druckerbesitzer interessant. Nach Drücken der Taste 9 wird man aufgefordert, diejenige Seite einzugeben, die ausgedruckt werden soll. Bei Eingabe von "0" gelangt man wieder ins Hauptmenü.

Um Bedienungsfehler zu vermeiden, sind im Programm auch Fehlermelder enthalten, die eine fehlerhafte Eingabe anzeigen und wieder ins Menü zurückführen, sollte es trotzdem zu einem Abbruch kommen, ist es einen Versuch wert, mit GO-TO 500 erneut zu starten.

Um ein ungewolltes Unterbrechen des Programmablaufes zu erschweren, ist die Break-Taste außer Funktion gesetzt.

Somit kann nur mit der Reset-Taste abgebrochen werden. Abgeschaltet wird die Break-Taste in Zeile 500. Um den Speicherplatz zu erhöhen, kann

man die REM-Zeilen löschen, da sie im Programm durch keine Sprungbefehle angesprochen werden.

Karl Wurm

Karteiverwaltung

```

1 REM VERWALTUNG
2 REM VON WURM KARL/NIEDERGLEINKERSTR.
32/4407 DIETACH
3 REM *****
100 DIM A$(20000),B$(256),DTA$(25),TIS$(110),C$(250),DMS(25):TI=10:DMS="*****"
120 OPEN #1,4,0,"K":GOTO 500
150 REM TITEL *****
200 GRAPHICS 17:POKE 87,0:SETCOLOR 0,4,10
250 PRINT " GEBEN SIE DIE":POSITION 0,1:PRINT " GEMEINSCHTEN TITEL":POSITION 0,2:PRINT " E I N "
300 TIS=" ":TIS(110)=TIS(1):TIS(2)=TIS(1)
350 FOR T=10 TO 100 STEP 10:POSITION 5,3:PRINT T/10;" . TITEL"
400 POSITION 5,7:PRINT ".....":POSITION 4,7:INPUT DTA$
450 TIS(T,T+9)=DTA$:NEXT T:POKE 87,8
499 REM MENUE *****
500 GRAPHICS 0:POKE 82,1:POKE 16,64:POKE 53774,64:X=1
600 SETCOLOR 2,6,10:SETCOLOR 4,6,8:SETCOLOR 1,12,4
650 PRINT CHR$(125):PRINT " *****"
700 PRINT " *** M E N U E ***"
710 PRINT " *****"
715 PRINT "0-TITEL-MEN":DMS(3,(LEN(DMS))) : POSITION 5,8
720 PRINT " 1.SPEICHERN,LADEN"
730 PRINT " 2.TITEL NEU ANLEGEN"
740 PRINT " 3.KARTEI NEU ANLEGEN"
750 PRINT " 4.KARTEI SUCHEN"
760 PRINT " 5.KARTEI AENDERN"
770 PRINT " 6.KARTEI EINFUEGEN"
775 PRINT " 7.KARTEI SORTIEREN"
780 PRINT " 8.KARTEI LESEN"
785 PRINT " 9.KARTEI AUSDRUCKEN"
790 POSITION 12,21:PRINT "WAEHLE AUS:"
800 GET #1,T
810 IF T=49 THEN 1000
815 IF T=50 THEN 200
820 IF T=51 THEN A$="":GOTO 3000
825 IF T=52 THEN 4000
830 IF T=53 THEN 8000
835 IF T=54 THEN 3000
840 IF T=55 THEN 7000
845 IF T=56 THEN 8000
847 IF T=57 THEN 10000
890 GOTO 800
999 REM LADEN /SPEICHERN
1000 PRINT CHR$(125):DMS=" "
1010 PRINT " LADEN /SPEICHERN "
1020 POSITION 9,6:PRINT " 0. LADEN VON DISKETTE":POSITION 9,8:PRINT " 0. SPEICHERN VON DISKETTE"
1025 POSITION 9,10:PRINT " 0. LADEN VON KASSETTE":POSITION 9,12:PRINT " 0. SPEICHERN VON KASSETTE"
1030 POSITION 13,20:PRINT "WAEHLE AUS:"
1035 GET #1,T
1040 IF T=49 THEN DMS="D":GOTO 1100
1050 IF T=50 THEN DMS="D":GOTO 1500
1052 IF T=51 THEN DMS="C":GOTO 1100
1054 IF T=52 THEN DMS="C":GOTO 1500
1060 DMS="*****":GOTO 500
1100 PRINT CHR$(125):POKE 752,1
1110 PRINT " L A D E N "

```

```

1120 POSITION 3,8:PRINT "D I S K E T T E / K A S S E T T E":PRINT "4 E I N L E G E N ! ! "
1130 POSITION 13,22:PRINT " T A S T E "
1135 GET #1,T
1132 DTA$="":POSITION 13,14:PRINT "DATEINAMEN":PRINT "4"
1134 DMS(LEN(DMS)+1)=DTA$:DMS(LEN(DMS)+1)="."
1135 TRAP 1210
1140 OPEN #2,4,0,DMS
1142 INPUT #2,TIS
1150 FOR I=1 TO 99000
1152 B$=" "
1160 TRAP 1210:INPUT #2,B$:A$(LEN(A$)+1)=B$
1170 NEXT I
1210 FEHLER=PEEK(195)
1220 IF FEHLER<136 THEN POSITION 13,17:PRINT "4FEHLER NR.":FEHLER:CLOSE #2:FOR I=0 TO 1000:NEXT I:GOTO 1000
1230 CLOSE #2:GOTO 500
1500 PRINT CHR$(125):POKE 752,1
1510 POSITION 5,2:PRINT " S P E I C H E R N "
1520 POSITION 3,8:PRINT "D I S K E T T E / K A S S E T T E":PRINT "4 E I N L E G E N ! ! "
1530 POSITION 13,22:PRINT " T A S T E "
1532 DTA$="":C$="":POSITION 13,14:PRINT "DATEINAMEN":PRINT "4"
1534 DMS(LEN(DMS)+1)=DTA$:DMS(LEN(DMS)+1)="."
1536 OPEN #2,0,1,DMS
1538 PRINT #2:TIS
1540 FOR I=1 TO LEN(A$) STEP 250
1542 PRINT #2:A$(I,I+249)
1544 NEXT I
1550 CLOSE #2:GOTO 500
2000 PRINT CHR$(125):GOTO 200
3000 GOSUB 9000
3010 REM EINGABE *****
3020 DTA$="":B$=" "
3030 FOR S=3 TO 21 STEP 2:POSITION 12,5:INPUT DTA$
3040 GOSUB 3100
3050 NEXT S
3055 IF T=42 THEN RETURN
3058 TRAP 6000
3060 A$(LEN(A$)+1)=B$
3065 PRINT:PRINT " NEUE EINGABE (N/A) "
3070 GET #1,T
3075 IF T=78 THEN 500
3080 GOTO 3000
3100 FOR Z=1 TO 25:IF LEN(DTA$)<25 THEN DTA$(LEN(DTA$)+1)=" ":NEXT Z
3110 B$(LEN(B$)+1)=DTA$
3120 RETURN
3999 REM SUCHEN *****
4000 PRINT CHR$(125):SETCOLOR 2,6,10:SETCOLOR 4,6,8
4010 PRINT:PRINT " ***** S U C H E N E ***** "
4020 POSITION 4,8:PRINT " 0. BESTIMMTE SEITE":PRINT "4"
4025 POSITION 12,21:PRINT "WAEHLE AUS:"
4030 GET #1,T
4040 IF T=49 THEN 4100
4050 IF T=50 THEN 4200
4060 GOTO 500
4100 POKE 752,0:POSITION 10,15:PRINT "WELCHE SEITE :":INPUT X
4110 GOTO 8000
4200 Z=0:POKE 752,0:POSITION 10,15:PRINT "WORT EINGEBEN!":POSITION 9,17:INPUT DTA$
4210 Z=Z+1
4230 TRAP 500:IF DTA$=A$(Z,Z+LEN(DTA$)-1) THEN X=INT(Z/250)+1:GOTO 8000
4240 GOTO 4210
4999 REM AENDERN *****

```

```

5000 GOSUB 3020
5010 Z=(X-1)*250+1:ZZ=Z+250
5020 A$(Z,ZZ)=B$:GOTO 8100
5099 REM SPEICHERUEBERLAUF
6000 PRINT CHR$(125):POSITION 10,9:PR
NT "SPEICHERUEBERLAUF":PRINT "++
DÄTEN ABSPREICHEN I"
6010 FOR I=0 TO 1000:NEXT I
6020 GOTO 500
6099 REM SORTIEREN *****
7000 POSITION 6,21:PRINT " B I T T E
M A R T E N"
7100 FOR I=1 TO LEN(A$) STEP 250
7110 B$="":C$="":TRAP 7150
7120 B$=A$(I,I+249):C$=A$(I+250,I+499)
7130 IF B$<C$ THEN A$(I,I+249)=C$:A$(I
+250,I+499)=B$:POP:GOTO 7100
7150 NEXT I
7160 GOTO 500
7990 REM KARTEI LESEN *****
8000 GOSUB 9000
8010 REM AUSGABE *****
8020 XX=(X-1)*250+1
8030 TRAP 500
8040 FOR S=3 TO 19 STEP 2
8050 POSITION 13,5:PRINT A$(XX,XX+24)
8055 XX=XX+25
8060 NEXT S
8070 POSITION 13,21:PRINT A$(XX,XX+24)
8090 POSITION 0,0:PRINT "      KOM
FOLETT NR.>":X;"(      ";
8100 POKE 752,1:POSITION 0,23:PRINT "
<>=BLÄTTERN *-BENDERN
";
8110 GET M1,T
8120 IF T=60 THEN X=X-1:IF X=0 THEN X=
LEN(A$)/250

```

```

8125 IF T=60 THEN 8000
8130 IF T=62 THEN X=X+1:IF X=LEN(A$)/2
50+1 THEN X=1
8135 IF T=62 THEN 8000
8140 IF T=42 THEN POKE 752,0:GOTO 5000
8150 GOTO 500
8999 REM EIN-AUSGABE MASKE *****
9000 PRINT CHR$(125):SETCOLOR 4,0,8:SE
TCOLOR 2,8,10
9010 TRAP 200:FOR S=3 TO 21 STEP 2:POS
ITION 1,5:PRINT TI$(TI,TI+9);": ";TI=
TI+10:NEXT S:TI=10
9020 RETURN
10000 PRINT CHR$(125)
10010 PRINT "      AUSDRUCK
E N      ":POSITION 0,20:PRINT " MIT
> 0 < ZUM MENUE":TRAP 10200
10020 POSITION 10,10:PRINT "WELCHE SEI
TE ":INPUT X
10025 IF X=0 THEN GOTO 500
10030 XX=(X-1)*250+1
10040 OPEN M2,0,0,"P:"
10050 FOR I=1 TO 10
10055 TRAP 10110
10057 C$=A$(XX,XX+24)
10060 PRINT M2;"      ";TI$(TI,TI+9)
;": ";A$(XX,XX+24)
10070 XX=XX+25:TI=TI+10
10080 NEXT I:TI=10:X=0
10085 CLOSE M2
10090 LPRINT:LPRINT:LPRINT
10100 GOTO 10000
10110 CLOSE M2:PRINT "+++ L E E R
E S E I T E !":FOR Z=0 TO 400:NEXT Z
:GOTO 10000
10200 POSITION 10,15:PRINT "FEHLER NR.
":PEEK(195):CLOSE M2:FOR I=0 TO 600:N
EXT I:GOTO 10000

```

Disk-Collector

Wer in seinen Datenwust endlich einmal Ordnung bringen will, der sollte sich einmal den Disk-Collector zu Gemüte führen, denn der speichert pro Datenbank die Directories von 32 Disketten mit je 63 Files. Nach dem Booten des Disk-Collectors (Autorun.Sys) wird automatisch eine Datenbank eingelesen, sofern die eingelegte Diskette eine solche enthält. Um die Datenbank zu wechseln, tauscht man einfacher die Diskette aus und drückt auf die START-Taste, wonach sich das Programm erneut startet.

Will man eine Programmbibliothek erstellen, so muß der Computer zunächst mit den Directories der einzelnen Disketten gefüttert werden. Um später das Auffinden von Programmen einfacher zu machen, gibt man jeder eingespeicherten Diskette einen Namen oder eine Nummer. Ich selbst bevorzuge zusätzlich die Erweiterung ".A" bzw. ".B" für Vorder- und Rückseite. Das Einlesen der Disketten erfolgt über Funktion 1 des Disk-Collector-Menüs. Ein Verzeichnis aller ein-

gelesenen Disketten erhält man mit Funktion 4.

Disketten, die noch keine Files besitzen, werden vom Disk-Collector nicht gespeichert. Es ist natürlich möglich, eine bereits gesicherte Directory über Funktion 1 zu erneuern oder über Funktion 3 ganz aus der Bibliothek zu löschen.

Einen Blick auf die Files einer Diskette kann man durch Funktion 2 erlangen. Will man jedoch die zugehörige Diskette zu bestimmten Files finden, so wählt man Funktion 5. Hier wird zunächst der Name des Programmes eingetragen (ohne D: aber mit .), nach dem gesucht werden soll. Wildcards (nur *) können wie im DOS benutzt werden. So listet z.B. die Eingabe "A*.BAS" alle Basicprogramme mit den entsprechenden Disketten auf, die mit A beginnen. Mit einem solchen Suchprogramm lassen sich Files und Disketten schnell und problemlos finden.

Über Funktion 6 kann der Disk-Collector verlassen und das DOS-Menü aktiviert wer-

den. Hat man eine Programm-Bibliothek erweitert, so sollte man sie jetzt auf einer Disk-Collector-Diskette sichern. Je-

de Datenbank belegt bis zu 199 Sektoren und erhält den Filenamen COLL.DAT.

Stefan Funk

```

10 REM *** Disk-Collector
15 REM *** erstellt mit QBJ-LST-3
20 REM *** (c) 1986 by Fys
25 REM ***
30 REM *** als Autorun.Sys
35 REM *** Modulfrei booten!
40 REM ***
45 REM *** in einer Disk-Collector
50 REM *** Programmansammlung haben
55 REM *** 32 Disk-Directories
60 REM *** platz
65 REM ***
70 REM *** einzelne Files koennen
75 REM *** schnell gefunden werden
80 REM ***
85 REM *** DOS-Filenamen (mit Wild-
90 REM *** cards) sind beim Suchen
95 REM *** sehr hilfreich
100 DIM X$(64),Y$(64),DISK$(20),Z(16)
110 I=110-999:DISK$="D:AUTORUN.SYS" "Y$="A06"
120 GOSUB 800:ANZAHL=Y
200 ? CHR$(125):? CHR$(253):? "Datentest...":? :FOR I=1 TO
300:NEXT I:POKE 559,0
210 TRAP 950:RESTORE
220 FOR I=1 TO ANZAHL
230 READ X$,Y$:IF I+999<>PEEK(183)+256*PEEK(184) THEN 940

```



```

232 IF LEN(X$)<>32 THEN 970
234 IF LEN(Y$)<>3 THEN 980
236 GOSUB 700
238 GOSUB 800
240 IF X<Y THEN 990
250 NEXT I:POKE 559,34
252 TRAP 260
254 READ X$
256 GOTO 960
260 ? CHR$(125):? CHR$(253):? "Diskette fuer ";DISK$:" ei
nlegen und 'START' drucken..."
270 IF PEEK(53279)<>6 THEN 270
300 ? CHR$(125):? CHR$(253):? "Schreibe File...":?
310 TRAP 900:RESTORE :OPEN #1,B,0,DISK$
320 FOR I=1 TO ANZAHL
330 READ X$,Y$:GOSUB 710
340 FOR J=1 TO 16
342 IF Z(J)>255 THEN 400
344 PUT #1,Z(J):NEXT J
350 NEXT I
400 ? CHR$(125):? CHR$(253):? "Arbeit beendet...":? DISK$;
"-File erstellt!!!"
410 CLOSE #1
420 END
700 X=0
702 FOR Q=1 TO 32:IF X$(Q,Q)<"A" OR X$(Q,Q)>"Q" THEN 920
704 IF X$(Q,Q)="Q" THEN X$(Q,Q)="A"
706 NEXT Q
710 FOR K=1 TO 16
720 Z(K)=16*(ASC(X$(K*2-1,K*2-1))-65)+(ASC(X$(K*2,K*2))-65
):X=X+Z(K)
730 NEXT K:RETURN
800 Y=256*(ASC(Y$(1,1))-65)+16*(ASC(Y$(2,2))-65)+(ASC(Y$(3
,3))-65)
802 FOR Q=1 TO 3:IF Y$(Q,Q)<"A" OR Y$(Q,Q)>"P" THEN 930
804 NEXT Q
810 RETURN
900 ? "Fehlermeldung...":PEEK(195)
910 ? CHR$(253):POKE 559,34:END
920 ? "1.Datenkette/";Q$".Zeichen (";X$(Q,Q);") falsch":?
"in Zeile...":LIST I+999:GOTO 910
930 ? "2.Datenkette/";Q$".Zeichen (";Y$(Q,Q);") falsch":?
"in Zeile...":LIST I+999:GOTO 910
940 ? "Zeile ";I+999;" fehlt !!!":GOTO 910
950 ? "Zu wenig Daten !!!":GOTO 910
960 ? "Zu viele Daten !!!":GOTO 910
970 ? "1.Datenlaenge falsch in Zeile...":LIST I+999:GOTO 9
10
980 ? "2.Datenlaenge falsch in Zeile...":LIST I+999:GOTO 9
10
990 ? "Zeichenfolge falsch in Zeile...":LIST I+999:GOTO 91
0
1000 DATA PPPPAKICKLFLKJAAIFEBKJABIFIEKJDA,HIA
1001 DATA IFIFKAAKJAAJJAAGIINAPKKFIFIFIC,HMF
1002 DATA M6ICKJAAIFIAKCAIBIBKAAJBIAMINA,I11
1003 DATA PLMKOEICNAPCKJEHINDACKJLLINDAC,IMA
1004 DATA KJPKINMGACKJAAINMFACKJCCINCPACKC,HCA
1005 DATA AAKAAIINAPNMKNAPKCANELECAJILED,IMN

```

```

1006 DATA ADCADKLECAIDLEKJAHKAIKACLECAFME,HKJ
1007 DATA KJAAINCPACKJAJINAAACKJKJINABACKJ,FDD
1008 DATA MAINAONEKJOGINDACKJKIINDBACKJCD,HFK
1009 DATA INCPACKJDAINMGACKJAOINMFACKJPPIN,HCM
1010 DATA PMACKNBPNAMJAGNAADEMAAKIKNPMACKJ,HKE
1011 DATA PPPADPKMNDKIKCABNNNDKINABDLNFKI,LBN
1012 DATA EILNNEKIEIKJPPINPMACKJEAJNAONEGA,ILE
1013 DATA DIOIOIINADCEJHKIAGBPMFKLBQJOLA,JHO
1014 DATA BKEELBBIPFLBNDHLCBLBALDHAAHAAHA,8MB
1015 DATA ECEOKJBAACBAACHAHAPACHAPACBAAC,EKD
1016 DATA BAACBAACBAACBAACBAHAPACEBOKIEI,DNB
1017 DATA KJCCINBINAKJCCINAAEKJCAINAAACKJ,8GN
1018 DATA KJINABACGIEAEIKJBCINBINAKJCBINAA,FLA
1019 DATA NEKJDHINAAACKJJKJINABACGIEAEIKJCC,FDA
1020 DATA INBINAKJCCINAAEKJAJINAAACKJJKJIN,8MB
1021 DATA ABACGIEAAAAAAAAAAAAAAAAAAAKAK,ALP
1022 DATA AKAAACECJDDCLAAANAAACDPCPCMCN,8BM
1023 DATA CFCDDCEPDCAAAAKAKAKAAAAAAAAAAAA,APL
1024 DATA AAAAAAAAAAAAAAAAAAAAAAABDBPHAHJ,BLL
1025 DATA HCBJ6HGIHEAIBDAJAABCHJAACBHF60BL,E0B
1026 DATA HJHDBP6GHEANC6HJHDAHBIB6AAAAAAA,D1J
1027 DATA AAAAAAAAAAAAAAAAAAAAAAABDBPHAHJ,BLL
1028 DATA AAGDHCBFBHEGFBGAABCHJAADDHEGFB6,FCF
1029 DATA B6GOACBHF60BLAAAAAAAAAAAAAAAAAA,CED
1030 DATA AAAAAAAAAAAABMBMBMAAAC6GJHDBLAN,BMH
1031 DATA C06P6BMB6F6DHEGPHCAAAACIB6HFAHE,F6J
1032 DATA 6N6F60HFBFAAAAB0BDBDAAAADLBBDNAO,DAL
1033 DATA A0A0CE6JHCBF6DHEGPHCHJAAGFBJ60BF,FFC
1034 DATA HCAACE6JHDBLAAGMB6B6F60DLBCDNO,EHJ
1035 DATA A0A0CE6JHCBF6DHEGPHCHJAAGBHFHDBH,F6B
1036 DATA 6FBC6F60AAAAAAADLBBDNAO,CDD
1037 DATA A0A0CE6JHCBF6DHEGPHCHJAAGMB6F6H,F6E
1038 DATA 6D616F60AAAAAAADLBBDNAO,CDI
1039 DATA A0A0CE6JHDBLF6HEGFBGAAB6HFB6BM,FEP
1040 DATA 6JHDB6F60AAAAAAADLBBDNAO,CLO
1041 DATA A0A0AHCB6PHHC6B6NGNAADHFB6D616F,FFJ
1042 DATA 60AAAAAAADLBBDNAO,BAK
1043 DATA A0A0CD6P6MB6F6DHEGPHCAAH6FHC6M,FFM
1044 DATA 6BHDHDBF60AAANBOAAC6CPDAAAAC6J,DFB
1045 DATA HEM6FAACBHF60BLHEGJ6P60AABNB6HE,FMF
1046 DATA AAD6BHD6F6AAB6F6DHB6B6F616MB,FKA
1047 DATA 60ABAAA6CBKJAMJNECAD6F60EEIJ1,F60
1048 DATA EIIKEIKCBKJADJNECAD6IJNEAD6IJN,FKL
1049 DATA EFAD6IJNEKADKJAAJNELAD6F60EKCBA,F6B
1050 DATA KJAFJNECADKJAAJNEEADKJCAJNEFADKJ,FHE
1051 DATA BEJNEIADKJAAJNEJADEMF60EKCBAJAH,FH6
1052 DATA JNECADKJAAJNEIADJNEJADEMF60EELDK,F6H
1119 DATA A0DD6P6MB6AACECPDANC6F60HFB6FAA,DPF
1120 DATA 6H6F6MB6B6F60AAH6F6H6G6F60DLCK,FLK
1121 DATA DNB6AADLCODN6F6J60DD6P6MB6AACE6B,EWP
1122 DATA H6B6F60HDB6HEHKAAG6F6D6B6H6CADA,FJD
1123 DATA H6B6F6H6G6F60DLCKD6B6AADLCODN6F6J,EPH
1124 DATA 60DAHCB6PHHC6B6NG6B6NG6FAAHKF,BCD
1125 DATA BNAADH6F6D616F60AAG6J60G6H6F6CBF,FIC
1126 DATA 60AND6F6HEH6F60AHAH6F6NAACN6F,EMK
1127 DATA 60H6F6AN60AAAAAAADLBBDNAO,BHD
1128 DATA AAKNEELBNAB6KMEDLBKCAHKJAAJJAAG,6BJ
1129 DATA NIMKBAPJMBIF6BIF6BIF6BIE6AKJIAIN,JNN

```

```

1056 DATA AABC6F8M6F9HHEABAABOKJININDAACKJ,FDA
1057 DATA LLINDBACKJCINCIPACKJPKINAANCKJPL,HKK
1058 DATA INACNCKCAAKAAKNALNEBISGFBEINAKNE,BCL
1059 DATA INBHNAIKCJPAEKEKEKEKBIBJKAINABNC,GMA
1060 DATA INADNCIINAQBHKMANDEMSKKIKJHAINJF,JKM
1061 DATA LKINJHLKINKDLKKJAINEELBKJIKINDA,IJN
1062 DATA ACKJLKINDBACKJCBINCPACCAANLEMEIE,FNG
1063 DATA NABBKFIEHJCBAJCBOIFOBIFOBIFOBIE,KCB
1064 DATA EMBLKMJIETODEELBKJACINJFLKINJHLK,IDL
1065 DATA CAINLANJEKPAAIMJEONAPFBIEMMGKLGI,IML
1066 DATA KIJIEIIJIAKAKAKKIIMEDLBKCAALNBD,GBA
1067 DATA KNJJAAAGMIODAIANAPEKJJLLINDAACKJ,IFE
1068 DATA LKINDBACKJCIBINCPACCAINLAKJAGKCEL,FPL
1069 DATA KAKLCAPIKKBAAAGCAPLKPEMBHLAKJACIN,HDI
1070 DATA KDLKKJAAIFICBIKICAFKLEKFIbineclB,HPB
1071 DATA CABIKLDADCKNABCANJCANADDKCACKAAA,FPD
1072 DATA DAAKNAAFKJAJBJIAMILNAACACANDLAJB,GPC
1073 DATA IAHIOIOAANNADJFKJABIJBAMIFIAKFIB,ILD
1074 DATA GJAAIFIBOBIKFCIMJDPPAADENSKKMC,AHL
1075 DATA OOKKKFICNABFCAPLKPKJKHINDAACKJLK,IOA
1076 DATA INDBACKJCINCIPACENJDKLKJAAIFIANK,BCO
1077 DATA ECLBIFIBOGIBOGIBKAPPKFCIBIAEHMB,JLA
1078 DATA KLCEGJHDGLGFHEHEGFBOBGCFHKAADAANK,FPP
1079 DATA HFGNAACMBBEGFBOAAAAAAAAAAAAAAAAAAAA,CKG
1080 DATA BKCNSJHEAADCGFHEHFHCBOAAHKHFBNAA,EDA
1081 DATA CNGFBOHFGBFAANANBQAAAAAAAAAAAAAAAA,CBC
1082 DATA AACESJHDGLGFHEHEBFAGSJHDHEAABCGF,FDE
1083 DATA HCGFBJHEHDAAGHBGFMDHABFBJSDSIBFHC,SEG
1084 DATA HECOGFHFGFAACEBJHCGFBDHEBPCHJAA,FHB
1085 DATA GBGBEGFBOAADLCKDNGBAADLCOGNBFGJ,EHL
1086 DATA BOCEGJHDBLAGFBJSGOMBFGHBFOAAHF,FKD
1087 DATA BOGEAADEGBHDHEGFAGAEHCHFGBDGLBF,FJG
1088 DATA BOCOGFHFGFAACEBJHCGFBDHEBPCHJAA,FHA
1089 DATA HHGJHCSEAAGHBFGMBGEGFBOAAAAAOAQ,EKC
1090 DATA ADCEGJHDBLAGBSGFBIGMFHCAABKAACB,EDN
1091 DATA HFGBGLHEGJGPBOAAGBGJGDGIEHAAGBMF,FPE
1092 DATA HDGBHFGFBIHCGBBHCCGEGJHCGFBDHEBP,GBM
1093 DATA HCHJAAAGPBGIBGFAACBGJSMGFHDAANAA,EHF
1094 DATA CLFGJGDBGFAADDHAGFBJSDSIBFHCXFBD,FMC
1095 DATA SHDGGFHCHKFGFGJSDSIBOGJHDAAGSEFHC,GAH
1096 DATA AAGFBJGOGHGFEHCBBGHFBDFGOAAACE,FIA
1097 DATA GJHDBLGFEHEHEGFBOBKAAAHDCBFHEHFHC,FHK
1098 DATA GOAHAAGBHFSGFIHCHEAAHKHFHCHFGFBD,FKB
1099 DATA GLAAHKHFGNAACISBHFHAMENBFGBOHFGF,FMD
1100 DATA AACJGNAADDHAGFBJSDSIBFHCAAHDGJGD,EPD
1101 DATA GEAABLGFGJBOGFAACFBJBOGBHBGGCFBD,FBJ
1102 DATA ABCEGJHDBLGFEHEHEBFBOBGBFMKADAANK,FFF
1103 DATA HFGBAACMBGPFHDBDSIBFGAAAAAAAAAAAAA,DPD
1104 DATA BKCFBJGOGHBFGHBFGCFBGOFBGAACEBJHD,FEI
1105 DATA BLAOGBGJGDGIEHAAGBPHCBISBBOGEBF,FGB
1106 DATA BOCEGJHCGFBDHEBPCHJAGABGPGFHDGD,GBJ
1107 DATA SIBFOAABPAADLCKDNGBAADLCOGNBFGJ,DNB
1108 DATA BOCEGJHDBLGFEHEHEGFBOBGMKADAANK,FMC
1109 DATA HFGNAACBHFHDGMBFGCFBGAAAAAAAAAAAAA,DOM
1110 DATA BKCEGJHCGFBDHEBPCHJAGABGFMCAACE,FAQ
1111 DATA GJHDBLGFEHEHEGFBOBQAAAAAAAAAAAAAAAA,DDF
1112 DATA AAAAAAAAAAAAAADHGFBJHEHGFHCAABNGJ,DCG
1113 DATA HEAAAHDCGFHEHFHCBOAHABAAAAAAAAAAAAA,COD

```



```

1114 DATA AACB9JHDLGANCDBP6M6M6F8DH6PHCAA,EPP
1115 DATA CEB9HEGF60HDBBHEHKAHNGJHC6EAGH,FKL
1116 DATA 9F6M6FND6F60ADADACEB9HDLGANCDBP,ELA
1117 DATA 6M6M6F8DH6PHCAAACEB9HEGF60HDBBHE,GAJ
1118 DATA HKAHNGJHC6EAGH6FCHDH6GF6M6M6HEAD,FKN
1119 DATA CAINLAMJEOPADHMEKMAPFKJFBINDAAC,ICH
1120 DATA KJLLINDBACKJPKINMGACKJAAINMFACKJ,HMC
1121 DATA CCINCPACKCAAKAAIINAPNKNAPKCANP,IAK
1122 DATA LECAJILEBDFCAOKLEDBACAIOLKJPP,HBN
1123 DATA INPMACIF6BMAKAAEMGKKIKJAEINMBAC,6CH
1124 DATA CAIOLEMBHLACAINLANJLJLAPJEMGKKI,IFN
1125 DATA KAALKJAAJOPKPIIBAPKKOPKAKPKJAM,ILC
1126 DATA CACHLAKAALLJOPKPAIIBAPIGI6IEM,HIC
1127 DATA 6KKIBAKJAEINMBACKJAAKAHJJBDKNII,6KF
1128 DATA BAPKKCBKAKNKJACACHLAKAHLJBDKN,6NJ
1129 DATA NAAIIBAPIGI6IEMGKKIKAPPHINEIEPA,JDF
1130 DATA BJJIEIAKAKAKKKKAAALJBDKNNAAGNA,FIN
1131 DATA AKOIMIAINAPCGIKIMIGASIKIEM6LE,INC
1132 DATA KJAAIFIAKJDAIF18IIPAAJ0618081808,IMC
1133 DATA IBEM6CLEBAKAAKJAAJJAACAJAACBJJ,FJI
1134 DATA AACCMINAPESANIKJAJINAAACKJKJINAB,HAH
1135 DATA ACEN6CDEKCAKJAMJNECDEMFOEKCCA,6DF
1136 DATA KJADJNECACKJFBJNEEADKJLJNEFADKJ,6ED
1137 DATA AEJNEKADKJAAJNELADEMF60EKCAKJAH,FKH
1138 DATA JNECACKJIAJNEEADKJFBJNEFADKJCN,6LK

```

```

1205 DATA ETADKFIDJNEJADEMF60EKJAHINLJLEKJ,HDF
1206 DATA AEINKKLEBAKJALINLJLEKJAIINKKLEGA,HPJ
1207 DATA KJIEIFIAKJAAIFIBKJACIFICKJAAIFID,HEE
1208 DATA CALGLEDACKJAAIFIAKJAGIFIBKJAAIF,6HG
1209 DATA ICKJABIFIDCALGLEDABGKJAAIFIAKJDA,6IL
1210 DATA IFIBKJAAIFICKJFIDIOJDAIFIDCALBLE,HMD
1211 DATA 6APPPPIALKJHLLHAHAHAHAECJLKNEBIA,IPF
1212 DATA LKHAHAHAHAECNLKMBAAACHAHABABABAE,FKG
1213 DATA IKLKHHAHAHAECFLKMBABAEJLLKHAHA,6OE
1214 DATA HAAECMDKNEBKHLKHAHAHAHAHAECOLKN,IDO
1215 DATA HAAECAACABAACBAACBAACBAACBAACBA,6KN
1216 DATA ACBAACHAHAECBDKOEJBLKHAHAHAHAEC,FKF
1217 DATA DLKOEJBLKHAHAHAHAECFLKOBACPLKN,HLM
1218 DATA EBNOLKHAHAHAHAECFLKOEJBLKHAHAHA,IDL
1219 DATA HAECJLKOEJBLKHAHAHAHAECFLKOBAC,6KJ
1220 DATA PLKNEBAILHAHAHAECNLKOBACAAACAA,6PA
1221 DATA ACAACAAACAAACAAACAAACAAACAAACAA,ABA
1222 DATA ACAACAAACAAACAAACAAACAAACAAACAA,ABA
1223 DATA ACAACAAACAAACBAECPLKOEBAOLLHAHA,6ON
1224 DATA HAAECBLKPEBEHLLHAHAHAHAECEDKPEB,6GE
1225 DATA FBLHAHAHAHAECBLKPHAEFBLHAHAHA,6DP
1226 DATA HAAHAHAHAECLLKPBAAACHAHAECACAAAC,6MC
1227 DATA AAACAAACAAACAAACAAACAAACHAHAECPL,CCJ
1228 DATA KOEBHLLHAHAHAHAECFNLKLEBINLLAAPP,6KC
1229 DATA PPOACOBACAAKIQ0000000000000000,6GM

```

Das MIDI-Disk-Programm für den Atari 800 XL

Wer das Atari-MIDI-Interface aus der CK 6-7/1986 nachgebaut hat, wird sich sicherlich mehr MIDI-Software dazu wünschen. Deshalb stellen wir hier ein weiteres Programm vor.

Grundsätzliches

Zunächst einmal besteht das Programm aus zwei Teilen: Einem Kernteil, der in Assembler geschrieben ist, und einem Rahmenprogramm in Basic. Basic ist bekanntlich einfacher, aber für die Verarbeitung von MIDI-Daten leider zu langsam. Deshalb wurde der ganze zeitskritische Teil – wie Senden und Empfangen, Schreiben und Lesen – in Maschinensprache erstellt. Darüber hinaus werden die Assembler-Programmenteile für den Empfang von MIDI-Daten Interrupt-gesteuert, das heißt, daß der Computer alle anderen Arbeiten sofort unterbricht, sobald das Interface MIDI-Daten vom Keyboard empfängt.

Die Interrupt-Steuerung erfolgt in zwei Ebenen: Beim Datenempfang wird vom UART des Interfaces der normale Systeminterrupt (IRQ) ausgelöst. Beim Senden von Daten an das Keyboard steuert ein Zeittakt die Datenausgabe. Die dafür erforderlichen Zeittaktwerte werden im sogenannten VBI (Vertical-Blank-Interrupt) fortgeschrieben. Mit dieser Steuerung erhält die MIDI-Datenübertragung Vorrang vor allen anderen Aufgaben.

Das Basic-Programm übernimmt im wesentlichen den Bildschirmaufbau, das Directory sowie das Speichern und Laden von MIDI-Daten auf Diskette, wobei für letzteres ein Maschinen-Unterprogramm eingesetzt wird.

Das Assembler-Programm übernimmt die Kommunikationsaufgaben mit dem Keyboard (über das MIDI-Interface) mit den Funktionen "Record/Play" sowie der Datenausgabe auf dem Bildschirm.

Programm-Funktion

Wurde das Listing richtig eingegeben, sollte nach dem Initialisierungsbild (für die Zeitspanne, in der die Maschinendaten eingepokt werden) das folgende Hauptmenü erscheinen:

```

REC.: TASTER
PLAY: TASTE P
DIR.: TASTE D
SAVE: TASTE S
LOAD: TASTE L

```

Anhand dieses Menüs lassen sich die Programm-Möglichkeiten recht gut beschreiben. REC steht natürlich für Record und heißt, daß nach Aufruf dieser Funktion MIDI-Daten vom Keyboard empfangen und in einem festen Datenspeicherbereich des Atari (dez. 24000-38000) abgelegt werden. Ein Einspeicher-Vorgang wird durch Drücken der Leertaste oder bei Erreichen der Speicherkapazität beendet und führt zurück zum Hauptmenü.

Mit PLAY läßt sich das eben Eingespielte anhören. Auch hier kann mit der Leertaste diese Funktion verlassen werden.

Die wesentlichen MIDI-Daten werden sowohl beim Einspielen vom Keyboard als auch beim Senden an das Keyboard, z.B. nach dem Laden von der Diskette, angezeigt. In der REC-Funktion wird zunächst im Kopf die Zeile "KEYBOARD SENDEBEREIT" angezeigt. Diese Echtzeit-Meldung erscheint aber nur, wenn das Keyboard tatsächlich sendebereit ist und alle MIDI-Kabel richtig angeschlossen sind. Da für diese Meldung die MIDI-Commands "FE" und "F8" verwendet werden, erscheint diese Kopfzeile natürlich nur dann, wenn das Keyboard diese Statusbytes sendet. Zum Zeitpunkt der Programmherstellung stand mir nur das Yamaha-Keyboard PS6100 zur Verfügung, und das sendet diese Daten.

Im Hauptfeld erscheinen dann fünf Datenspalten mit folgender Bedeutung:

Spalte 1: Adresse. Diese wird in dezimaler Form angezeigt. Das hat den Vorteil, daß man im Basic-Steuerungsprogramm nicht ständig umrechnen muß. Und falls dieses Programm einmal um einen Editor erweitert wird, vereinfacht das die Arbeit beträchtlich. Die angezeigte Adresse gilt immer für das erste MIDI-Byte in der Spalte "CMD". Die Adresssprünge kommen daher, daß ja mit jedem Statusbyte immer zwei Zeittaktwerte mitgespeichert werden, die aber am Bildschirm nicht ersichtlich sind.

Spalte 2: CMD wie Command. Hier stehen die MIDI-Statusbytes in hexadezimaler Form. Um damit etwas anfangen zu können, ist es unerlässlich, sich mit den MIDI-Spezifikationen auseinanderzusetzen. Handelt es sich hier allerdings um einen NOTE ON- oder NOTE OFF- Befehl, dann stehen in dieser Spalte nicht die Hex-Bytes 90 bzw. 80, sondern die gespielte Note in Klarschrift, also C, Dis oder Cis 3.

Spalte 3: DAT1. Dies ist das erste MIDI-Daten-Byte und zwar wiederum in Hex-Form. Im Falle eines NOTE-ON- oder NOTE-OFF-Bytes steht hier noch einmal die gespielte Note bzw. gedrückte Taste als hexadezimale Zahl.

Spalte 4: DAT2. Hier handelt es sich um das zweite MIDI-Daten-Byte, sofern zum entsprechenden Statusbyte ein solches überhaupt gesendet wird. Es sind ebenfalls Hex-Zahlen, die bei NOTE ON den Anschlags-Dynamikwert und bei NOTE OFF den Wert 00 anzeigen. Ist das Keyboard nicht anschlagsdynamisch, wird bei NOTE ON eine hexadezimale "40" angezeigt (= dez. 64 = die Hälfte zwischen 0 und 127; das entspricht nach der MIDI-Spezifikation einem Anschlagswert von Null).

Spalte 5: Kommentar. Hier werden nur die Hauptbegriffe angezeigt, also z.B. CONTROLLER, aber nicht SUSTAIN, falls gerade dieser mit der CONTROLLER-Meldung verändert wurde.

Im unteren Teil des Bildschirms werden Bedienungsanweisungen (z.B. DRUECKE LEERTASTE) oder Fehlermeldungen (z.B. DATENSPEICHER LEER) angezeigt. In der Funktion REC wird die Größe des Datenspeichers (24000-38000) angegeben, so daß man im Vergleich mit den laufenden Adressen im Hauptfeld immer den Überblick hat, wieviel schon "verbraten" ist.

Das nächste wichtige Feature heißt SAVE und beinhaltet das Speichern von MIDI-Daten, die sich im Atari-Datenspeicherbereich befinden, auf Diskette. Das entsprechende Untermenü erwartet die Eingabe eines 8stelligen File-Namens und beginnt das Schreiben auf Diskette nach dem Drücken der Return-Taste. Einige mögliche Fehler – wie z.B. ein zu langer File-Name oder ein leerer Datenspeicher – werden abgefangen und angezeigt. Das Ende des Schreibvorgangs erfolgt mit der Aufforderung, die Leertaste zu drücken.

Mit LOAD können die so gesicherten Daten in den Rechner geladen werden, um sie dann z.B. mit PLAY wieder an das Keyboard zu senden. Dabei kann man sich jederzeit mit DIR wie Directory einen Überblick darüber verschaffen, was sich schon auf der Diskette befindet oder wieviel Platz noch verfügbar ist. Und wie immer führt auch hier der Weg zurück zum Hauptmenü über die Leertaste.

Wer sich mit MIDI schon näher beschäftigt hat, weiß, daß sich MIDI-Daten über 16 verschiedene Kanäle übertragen lassen. Allerdings können viele Keyboards der unteren und mittleren Preisklasse nur auf Kanal 1 senden. In diesem Programm wird zwar in der zweiten Stelle des Statusbytes der jeweils aktuelle Kanal angezeigt, es wird intern aber nur Kanal 1 verarbeitet. Beispiele:

Statusbyte: 90
9 = NOTE ON
0 = Kanal 1

Statusbyte: 8F
8 = NOTE OFF
F = Kanal 16

Eine Erweiterung auf die Kanäle 2-16 erfolgt ggf. später.

Wichtig für die praktische Arbeit mit diesem Programm ist auch die Größe des Arbeitsspeichers im Rechner, der die MIDI-Daten aufnehmen soll. Dazu folgendes: Ein MIDI-Ereignis besteht aus max. 5 Bytes:

1. Statusbyte, z.B. NOTE ON
2. MIDI-Byte 1, z.B. Note Cis
3. MIDI-Byte 2, z.B. Dynamik 7B
4. Zeittakt 1
5. Zeittakt 2

(Tatsächlich werden die beiden Zeittaktwerte direkt nach dem Statusbyte gespeichert.)

Diese 5 Bytes reichen z.B. für eine Controller-Meldung aus. Im Falle einer gedrückten Taste verdoppeln sich die Daten auf nun insgesamt 10 Bytes, nämlich 5 Bytes für "Taste drücken" (NOTE ON) und 5 Bytes für "Taste loslassen" (NOTE OFF). Damit könnte der Speicher etwa 1400 Noten aufnehmen, nämlich $38000 - 24000 = 14000$: 10 Bytes = 1400 Noten. Da aber auch immer wieder Controller-, Programm- und Systemnachrichten gesendet und gespeichert werden, kann man sagen, daß die Speicherkapazität für etwa 1000 bis 1200 Noten reicht, also für einen mittellangen Song groß genug ist.

Programm-Aufbau

Die Steuerung der genannten Funktionen erfolgt in einem Basic-Rahmenprogramm mit den Zeilennummern 10-7090. Das Senden und Empfangen sowie Ein- und Auslesen von MIDI-Daten erledigt ein Assembler-Programm, das von Basic aus angerufen wird.

Das Basic-Steuerprogramm als Hauptprogramm des Basic-teils umfaßt neben der Dimensionierung der Variablen in den Zeilen 180-260 das erste Bild, das solange auf dem Bildschirm steht, bis durch GOSUB 7000 und 6000 alle Maschinendaten eingepokt sind. Da das einige Sekunden dauert, zeigt während dieser Zeit ein Zähler den Stand auf dem Bildschirm an (Zeilen 7010-7090). Bei "00" startet das Programm dann mit den Zeilen 300-390 das Hauptmenü und mit 400-540 wird die Tastatur nach der gewählten Funktion abgefragt und entsprechend in folgende Unter-

Routinen verzweigt: 1000-1170 für den Bildaufbau REC und PLAY, 2000-2150 für das Directory, 3000-3220 für das Speichern von Daten auf Diskette und 4000-4200 für das Laden von Daten von der Diskette.

Die Zeilen 700-890 enthalten einige häufiger benutzte Befehlsfolgen, die von verschiedenen Routinen aufgerufen werden. Im Bereich 6000-6080 liegt ein kleines Maschinenprogramm zur Ausführung der Disketten-Funktionen. Dieser Teil wird bereits in der Startphase mit GOSUB 6000 aktiviert. In den Zeilen 7000-7090 werden die Maschinendaten für das MIDI-Kernprogramm eingelesen und der Zähler für die Anzeige während der Initialisierung aktiviert.

Ab 7100 folgen dann die Data-Werte des Maschinenprogramms.

Es muß wohl nicht besonders betont werden, daß das Übertragen der Maschinendaten vom Listing in den Computer sorgfältig erfolgen sollte. Trotzdem der Hinweis: Die Zahlen müssen absolut fehlerfrei eingegeben werden. Da ein Fehler später nur schwer zu finden ist, bedeutet eine korrekte und langsame Eingabe in jedem Fall den geringsten Zeitaufwand! Grundsätzlich gilt das für das gesamte Programm: Im Zweifel hat auch jede einzelne Leerstelle ihre Bedeutung. Wie schon im ersten Teil erwähnt, hat hier derjenige die wenigste Arbeit, der über den MAC/65 verfügt und mit einem Hilfsprogramm den Object-Code in Data-Werte umwandelt.

Weniger der Funktionstüchtigkeit als dem Komfort dient die Bildschirmaufbereitung. Dazu sollten die folgenden Zeilen mit Hilfe der jeweils angegebenen Sondertasten eingegeben werden:

- 190, 210, 240: Invers
- 260: Invers und CAPS
- 310: Invers
- 330-370: Die Kleinbuchstaben r-l Invers und CAPS
- 400: Invers, 10 x Leertaste, Text, 10 x Leertaste
- 730: Invers (einschl. der Leerstellen)
- 750: 38 x Invers und Leertaste
- 800: CTRL und R

810: Invers (einschl. der Leerstellen)
 840: 38 x Invers und Leertaste
 850, 890: Invers (einschl. der Leerstellen)
 1140, 3120, 3200, 4120, 4190: Invers.

Und noch etwas: Die Zeilennummern sollten unverändert übernommen werden, da freie, jetzt noch nicht benutzte Zeilen späteren Erweiterungen dienen.

An zwei Stellen im Programm wird mit X=USR... das Basic verlassen und der Assembler aufgerufen: Zeile 1050 für REC und Zeile 1160 für PLAY. In diese beiden Hauptprogrammteile wird im Assemblerprogramm über die Basic-Start-Leiste (0440-0510) verzweigt.

MIDI-Recorder

In der ersten Hauptroutine RECORD wird ein MIDI-Byte, das nach dem Empfang zwischengespeichert wurde, in den Datenspeicherbereich übertragen und – soweit es sich um Statusbytes handelt – mit Zeittaktwerten versehen. Um diese Hauptschleife herum liegen einige Hilfs-Routinen:

- RECI zur Initialisierung von Interrupt, UART und Buffer-Bereichen;
- RECE zum Auslesen der MIDI-Daten aus dem Zwischenspeicher;
- EX1 zum ordnungsgemäßen Abschluß der Record-Funktion.

Der Teil "IRQ" ist eine MIDI-Daten-Empfangsroutine, die immer dann aufgerufen wird, wenn MIDI-Daten entstehen. Dies weiß zuerst der UART, der aus diesem Grunde genau jetzt einen Interrupt auslöst. Für den Atari bedeutet dies eine Art Vollbremsung: Womit er auch immer gerade beschäftigt ist, mit dem Interrupt unterbricht er seine Tätigkeit, um die Routine IRQ anzuspringen und abzuarbeiten. Anschließend setzt er seine unterbrochene Arbeit fort.

Diese Interrupt-gesteuerte Empfangsroutine beginnt mit einem Sprung in die Unteroutine ZEIT, wo die Zeittaktwerte fortgeschrieben werden,

dann wird der UART abgefragt und das MIDI-Byte zwischengespeichert. Die weitere Behandlung vor dem Rücksprung aus dem Interrupt hängt von der Art des MIDI-Bytes ab.

MIDI-Player

Der zweite Hauptprogrammteil ist ganz ähnlich aufgebaut: Hier gibt es mit PLAY eine Hauptschleife zum Auslesen der MIDI-Bytes und einige Hilfsroutinen wie

- PLAI zur Initialisierung von Interrupt (dieses Mal der VBI), UART und zum Einrichten der Buffer;
- TRAN zum Senden der MIDI-Daten an das Keyboard;
- EX2 zum Abschließen der Play-Routine und
- VBI, die vom Vertikal-Blank-Interrupt aufgerufene Routine für die Zeittakt-Steuerung beim Senden.

So wie bei RECORD die MIDI-Bytes in den Datenspeicher eingeschrieben wurden, werden sie mit PLAY wieder ausgelesen. Vorher selektiert das Programm jedoch die Zeittaktwerte, denn das Keyboard benötigt diese Daten nicht.

Bildschirm-Anzeigen

Dieser dritte Programmteil beinhaltet die Aufbereitung der MIDI-Daten für die Ausgabe auf dem Monitor und ist in einigen Teilen identisch mit dem Maschinenprogramm aus CK 6-7/1986. Neu sind im wesentlichen die Routinen, die sich mit der Darstellung des Speicherplatzes und des Notenzeichens befassen. Wie schon weiter oben erwähnt, beginnt jede MIDI-Zeile mit der Angabe der Startadresse (in dezimaler Form) für jeweils 5 Bytes. Daß durch das Hochzählen der Adressen und deren Anzeige am Bildschirm der verbrauchte Speicherplatz angegeben wird, ist eigentlich nur ein Nebeneffekt. Primär ist jedoch daran gedacht, diese Adreßangaben später für Editier-Zwecke zu verwenden.

Die Darstellung von gespielten Tasten mit ihren Namen (C, Cis2, G usw.) anstelle von Hex-Zahlen ist eine bedeutende qualitative Verbesserung für den Anwender, insbesondere

dann, wenn dieser nicht vom Computer, sondern von der Musik aus an das Thema "MIDI" herangeht. Die Darstellung in Notenbezeichnungen erfolgt in dieser Programmversion allerdings nur dann, wenn das Keyboard auf Kanal 1 sendet.

Aufmerksamkeit ist der Notentabelle zu widmen. Eine richtige Anzeige ergibt sich nur dann, wenn die Notenbezeichnungen genau in der Form wie im Listing dargestellt eingegeben werden – auch wenn manches etwas unsinnig erscheint. Im Falle eines Eingabefehlers kann der ganze Bildschirm in Verwirrung geraten.

Ausblick

Das vorliegende Programm bietet schon eine Reihe interessanter Möglichkeiten, obwohl

zum Beispiel das Speichern von MIDI-Daten bei vielen Keyboards mittels eines eingebauten Cassetten-Interfaces einfacher und billiger ist. Aber dieses Programm ist die Basis eines Sequenzer-Moduls und kann nach entsprechender Erweiterung viel mehr, als nur Daten speichern. Insbesondere mit der Editier-Funktion ist es möglich, MIDI-Daten zu verändern. Z.B. kann man Fehler verbessern oder auch MIDI-Daten über die Rechnertasten (Step for Step) eingeben. Eine andere Möglichkeit ist die Selektion der MIDI-Daten nach Kanälen. Von hier ist es dann nur noch ein Schritt zur "16-Spur-Software-Tonbandmaschine".

Doch zunächst viel Erfolg beim Abtippen und bei der Arbeit mit diesem Programm.

Karlheinz Metscher

MIDI-Disk

```

10 REM *****
11 REM *
12 REM *      MIDIDISK.BAS      *
13 REM *      BASIC-PROGRAMM FUER *
14 REM *      ATARI 800 XL      *
15 REM *
16 REM *      VERSION 2        *
17 REM *
18 REM *      C MEKASOUND 1985  *
19 REM *
20 REM *****
21 REM
90 REM -----
91 REM *      HAUPTPROGRAMM      *
92 REM -----
100 DIM DAT$(17), FILE$(15), FS$(15), TIS$(
8), TAS$(2), AAS$(4), BBS$(4), ES$(4): ES=""
140 BA=24000:REM BUFFER-ANFANG
180 GRAPHICS 18:POKE 710,234:POKE 712,
2:POKE 708,10
190 POSITION 2,0:? #6;"MIDI-DISK VERS.
2"
200 POSITION 0,1:? #6;"
"
210 POSITION 1,2:? #6;"FOR ATARI COMPU
TER"
220 POSITION 3,5:? #6;"MASCHINENDATEN"
230 POSITION 3,6:? #6;"READY BIS"
240 POSITION 1,9:? #6;"PROGRAMM-START:
00"
250 POSITION 0,10:? #6;"
"
260 POSITION 2,11:? #6;"C MEKASOUND 8.1
"
261 REM FUER "1985" EINGEBEN:INVERS UN
D CTRL 1=0,9=Y,8=X,5=U
270 GOSUB 7000
280 GOSUB 6000
300 GRAPHICS 18:POKE 710,232:POKE 712,
2:POKE 708,12
310 POSITION 3,0:? #6;"ATARI MIDIDISK"
320 POSITION 0,1:? #6;"
"
330 POSITION 2,3:? #6;"REC.: TASTE 0
"
```

```

340 POSITION 2,4:?" M6;" PLAY: TASTE 0
"
350 POSITION 2,6:?" M6;" DIR.: TASTE 0
"
360 POSITION 2,8:?" M6;" SAVE: TASTE 0
"
370 POSITION 2,9:?" M6;" LOAD: TASTE 0
"
390 GOTO 410
400 POKE 752,1:POSITION 1,22:?"
DRUECKE LEERTASTE
410 POKE 764,255
420 TASTE=PEEK(764)
430 IF TASTE=33 THEN 300
440 IF TASTE=40 THEN 1000:REM R
450 IF TASTE=10 THEN 1100:REM P
460 IF TASTE=50 THEN 2000:REM D
470 IF TASTE=62 THEN 3000:REM S
480 IF TASTE=0 THEN 4000:REM L
540 GOTO 420
700 REM -----
701 REM * HILFS-ROUTINEN *
702 REM -----
710 GRAPHICS 0:POKE 710,8:POKE 712,226
:POKE 709,0
720 POKE 752,1:POKE 764,255
730 POSITION 0,0:?" MEDIADISK VERS.0
2 C MEKASOUND 1985
740 POSITION 1,5:?"
"
750 POSITION 1,22:?"
"
790 RETURN
800 POSITION 10,11:?"
"
810 POSITION 1,22:?" MIT RETURN
ABSCHLIESSEN!
820 CLOSE #1:OPEN #1,4,0,"K:"
830 POSITION 10,10:?"FILENAME D:":IN
PUT FILES:CLOSE #1:TRAP 890
840 POSITION 1,22:?"
"
850 IF LEN(FILES)>8 THEN POSITION 10,2
0:?" FILENAME ZU LANG":GOTO 400
860 TIS=FILES
870 FS="D:":FILES=LEN(FILES)+1:".DAT"
880 FS=LEN(FS)+1:FILES=RETURN
890 POSITION 9,20:?" FILE NICHT VORHA
NDEN":TIS="":GOTO 400
1000 REM -----
1001 REM * RECORD/PLAY *
1002 REM -----
1010 GOSUB 700
1020 POSITION 14,2:?" R E C O R D":TIS
="NEU!"
1030 POSITION 0,4:?" ADRES CMD D
AT1 DAT2 KOMMENTAR "
1040 POSITION 5,22:?" DATENSPEICHER:
24000 - 38000 "
1050 X=USR(R):D=0
1060 GOTO 300
1100 GOSUB 700
1110 POSITION 11,2:?" P L A Y :":TIS
1120 POSITION 0,4:?" ADRES CMD D
AT1 DAT2 KOMMENTAR "
1130 D=PEEK(254)+PEEK(255)*256-BA
1140 IF B<=1 AND D=0 THEN POSITION 10,
20:?" DATENSPEICHER LEER":GOTO 400
1150 POSITION 7,22:?" STOP: DRUECKE L
EERTASTE "
1160 X=USR(R+4)
1170 GOTO 400
2000 REM -----
2001 REM * DIRECTORY *
2002 REM -----
2030 GOSUB 700
2040 POSITION 11,3:?" D I R E C T O R
Y"?
2060 POKE 82,11:?"
2100 CLOSE #1:OPEN #1,6,0,"D:*.?"
2110 TRAP 2140
2120 INPUT #1,DATS:?" DATS
2130 GOTO 2120
2140 ?

```

```

2150 CLOSE #1:POKE 82,0:GOTO 400
3000 REM -----
3001 REM * SPEICHERN *
3002 REM -----
3010 GOSUB 700
3040 POSITION 5,3:?" S O N G S S P E
I C H E R N "
3050 POSITION 1,18:?"
"
3060 GOSUB 800
3100 LN=PEEK(254)+PEEK(255)*256-BA
3120 IF LN<=0 THEN POSITION 10,20:?"
DATENSPEICHER LEER":TIS="":GOTO 400
3130 CLOSE #3:OPEN #3,8,0,F$
3140 PUT #3,255:PUT #3,255
3150 PUT #3,192:PUT #3,93
3160 PUT #3,PEEK(254):PUT #3,PEEK(255)
3170 POSITION 12,12:?" wird gespeichert
"
3180 X=USR(MD,1,BA,LN)
3190 CLOSE #3
3200 IF X<128 THEN POSITION 10,20:?"
DISK-FEHLER":X;"":GOTO 400
3220 GOTO 300
4000 REM -----
4001 REM * LADEN *
4002 REM -----
4020 GOSUB 700
4040 POSITION 10,3:?" S O N G S L A D
E N "
4050 POSITION 1,18:?"
"
4060 GOSUB 800
4100 CLOSE #3:OPEN #3,4,0,F$
4110 GET #3,X1:GET #3,X2
4120 IF X1<255 OR X2<255 THEN CLOSE
#3:POSITION 10,20:?" KEIN BINDER-FILE
":TIS="":GOTO 400
4130 GET #3,AL:GET #3,AH:AADR=AL+AH*25
6
4140 GET #3,AL:GET #3,AH:EADR=AL+AH*25
6
4150 LAE=EADR-AADR
4160 POSITION 14,12:?" wird geladen"
4170 X=USR(MD,0,AADR,LAE):D=1
4180 CLOSE #3
4190 IF X<128 THEN POSITION 10,20:?"
DISK-FEHLER":X;"":GOTO 400
4200 GOTO 1100
6000 REM -----
6001 REM * I/O-MASCHINENDATEN *
6002 REM -----
6010 S=0:RESTORE 6060
6020 FOR B=1536 TO 1576:READ D:POKE B,
D:S=S+D:NEXT B
6030 IF S<4119 THEN ? "I/O-DATEN-FEHL
ER":STOP
6040 MD=1536
6050 RETURN
6060 DATA 104,162,48,104,160,7,104,240
,2,160,11,152,157,86,3,104
6070 DATA 157,69,3,104,157,68,3,104,15
7,73,3,104,157,72,3,32,86,228
6080 DATA 132,212,169,0,133,213,96
7000 REM -----
7001 REM * MASCHINENDATEN LADEN *
7002 REM -----
7010 S=0:RESTORE 7100
7020 C=21921-20400:R=20430
7030 FOR A=20400 TO 21921:READ D:POKE
A,D:S=S+D
7035 N5=4:N=C:BB$=STR$(N):AA$=E$:AA$(N
5-LEN(BB$)+1,N5)=BB$
7040 C=C-1
7045 POSITION 12,6:?" M6;"":AA$(1,N5)
7050 NEXT A
7060 IF S<135905 THEN ? "DATEN-FEHLER
IN ZEILE 7930":STOP
7090 RETURN
7100 DATA 104,76,25,80,104,76,106,81,0
,232,0,0,37,20,0,0,3,176,64
7110 DATA 0,2,161,94,0,0,169,192,133,2
54,169,93,133,255,32,137,80

```


7120 DATA 32,132,83,173,252,2,201,33,2
 08,3,76,75,81,32,46,82,32,201
 7130 DATA 80,144,238,160,0,145,254,201
 ,128,144,10,166,254,142,21,80
 7140 DATA 166,255,142,22,80,72,32,122,
 82,104,230,254,208,2,230,255
 7150 DATA 166,255,224,148,144,3,76,75,
 81,201,128,144,196,160,0,177
 7160 DATA 222,230,222,145,254,177,222,
 230,222,200,145,254,165,254
 7170 DATA 24,105,2,133,254,165,255,105
 ,0,133,255,201,148,144,3,76
 7180 DATA 75,81,76,39,80,120,169,217,1
 41,22,2,169,80,141,23,2,169
 7190 DATA 3,141,0,209,169,146,141,0,20
 9,169,0,133,206,133,210,169
 7200 DATA 88,133,207,133,211,169,0,133
 ,220,133,222,169,89,133,221
 7210 DATA 133,223,169,0,141,9,80,141,1
 0,80,141,11,80,141,15,80,141
 7220 DATA 12,80,88,96,165,206,197,210,
 240,8,160,0,177,210,230,210
 7230 DATA 56,96,24,96,120,72,138,72,15
 2,72,32,30,81,173,2,209,74,144
 7240 DATA 46,160,0,173,3,209,145,206,1
 6,29,141,8,80,201,248,240,24
 7250 DATA 201,254,240,20,140,8,80,173,
 9,80,145,220,200,173,10,80,145
 7260 DATA 220,230,220,230,220,230,206,
 104,160,104,170,104,64,104,160
 7270 DATA 104,170,104,76,48,192,169,2,
 36,20,240,20,44,11,80,16,33
 7280 DATA 230,9,80,200,3,238,10,80,169
 ,0,141,11,80,240,18,44,11,80
 7290 DATA 48,13,238,9,80,200,3,238,10,
 80,169,255,141,11,80,96,160
 7300 DATA 8,169,255,145,254,152,200,14
 5,254,200,145,254,120,169,3
 7310 DATA 141,0,209,169,40,141,22,2,16
 9,192,141,23,2,88,96,32,132
 7320 DATA 83,32,214,81,173,252,2,201,3
 3,208,3,76,23,82,160,2,173,10
 7330 DATA 80,209,254,144,237,136,173,9
 ,80,209,254,144,229,136,177
 7340 DATA 254,201,255,200,3,76,23,82,3
 2,3,82,72,165,254,141,21,80
 7350 DATA 165,255,141,22,80,104,32,122
 ,82,165,254,24,105,3,133,254
 7360 DATA 165,255,105,0,133,255,201,14
 8,144,3,76,23,82,160,0,177,254
 7370 DATA 48,175,32,3,82,32,122,82,230
 ,254,200,240,230,255,165,255
 7380 DATA 224,148,144,232,76,23,82,120
 ,216,162,82,160,16,169,7,32
 7390 DATA 92,228,169,3,141,0,209,169,1
 0,141,0,209,169,192,133,254
 7400 DATA 169,93,133,255,169,0,141,9,8
 0,141,10,80,141,15,80,141,12
 7410 DATA 80,80,96,72,173,2,209,74,74,
 144,249,104,141,1,209,96,120
 7420 DATA 32,30,81,76,98,228,169,252,3
 2,3,82,169,255,32,3,82,169,255
 7430 DATA 32,3,82,169,255,32,3,82,76,8
 8,81,173,0,80,201,240,240,14
 7440 DATA 201,254,240,20,197,20,240,1,
 96,162,20,76,84,82,169,0,141
 7450 DATA 8,80,162,16,76,84,82,169,0,1
 41,8,80,162,18,189,132,85,141
 7460 DATA 105,82,232,189,132,85,141,10
 6,82,169,0,141,13,80,160,0,185
 7470 DATA 238,04,174,13,80,157,154,156
 ,238,13,80,200,192,20,200,239
 7480 DATA 96,174,15,80,157,17,80,240,1
 0,201,128,144,22,169,0,141,15
 7490 DATA 80,96,201,128,144,251,173,17
 ,80,41,127,74,74,74,74,141,16
 7500 DATA 80,238,15,80,172,16,80,185,1
 54,85,205,15,80,176,225,160
 7510 DATA 173,17,80,201,144,208,10,173
 ,19,80,201,0,208,3,141,16,80
 7520 DATA 169,2,141,12,80,173,127,83,2

01,159,200,13,173,126,83,201
 7530 DATA 120,144,6,32,132,83,76,227,8
 2,173,126,83,24,105,40,141,126
 7540 DATA 83,144,3,238,127,83,32,171,8
 3,165,213,32,108,83,165,214
 7550 DATA 32,99,83,165,215,32,99,83,16
 9,10,141,12,80,162,0,189,17
 7560 DATA 80,142,20,80,201,144,208,10,
 152,72,32,192,83,104,168,76
 7570 DATA 25,83,32,99,83,238,12,80,238
 ,12,80,238,12,80,238,12,80,238
 7580 DATA 12,80,174,20,80,232,224,2,20
 8,6,206,12,80,206,12,80,136
 7590 DATA 16,201,169,0,141,15,80,169,2
 7,141,12,80,173,16,80,10,170
 7600 DATA 189,132,85,141,82,83,232,189
 ,132,85,141,83,83,160,0,185
 7610 DATA 238,84,170,73,64,240,7,138,3
 2,122,83,200,208,241,76,139
 7620 DATA 82,72,74,74,74,32,110,83,
 104,41,15,201,10,144,5,24,105
 7630 DATA 23,208,3,24,105,16,174,12,80
 ,157,48,157,238,12,80,96,169
 7640 DATA 0,133,77,162,48,142,126,83,1
 69,157,141,127,83,162,0,169
 7650 DATA 0,157,48,157,232,208,250,157
 ,48,158,232,208,250,157,48,159
 7660 DATA 232,224,126,208,248,96,152,7
 2,32,60,218,173,21,80,133,212
 7670 DATA 173,22,80,133,213,32,170,217
 ,104,168,96,162,6,169,108,56
 7680 DATA 233,12,202,240,7,205,18,80,2
 40,2,176,243,141,24,80,173,18
 7690 DATA 80,56,237,24,80,10,10,224,0,
 240,6,24,105,48,202,208,250
 7700 DATA 168,169,4,141,24,80,185,250,
 83,32,122,83,200,206,24,80,208
 7710 DATA 244,96,35,0,0,0,35,41,51,0,3
 6,0,0,0,36,41,51,0,37,0,0,0
 7720 DATA 38,0,0,0,38,41,51,0,39,0,0,0
 ,39,41,51,0,33,0,0,0,34,0,0
 7730 DATA 0,40,0,0,0,99,0,0,99,105,1
 15,0,100,0,0,0,100,105,115,0
 7740 DATA 101,0,0,0,102,0,0,0,102,105,
 115,0,103,0,0,0,103,105,115
 7750 DATA 0,97,0,0,0,98,0,0,0,104,0,0,
 0,99,17,0,0,99,105,115,17,100
 7760 DATA 17,0,0,100,105,115,17,101,17
 ,0,0,102,17,0,0,102,105,115
 7770 DATA 17,103,17,0,0,103,105,115,17
 ,97,17,0,0,98,17,0,0,104,17
 7780 DATA 0,0,99,18,0,0,99,105,115,18,
 100,18,0,0,100,105,115,18,101
 7790 DATA 18,0,0,102,18,0,0,102,105,11
 5,18,103,18,0,0,103,105,115
 7800 DATA 18,97,18,0,0,98,18,0,0,104,1
 8,0,0,99,19,0,0,99,105,115,19
 7810 DATA 180,19,0,0,100,105,115,19,10
 1,19,0,0,102,19,0,0,102,105
 7820 DATA 115,19,103,19,0,0,103,105,11
 5,19,97,19,0,0,98,19,0,0,104
 7830 DATA 19,0,0,99,20,0,0,46,47,52,37
 ,0,47,38,38,64,46,47,52,37,0
 7840 DATA 47,46,64,48,47,44,57,13,48,5
 0,37,51,51,64,35,47,46,52,50
 7850 DATA 47,44,44,37,50,64,48,50,47,3
 9,14,35,40,33,46,39,37,64,35
 7860 DATA 40,33,46,46,14,48,50,37,51,5
 1,64,48,41,52,35,40,0,34,37
 7870 DATA 46,36,64,51,57,51,52,37,45,1
 3,41,46,38,47,64,0,38,24,0,13
 7880 DATA 0,52,41,45,41,46,39,0,15,0,3
 5,44,47,35,43,0,64,43,37,57
 7890 DATA 34,47,33,50,36,0,51,37,46,36
 ,37,34,37,50,37,41,52,64,0,0
 7900 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0
 ,0,0,0,0,64,238,84,247,84,255
 7910 DATA 84,10,85,21,85,33,85,45,85,5
 6,85,68,85,90,85,111,85,2,2
 7920 DATA 2,2,1,1,2,0

MicroMon

Ein Maschinensprache-Monitor für den Atari 600/800 XL mit 64 K + Cassette

MicroMon ist ein Monitor, mit dem man recht komfortabel Programme in Assembler schreiben kann. Darüber hinaus arbeitet er parallel zu Basic, wodurch man nach Belieben zwischen Basic und Maschinensprache wechseln kann.

Damit keine Probleme wegen des Arbeitsspeichers entstehen, wurde der Monitor hinter das Basic-ROM gelegt, so daß Basic-Programme vom Monitor in keinsten Weise gestört werden. Auch die gern verwendete Page 6 bleibt für eigene Anwendungen frei.

Zuerst muß man den Basic-Loader abtippen. Alle verwendeten Steuerzeichen werden im Kopf erläutert. Dabei steht CTRL für die Control-Taste. Ist das Programm abgetippt, kann man es mit RUN starten. Nach einiger Zeit müßte der Titel sowie die Meldung "Load-Routine" mit einer Zahl dahinter erscheinen. Diese Zahl gibt an, wieviel Datenzeilen noch eingelesen werden müssen. Sind die Daten in Ordnung, erscheint nach einiger Zeit die Meldung "Monitor", ebenfalls mit einer Zahl dahinter. Andernfalls wird man mit einer Fehlermeldung konfrontiert, wobei die fehlerhafte Zeile angezeigt wird.

Arbeitet das Programm fehlerfrei, erscheint die Angabe "Load-Routine save ...", worauf sich die übliche Save-Prozedur abspielt. Nachdem auch der eigentliche Monitor gesaved wurde, wäre die Arbeit fürs erste beendet.

Die Bedienung

Um den Monitor zu laden, muß man den Computer einschalten und dabei die Start-Taste gedrückt halten, bis der gewohnte Tröter ertönt. Danach spielt sich die übliche Lade-Prozedur ab. Mit geladenem Monitor wird zunächst ein Kaltstart durchgeführt, um sicherzustellen, daß alle Register richtig initialisiert wurden. Sobald "Ready" erscheint, kann man durch Drücken der Reset-Taste in ein kleines Menü gelangen, in dem

mit den Tasten "1" und "2" entweder der Monitor oder das Basic aufgerufen wird.

Die Programmierung mit MicroMon ist nicht besonders schwer, da alle Anweisungen nur aus einem Buchstaben oder Zeichen mit den entsprechenden Angaben dahinter bestehen. Man kann den Speicherinhalt entweder bitweise, als Hexdump oder als Assembler-Listing auflisten oder verändern.

Nachfolgend nun alle Anweisungen mit einem Beispiel. Statt der Klammern muß man natürlich die entsprechenden Daten eingeben. Auch die Blanks sind entsprechend zu setzen.

M. (Startadresse)

Diese Eingabe listet den Speicherinhalt als Hexdump auf. Mit der Leertaste wird die Ausgabe fortgesetzt, mit jeder anderen unterbrochen. Z.B.: M_1C00.

D. (Startadresse)

Mit diesem Befehl wird der Speicherinhalt disassembliert, d.h., es werden die entsprechenden 6502-Anweisungen ausgegeben. Ansonsten arbeitet diese Funktion wie die erste. Z.B.: D_1CEA.

@. (Startadresse)

Auf diese Weise kann der Speicherinhalt mit den entsprechenden Bitmustern wiedergegeben werden. Damit lassen sich zum Beispiel Zeichensätze oder Sonderzeichen aufzeigen. Auch diese Funktion arbeitet wie "M" und "D". Z.B.: @_CC00.

L. (Startadresse)

Mit dieser Anweisung wird der Speicherinhalt als Text ausgegeben. Das kann sehr hilfreich sein, wenn man Überschriften oder dergleichen sucht. Ansonsten arbeitet diese Funktion wie "M", "D" und "@". Z.B.: L_D800.

; (Aktuelle Adresse) (Hex byte 1-8)

Hiermit werden insgesamt acht Hexbytes an die angegebene Adresse gelesen. Es lassen sich auch bestehende Hex-Li-

stings verändern. Z.B.: ;_2000 A9 0A 8D FC 02 60 EA EA.

.. (Aktuelle Adresse) (6502-Anweisung)

Hiermit wird ein Assembler-Programm eingegeben und die nächsthöhere Adresse automatisch berechnet. Auch indirekte Sprünge wie z.B. BNE etc. werden berechnet, soweit sie nicht unzulässig sind. Z.B.: ..1C00 LDA #SEI.

Die Adressierungs-Arten müssen wie folgt angegeben werden:

```
LDA #1C    unmittelbar
LDA $02FC  absolut
STY $2000,X absolut mit X-Register
STX $2000,X absolut mit Y-Register
STA $88    Zero-Page
LDA $88,X  Zero-Page mit X-Register
STA $88,Y  Zero-Page mit Y-Register
STY ($1A,X) indirekt mit X-Register
LDA ($1F),Y indirekt mit Y-Register
BNE $201F  indirekter Sprung (Branch)
JMP ($7000) indirekt
RTS        ein-Byte-Befehl
```

> (Aktuelle Adresse) (8-Bit-Muster)

Auf diese Weise kann man ein Bit-Muster in eine Adresse eingeben. Dabei wird für "0" das Zeichen "." und für "1" das Zeichen "*" verwendet. Z.B.: >_201A_*.***.

G. (Startadresse)

Damit wird eine Routine im Speicher aufgerufen, wobei die Routine mit RTS beendet werden muß. USR-Routinen lassen

sich hiermit nicht testen. Z.B.: G_1CBA.

S. (Startadresse) (Länge des Files)

Hiermit kann man einen bestimmten Speicherbereich als Binärfile auf Cassette save. Z.B. S_E000_0400

L. (gewünschte Startadresse) (maximale Länge des Files)

Nit dieser Eingabe wird ein Binärfile an eine bestimmte Startadresse geladen. Nach dem Ladevorgang zeigt das Programm die Anzahl der geladenen Bytes an. Z.B.: L_0600_0080

#. (Dezimalzahl)

Dient zur Umrechnung einer Dezimalzahl in die entsprechende Hexzahl. Z.B.: #_764.

\$. (Hexzahl)

Dient zur Umrechnung einer Hexzahl in die entsprechende Dezimalzahl. Z.B.: \$_00E7.

Um den Monitor wieder zu verlassen, muß man nur Reset drücken, wodurch man wieder in das Hauptmenü gelangt.

Zum Schluß noch ein kleines Programm-Beispiel:

```
.0600 LDA $D20A
.0603 STA $D40A
.0606 STA $D018
.0609 JMP $0600
.060C
G 0600
```

Arno Welzel

0 REM Basic-Loader

```
1 REM
2 REM  = Erst 1 x ESC, dann TAB
3 REM  = " , dann CTRL & -
4 REM  = " , dann CTRL & +
5 REM
10 RESTORE 20:FOR I=1536 TO 1606:READ
A:POKE I,A:NEXT I
20 DATA 162,48,169,3,157,66,3,169,68,1
57,68,3,169,6,157,69
30 DATA 3,169,8,157,74,3,169,128,157,7
5,3,32,86,228,162,48
40 DATA 184,184,157,69,3,184,157,68,3,
184,157,73,3,184,157,72
50 DATA 3,169,11,157,66,3,32,86,228,16
2,48,169,12,157,66,3
60 DATA 32,86,228,96,67,58,155
70 GRAPHICS 0:SETCOLOR 2,0,0
80 ? "MicroMon / Basic-Loader"
90 ? "(C) 1986 by Freak"
100 DIM A$(16):RESTORE 1000
110 ? "Load-Routine":H=6*4096
120 FOR I=0 TO 19:READ A$,B:C=0
130 FOR J=0 TO 7:D=J*2+1
140 E=ASC(A$(D,D)):F=ASC(A$(D+1,D+1))
150 G=16*(E-48-(E/57)*7)+(F-48-(F/57)*
7)
```



```

160 POKE H,G:H=H+1:C=C+G:NEXT J
170 IF C<>B THEN ? "Datenfehler in Zei
le ";1000:I:END
180 ? "+)";19-I;" ":NEXT I
190 ? "Monitor":H=7*4096
200 FOR I=0 TO 511:READ A$,B:C=0
210 FOR J=0 TO 7:D=J*2+1
220 E=ASC(A$(D,D)):F=ASC(A$(D+1,D+1))
230 G=16*(E-48-(E/57)*7)+(F-48-(F/57)*
7)
240 POKE H,G:H=H+1:C=C+G:NEXT J
250 IF C<>B THEN ? "Datenfehler in Zei
le ";1020:I:END
260 ? "+)";511-I;" ":NEXT I
270 ? "+Load-Routine saveen ... "
280 A=USR(1536,6*4096,154)
290 ? "Monitor: saveen ... "
300 POKE 764,12:A=USR(1536,7*4096,4096
):POKE 764,255
310 ? "4Es ist geschafft !!"
320 END
1000 DATA 0002000779074C0C,225
1001 DATA 07433A9BA93C8D02,659
1002 DATA 03A9838D01D3A90C,1045
1003 DATA 8DFC02A230A9039D,934
1004 DATA 420A9099D4403A7,644
1005 DATA 079D4503A9049D4A,640
1006 DATA 03A9809D48032056,653
1007 DATA E4A230A9079D4203,840
1008 DATA A9009D4403A909D,883
1009 DATA 4503A9FF9D4803A9,897
1010 DATA FF9D49032056E4A2,996
1011 DATA 30A90C9D42032056,573
1012 DATA E4A99A850E80E702,1072
1013 DATA A907850F80E802A7,868
1014 DATA 79850AA907850B18,600
1015 DATA 60AD9707C900D008,844
1016 DATA A9FF0D97074C9007,950
1017 DATA A9838D01D320D0AC,1076
1018 DATA A9818D01D3600000,747
1019 DATA 0000000000000000,0
1020 DATA 4252480AA4F524107,466
1021 DATA 3F3F3F0A3F3F3F0A,398
1022 DATA 3F3F3F0A4F524104,429
1023 DATA 41534C063F3F3F0A,429
1024 DATA 5048500AA4F524108,468
1025 DATA 41534C0A3F3F3F0A,433
1026 DATA 3F3F3F0A4F524101,426
1027 DATA 41534C013F3F3F0A,424
1028 DATA 42504C094F524108,465
1029 DATA 3F3F3F0A3F3F3F0A,398
1030 DATA 3F3F3F0A4F524105,430
1031 DATA 41534C0A3F3F3F0A,427
1032 DATA 434C430A4F524103,449
1033 DATA 3F3F3F0A3F3F3F0A,398
1034 DATA 3F3F3F0A4F524102,427
1035 DATA 41534C023F3F3F0A,425
1036 DATA 4A535201414E4A07,458
1037 DATA 3F3F3F0A3F3F3F0A,398
1038 DATA 42495404414E4A04,442
1039 DATA 524F4C043F3F3F0A,440
1040 DATA 5053500A414E4A00,464
1041 DATA 524F4C0A3F3F3F0A,446
1042 DATA 42495401414E4A01,436
1043 DATA 524F4C013F3F3F0A,437
1044 DATA 424D4909414E4A08,444
1045 DATA 3F3F3F0A3F3F3F0A,398
1046 DATA 524F4C0A414E4A05,415
1047 DATA 524F4C053F3F3F0A,441
1048 DATA 5345430A414E4A03,443
1049 DATA 3F3F3F0A3F3F3F0A,398
1050 DATA 3F3F3F0A414E4A02,412
1051 DATA 524F4C023F3F3F0A,438
1052 DATA 5254490A454F5207,486
1053 DATA 3F3F3F0A3F3F3F0A,398
1054 DATA 3F3F3F0A454F5204,433
1055 DATA 4C5352043F3F3F0A,444
1056 DATA 5048410A454F5200,457
1057 DATA 4C53520A3F3F3F0A,450
1058 DATA 4A4D5001454F5201,463
1059 DATA 4C5352013F3F3F0A,441
1060 DATA 42564309454F5208,466
1061 DATA 3F3F3F0A3F3F3F0A,398
1062 DATA 3F3F3F0A454F5205,434
1063 DATA 4C5352053F3F3F0A,445
1064 DATA 434C490A454F5203,459
1065 DATA 3F3F3F0A3F3F3F0A,398

```

```

1066 DATA 3F3F3F0A454F5202,431
1067 DATA 4C5352023F3F3F0A,442
1068 DATA 5254530A41444307,466
1069 DATA 3F3F3F0A3F3F3F0A,398
1070 DATA 3F3F3F0A41444304,403
1071 DATA 524F52043F3F3F0A,446
1072 DATA 504C410A41444300,431
1073 DATA 524F520A3F3F3F0A,452
1074 DATA 4A4D500541444301,443
1075 DATA 524F52013F3F3F0A,443
1076 DATA 4256530941444308,452
1077 DATA 3F3F3F0A3F3F3F0A,398
1078 DATA 3F3F3F0A41444305,404
1079 DATA 3F3F3F0A3F3F3F0A,398
1080 DATA 5345490A41444303,438
1081 DATA 3F3F3F0A3F3F3F0A,398
1082 DATA 3F3F3F0A41444302,401
1083 DATA 3F3F3F0A3F3F3F0A,398
1084 DATA 3F3F3F0A53544107,438
1085 DATA 3F3F3F0A3F3F3F0A,398
1086 DATA 5354590453544104,496
1087 DATA 535458043F3F3F0A,458
1088 DATA 4445590A3F3F3F0A,435
1089 DATA 5458410A3F3F3F0A,446
1090 DATA 5354590153544101,490
1091 DATA 535458013F3F3F0A,455
1092 DATA 4243430953544108,449
1093 DATA 3F3F3F0A3F3F3F0A,398
1094 DATA 5354590553544105,498
1095 DATA 535458063F3F3F0A,460
1096 DATA 5459410A53544103,483
1097 DATA 5458530A3F3F3F0A,464
1098 DATA 3F3F3F0A53544102,433
1099 DATA 3F3F3F0A3F3F3F0A,398
1100 DATA 4C445904C444107,449
1101 DATA 4C4458063F3F3F0A,431
1102 DATA 4C445904C444104,450
1103 DATA 4C4458043F3F3F0A,435
1104 DATA 5441590A4C444100,457
1105 DATA 5441580A3F3F3F0A,446
1106 DATA 4C4459014C444101,444
1107 DATA 4C4458013F3F3F0A,432
1108 DATA 424353094C444108,442
1109 DATA 3F3F3F0A3F3F3F0A,398
1110 DATA 4C4459054C444105,452
1111 DATA 4C4458063F3F3F0A,437
1112 DATA 434C560A4C444103,451
1113 DATA 5453580A3F3F3F0A,464
1114 DATA 4C4459024C444102,444
1115 DATA 4C4458033F3F3F0A,434
1116 DATA 4350590A434D5007,467
1117 DATA 3F3F3F0A3F3F3F0A,398
1118 DATA 43505904434D5004,468
1119 DATA 444543043F3F3F0A,407
1120 DATA 494E590A434D5000,474
1121 DATA 4445580A3F3F3F0A,434
1122 DATA 43505901434D5001,462
1123 DATA 444543013F3F3F0A,404
1124 DATA 424E4509434D5008,454
1125 DATA 3F3F3F0A3F3F3F0A,398
1126 DATA 3F3F3F0A434D5005,428
1127 DATA 444543053F3F3F0A,408
1128 DATA 434C440A434D5003,448
1129 DATA 3F3F3F0A3F3F3F0A,398
1130 DATA 3F3F3F0A434D5002,425
1131 DATA 444543023F3F3F0A,405
1132 DATA 4350580053424307,458
1133 DATA 3F3F3F0A3F3F3F0A,398
1134 DATA 4350580453424304,459
1135 DATA 494E43043F3F3F0A,421
1136 DATA 494E580A53424308,465
1137 DATA 4E4F500A3F3F3F0A,446
1138 DATA 4350580153424301,453
1139 DATA 494E43013F3F3F0A,418
1140 DATA 4245510953424308,449
1141 DATA 3F3F3F0A3F3F3F0A,398
1142 DATA 3F3F3F0A53424305,420
1143 DATA 494E43053F3F3F0A,422
1144 DATA 5345440A53424303,449
1145 DATA 3F3F3F0A3F3F3F0A,398
1146 DATA 3F3F3F0A53424302,417
1147 DATA 494E43023F3F3F0A,419
1148 DATA 3031323334353637,412
1149 DATA 3039414243444546,518
1150 DATA 3036303000000000,210
1151 DATA A9080D15A4E002F0,961
1152 DATA 23AD11A42068A4E0,913

```

```

1153 DATA FF0001608D15A4AD,1059
1154 DATA 10A42068A4E0FFD0,1167
1155 DATA 01608A0A00A00D15,171
1156 DATA A48D15A4AD13A420,878
1157 DATA 68A4E0FFD001608D,1193
1158 DATA 14A4AD12A42068A4,839
1159 DATA E0FFD001608A0A0A,814
1160 DATA 0A0D14A48D14A460,620
1161 DATA A200D00A4F008E8,1027
1162 DATA E010D0F6A2FF608A,1345
1163 DATA 60EAEAD015A429F0,1203
1164 DATA 4A4A4A4A0A0D00A4,819
1165 DATA 8D10A4AD15A429F0,735
1166 DATA AABD00A48D11A440,1018
1167 DATA 14A429F0A4A4A4A4,761
1168 DATA AABD00A48D12A440,1019
1169 DATA 14A429F0AABD00A4,763
1170 DATA 8D13A4608E14A48C,886
1171 DATA 15A4207BA4A200A5,831
1172 DATA 06C900D002A2020A,719
1173 DATA 48BD10A42080F268,995
1174 DATA AAE8EAD04F160A0,1348
1175 DATA 16A48500AD17A485,812
1176 DATA 01A93B2080F2A9FF,1103
1177 DATA 8506A600A401208A,682
1178 DATA A4A9008506A00098,784
1179 DATA 43A9202080F268A8,995
1180 DATA 48B100AA208A4A68,899
1181 DATA A8C8C008D0E9A99B,1333
1182 DATA 2080F2A500186908,752
1183 DATA 85009002E601A9FF,934
1184 DATA 8DFC02ADF0C2C9FF,1278
1185 DATA F0F9C921F0B3A9FF,1566
1186 DATA 8DFC0250AD16A485,983
1187 DATA 00AD17A48501A92E,709
1188 DATA 2080F2A9FF8506A6,1179
1189 DATA 00A40120B4A4A720,742
1190 DATA 2080F2A000B10027,828
1191 DATA C0182A2A2A8505B1,657
1192 DATA 000A0A850A405018,351
1193 DATA 69A08505A0009848,787
1194 DATA B1042080F268A8C8,1103
1195 DATA C00300F2A9202080,1054
1196 DATA F2A003B1040A0A0D,955
1197 DATA E5068D94A58DE6A6,1434
1198 DATA 8D95A520FFA5A99B,1231
1199 DATA 2080F2A500186907,747
1200 DATA 85009002E601A9FF,934
1201 DATA 8DFC02ADF0C2C9FF,1278
1202 DATA F0F9C921F0B3A9FF,1393
1203 DATA 8DFC0268A43EA500,794
1204 DATA 080B000000000000,19
1205 DATA 000A0A723200F2A9,823
1206 DATA 242080F2A9008506,794
1207 DATA A091B100AA208A4A,884
1208 DATA A902805760A92420,644
1209 DATA B0F2A9FF8506A001,1142
1210 DATA B100AAC0B100A820,924
1211 DATA 4A4A490385076020,784
1212 DATA E5A5A92C2080F2A9,1226
1213 DATA 582080F26020E5A5,1060
1214 DATA A92C2080F2A95920,953
1215 DATA B0F260A9242080F2,1169
1216 DATA A9008506A001B100,646
1217 DATA AA20B4A4A9028507,857
1218 DATA 602018A6A92C2080,742
1219 DATA F2A9582080F26020,1077
1220 DATA 18A6A92C2080F2A9,1025
1221 DATA 592080F260A92820,876
1222 DATA B0F22031A6A92920,907
1223 DATA B0F260A9282080F2,1173
1224 DATA 2018A6A9292080F2,885
1225 DATA A92C2080F2A95920,953
1226 DATA B0F260A901850760,920
1227 DATA A9282080F260E5A5,1085
1228 DATA A9292080F260A500,921
1229 DATA 8D14A4A5018D15A4,817
1230 DATA A9FF8506A92C2080,976
1231 DATA F2A9020507A001B1,891
1232 DATA 00C97F1021B10018,578
1233 DATA 6D14A48D14A4A903,765
1234 DATA EE15A4AD14A41869,909
1235 DATA 028D14A49003EE15,733
1236 DATA A44CE1A6EAEAA001,1260
1237 DATA B100A9FF8DC0A5CE,1209
1238 DATA C0A5AD14A438EDC0,1199
1239 DATA A58D14A4B003CE15,896

```



```

1240 DATA A420BAA460CAAE5,1238
1241 DATA A5FFA50DA61BA631,1005
1242 DATA A63FA64DA65BA686,1029
1243 DATA A673A678A6A2008A,1033
1244 DATA 48BD0FA720B0F268,997
1245 DATA AAEE01C00F1607D,1324
1246 DATA 406963726F406F6E,804
1247 DATA 9B28432920313938,497
1248 DATA 3620627920467265,622
1249 DATA 616B98A2008A488D,920
1250 DATA 3DA720B0F268AAE8,1184
1251 DATA E004D0F1609B4F4B,1082
1252 DATA 9B204AF2C99B00F9,1316
1253 DATA C001D0F5A5588500,1032
1254 DATA A5598501A654CA86,974
1255 DATA 04A200A500186504,460
1256 DATA 85009002E601E0E0,966
1257 DATA 28D0F0A002B10018,851
1258 DATA 6920979A7C8C026,1008
1259 DATA D0F3604720383630,800
1260 DATA 3020202020202020,272
1261 DATA 2020202020202020,256
1262 DATA 2020202020202020,256
1263 DATA 2020202020202020,448
1264 DATA 0620F0A6202BA720,733
1265 DATA 41A7A200AD7BA7D0,1078
1266 DATA BEADF008E3E01000,1291
1267 DATA F34CA7A78A00AA8D,1160
1268 DATA CEAD80CCA78DCAAD,1460
1269 DATA 8DCDA72026A08CA7,994
1270 DATA A74D443B2E475340,635
1271 DATA 4C08A817A844A81C,707
1272 DATA AB26A813A943AD40,069
1273 DATA A9A200BD7DA79D10,985
1274 DATA A4E8E004D0F5A200,1239
1275 DATA 2018A4AD14A48D16,740
1276 DATA A4AD15A48D17A460,946
1277 DATA 20E9A7E0FFD00160,1216
1278 DATA 20D7A4202BA76020,701
1279 DATA E9A7E0FFD0016020,1216
1280 DATA 3A5202BA76020E9,820
1281 DATA A7E0FFD00160AD16,1146
1282 DATA A48D3BA8AD17A48D,1033
1283 DATA 3CA8200006202BA7,508
1284 DATA 60EAEAEAA200B07C,1273
1285 DATA A79D10A4E8E004D0,1172
1286 DATA F5A2002018A4E0FF,1106
1287 DATA D00160AD14A48D00,795
1288 DATA AD15A48D01A00089,837
1289 DATA 01A78D12A4B982A7,1101
1290 DATA 0013A4A2022018A4,708
1291 DATA E0FFD001609848AD,1181
1292 DATA 1A4A000910068A8,761
1293 DATA E600D002E601C8C8,1071
1294 DATA C3C01000D2A93820,1094
1295 DATA D0F2A9FF8506A600,1147
1296 DATA A01208A4A492020,774
1297 DATA B0F260A230A9039D,1053
1298 DATA 4203A9FD9B4A403A9,888
1299 DATA A89D4503ADC0A59D,1084
1300 DATA 4A03A9809D4B0320,641
1301 DATA 56E4A230ADC1A59D,1212
1302 DATA 4203AD16A49D4A03,656
1303 DATA AD17A49D4503AD14,782
1304 DATA A49D4803AD15A49D,911
1305 DATA 49032056E4A230A9,801
1306 DATA 0C9D42032056E4A9,753
1307 DATA 038D0FD260433A9B,745
1308 DATA A200B082A79D10A4,985
1309 DATA E8E004D0F5A20020,1107
1310 DATA 18A46020E9A7E0FF,1195
1311 DATA D00160AD14A48D16,825
1312 DATA 4A0D15A48D17A428,882
1313 DATA 00A9E0FFD00160A9,1122
1314 DATA 068DCA0A5A908D0C1,1020
1315 DATA A520A8A8202BA760,874
1316 DATA 20E9A7E0FFD00160,1216
1317 DATA AD14A48D16A4AD15,878
1318 DATA A48D17A42000A9E0,917
1319 DATA FF00D160A9048D0C0,1066
1320 DATA A5A9070DC1A520A8,1043
1321 DATA A8A230BD4803D14,803
1322 DATA A4BD4903BD15A4A9,924
1323 DATA FF050620BAA4A200,938
1324 DATA 8A48BD93A920B0F2,1165
1325 DATA 68AAE8E007D0F120,1218
1326 DATA 2BA7602042797465,742
1327 DATA 739BA200BD7CA79D,1069
1328 DATA 10A4E8E004D0F5A2,1255
1329 DATA 002018A4AD14A485,710
1330 DATA 00AD15A4850168AD,761
1331 DATA 85A7C923D00CAD86,1063
1332 DATA A7C92AD0005A90085,919
1333 DATA 0760AD85A7C92AD0,1021
1334 DATA 0CAD88A7C9200005,934
1335 DATA A904850760AD85A7,802
1336 DATA C92AD0813AD88A7C9,1141
1337 DATA 2CD00CAD89A7C958,1030
1338 DATA D005A905850760AD,796
1339 DATA 85A7C92AD0813AD88,1073
1340 DATA A7C92CD00CAD89A7,1109
1341 DATA C959D005A9060507,818
1342 DATA 60AD81A7C942D00C,1052
1343 DATA AD85A7C92AD0005A9,1092
1344 DATA 07850760AD85A7C9,919
1345 DATA 20D005A90A850760,660
1346 DATA AD85A7C92AD0021AD,1128
1347 DATA 06A7C92AD081A0D89,1082
1348 DATA A7C92CD0813AD8A7,1117
1349 DATA C958D00CAD88A7C9,1189
1350 DATA 29D005A907850760,666
1351 DATA AD85A7C92AD0021AD,1128
1352 DATA 86A7C92AD081A0D89,1082
1353 DATA A7C929D0813AD8A7,1114
1354 DATA C92CD00CAD88A7C9,1145
1355 DATA 59D005A908850760,715
1356 DATA AD85A7C92AD0813AD,1114
1357 DATA 86A7C92AD00CAD88,1070
1358 DATA A7C929D005A90885,935
1359 DATA 0760AD85A7C92AD0,1021
1360 DATA 0CAD88A7C92AD0005,936
1361 DATA A901850760AD85A7,879
1362 DATA C92AD0813AD8A7C9,1143
1363 DATA 2CD00CAD88A7C958,1032
1364 DATA D005A902850760AD,793
1365 DATA 85A7C92AD0813AD8A,1075
1366 DATA A7C92CD00CAD88A7,1111
1367 DATA C959D005A9038507,815
1368 DATA 60A9FF850760A200,918
1369 DATA A0A0860A48A05A07,767
1370 DATA 3D8A47A9008506A0,908
1371 DATA 00B10AD981A7D006,908
1372 DATA C0C00AD0F460E606,1180
1373 DATA A504186904850490,503
1374 DATA 02E605A505C9A400,708
1375 DATA DEA902850660209A,814
1376 DATA A9E0FFD001602087,1168
1377 DATA A9A507C9FFD00160,1102
1378 DATA 20E6AA0A506C9020,1014
1379 DATA 0160A000A5069100,573
1380 DATA 0160A000A5069100,573
1381 DATA 51A8BD7CAB8D52AB,1130
1382 DATA 2006ACA506C9FFD0,1045
1383 DATA 0160A50018650685,526
1384 DATA 009002E601A92E20,624
1385 DATA B0F2A9FF8506A600,1147
1386 DATA A48120B4A4A92020,774
1387 DATA B0F26093AB87A887,1369
1388 DATA AB87ABE2ABE2ABE2,1545
1389 DATA AB93AB93AB36AC06,1039
1390 DATA AC0BACAD87A78D12,989
1391 DATA A4AD88A78D13A4A2,1126
1392 DATA 022018A4E0FFD003,912
1393 DATA 860600A001AD14A4,754
1394 DATA 9100A902850660A2,713
1395 DATA 00BD86A79D10A4E8,1059
1396 DATA E004D0F5A2002018,899
1397 DATA A4E0FFD003860660,1090
1398 DATA A001AD14A491080C,863
1399 DATA AD15A49100A90385,808
1400 DATA 0660AD85A78D12A4,899
1401 DATA AD87A78D13A4A202,963
1402 DATA 2018A4E0FFD00386,1044
1403 DATA 0660A001AD14A491,765
1404 DATA 00A902850660A901,576
1405 DATA 850660A200BD87A7,888
1406 DATA 9D10A4E8E004D0F5,1250
1407 DATA A2002018A4E0FFD0,1069
1408 DATA 03860660A001AD14,593
1409 DATA A49100C8AD15A491,1012
1410 DATA 00A903850660A200,569
1411 DATA BD86A79D10A4E8E0,1283
1412 DATA 04D0F5A2002018A4,839
1413 DATA E0FFD003860660A5,1091
1414 DATA 018585A500186902,435
1415 DATA 85049002E605AD14,711
1416 DATA A43506AD15A48507,801
1417 DATA A507C5053037A506,640
1418 DATA C5043031A50738E5,755
1419 DATA 050505A50638E504,603
1420 DATA 8504B002C605A505,680
1421 DATA C900F005A9FF8506,1009
1422 DATA 60A504C97F30034C,720
1423 DATA 8CA0A001A5049100,787
1424 DATA A902850660A505D3,632
1425 DATA E5078507A50438E5,830
1426 DATA 0605068002C607A5,693
1427 DATA 07C900F005A9FF85,1010
1428 DATA 0660A506C9823003,655
1429 DATA 4CB0ACA50649FF18,960
1430 DATA 5901A0019100A902,503
1431 DATA 850660A9008D0C02,745
1432 DATA A9148D0902A9028D,861
1433 DATA D0A2A2000A48BD30,829
1434 DATA AD20B0F268AAE8E0,1353
1435 DATA 13D0F1A9FF8D0FC0,1287
1436 DATA ADFC02C91EF01EC9,1129
1437 DATA 1FD0F5A9FF8D0FC0,1303
1438 DATA A9948D0C602A9288D,1008
1439 DATA D902A90500D0A02A9,923
1440 DATA 7D20B0F268A9FF8D,1236
1441 DATA FC0220A1A74C0B0A,874
1442 DATA 312D42415349439B,603
1443 DATA 322D4D6F6E69746F,725
1444 DATA 729B980000000000,424
1445 DATA 0000000000000000,0
1446 DATA 000020E9A7E0FFD0,1119
1447 DATA 0160AD14A48500AD,760
1448 DATA 15A48501A93CE200,758
1449 DATA F2A9FF8506A600A4,1135
1450 DATA 0120B4A4A9202000,786
1451 DATA F2A9000506A000B1,1015
1452 DATA 002506F008A9A2A0,534
1453 DATA B0F24C92ADA92E20,1060
1454 DATA B0F24606A506C900,866
1455 DATA D0E3A99820B0F2E6,1439
1456 DATA 00D002E601A9FF8D,1006
1457 DATA FC02ADFC02C9FFD0,1377
1458 DATA F9C921F0A9F202BA7,1140
1459 DATA A9FF8D0FC02604D4,1060
1460 DATA 3B2E4753404C3E23,496
1461 DATA 24492020202000A5,413
1462 DATA 17A84A81CAB26A8,832
1463 DATA 13A952AD40A9EEAD,1087
1464 DATA 72AE93AECEAE0DAD,1397
1465 DATA BDADBDADBDADAD200,1248
1466 DATA BD7CA79D10A4E8E0,1273
1467 DATA 04D0F5A2002018A4,839
1468 DATA E0FFD00160AD14A4,1141
1469 DATA 8500AD15A48501A9,794
1470 DATA 808506A9008507A2,738
1471 DATA 00B081A7C92AD009,945
1472 DATA A506050785074C2D,444
1473 DATA AEC92ED0074606E8,1152
1474 DATA E008D0E5A000A507,1001
1475 DATA 9100E600D002E601,816
1476 DATA A93CE2008F2A9FF85,1230
1477 DATA 06A600A40120B4A4,713
1478 DATA A92020B0F260EAE,1215
1479 DATA EAEAEAEAEAEAEAE,1872
1480 DATA EAEAEAEAEAEAEAE,1872
1481 DATA EAEAEAEAEAEAEAE,1872
1482 DATA EAEAA27CA0A786F3,1458
1483 DATA 34FA420086F22000,946
1484 DATA D890016020D2D990,1060
1485 DATA 0160A92420B0F2A9,921
1486 DATA FF8506A6D4A4D520,1101
1487 DATA B4A4A99820B0F220,1150
1488 DATA 2BA760A200BD7CA7,948
1489 DATA 9D10A4E8E004D0F5,1250
1490 DATA A2002018A4E0FFD0,1069
1491 DATA 0160AD14A485D4AD,972

```



```

1492 DATA 15A485D520A0D920,982
1493 DATA E6D8A92320B0F2A0,1260
1494 DATA FFC69840B1F3297F,1267
1495 DATA 20B0F268A8B1F3C9,1343
1496 DATA 7F90EEA99820B0F2,1283
1497 DATA 202BA76020E9A7E0,994
1498 DATA FFD00160AD14A485,1050
1499 DATA 00AD15A48501A9FF,916
1500 DATA 8506A600A40120B4,682
1501 DATA A4A92D20B0F2A000,988
1502 DATA 9848A91B20B0F268,974
1503 DATA A048B100C99B0002,983
1504 DATA A92020B0F268A8C8,1123
1505 DATA C020D0E4A99820B0,1192
1506 DATA F2A500186920B050,701
1507 DATA 9002E601A9FF80FC,1194
1508 DATA 02ADFC02C9FF0F9,1374
1509 DATA C921F0B6202BA7A9,1067
1510 DATA FF8DFC0260000000,746
1511 DATA 0000000000000000,0
1512 DATA 0000000000000000,0
1513 DATA 0000000000000000,0
1514 DATA 0000000000000000,0
1515 DATA 0000000000000000,0
1516 DATA 0000000000000000,0
1517 DATA 0000000000000000,0
1518 DATA 0000000000000000,0
1519 DATA 0000000000000000,0
1520 DATA 0000000000000000,0
1521 DATA 0000000000000000,0
1522 DATA 0000000000000000,0
1523 DATA 0000000000000000,0
1524 DATA 0000000000000000,0
1525 DATA 0000000000000000,0
1526 DATA 0000000000000000,0
1527 DATA 0000000000000000,0
1528 DATA 0000000000000000,0
1529 DATA 0000000000000000,0
1530 DATA 0000000000000000,0
1531 DATA 0000000000000000,0
32767 REM

```

*** BASIC-Zauberer ***

Die Super-Hilfsdiskette für Ihren ATARI!

Über 40 Files helfen Ihnen bei vielen Programmierproblemen - jeden Tag!

BASIC-Zauberer - das sind:

* BASIC-Erweiterungen wie

- Autonumber
- REM Killer
- REM Marker & -Entmarker
- COLOR-Maker
- HEX, BIN->DEC -Wandlung
- Block-Zeilen-Delete
- Variablen-Liste...

* Tolle Hilfsprogramme wie

- Zeichensatzeditor
- Farb-Zeicheneditor
- TT-Titelmaschine

* nützliche MS-Programme wie

- Zeichensatz LOAD/SAVE/COPY
- 128-Farben-Effekt
- Bilder-Laderoutine

und vieles mehr! nur DM 29,-

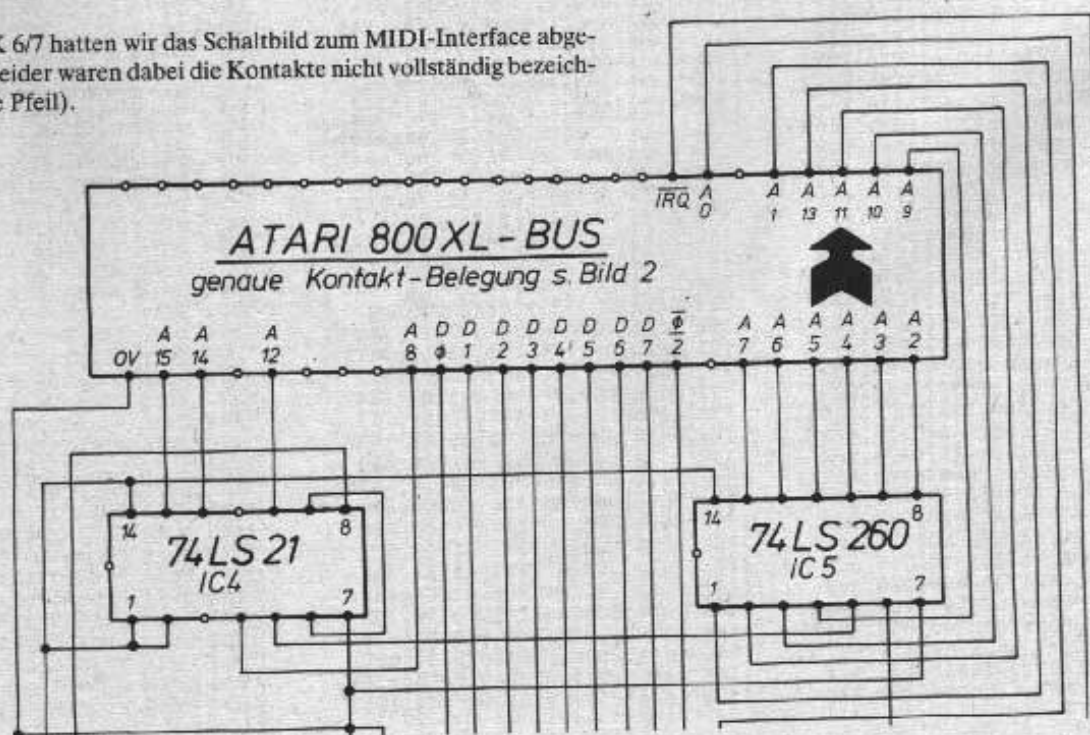


Bitte bestellen Sie per Nachnahme oder Scheck bei:

Thomas Tausend, Aied 17
Tel.: (08376)/1669
8961 Sulzberg

Bild 1: MIDI-Interface für ATARI 800XL

In der CK 6/7 hatten wir das Schaltbild zum MIDI-Interface abgedruckt. Leider waren dabei die Kontakte nicht vollständig bezeichnet (siehe Pfeil).



3-D Flying Ace

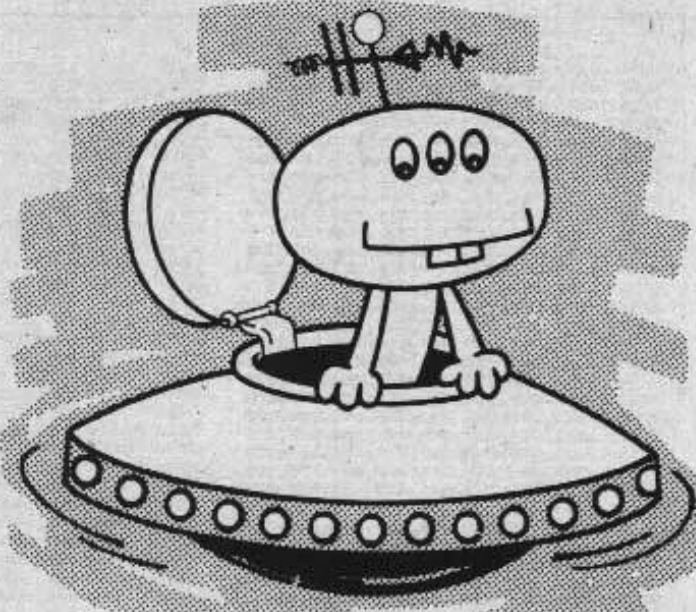
Achtung! Besitzer eines STs mit nur 512 K müssen vor dem Abtippen den Grafik-Puffer abschalten. Das Spiel läuft nur mit einem SW-Monitor.

Wenn Sie das Programm richtig abgetippt und abgespeichert haben, wird nach dem Start das Programm RADDISH.TOS auf Diskette geschrieben, das Sie in Zukunft nur noch vom Desktop aus laden müssen. Nach dem Laden des Spieles muß noch ein Joystick in den Maus-Anschluß gesteckt werden.

Sie sitzen nun in Ihrem Raumjäger und sehen einen

Planeten unter sich hinwegscrollen. Ihr Ziel ist es jetzt, so lange wie möglich den gegnerischen Angriffen, die pausenlos stattfinden, standzuhalten. Sie können Ihr Fadenkreuz über einen auf Sie zuffliegenden Gegner steuern und durch Knopfdruck versuchen, ihn zu treffen. Doch passen Sie auf. Wenn ein Gegner so weit an Sie herankommt, daß er abdrehet, so schießt dieser auf Sie, und Ihr Energie-Pegel sinkt. Das Spiel ist zu Ende, wenn Ihre Energie aufgebraucht ist. Punkte erhalten Sie für das Abschießen eines Gegners (1000) und für jede Sekunde, die Sie durchhalten (70).

Christian Rduch



```

1  *****
2  *
3  * 3-D FLYING ACE *
4  *
5  * (C) BY RADDISH-SOFT GERMANY *
6  *
7  * Written by Christian Rduch *
8  * Tel. 02365 / 59000 *
9  *****
23 dim ps(85)
24 restore for i = 1 to 85
25 read ps(i):next
27 restore 1000:for i = 1 to 84
28 s=0:for t = 0 to 69:read w
29 s=s+w
30 next if s < ps(i) then ?"DATAFEH
LER IN "1000+i*100:" BIS "1090+i*100
40 if s < ps(i) then stop
50 next ?"DATAS WERDEN GESCHRIEBEN"
60 restore 1000
70 open "R:",#1,"RADDISH.TOS",2
75 field #1, 2 as d$
77 z=0
78 z=z+1:read b:if b<0 then 95
82 read ciif c<0 then 95
85 w=256+b+c:rlset d$=skis(w)
90 put #1,z
92 goto 80
95 close:"FERTIG !!!!!!!":end
100 data 3794,4235,4929,3522,6124
110 data 4040,6115,4652,4297,4031
120 data 3998,6275,3051,3279,3316
130 data 4074,3698,3814,3880,3977
140 data 3804,3400,4153,3922,3519
150 data 4039,3837,3755,3590,3922
160 data 3670,3692,2419,2200,3307
170 data 3587,3024,4640,5234,2560
180 data 2615,1698,2314,4070,4577
190 data 3824,3641,5094,4951,5163
200 data 2193,1014,702,64,402,727
210 data 697,1037,1511,19,2693,417
220 data 53,2733,977,301,3245,1295
230 data 1088,3181,1917,33,0,831
240 data 3607,385,1681,3413,2499,514
250 data 084,480,496,710,390
1000 data 096,026,000,000,021,150,000
1010 data 000,000,000,000,000,000,000
1020 data 000,000,000,000,000,000,000
1030 data 000,000,000,000,000,000,000
1040 data 047,060,000,000,012,052,063
1050 data 060,000,009,078,065,092,143
1060 data 032,060,000,000,039,016,034
1070 data 060,000,000,001,244,081,201
1080 data 235,254,001,200,235,244,063
1090 data 060,235,255,047,060,000,015
1100 data 120,000,047,060,000,015,128
1110 data 000,063,060,000,005,078,078
1120 data 223,252,000,000,000,012,063
1130 data 060,000,020,063,060,000,000
1140 data 063,060,000,021,070,078,092
1150 data 143,066,167,063,060,000,032
1160 data 078,065,092,143,019,252,000
1170 data 057,000,000,010,168,019,252
1180 data 000,057,000,000,010,167,019
1190 data 252,000,057,000,000,010,170
1200 data 032,124,000,000,010,194,017
1210 data 060,000,048,177,252,000,000
1220 data 010,187,102,000,255,244,035
1230 data 252,000,000,010,196,000,000
1240 data 013,180,035,252,000,000,010
1250 data 196,000,000,013,216,051,252
1260 data 000,000,000,000,010,142,047
1270 data 060,000,000,013,184,063,060
1280 data 000,009,078,065,092,143,063
1290 data 060,000,034,078,078,084,143
1300 data 065,249,000,000,000,000,067
1310 data 249,000,000,010,042,035,192
1320 data 000,000,010,148,035,240,000
1330 data 024,000,000,010,144,033,137
1340 data 000,024,050,060,000,020,051
1350 data 196,000,000,010,154,077,000
1360 data 009,046,036,124,000,000,013
1370 data 226,030,124,000,000,014,000
1380 data 032,124,000,000,014,056,034
1390 data 124,000,000,014,020,032,060
1400 data 000,000,000,013,052,219,040
1410 data 217,001,200,255,250,032,124
1420 data 000,015,120,000,046,060,000
1430 data 000,031,063,032,252,255,255
1440 data 255,255,032,060,000,000,001
1450 data 044,001,200,255,254,001,207
1460 data 255,230,032,121,000,000,004
1470 data 006,012,152,000,000,000,000
1480 data 102,000,255,240,033,060,000
1490 data 000,002,000,032,124,000,015
1500 data 171,192,030,060,255,255,255
1510 data 255,046,060,000,000,000,013
1520 data 044,124,000,000,013,226,012
1530 data 121,000,000,000,000,013,224
1540 data 103,000,000,020,051,252,000
1550 data 000,000,000,013,224,060,060
1560 data 170,170,079,249,000,000,001
1570 data 120,051,252,000,001,000,000
1580 data 013,224,060,060,000,000,050
1590 data 030,040,060,000,000,000,035
1600 data 040,190,001,204,255,252,209
1610 data 252,000,000,000,000,002,131
1620 data 102,130,102,000,255,232,001
1630 data 207,255,104,012,121,000,001
1640 data 000,000,010,142,102,000,255
1650 data 246,032,121,000,000,004,006
1660 data 012,152,000,000,002,000,102
1670 data 000,255,240,033,060,000,000
1680 data 000,000,032,124,000,000,010
1690 data 194,034,124,000,000,011,239
1700 data 019,032,177,252,000,000,010
1710 data 107,102,000,255,246,047,060
1720 data 000,000,011,122,063,060,000
1730 data 009,078,065,092,143,016,057
1740 data 000,000,010,141,002,000,000
1750 data 120,012,000,000,120,102,000
1760 data 255,240,078,249,000,000,000
1770 data 000,036,124,000,000,021,158
1780 data 160,012,036,124,000,000,023
1790 data 000,160,012,036,124,000,000
1800 data 023,016,160,012,036,124,000
1810 data 000,022,190,160,012,036,124
1820 data 000,000,022,124,160,012,036
1830 data 124,000,000,021,232,160,012
1840 data 036,124,000,000,022,050,160
1850 data 012,012,121,000,000,000,000
1860 data 013,254,103,000,000,020,050
1870 data 060,170,170,051,252,000,001
1880 data 000,000,013,224,078,249,000
1890 data 000,002,090,050,060,000,000
1900 data 051,252,000,000,000,000,013
1910 data 224,044,124,000,000,013,226
1920 data 042,124,000,000,014,056,040
1930 data 124,000,000,014,020,046,060
1940 data 000,000,000,013,012,121,000
1950 data 000,000,000,013,224,103,000
1960 data 000,018,066,121,000,000,013
1970 data 224,050,060,170,170,078,249
1980 data 000,000,002,162,051,252,000
1990 data 001,000,000,013,224,050,060
2000 data 000,000,003,003,012,005,000
2010 data 000,102,000,000,110,107,252
2020 data 000,000,014,056,103,000,000
2030 data 000,078,249,000,000,003,002
2040 data 032,060,000,000,000,013,032
2050 data 124,000,000,013,224,034,124
2060 data 000,000,014,000,036,124,000
2070 data 000,014,056,036,124,000,000
2080 data 014,020,040,217,052,219,001
2090 data 200,255,250,012,121,000,000
2100 data 000,000,013,254,103,000,000
2110 data 014,066,121,000,000,013,254
2120 data 078,249,000,000,003,002,051
2130 data 252,000,001,000,000,013,254
2140 data 050,140,040,022,192,252,000
2150 data 000,032,124,000,015,171,192
2160 data 209,192,044,060,000,000,000
2170 data 035,040,197,001,206,255,252
2180 data 002,006,004,142,004,141,004
2190 data 140,001,207,255,002,032,121
2200 data 000,000,014,000,036,124,000
2210 data 000,022,050,040,057,000,000
2220 data 014,004,050,057,000,000,014
2230 data 006,160,013,032,121,000,000
2240 data 014,092,036,124,000,000,021
2250 data 232,040,057,000,000,014,004
2260 data 050,057,000,000,014,006,160
2270 data 013,032,121,000,000,014,096
2280 data 036,124,000,000,022,124,040
2290 data 057,000,000,014,004,050,057
2300 data 000,000,014,006,160,013,032
2310 data 121,000,000,013,166,036,124
2320 data 000,000,022,190,040,057,000
2330 data 000,013,000,050,057,000,000
2340 data 013,002,160,013,032,121,000
2350 data 000,013,170,036,124,000,000
2360 data 023,016,040,057,000,000,013
2370 data 000,050,057,000,000,013,002
2380 data 160,013,032,121,000,000,013
2390 data 174,036,124,000,000,023,090
2400 data 040,057,000,000,013,000,050
2410 data 057,000,000,013,002,160,013
2420 data 012,121,000,001,000,000,013
2430 data 070,103,000,000,010,035,252
2440 data 000,000,013,092,000,000,013
2450 data 004,078,249,000,000,003,236
2460 data 035,252,000,000,019,152,000
2470 data 000,013,004,052,121,000,000
2480 data 013,004,036,124,000,000,021
2490 data 150,040,057,000,000,013,000
2500 data 050,057,000,000,013,090,160
2510 data 013,097,000,006,092,012,121
2520 data 000,001,000,000,013,170,102
2530 data 000,000,150,003,121,000,000
2540 data 013,220,012,121,000,000,000
2550 data 000,013,220,102,000,000,140
2560 data 051,252,000,023,000,000,013
2570 data 220,032,121,000,000,013,100
2580 data 035,216,000,000,014,000,035
2590 data 216,000,000,014,092,035,216

```



```

2600 data 000,000,014,076,035,200,000
2610 data 000,013,180,012,185,000,000
2620 data 011,040,000,000,013,180,102
2630 data 000,000,000,051,252,000,000
2640 data 000,000,013,178,012,121,000
2650 data 000,000,000,013,224,103,000
2660 data 000,030,035,252,000,000,019
2670 data 226,000,000,014,000,035,252
2680 data 000,000,020,044,000,000,014
2690 data 072,035,252,000,000,020,118
2700 data 000,000,014,076,078,249,000
2710 data 000,004,178,035,252,000,000
2720 data 020,192,000,000,014,000,035
2730 data 252,000,000,021,010,000,000
2740 data 014,092,035,252,000,000,021
2750 data 004,000,000,014,076,003,121
2760 data 000,000,013,222,012,121,000
2770 data 000,000,000,013,222,102,000
2780 data 000,016,051,252,000,002,000
2790 data 000,013,222,002,121,000,000
2800 data 014,006,012,185,000,000,010
2810 data 200,000,000,013,180,102,000
2820 data 000,010,051,252,000,120,000
2830 data 000,014,006,012,121,000,000
2840 data 000,000,013,178,102,000,000
2850 data 198,012,185,000,000,019,226
2860 data 000,000,014,000,102,000,000
2870 data 076,004,121,000,000,014,004
2880 data 012,121,000,075,000,000,014
2890 data 004,109,000,000,166,035,252
2900 data 000,000,019,070,000,000,014
2910 data 000,035,252,000,000,014,100
2920 data 000,000,014,072,035,252,000
2930 data 000,019,152,000,000,014,076
2940 data 051,252,000,001,000,000,013
2950 data 178,063,060,000,017,078,078
2960 data 004,143,019,252,000,001,000
2970 data 000,014,004,019,192,000,000
2980 data 014,005,035,252,000,000,010
2990 data 200,000,000,013,180,078,249
3000 data 000,000,005,184,005,121,000
3010 data 000,014,004,012,121,000,000
3020 data 000,000,014,004,078,000,000
3030 data 072,051,252,000,001,000,000
3040 data 013,178,035,252,000,000,019
3050 data 070,000,000,014,000,035,252
3060 data 000,000,014,100,000,000,014
3070 data 072,035,252,000,000,019,152
3080 data 000,000,014,076,063,060,000
3090 data 017,078,078,004,143,019,252
3100 data 000,000,010,000,014,004,019
3110 data 192,000,000,014,005,035,252
3120 data 000,000,010,200,000,000,013
3130 data 100,012,121,000,001,000,000
3140 data 013,074,102,000,000,150,003
3150 data 121,000,000,013,076,012,121
3160 data 000,000,000,000,013,076,102
3170 data 000,000,140,051,252,000,030
3180 data 000,000,013,076,032,121,000
3190 data 000,013,216,035,216,000,000
3200 data 013,166,035,216,000,000,013
3210 data 170,035,216,000,000,013,216,012
3220 data 035,200,000,000,013,216,012
3230 data 185,000,000,011,040,000,000
3240 data 013,216,102,000,000,000,051
3250 data 252,000,000,000,000,013,074
3260 data 012,121,000,000,000,000,013
3270 data 224,103,000,000,038,035,252
3280 data 000,000,019,226,000,000,013
3290 data 166,035,252,000,000,020,044
3300 data 000,000,013,178,035,252,000
3310 data 000,020,110,000,000,013,174
3320 data 078,249,000,000,006,076,035
3330 data 252,000,000,020,192,000,000
3340 data 013,166,035,252,000,000,021
3350 data 010,000,000,013,178,035,252
3360 data 000,000,021,004,000,000,013
3370 data 174,003,121,000,000,013,078
3380 data 012,121,000,000,000,000,013
3390 data 078,102,000,000,016,051,252
3400 data 000,003,000,000,013,078,002
3410 data 121,000,000,013,002,012,105
3420 data 000,000,014,200,000,000,013
3430 data 216,102,000,000,010,051,252
3440 data 000,120,000,000,013,002,012
3450 data 121,000,000,000,000,013,074
3460 data 102,000,000,206,012,185,000
3470 data 000,019,226,000,000,013,166
3480 data 102,000,000,076,004,121,000
3490 data 000,013,000,012,121,002,073
3500 data 000,000,013,000,109,000,000
3510 data 174,035,252,000,000,019,078
3520 data 000,000,013,166,035,252,000
3530 data 000,014,100,000,000,013,170
3540 data 035,252,000,000,019,152,000
3550 data 000,013,174,051,252,000,001
3560 data 000,000,013,074,063,060,000
3570 data 017,078,078,004,143,019,252
3580 data 000,001,000,000,013,000,019
3590 data 192,000,000,013,001,035,252
3600 data 000,000,010,200,000,000,013
3610 data 216,078,249,000,000,007,110
3620 data 005,121,000,000,013,000,012
3630 data 121,000,060,000,000,013,000
3640 data 070,000,000,000,051,252,000
3650 data 001,000,000,013,074,035,252
3660 data 000,000,019,078,000,000,013
3670 data 166,035,252,000,000,014,100
3680 data 000,000,013,178,035,252,000
3690 data 000,019,152,000,000,013,174
3700 data 063,060,000,017,078,078,004
3710 data 143,019,252,000,001,000,000
3720 data 013,000,019,192,000,000,013
3730 data 001,051,252,000,200,000,000
3740 data 013,000,035,252,000,000,010
3750 data 200,000,000,013,216,030,057
3760 data 000,000,010,141,002,007,000
3770 data 120,012,007,000,120,103,000
3780 data 000,000,078,249,000,000,000
3790 data 130,012,121,000,000,000,000
3800 data 013,078,103,000,000,242,051
3810 data 252,000,050,000,000,013,072
3820 data 051,252,000,001,000,000,013
3830 data 070,040,057,000,000,013,000
3840 data 176,121,000,000,014,004,070
3850 data 000,000,078,006,064,000,032
3860 data 176,121,000,000,014,004,109
3870 data 000,000,004,040,057,000,000
3880 data 013,070,176,121,000,000,014
3890 data 006,109,000,000,060,004,064
3900 data 000,000,176,121,000,000,014
3910 data 000,070,000,000,054,035,252
3920 data 000,000,011,040,000,000,014
3930 data 000,035,252,000,000,012,178
3940 data 000,000,014,072,035,252,000
3950 data 000,012,252,000,000,014,076
3960 data 035,252,000,000,010,156,000
3970 data 000,013,100,051,252,000,100
3980 data 000,000,013,220,077,000,002
3990 data 046,040,057,000,000,013,000
4000 data 176,121,000,000,013,000,090
4010 data 000,000,078,006,064,000,032
4020 data 176,121,000,000,013,000,109
4030 data 000,000,004,040,057,000,000
4040 data 013,070,176,121,000,000,013
4050 data 002,109,000,000,054,035,252
4060 data 000,000,176,121,000,000,013
4070 data 002,078,000,000,054,035,252
4080 data 000,000,011,040,000,000,013
4090 data 166,035,252,000,000,012,178
4100 data 000,000,013,170,035,252,000
4110 data 000,012,252,000,000,013,174
4120 data 035,252,000,000,010,196,000
4130 data 000,013,216,051,252,000,100
4140 data 000,000,013,076,077,000,001
4150 data 190,012,121,000,000,000,000
4160 data 013,070,103,000,000,028,003
4170 data 121,000,000,013,072,012,121
4180 data 000,000,000,000,013,072,102
4190 data 000,000,010,051,252,000,000
4200 data 000,000,013,070,030,057,000
4210 data 000,010,141,002,007,000,001
4220 data 012,007,000,001,103,000,000
4230 data 000,078,249,000,000,000,210
4240 data 012,121,000,150,000,000,013
4250 data 070,109,000,000,000,005,121
4260 data 000,000,013,070,030,057,000
4270 data 000,010,141,002,007,000,002
4280 data 012,007,000,002,102,000,000
4290 data 020,012,121,001,074,000,000
4300 data 113,070,070,000,000,000,004
4310 data 121,000,000,013,070,030,057
4320 data 000,000,010,141,002,007,000
4330 data 000,020,012,121,000,050,000
4340 data 000,013,000,109,000,000,000
4350 data 007,121,000,000,013,000,030
4360 data 057,000,000,010,141,002,007
4370 data 000,000,012,007,000,000,102
4380 data 000,000,020,012,121,002,038
4390 data 000,000,013,000,070,000,000
4400 data 000,006,121,000,000,013,000
4410 data 012,103,000,000,020,192,000
4420 data 000,014,000,103,000,000,050
4430 data 012,103,000,000,020,192,000
4440 data 000,013,166,103,000,000,036
4450 data 012,103,000,000,019,226,000
4460 data 000,013,166,103,000,000,022
4470 data 000,014,000,103,000,000,000
4480 data 012,103,000,000,019,226,000
4490 data 078,249,000,000,009,226,003
4500 data 057,000,000,010,170,012,057
4510 data 000,047,000,000,010,170,102
4520 data 000,000,006,019,252,000,057
4530 data 000,000,010,170,003,057,000
4540 data 000,010,160,012,057,000,047
4550 data 000,000,010,160,102,000,000
4560 data 060,019,252,000,057,000,000
4570 data 010,160,003,057,000,000,010
4580 data 167,012,057,000,047,000,000
4590 data 010,167,102,000,000,034,019
4600 data 252,000,040,000,000,010,167
4610 data 019,252,000,040,000,000,010
4620 data 160,019,252,000,040,000,000
4630 data 010,170,051,252,000,001,000
4640 data 000,010,142,003,121,000,000
4650 data 010,130,012,121,000,000,000
4660 data 000,010,130,103,000,000,004
4670 data 078,117,051,252,000,070,000
4680 data 000,010,130,047,060,000,000
4690 data 010,156,063,060,000,007,078
4700 data 065,072,143,070,117,019,252
4710 data 000,000,000,000,010,140,072
4720 data 121,000,000,010,153,063,060
4730 data 000,000,063,060,000,025,078
4740 data 078,000,143,070,117,016,060
4750 data 000,002,067,249,000,000,010
4760 data 140,010,216,003,000,102,000
4770 data 255,250,078,117,032,124,000
4780 data 000,010,190,062,016,012,016
4790 data 000,050,102,000,000,010,016
4800 data 100,000,040,003,136,002,024
4810 data 003,136,177,252,000,000,010
4820 data 107,102,000,255,230,070,117
4830 data 032,124,000,000,010,193,002
4840 data 016,012,016,000,050,102,000
4850 data 000,010,016,100,000,040,003
4860 data 136,002,024,003,136,177,252
4870 data 000,000,010,107,102,000,255
4880 data 230,078,117,000,001,000,000
4890 data 000,000,000,000,000,000,000
4900 data 000,000,000,000,020,020,020
4910 data 013,069,078,069,032,071,073
4920 data 069,032,050,032,057,057,046
4930 data 057,032,037,032,032,032,032
4940 data 032,000,005,070,075,004,069
4950 data 032,050,032,040,040,040,040
4960 data 040,040,040,000,000,000,000
4970 data 019,070,000,000,014,100,000
4980 data 000,019,152,000,000,019,078
4990 data 000,000,014,174,000,000,019
5000 data 152,000,000,019,078,000,000
5010 data 014,240,000,000,019,152,000
5020 data 000,019,078,000,000,015,066
5030 data 000,000,019,152,000,000,019
5040 data 070,000,000,015,140,000,000
5050 data 019,152,000,000,015,214,000
5060 data 000,016,032,000,000,016,106
5070 data 000,000,016,100,000,000,016
5080 data 254,000,000,017,072,000,000
5090 data 017,146,000,000,017,220,000
5100 data 000,010,030,000,000,010,112
5110 data 000,000,010,106,000,000,019
5120 data 004,000,032,000,000,000,000
5130 data 000,000,000,001,000,000,000
5140 data 000,000,000,000,000,000,000
5150 data 000,000,000,000,000,000,000
5160 data 000,000,000,000,000,000,000
5170 data 000,000,000,000,000,000,000
5180 data 000,214,000,214,001,073,001
5190 data 073,002,050,002,050,000,172
5200 data 000,172,013,133,013,133,002
5210 data 070,002,070,005,129,005,129
5220 data 000,052,000,052,003,064,003
5230 data 064,000,000,000,000,010,010
5240 data 010,010,010,010,010,010,010
5250 data 010,010,010,010,010,010,010
5260 data 010,010,010,010,010,010,010
5270 data 010,010,010,010,010,010,010
5280 data 013,060,065,003,032,007,065
5290 data 002,039,003,032,060,065,070
5300 data 070,032,007,077,072,074,032
5310 data 033,033,010,010,013,003,073
5320 data 069,032,003,073,078,060,032
5330 data 060,069,077,032,071,069,071
5340 data 070,069,002,032,069,002,076
5350 data 069,071,069,078,032,033,010
5360 data 010,010,013,065,076,076,069
5370 data 002,060,073,078,071,003,032
5380 data 072,065,066,069,078,032,003
5390 data 073,069,032,040,040,040,040
5400 data 040,040,040,032,000,000,000
5410 data 075,004,069,032,071,069,072
5420 data 079,076,004,032,033,010,010
5430 data 010,010,013,060,002,154,067
5440 data 075,069,078,032,003,073,069
5450 data 032,060,069,078,032,066,005
5460 data 004,004,079,078,032,070,154
5470 data 002,032,069,073,078,069,078
5480 data 032,078,069,005,069,078,032
5490 data 006,069,002,003,005,067,072
5500 data 013,000,007,069,076,076,079
5510 data 077,069,032,032,004,079,032
5520 data 032,004,072,069,032,032,003
5530 data 004,039,003,010,010,013,032
5540 data 032,051,045,060,032,078,076
5550 data 009,073,078,071,032,065,067
5560 data 069,010,010,013,004,072,073
5570 data 003,032,071,065,077,069,032
5580 data 073,003,032,040,067,041,032
5590 data 066,009,032,010,032,002,065
5600 data 060,060,073,003,072,045,003
5610 data 079,070,004,032,071,069,002
5620 data 077,065,078,009,010,010,010
5630 data 007,002,073,004,004,069,070
5640 data 032,060,009,032,067,072,002
5650 data 073,003,004,073,065,078,032
5660 data 002,060,005,067,072,032,073
5670 data 078,032,049,057,056,054,013
5680 data 000,000,000,016,000,000,000

```



```

5690 data 000,000,000,000,001,000,000
5700 data 000,000,000,000,000,000,000
5710 data 214,000,214,000,018,000,018
5720 data 001,072,001,072,004,182,004
5730 data 182,018,201,018,201,213,042
5740 data 213,042,041,116,041,116,150
5750 data 069,190,069,069,187,069,187
5760 data 082,180,082,180,181,151,181
5770 data 151,076,153,076,153,027,044
5780 data 027,044,000,076,000,076,000
5790 data 000,000,000,000,000,000,000
5800 data 000,001,000,000,000,000,000
5810 data 000,000,000,000,000,000,000
5820 data 023,000,023,000,148,128,148
5830 data 128,106,064,106,064,053,064
5840 data 053,064,148,128,148,128,253
5850 data 160,253,160,005,000,005,000
5860 data 170,240,170,240,186,064,186
5870 data 064,069,160,069,160,026,128
5880 data 026,128,000,000,000,000,000
5890 data 000,000,000,000,000,000,000
5900 data 000,001,000,100,000,010,000
5910 data 200,000,100,000,000,013,092
5920 data 001,064,001,044,000,000,000
5930 data 000,000,000,000,000,000,001
5940 data 000,192,000,192,000,192,000
5950 data 192,000,192,000,192,000,192
5960 data 000,192,000,192,000,192,000
5970 data 192,000,192,063,255,063,255
5980 data 063,255,063,255,000,192,000
5990 data 192,000,192,000,192,000,192
6000 data 000,192,000,192,000,192,000
6010 data 192,000,192,000,192,000,192
6020 data 000,000,000,000,000,000,000
6030 data 000,000,000,014,100,000,000
6040 data 014,100,000,000,014,100,000
6050 data 001,000,000,010,016,010,010
6060 data 010,010,010,010,010,010,010
6070 data 010,010,010,010,010,010,010
6080 data 010,010,010,010,010,010,010
6090 data 010,010,010,010,010,010,010
6100 data 000,000,000,010,196,000,000
6110 data 100,000,010,000,001,000,000
6120 data 000,000,000,000,000,000,000
6130 data 000,000,000,000,000,000,000
6140 data 000,000,000,000,000,000,000
6150 data 000,000,000,000,000,000,000
6160 data 000,000,000,001,000,003,000
6170 data 000,000,010,000,016,000,024
6180 data 000,034,000,046,000,061,000
6190 data 001,000,105,000,135,000,175
6200 data 000,120,000,060,000,040,000
6210 data 030,000,020,000,015,000,012
6220 data 000,010,000,000,000,004,000
6230 data 005,000,004,000,003,000,002
6240 data 000,000,000,000,000,000,000
6250 data 000,000,000,000,000,000,000
6260 data 000,000,000,000,000,000,000
6270 data 000,000,000,000,000,000,000
6280 data 001,044,000,100,000,000,014
6290 data 100,000,000,014,174,000,000
6300 data 016,032,000,016,000,000,000
6310 data 000,000,000,000,000,000,000
6320 data 000,000,000,000,000,000,000
6330 data 000,000,000,000,000,000,000
6340 data 000,000,000,000,000,000,000
6350 data 000,000,000,000,000,000,000
6360 data 000,000,000,000,000,000,000
6370 data 000,000,000,000,000,000,000
6380 data 000,000,000,000,000,000,000
6390 data 000,000,000,000,000,000,000
6400 data 000,000,001,192,001,192,000
6410 data 016,000,000,000,000,000,000
6420 data 000,000,000,000,000,000,000
6430 data 000,000,000,000,000,000,000
6440 data 000,000,000,000,000,000,000
6450 data 000,000,000,000,000,000,000
6460 data 000,000,000,000,000,000,000
6470 data 000,000,000,000,000,000,000
6480 data 000,000,000,000,000,000,000
6490 data 000,000,000,000,000,000,000
6500 data 000,000,000,128,000,128,003
6510 data 224,003,224,000,016,000,000
6520 data 000,000,000,000,001,000,000
6530 data 000,000,000,000,000,000,000
6540 data 000,000,000,000,000,000,000
6550 data 000,000,000,000,000,000,000
6560 data 000,000,000,000,000,000,000
6570 data 000,000,000,000,000,000,000
6580 data 000,000,000,000,000,000,000
6590 data 000,000,000,000,000,000,000
6600 data 000,000,000,120,000,120,007
6610 data 240,007,240,001,064,001,064
6620 data 000,016,000,000,000,000,000
6630 data 000,000,001,000,000,000,000
6640 data 000,000,000,000,000,000,000
6650 data 000,000,000,000,000,000,000
6660 data 000,000,000,000,000,000,000
6670 data 000,000,000,000,000,000,000
6680 data 000,000,000,000,000,000,000
6690 data 000,000,000,000,000,000,000
6700 data 000,000,000,000,000,000,001
6710 data 192,001,192,031,252,031,252

```

```

6720 data 002,032,002,032,000,016,000
6730 data 000,000,000,000,000,000,001
6740 data 000,000,000,000,000,000,000
6750 data 000,000,000,000,000,000,000
6760 data 000,000,000,000,000,000,000
6770 data 000,000,000,000,000,000,000
6780 data 000,000,000,000,000,000,000
6790 data 000,000,000,000,000,000,000
6800 data 000,000,000,000,000,000,000
6810 data 120,000,120,001,192,001,192
6820 data 127,255,127,255,012,024,012
6830 data 024,000,032,000,000,000,000
6840 data 000,000,000,001,000,000,000
6850 data 000,000,000,000,000,000,000
6860 data 000,000,000,000,000,000,000
6870 data 000,000,000,000,000,000,000
6880 data 000,000,000,000,000,000,000
6890 data 000,000,000,000,000,000,000
6900 data 000,000,000,000,000,000,000
6910 data 000,000,000,000,000,000,000
6920 data 000,001,000,001,000,000,000
6930 data 000,000,000,000,000,000,016
6940 data 000,000,000,000,000,000,000
6950 data 001,000,000,000,000,000,000
6960 data 000,000,000,000,000,000,000
6970 data 000,000,000,000,000,000,000
6980 data 000,000,000,000,000,000,000
6990 data 000,000,000,000,000,000,000
7000 data 000,000,000,000,000,000,000
7010 data 000,000,000,001,192,001,192
7020 data 007,112,007,112,255,255,255
7030 data 255,247,247,247,247,024,004
7040 data 024,006,000,000,000,000,000
7050 data 000,000,000,000,001,000,000
7060 data 000,000,000,000,000,000,000
7070 data 000,000,000,000,000,000,000
7080 data 000,000,000,000,000,000,000
7090 data 000,000,000,000,000,000,000
7100 data 000,000,000,000,000,000,000
7110 data 000,000,000,000,000,000,000
7120 data 000,000,000,000,000,000,000
7130 data 000,192,000,192,000,000,000
7140 data 000,000,000,000,000,000,000
7150 data 032,000,000,000,000,000,000
7160 data 000,001,000,000,000,000,000
7170 data 000,000,000,000,000,000,000
7180 data 000,000,000,000,000,000,000
7190 data 000,000,000,000,000,000,000
7200 data 000,000,000,000,000,000,000
7210 data 000,000,000,000,000,000,000
7220 data 000,000,000,000,000,000,000
7230 data 000,000,000,000,000,000,015
7240 data 000,015,000,003,000,003,000
7250 data 000,000,000,000,014,000,000
7260 data 000,000,000,000,000,001,000
7270 data 000,000,000,000,000,000,000
7280 data 000,000,000,000,000,000,000
7290 data 000,000,000,000,000,000,000
7300 data 000,000,000,000,000,000,000
7310 data 000,000,000,000,000,000,000
7320 data 000,000,000,000,000,000,000
7330 data 000,001,192,001,192,007,112
7340 data 007,112,255,255,255,255,239
7350 data 251,239,251,040,006,040,006
7360 data 000,000,000,000,000,000,000
7370 data 000,000,001,000,000,000,000
7380 data 000,000,000,000,000,000,000
7390 data 000,000,000,000,000,000,000
7400 data 000,000,000,000,000,000,000
7410 data 000,000,000,000,000,000,000
7420 data 000,000,000,000,000,000,000
7430 data 000,000,000,000,000,000,000
7440 data 000,000,000,000,000,000,240
7450 data 000,240,000,224,000,224,000
7460 data 000,000,000,000,000,032,000
7470 data 000,000,000,000,000,000,001
7480 data 000,000,000,000,000,000,000
7490 data 000,000,000,000,000,000,000
7500 data 000,000,000,000,000,000,000
7510 data 000,000,000,000,000,000,000
7520 data 000,000,000,000,000,000,000
7530 data 000,000,000,000,000,000,000
7540 data 000,000,000,000,127,000,000
7550 data 000,015,000,015,000,000,000
7560 data 000,000,016,000,000,000,000
7570 data 000,000,000,001,000,000,000
7580 data 000,000,000,000,000,000,000
7590 data 000,000,000,000,000,000,000
7600 data 000,000,000,000,000,000,000
7610 data 000,000,000,000,000,000,000
7620 data 000,000,000,000,000,000,000
7630 data 000,000,000,000,000,000,000
7640 data 000,000,000,120,000,120,001
7650 data 192,001,192,007,112,007,112
7660 data 255,255,255,255,191,254,191
7670 data 254,224,003,224,003,000,000
7680 data 000,000,000,000,000,000,000
7690 data 001,000,000,000,000,000,000
7700 data 000,000,000,000,000,000,000
7710 data 000,000,000,000,000,000,000
7720 data 000,000,000,000,000,000,000
7730 data 000,000,000,000,000,000,000
7740 data 000,000,000,000,000,000,000

```

```

7750 data 000,000,000,000,000,000,000
7760 data 000,000,000,000,255,000,255
7770 data 000,240,000,240,000,120,000
7780 data 120,000,000,032,000,000,000
7790 data 000,000,000,000,001,000,000
7800 data 000,000,000,000,000,000,000
7810 data 000,000,000,000,000,000,000
7820 data 000,000,000,000,000,000,000
7830 data 000,000,000,000,000,000,000
7840 data 000,000,000,000,000,000,000
7850 data 000,000,000,000,000,000,000
7860 data 000,000,000,000,000,000,000
7870 data 000,015,255,015,255,001,251
7880 data 001,251,000,014,000,014,000
7890 data 016,000,000,000,000,000,000
7900 data 000,001,000,000,000,000,000
7910 data 000,000,000,000,000,000,000
7920 data 000,000,000,000,000,000,000
7930 data 000,000,000,000,000,000,000
7940 data 000,000,000,000,000,000,000
7950 data 000,000,000,000,000,000,000
7960 data 001,120,001,120,007,224,007
7970 data 224,030,100,030,100,255,255
7980 data 255,255,255,255,255,255,000
7990 data 000,000,000,000,000,000,000
8000 data 000,000,000,000,000,000,000
8010 data 000,000,000,000,000,000,000
8020 data 000,000,000,000,000,000,000
8030 data 000,000,000,000,000,000,000
8040 data 000,000,000,000,000,000,000
8050 data 000,000,000,000,000,000,000
8060 data 000,000,000,000,000,000,000
8070 data 000,000,000,000,000,000,000
8080 data 000,000,255,240,255,240,223
8090 data 120,223,120,112,000,112,000
8100 data 000,032,000,000,000,000,000
8110 data 000,000,001,000,000,000,000
8120 data 000,000,000,000,000,000,000
8130 data 000,000,000,000,000,000,000
8140 data 000,000,000,000,000,000,000
8150 data 000,000,000,000,000,000,000
8160 data 000,000,000,000,000,000,000
8170 data 000,000,000,000,000,000,000
8180 data 000,000,000,000,000,000,000
8190 data 000,000,000,000,000,000,000
8200 data 000,000,000,000,000,000,000
8210 data 000,000,000,000,000,000,000
8220 data 000,000,000,000,000,000,000
8230 data 000,000,000,000,000,000,000
8240 data 000,000,000,000,000,000,000
8250 data 000,000,000,000,000,000,000
8260 data 000,000,000,000,000,000,000
8270 data 000,000,000,000,000,000,000
8280 data 000,000,000,000,000,000,000
8290 data 000,000,000,000,000,000,000
8300 data 000,000,000,000,000,000,000
8310 data 000,000,032,000,000,000,000
8320 data 000,000,000,001,000,192,000
8330 data 192,000,112,000,112,000,060
8340 data 000,060,000,031,000,031,000
8350 data 003,000,003,000,001,000,001
8360 data 000,000,000,000,000,000,000
8370 data 000,000,000,000,000,000,000
8380 data 000,000,000,000,000,000,000
8390 data 000,000,000,000,000,000,000
8400 data 000,000,000,000,000,000,000
8410 data 000,000,000,000,000,000,016
8420 data 000,000,000,000,000,000,000
8430 data 001,000,000,000,000,000,000
8440 data 000,000,000,000,000,000,120
8450 data 000,120,000,224,040,224,040
8460 data 123,240,123,240,223,220,223
8470 data 220,015,239,015,239,001,255
8480 data 001,255,000,063,000,063,000
8490 data 007,000,007,000,001,000,001
8500 data 000,000,000,000,000,000,000
8510 data 000,000,000,000,000,000,000
8520 data 000,000,000,000,000,000,000
8530 data 000,000,000,000,001,000,000
8540 data 000,000,000,000,000,000,000
8550 data 000,000,000,000,000,000,000
8560 data 000,000,000,000,000,000,000
8570 data 000,000,000,000,000,000,000
8580 data 000,000,000,000,000,000,000
8590 data 000,000,000,192,000,192,000
8600 data 240,000,240,000,040,000,040
8610 data 000,063,120,063,120,011,240
8620 data 011,240,000,000,000,000,000
8630 data 032,000,000,000,000,000,000
8640 data 000,001,000,000,000,000,000
8650 data 000,000,000,000,000,000,000
8660 data 000,000,000,000,000,000,000
8670 data 000,000,000,000,000,000,000
8680 data 000,000,000,000,000,000,000
8690 data 000,000,000,000,000,000,000
8700 data 000,003,000,003,000,031,000
8710 data 031,000,116,000,116,001,252
8720 data 001,252,015,200,015,200,015
8730 data 000,026,000,000,016,000,000
8740 data 000,000,000,000,000,001,000
8750 data 000,000,000,000,000,000,000
8760 data 000,000,000,000,000,001,000
8770 data 001,012,007,012,007,032,006

```



```

0700 data 032,004,059,251,059,251,055
0770 data 240,055,240,255,120,255,120
0800 data 252,000,252,000,224,000,224
0810 data 000,120,000,120,000,000,000
0820 data 000,000,000,000,000,000,000
0830 data 000,000,000,000,000,000,000
0840 data 000,000,000,000,000,000,000
0850 data 000,000,001,003,000,003,000
0860 data 014,000,014,000,060,000,060
0870 data 000,240,000,240,000,192,000
0880 data 192,000,120,000,120,000,000
0890 data 000,000,000,000,000,000,000
0900 data 000,000,000,000,000,000,000
0910 data 000,000,000,000,000,000,000
0920 data 000,000,000,000,000,000,000
0930 data 000,000,000,000,000,000,000
0940 data 000,000,000,000,000,000,000
0950 data 002,070,000,000,000,000,010
0960 data 004,000,004,000,000,020,000
0970 data 000,014,010,000,000,000,000
0980 data 024,000,012,010,000,044,016
0990 data 016,000,000,010,014,010,000
1000 data 000,000,000,000,000,000,010

```

```

9010 data 016,000,012,000,000,000,014
9020 data 010,010,000,020,010,012,000
9030 data 000,000,010,010,000,000,040
9040 data 000,000,000,000,000,000,000
9050 data 000,000,000,000,000,000,000
9060 data 000,000,000,000,000,000,000
9070 data 000,000,010,010,004,000,000
9080 data 004,000,000,000,000,014,010
9090 data 000,012,000,000,000,000,000
9100 data 000,004,012,000,010,004,000
9110 data 004,000,004,000,000,004,000
9120 data 004,000,004,000,000,012,000
9130 data 000,004,012,000,010,004,010
9140 data 000,010,004,000,004,000,004
9150 data 000,010,000,000,004,000,000
9160 data 000,012,000,000,000,004,000
9170 data 004,010,000,000,004,000,010
9180 data 000,012,000,000,000,000,000
9190 data 000,004,012,000,010,004,000
9200 data 004,000,004,000,000,004,000
9210 data 004,000,004,000,000,012,000

```

```

9220 data 000,004,012,000,010,004,010
9230 data 000,010,004,000,004,000,004
9240 data 000,010,000,000,004,000,000
9250 data 000,012,000,004,000,004,000
9260 data 004,010,000,000,000,004,000
9270 data 010,000,012,000,000,000,014
9280 data 010,000,014,010,004,000,004
9290 data 000,004,000,004,000,010,000
9300 data 014,010,000,014,010,004,000
9310 data 004,000,004,000,004,000,012
9320 data 010,000,012,000,010,000,010
9330 data 000,020,010,000,020,010,000
9340 data 020,010,000,004,010,004,010
9350 data 004,010,004,010,000,000,012
9360 data 000,000,012,000,000,012,000
9370 data 000,000,000,000,014,000,010
9380 data 000,024,010,020,012,020,060
9390 data 004,004,004,004,004,004,004
9400 data 004,004,004,004,004,004,004
9410 data 004,004,004,004,004,004,004
9420 data 004,004,004,004,004,001,001
9430 data 044,002,004,004,000,036,120
9440 data 004,004,000,-1

```

Textomat ST – der Nachfolger des bekannten C 64 Programms

Der Erfolg begann auf dem C 64: Wolfgang Schellenberger schrieb das erfolgreiche Textverarbeitungsprogramm Textomat, den Nachfolger Textomat Plus und genau passend dazu den Datamat. Seine Auferstehung feiert der Textomat nun auf dem Atari ST und zwar nicht nur in völlig neuem Gewand und deutlich erweiterten Funktionen, sondern auch mit einem neuen Programmier-team.

Der neue Textomat hat sich ganz den Gegebenheiten der Atari Hard- und Software angepaßt. Er präsentiert sich mit Pull-Down-Menüs, 77-Zeichendarstellung und Maussteuerung. Eine 80-Zeichendarstellung konnte aufgrund des Windowrahmens nicht realisiert werden. Doch wenn es darauf ankommt, kann das Programm durch horizontales Scrolling bis zu 180 Zeichen pro Zeile verarbeiten.

Wenn Sie zum ersten Mal mit Textomat arbeiten, werden Sie wahrscheinlich erst einmal mit den Menüs spielen und die Buchstaben beobachten, die sich erstaunlicherweise nach jedem Tastendruck auf dem Bildschirm zeigen. Wer sich dann eingehender mit dem Handbuch beschäftigt, wird bald feststellen, daß es sehr gut geschrieben und mit vielen Abbildungen versehen ist. Außer diesem objektiven Vorteil fallen auch noch viele praktische Tips auf,

die jedem bei seiner täglichen Arbeit nützen werden. Da die Programmierer ihr Produkt selbst am besten kennen, wundert es eigentlich, daß solche Ratschläge so selten in Handbüchern zu finden sind.

Wer mit Textomat seinen ersten Text erstellen will, muß vorher genau überlegen, welchen Modus er hierfür wählt. Neben dem Brief-Modus – dem eigentlichen Textomat-Modus – ist noch ein besonderes Bonbon für die C-Programmierer vorhanden. Wenn Sie Ihren Sourcecode im C-Modus erstellen, werden nach den geschweiften Klammern automatisch drei Leerzeichen nach- bzw. vorangestellt. So ist eine korrekte Syntax garantiert, die auch von sehr genauen Compilern akzeptiert wird. Spuckt das Programm trotzdem Fehlermeldungen aus, geschieht das in der Regel mit Zeilennummern.

Durch die "Goto Zeile"-Option kann der Programmierer dann ganz einfach und ohne viel Sucherei an die entsprechende Stelle springen. Als weitere Möglichkeit zur ausgiebigen Nutzung von Textomat-Files erweist sich der Quick-Modus. Dokumente, die hier erstellt werden, gelangen als reines ASCII-File auf die Diskette. Steuerzeichen und sonstige programminterne Parameter werden vollkommen weggelassen.

Und endlich hat ein Programm auch eine sinnvolle Ver-

wendung für die Funktionstasten, bei denen sich die Designer so viel Mühe gegeben haben. So kann der Anwender jede der 10 Tasten auf drei Ebenen (mit Shift und Alternate) mit je 160 Zeichen belegen (Bild 1). Bei der Verwendung als Floskeltasten mit immer wiederkehrenden Formulierungen kann man die Inhalte abspeichern und vor jeder neuen Arbeit wieder einladen. Belegungen mit Adressen sind ebenfalls denkbar. Diese werden in Serienbriefen je nach Bedarf eingeblendet. Otto Normalverbraucher dürfte mit den möglichen 30 Adressen auskommen.

Bei der Texteingabe wird natürlich die Tastatur benutzt, und die Maus hat Pause. Für ein schnelles Positionieren im File leistet der Pfeil jedoch gute Dienste. Wer allerdings weiter

auf die konservative Art arbeiten will, kann auch nicht meckern, denn über Kontrolltasten sind die üblichen Cursorbewegungen (Textanfang, rechtes Wort, nächster Absatz usw.) möglich.

Zur lästigen Wechselei zwischen Tastatur und Maus kommt es allerdings, wenn man sehr oft die Menüs benötigt, um Funktionen aufzurufen. Aber auch hier haben sich die Programmierer etwas einfallen lassen. Bei einer langsameren Einarbeitung kann man zunächst noch die Menüs in Anspruch nehmen. Die meisten Menüpunkte sind nach dem Druck von ESC auch über eine bestimmte Taste erreichbar, so daß der eingearbeitete Schreiber nicht mit lästigen "Mauseereien" aufgehalten wird. Für Anwender, die andere Dinge



Bild 1: Funktionstasten belegen

im Kopf haben als Abkürzungen, ein Tip: Klappen Sie nacheinander die Menüs herunter, und machen Sie sich durch ALTERNATE + HELP Hardcopy. Schneiden Sie die Menüs dann aus und kleben sie auf dünne Pappe. So ergibt sich eine Referenzkarte, die schnell gemacht ist und im täglichen Einsatz ihre Nützlichkeit beweisen wird.

Das Trennen von Wörtern wird von einem speziellen Algorithmus übernommen. Er beherrscht natürlich aber nicht alle Sonderfälle und Tücken der deutschen Sprache. "CK" und "ß" trennt er korrekt, das Programm hält sich auch daran, "st" nicht auseinander zu reißen.

Natürlich lassen sich auch Steuerzeichen zum Fettdruck, Unterstreichen usw. setzen. Dabei können 10 Zeichen mit beliebigen Codesequenzen für den Drucker definiert werden. Neben der erwähnten Möglichkeit des Auffindens über die Menüs, lassen sich auch die Steuerzeichen mit ESC und einer Befehlstaste anwählen. Daraufhin erscheint eine Box, in der man durch + oder - zu verstehen gibt, ob der Effekt ein- oder ausgeschaltet werden soll. Diese Art der Eingabe ist sehr sinnvoll und begrüßenswert. Enttäuschend ist es indessen, daß selbst die einfachsten Effekte wie Fettdruck oder Unterstreichen nicht im Editiermodus sichtbar sind. Das Programm "Ist Word" geht hier mit gutem Beispiel voran. Selbst Hoch- und Tiefstellen ist nach dem Setzen der Steuerzeichen sichtbar. Bei Textomat muß man zu diesem Zweck erst den Anzeigemodus (Bild 2) bemühen. Am C64 noch verständlich, kann er am Atari ST nur

Kopfschütteln auslösen, zumal nicht einmal zentrierte Zeilen sichtbar sind. Über einen entsprechenden Menüpunkt zeigt der Bildschirm die Wirkung aller Steuerzeichen sowie eventuell gesetzte Fuß- und Kopfzeilen und Seitenzahlen an. Das Dokument ist so sichtbar, wie es nachher auf dem Drucker aussehen wird. Bei den Möglichkeiten, die der Computer selbst bietet, hätte aber dem Anwender das lästige Hin- und

Herschalten zwischen den Modi erspart werden können.

Neben den üblichen Blockfunktionen wie "Suchen und Ersetzen" und "Anzeige des freien Speicherplatzes" (beim 520 ST sind 550 KByte frei) fallen noch vier Dinge auf, die das Programm von anderen seiner Preisklasse deutlich abheben. Da die Programmierer auch eifrige Buchautoren sind, haben sie Funktionen zum Erstellen eines Stichwort- und Inhaltsverzeichnisses eingebaut. Die zu

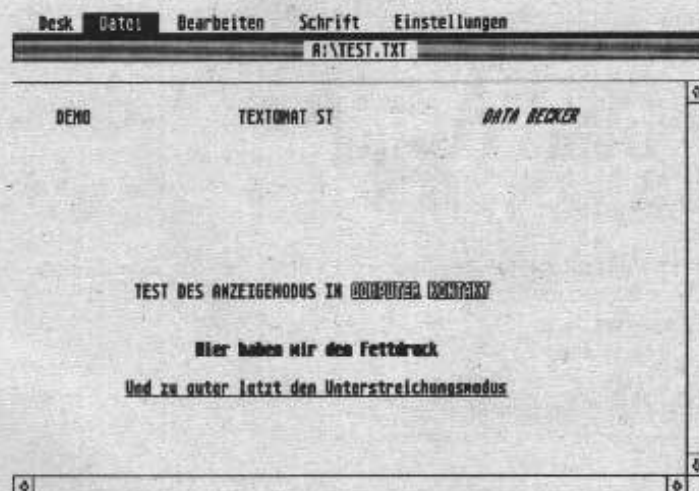


Bild 2: Einige Schriftarten

kennzeichnenden Ausdrücke bzw. Überschriften müssen beim Schreiben markiert werden und werden dann vom Output-Programm herausgesucht und mit Seitenzahlen in ein File geschrieben. Zur Abrundung des Ganzen können Dokumente noch zeilenweise nach dem Alphabet sortiert werden, so daß Stichwortverzeichnisse mit ungeheurer Zeitersparnis erstellt werden.

Nicht immer selbstverständlich für ST-Programme ist die Ausnutzung der internen Uhr, die recht genau und inklusive Datum arbeitet. Mit entsprechenden Hardwarezusätzen kann sie batteriegepuffert werden, so daß die Neueinstellung nach jedem Reset entfällt. So ausgerüstet sind die Datum- und Uhrzeit-Steuerzeichen von Textomat wohl am effektivsten. Besonders beim Briefe Schreiben genügt es nun, einfach das Steuerzeichen zu setzen, und das Datum wird ausgelesen und eingesetzt. Gleiches gilt auch für die Uhrzeit, die man in Schriftstücken aber wohl seltener benötigt.

Als sehr nützlich ist das sog. Formular einzuschätzen, das Grundeinstellungen für das Schreibformat beinhaltet (Bild 3). Hier wird auch ersichtlich, daß man in mehreren Spalten schreiben kann (max. 5), die in Breite und Position festgelegt werden. Ebenso ist der Inhalt der Kopf- und Fußzeilen Bestandteil dieses Formulars, das mit jedem Text (außer im Quick- und C-Modus) abgespeichert wird.

Ein wichtiger Punkt fehlt uns

noch bei der Besprechung aller Besonderheiten dieses einmaligen Textprogramms: das Output-Programm. Es eröffnet ei-

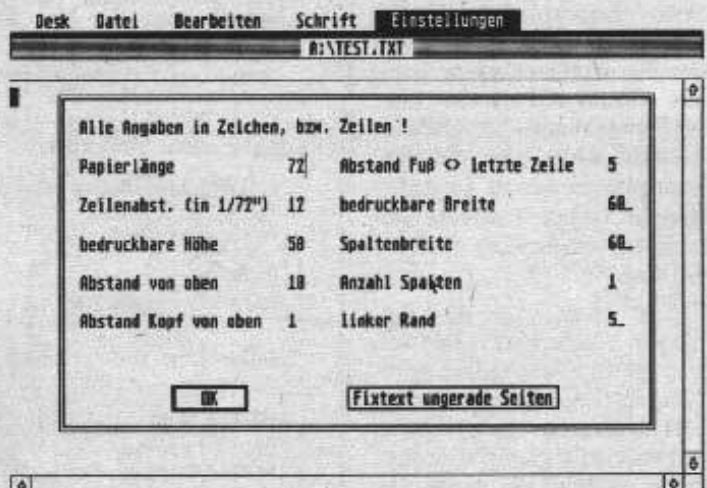


Bild 3: Einstellen des Druckformates

ne Fülle von weiteren Möglichkeiten für den Ausdruck der Texte. Neben der Erstellung der erwähnten Stichwort- und Inhaltsverzeichnisse können Texte wahlweise auch vertikal ausgedruckt sowie Serienbriefe erstellt werden. Weiterhin erhält man eine speicherbare Druckliste, die Dokumente in

der angegebenen Reihenfolge ausdruckt, während man einen Kaffee trinkt. Das Programm ist auch als Schnittstelle zu dem Programm "Text Design" vorgesehen, über das wir sofort nach Erscheinen ebenfalls berichten werden.

Auf der Diskette befinden sich noch zwei sehr nützliche Hilfsprogramme, von denen das eine "Ist Word"-Texte in das Textomat-Format umwandelt. Das andere ist wohl nur für Atarianer mit 512 K interessant. Sehr lange Dokumente, die auf dem aufgerüsteten Computer geschrieben wurden, kann man hier so lange spalten, bis sie in den eigenen Speicher passen.

Leider sind die komfortable Druckeranpassung, die Möglichkeit des Sendens und Empfangens über die RS 232-Schnittstelle sowie diverse andere Funktionen gar nicht zur Sprache gekommen. Trotzdem ist Textomat Plus für den Preis von 99.- DM unschlagbar und übertrifft wohl auch bedeutend teurere Programme (siehe Ist Word) um Längen. Einziger Mangel: der ärgerliche Anzeigemodus. Sonst stimmt alles: optimale Ausnutzung der Hardware, vorbildliches Handbuch, Bedienungskomfort, der seinesgleichen sucht, und last,

but not least ein Preis, der in wirklich keinem Verhältnis zu der Fülle von gebotenen Leistungen steht.

System: Atari 520 ST, 520 ST+, 260 ST
Hersteller: Data Becker
Preis: 99.- DM
Bezugsquelle: Data Becker
Thomas Tai

Das GfA-Basic ist da!

Obwohl im Zusammenhang mit dem Atari ST ständig über C, Modula2 und Assembler gesprochen wird, hat das gute alte Basic neuerdings Chancen, eine Wiedergeburt zu feiern. Da das Digital-Research-Basic, das Atari mitliefert, durch ständige Abstürze, umständliche Editfunktionen, gemühtes Arbeitstempo und dürftigen Befehlssatz ständig unangenehm auffiel, hat wohl kaum jemand Basic für ernsthafte Programmobjekte in Betracht gezogen (abgesehen davon, daß Basic als "unstrukturierte" Programmiersprache bei Schulen, Unis und "echten" Programmierern einen sehr schlechten Ruf genießt). Das neue Basic stammt nun keineswegs aus den USA, sondern ist eine deutsche Entwicklung, auf die man in Fachkreisen seit der Ankündigung auf der CeBit gespannt gewartet hatte. Um es gleich vorweg zu sagen: Meine kühnsten Erwartungen wurden übertroffen!

Der GfA-Basic-Interpreter wird in einem schwarzen Ringordner geliefert, in dem die Programmdiskette und die Dokumentation enthalten ist – natürlich auf deutsch! Die eigentliche Anleitung nimmt dabei 33 Seiten ein, auf denen vor allem der Editor beschrieben wird. Als Anhang A folgen auf über 200 Seiten die Befehle und Funktionen in alphabetischer Reihenfolge, immer aufgegliedert in Syntax, Beispiel, Verwendungsmöglichkeit und Erklärung.

Legt man jetzt die Programmdiskette ein, so fällt auf, daß gleich über 20 Demoprogramme mitgeliefert werden. Der Interpreter belegt dabei nur ca. 50 K! (Zum Vergleich: Beim ST-Basic ist bereits die kurze Version 130 K lang.) Mit dem üblichen Doppelklick auf das Programm-Icon wird das Basic geladen, und es meldet sich mit dem Editor. Im Gegensatz zum Atari-Basic besteht dieser jedoch nicht aus einem Fenster, sondern aus einem makellos weißen Bildschirm, bei dem die ersten beiden Zeilen durch 20 inverse Befehle eingenommen werden. Klickt man

zum Beispiel auf LOAD (mit der Maus oder F1), so erscheint der Fileselector. Jetzt kann man eines der Demoprogramme einladen.

Bild 1 zeigt den Editor nach dem Laden des Demoprogrammes RECHT1. Was dem geübten Basic-Programmierer wohl als erstes auffällt, sind die fehlenden Zeilennummern. Tatsächlich werden im GfA-Basic keine Zeilennummern verwendet. Wer jetzt im Editor das ge-

wohnte RUN eingeben will, hat leider Pech – die Zeile wird einfach mit ins Programm aufgenommen. Auch hier greift man am besten zur Maus oder drückt SHIFT+F10.

Bild 2 zeigt das Ergebnis dieses Programmes, nachdem es mit BREAK unterbrochen wurde. Da der Atari keine BREAK- oder STOP-Taste kennt, wird hierfür die Tastenkombination CONTROL-SHIFT-ALTERNATE ver-

wendet (einfach mit der flachen Hand links unten auf's Keyboard patschen...). Höflich fragt dann der Interpreter, ob er seine Arbeit fortsetzen oder abbrechen soll. Auch das Programmende wird dann durch eine solche Alarmbox angekündigt.

Aber sehen wir uns die Programmerstellung zuerst etwas näher an. Das GfA-Basic enthält einen sehr komfortablen Editor, der sehr schnell arbei-

Demo-Listing

Dim M\$(30)

* Einlesen der Menuestruktur

For N=0 To 30

Read M\$(N)

Exit If M\$(N)="***"

Next N

M\$(N)=""

* Date fr das Menu. Jede Box wird mit einem Leerstring abgeschlossen

Data DESK, About Demo,-----,1,2,3,4,5,6,"

Data EINTRAEGE, Text 1, Text 2, Text 3, Text 4, ENDE,"***"

* Darstellen der Menueleiste in M\$

Menu M\$()

* Prozedurname, die bei Menueereignis angesprungen werden soll

On Menu Gosub Menue

* Eigentliches Programm (hier Endlosschleife)

Do

On Menu

Loop

* Auswertungsprozedur fr Menue

Procedure Menue

Menu Off

As="Sie haben den Punkt! >> "+M\$(Menu(0))+> "<<gewählt"

Alert 0,As,1,"Aha!",T

If M\$(Menu(0))=" ENDE"

End

Else

If M\$(Menu(0))=" About Demo"

Alert 1,"Dies ist nur ein!Demo fr das GfA-BASIC!Wollen Sie weitermachen?",1,"Oh ja!! Nein ",T

If T=2

End

Endif

Endif

Endif

Return

tet, da das bremsende Fenster weggelassen wurde. Er arbeitet vollkommen bildschirmorientiert: Soll in den Programmtext eine Zeile eingefügt werden, muß man den Cursor nur an diese Stelle bewegen. Ein Druck auf INSERT schiebt den Text dann auseinander. Der Editor formatiert das Programm dabei vollautomatisch sehr übersichtlich. Der Inhalt von Schleifen oder Verzweigungen wird um eine Stelle nach rechts verschoben, so daß es schon schwerfällt, eine Schleife nicht zu schließen. Auch die Syntax (Grammatik) der Zeile wird sofort überprüft. Ist diese nicht in Ordnung, so erscheint in der 2. Zeile die Nachricht "Syntax Error", und der Cursor wird an die Stelle bewegt, an der der Interpreter ausgestiegen ist.

Zu Beginn sind natürlich kleine Anfangsschwierigkeiten zu überwinden, wenn man zum Beispiel versucht, mehrere Befehle in eine Zeile zu schreiben. Das GfA-Basic läßt nämlich nur einen Befehl pro Zeile zu! Dadurch wird der Text zwar länger, aber dafür enorm übersichtlich. Warum sollte man auch alles in eine Zeile quetschen? Bei den altmodischen Zeilennummern war dies oft notwendig, andernfalls mußte regelmäßig das ganze Programm neu nummeriert werden.

Um mehr Übersicht zu erreichen, kann auch auf den 8-Pixel-Zeichensatz, der normalerweise auf dem Farbmonitor verwendet wird, umgeschaltet werden. So ist es möglich, doppelt soviel Programmtext gleichzeitig zu betrachten. Mit Flip kann auf den Ausgabebildschirm umgeschaltet werden, denn GfA-Basic arbeitet mit 2 Bildschirmen: einmal zum Editieren und einmal für die Ausgabe. Mit "Direct" werden Befehle direkt eingegeben, die der Rechner dann sofort ausführt (ähnlich dem Command-Window beim ST-Basic). Auch der Direktmodus wird im Ausgabefenster ausgeführt. Wie in jedem guten Editor findet man auch beim GfA-Basic Blockbefehle, um Programmteile zu markieren, zu kopieren und zu löschen. Obwohl ich eine Menge GEM-Editoren kenne, habe ich beim GfA-Editor nichts vermisst.

Aber auch der tollste Editor macht aus einem Basic-Interpreter noch lange kein Super-Basic.

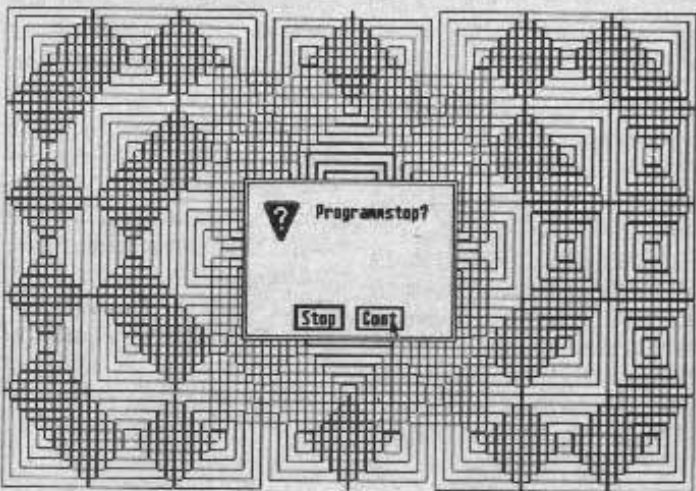
Das GfA-Basic kann aber noch viel mehr. Es ermöglicht strukturiertes Programmieren, das sehr an Pascal oder C erinnert. Hierfür stehen zum Beispiel DO ... LOOP, WHILE ... WEND, REPEAT ... UNTIL, Prozeduren mit lokalen Variablen und IF ... THEN ... ELSE ... ENDIF zur Verfügung. GfA-Basic unterstützt in vorbildlicher Weise GEM und TOS. So können problemlos Alarmboxen, Windows oder Menüleisten programmiert werden. Auch Befehle wie BIOS, XBIOS und GEMDOS, mit denen sich die verschiedenen Betriebssystem-Funktionen direkt nutzen lassen, fehlen nicht.

GfA-Basic fängt nahezu jeden Fehler auf. Bei einem POKE an eine ungerade Adresse erscheinen nun keineswegs die gewohnten 3 Bomben am linken Bildschirm, sondern eine Alarmbox mit dem Text "3 Bomben - Address error. Ungerade Wort-Adresse. Bei Dpoke, Dpeek, Lpoke oder Lpeek?". Nach Anklicken von "O.K." kann weitergearbeitet werden, als sei nichts geschehen. Um den Interpreter zu echten Bomben oder zu einem Absturz zu veranlassen, muß man sich schon echt anstrengen.

Um die enormen Möglichkeiten des GfA-Basic zu verdeutlichen, wollen wir nachfolgend einige Befehle näher vorstellen. ADD var,n: Erhöht den Wert von var um n. Arbeitet doppelt so schnell wie var = var + n. Alert a, m\$, b, b\$, var: Stellt eine Alarmbox mit dem Icon Nummer a, dem Text m\$, dem Default-Knopf b und der Knopfleiste b\$ dar und gibt die Nummer des gedrückten Knopfes in var zurück. ARRAYFILL: Füllt ein Feld mit einem vorgegebenen Wert. C: : Ruft ein Unterprogramm in kompiliertem C auf (mit Parameterübergabe). CHDIR: Wechselt den aktuellen Ordner. DEFFILL: Legt Füllmuster fest bzw. definiert eigene Muster.

```
Save Save.A Quit New |Bk St|Replace Pg up |Text 16|Direct Run
Load Merge LList Block |Bk End| Find |Pg down|Insert Flip Test
Alert 1,"Rechtecke".1,"Return",0
Graphmode 1
Do
  Box 1 Mod 640,1 Mod 400,639-1 Mod 640,399-1 Mod 400
  Add 1,0
Loop
! Dieses Demo ist als "RECT1.BAS" auf der Programmdiskette enthalten!
```

RECT1.BAS geladen und gelistet



Ergebnis von RECT1.BAS

```
Save Save.A Quit New |Bk St|Replace Pg up |Text 16|Direct Run
Load Merge LList Block |Bk End| Find |Pg down|Insert Flip Test
Print "Benchmark 0 : "
T=Timer
K=0
Repeat
  K=K+1
  A=K^2
  B=Log(K)
  C=Sin(K)
Until K>=1000
Print (Timer-T)/200
! Der ST benötigt für diesen Benchmark nur 3.02s
! das "normale" BASIC von DR dagegen 3.5s
! außer Konkurrenz lief der 130XE mit 7min
```

Benchmark-Test

```
Save Save.A Quit New |Bk St|Replace Pg up |Text 16|Direct Run
Load Merge LList Block |Bk End| Find |Pg down|Insert Flip Test
T=Timer
For I=1 To 1000
  Print I
Next I
Print (Timer-T)/200;" s",A
! Für diese Schleife benötigt der ST 34.08s
! Der 130XE benötigt dagegen nur 21s!
! allerdings muß der 130er nur 960 Bytes scrollen, der ST hingegen
! volle 32000 Bytes!
```

Noch ein Benchmark-Test

```
Save Save.A Quit New |Bk St|Replace Pg up |Text 16|Direct Run
Load Merge LList Block |Bk End| Find |Pg down|Insert Flip Test
Print "nc-Benchmark 2001 Divisionen 10/3"
Print "Zeit","Ergebnis"
T=Timer
For I=0 To 2000
  A=10/3
Next I
Print (Timer-T)/200;" s",A
! Diese 2001 Divisionen absolviert der ST in durchschnittlich 0.8s
! Der 130XE benötigt dagegen 25s!
```

Und noch ein Benchmark-Test

DECLINE: Legt den Linientyp fest (z.B. als Pfeil...).

DEFMOUSE: Legt den Mauscursor fest (auch eigene Definitionen).

DFREE: Gibt den freien Speicherplatz der Diskette aus.

DIM?: Bestimmt die Anzahl der Elemente eines Feldes.

DIV: Sorgt ähnlich wie ADD für eine Geschwindigkeitssteigerung bei Division.

DRAW [TO] x,y: Zeichnet eine oder mehrere Linien.

EXIST: Ermittelt, ob ein vorgegebener Filename auf Disk existiert.

FILESELECT: Stellt die Fileselector-Box dar und wartet auf Tastendruck. In der übergebenen Stringvariablen kommt der gewählte Filename zurück.

FOR...NEXT: Fast wie gewohnt. TO zählt aufwärts, DOWNTOW abwärts. So ist FOR N=100 TO 1 STEP -1 und FOR N= 100 DOWNTOW 1 identisch, aber schneller.

FORMINPUT: Nimmt für einen String nur eine vorgegebene Länge entgegen.

GET: Kopiert einen beliebigen Bildschirmabschnitt in einen String.

GOSUB name [Parameterliste]: Ruft eine Prozedur auf, der beliebig Parameter übergeben werden können. Die Variablen einer Prozedur sind lokal.

**IF bedingung
programmblock
[ELSE]
[programmblock]
ENDIF.**

So können im GfA-Basic IF-THEN-Vergleiche aussehen. Da der "programmblock" aus beliebig vielen Zeilen bestehen darf, ist es nicht notwendig, mehrere Befehle in eine Zeile zu packen.

LOF: Ermittelt die Länge einer Datei.

MENU: Ermöglicht die Erstellung und Modifikation von Menüleisten (siehe Demo!).

MKDIR: Legt einen neuen Ordner an.

MOUSE: Ermittelt Mausposition und Tastenstatus.

ON BREAK...: Fängt die BREAK-Tasten ab und reagiert wunschgemäß.

ON MENU GOSUB: Enthält praktisch den GEM-Multi-

Event. Verwaltet Menüaufrufe, Tastenbetätigungen, Mausbewegungen in und aus Rechtecken und Nachrichten (z.B. bei der Arbeit mit Fenstern [Mover, Closer, Slider...]).

POLYLINE: Entspricht der VDI-Polyline-Funktion.

PUT: Gegenstück zu GET.

SETTIME: Stellt Uhrzeit und Datum.

SPOKE: POKE im Supervisor-Mode.

SPRITE: Stellt ein in einem String abgelegtes Sprite dar oder löscht es.

SWAP: Vertauscht den Inhalt zweier Variablen oder ganzer Felder.

TEXT: Stellt an den angegebenen Koordinaten Grafiktext dar. Dabei kann die Länge in Pixeln angegeben werden, auf die der Text dann ausgedehnt wird.

TITELW: Bestimmt den Titel eines Fensters.

VARPTR: Bestimmt die Adresse bzw. Anfangsadresse einer Variablen.

Natürlich sind auch die AES- und VDI-Arrays angelegt und mit Systemvariablen versehen, so daß auch weitergehende GEM-Funktionen programmiert werden können. So ist es mir z.B. schon beim ersten Anlauf gelungen, ein RSC-File einzuladen und verschiedene Formulare darzustellen. Für die Windowverwaltung steht der Befehl WINDTAB zur Verfügung, in den z.B. die Fensterattribute eingetragen werden können.

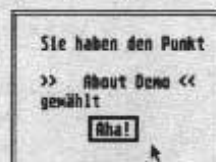
Wer nun glaubt, daß man sich diesen Komfort mit entsprechenden Programmlaufzeiten erkaufen muß, der hat sich gründlich getäuscht. Auch ohne die speziellen Hochgeschwindigkeitsbefehle wie ADD, INC, MUL, DIV, SUB, SWAP usw. ist das GfA-Basic um einige Faktoren schneller als andere Dialekte. Bild 3 zeigt einen Geschwindigkeitstest, in dem 1000 Durchläufe zu absolvieren waren, was den GfA-Interpreter nur 3.02 Sekunden beschäftigte. Dabei wurde auf Spezialbefehle wie INC oder Integervariablen verzichtet. Das ST-Basic benötigte immerhin dreimal so lange. Damit auch der 130 XE etwas zu tun hatte, durf-

DESK EINTRÄGE



Menüs wie im Desktop

DESK EINTRÄGE



Alarmbox, selbst gemacht

DESK EINTRÄGE



So wird das Demoprogramm beendet

```
Save | Save.R | Quit | New | Blk Sta Replace | Pg up | Text 16 | Direct | Run
Load | Merge | List | Block | Blk End | Find | Pg down | Insert | Flip | Test

Do
On Menu
Loop
' Auswertungsprozedur für Menue
'
Procedure Menue
Menu Off
AS="Sie haben den Punkt | >> "+$(Menu(0))+<< "gewählt"
Alert 0,AS,1,"Alia!",1
If $(Menu(0))=" ENDE"
End
Else
If $(Menu(0))=" About Demo"
Alert 1,"Dies ist nur ein Demo für das GfA-BASIC|Wollen Sie weitermachen?"
If T=2
End
Endif
Endif
Endif
Return
Ausschnitt des Demolistings
```

te er es ebenfalls versuchen. Sieben Minuten war seine Zeit (bekanntlich gehört ja die Potenzierung beim ST zu den eher gemütlichen Funktionen). Dabei darf jedoch auch die Genauigkeit, mit der das GfA-Basic arbeitet, nicht unerwähnt bleiben. Während man beim ST-Basic öfters den Eindruck hat, die Ergebnisse kämen aus dem Zufallsgenerator, ist das GfA-Basic sehr exakt. Dies liegt an der internen Speicherung der Variablen, die bei GfA mit 6 Bytes um einiges genauer ist als die 4 Bytes, mit denen das ST-Basic auskommt.

Der Wertebereich läßt beim GfA-Interpreter für Gleitkom-

mazahlen Exponenten bis 154 zu. Integers sind bis ± zwei Milliarden erlaubt. Stringvariablen dürfen bis 32767 Zeichen enthalten – genug also, um z.B. mit der GET-Funktion einen ganzen Bildschirm zwischenzuspeichern. Anders als beim ST-Basic dürfen auch Felder zusammen mehr als 32 KByte belegen. Auf Wunsch stellt das GfA-Basic den gesamten freien Speicher für Arrays zur Verfügung.

Vor allem die sehr langsame Bildschirmausgabe, die sich zudem nicht genau positionieren ließ, haben das ST-Basic in Ungnade fallen lassen. Daran war natürlich die Fenstertechnik

schuld. Das GfA-Basic arbeitet auch hier um einiges schneller. Bild 4 zeigt einen interessanten Vergleich zwischen dem ST und dem 130XE. Es war abzusehen, daß hier der XE schneller ist, was vor allem mit den unterschiedlichen Bytebewegungen beim Scrollen zusammenhängt. Bei einer leeren Schleife (Bild 5) ist der ST wiederum ungeschlagener Sieger.

Sollten Sie vom ST-Basic auf das GfA-Basic umsteigen wollen, so können Sie die alten Basic-Programme an den GfA-Interpreter anpassen. Auf der Programmdiskette befindet sich ein Basic-Demo, das die Programmzeilen entfernt, so daß es mit MERGE eingelesen werden kann. Kleinere Änderungen sind jedoch meist nicht zu vermeiden. In der Praxis hat sich gezeigt, daß es oft effektiver ist, das Programm neu zu schreiben, da oft ganze Zeilenblöcke des ST-Basic durch einen einzigen GfA-Befehl ersetzt werden können.

Sollten Sie das GfA-Basic bereits besitzen oder vielleicht aufgrund dieses Artikels anschaffen wollen, so wird Sie das Demolisting interessieren. Es erzeugt eine Menüleiste und verwaltet diese. Jeder gewählte Eintrag wird in einer Alarmbox wiedergegeben. "ENDE" beendet das Programm, "About Demo" stellt eine erklärende Alarmbox dar.

Außer für den Hobbyprogrammierer ist das GfA-Basic auch für die professionelle Softwareerstellung geeignet, da ein "Run Only Interpreter" zum Lieferumfang gehört. Dieser Interpreter kann, wie der Name schon sagt, ein Programm nur ausführen, nicht aber auflisten oder gar verändern. Der "Run Only Interpreter" darf frei kopiert werden, so daß man seine in GfA-Basic erstellte Software ohne Probleme verkaufen kann, auch wenn der Kunde selbst keine GfA-Basic besitzt!

Auch für uns als Computerzeitschrift ist das GfA-Basic der ideale Programmträger. Programme, die mit dem "alten" ST-Basic geschrieben wurden, waren in der Regel recht lang und absturzgefährdet. Programme in C scheiterten (außer an der Länge) an den vielen unterschiedlichen Compilern, die zudem sehr wenig verbreitet sind. Bei Assemblerprogrammen sieht es ähnlich aus; als Quellcode gibt es mit den unterschiedlichen Assemblern Ärger, und als endlose Basic-DATA-Wüsten mag die Programme niemand abtippen. Gute Programme enthalten in der Regel ein Resource-File, das das Aussehen der Menüleiste und der Formulare enthält. Dieses kann jedoch praktisch nicht abgedruckt werden.

Mit dem GfA-Basic ist nun die Lösung gefunden: Eine Programmiersprache, mit der sich Programme kurz und übersichtlich abdrucken lassen, wobei man auf Komfort wie Menüs und Fenster nicht verzichten braucht. Zwar können wir auf unseren Programmdisketten zum ST, die es vielleicht bald geben wird, auch den "Run Only Interpreter" mit anbieten, aber für eigene Projekte oder Änderungen/Verbesserungen

sollten Sie sich schon den "richtigen" Interpreter kaufen – sicher die beste Investition zum ST!

Für den Herbst ist übrigens ein Compiler von GfA angekündigt, der die Programme noch schneller machen wird. Ich bin sicher, daß man in Zukunft noch einiges von GfA hören wird!

Ab sofort sind Programme in GfA-Basic willkommen! Bedingungen siehe "Bei uns können Sie mitmachen" auf Seite 2.

System: alle ATARI ST
Anleitung: ca. 250 Seiten
DIN A5 (deutsch)
Preis: 149,- DM

Hersteller:
GfA Systemtechnik
Am Hochhofen 108
4000 Düsseldorf 11

Bezugsquelle:
EDV-Schweizer
Benninger Straße 34
8940 Memmingen

Thomas Tausend

Achtung: In der ersten Auflage des GfA-BASIC waren einige Fehler enthalten, die jedoch bereits behoben sind (VARPTR, ARRPTR und PRINT USING zum Beispiel). Wer noch eine alte Version besitzt, der erhält gegen Einsendung der Originaldiskette eine verbesserte Version.

Durch dieses kleine Ärgernis sollte man sich jedoch keinesfalls von diesem Superprogramm abbringen lassen – der Autor Frank Ostrowski hat hier in kurzer Zeit Erstaunliches geleistet. Auch Turbo-Basic und der zugehörige Compiler sind von ihm.

Softwaretest: Cards

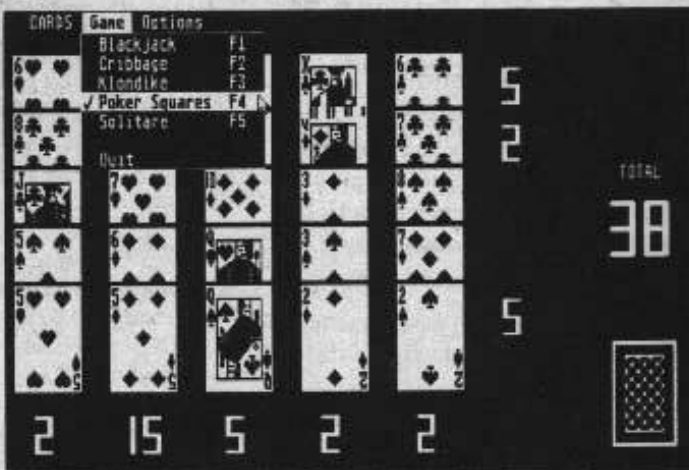
All jene, die ihrer Frau versprochen haben, daß ein Atari ST den regelmäßigen Stammtischbesuch ersetzen kann, brauchen jetzt nicht mehr auf das Kartenspielen zu verzichten. MichTron hat ein weiteres microdeal-Programm auf den Markt gebracht, das 5 (in Großbritannien) bekannte Kartenspiele in sich vereinigt: Blackjack, Cribbage, Klondike, Poker Squares und Solitaire. Bei uns verbreitete Spiele wie Skat oder Schafkopf (regional natürlich unterschiedlich) sucht man freilich vergeblich.

Einzig Blackjack (17 und 4), das bis zu 4 Personen gleichzeitig spielen können, war mir als Nichtspieler auf Anhieb geläufig. Schade nur, daß der Spiel-

verlauf über die Tastatur bestimmt wird. Mit den Zifferntasten bestimmt man zum Beispiel, ob man wetten, halten oder kaufen möchte. Bei "Cards" wurden die Fähigkeiten des GEM nur zum Teil genutzt (Menüleiste und bei einigen Spielen die Positionierung der Karten).

Glücklicherweise läuft "Cards" sowohl mit S/W- als auch mit Farbmonitor, auf dem die ansehnlich gestalteten Karten besonders zur Geltung kommen. Besonders begeistern kann dieses Programm nicht.

System: alle Atari ST
Vertrieb: MichTron (microdeal)
Preis: 19.95 englische Pfund
Thomas Tausend



PADERCOMP – Walter Ladz
Erzbergerstraße 27
4790 Paderborn
Telefon 052 51 / 363 96

Floppystationen für ATARI ST®

PADERCOMP FL 1 498,-
3,5", 1 MB, singeb. Netzteil, NEC-Laufwerk, Abm. 240x105x40 mm, anschlussfertig mit Industrie-Floppystecker, graues Metallgehäuse, ohne Schrauben an den Seiten. Ein Qualitätsprodukt, das schon Hunderte von zufriedenen Kunden gefunden hat.

PADERCOMP FL 2 798,-
Unsere Neue, Doppellaufwerk überlappend, sonst wie FL 1

NEC FD 1036 A 3,5", 1 MB,
32 mm Bauhöhe, Aktionspreis 269,-
dto. ST modifiziert 289,-

Industrie-Floppystecker
jetzt lieferbar nur 9,90
ST-Kabel an Shugart-Bus 3,5" 29,90

Drucker

Panasonic KX-P 1080
100 Z/s, NLO 648,-

Panasonic KX-P 1091
120 Z/s, NLO 768,-

Panasonic KX-P 1092
180 Z/s, NLO 1049,-

OKIDATA ML 192 inkl.
vollautom. Einzelblattzug 1498,-

Citizen 120D der Einsteiger 595,-

STAR NL 10 inkl. Interface 788,-

Zubehör

Eprommer für ST 2764-
27256, Software unter GEM 195,-

3,5" Disketten ab 3,49

Disk-Box SS-50
für 50 3,5" Disketten 24,90

Druckerkabel ST 34,90

Preisliste kostenlos

Alle Angebote solange Vorrat reicht. Bestellungen per Nachnahme oder Vorkasse ab 30 DM. Auslandsbestellungen nur gegen Vorwissen. Ungetragenes Warenschilder: ATARI ST. Die Preise können günstiger liegen. Rufen Sie an! Händleranfragen erwünscht.



TI 99/4A

Liebe TI-Freunde!

Die Urlaubszeit ist nun vorbei, und damit der Einstieg in den grauen Alltag nicht zu schwer fällt, ist gerade rechtzeitig die neue CK fertig geworden. Diesmal habe ich für euch 17 TI-Seiten voller nützlicher Utilities und unterhaltender Spiele zusammengestellt, die nur darauf warten, euren TI zum Leben zu erwecken. Natürlich fehlt auch der von vielen erwartete 2. Teil unseres Superprogrammes Monopoly nicht. Also die restlichen Zeilen ein-

getippt und los geht die Jagd nach Schloßallee und Parkstraße, allein oder zu mehreren gegen euren Blechkameraden.

Immer wieder erhalte ich Briefe, in denen Programmautoren wissen wollen, welche Programme die größten Chancen für einen Abdruck in der CK haben. Darum will ich an dieser Stelle kurz einige Tips geben. Die meisten Programme, die uns erreichen, sind Spielprogramme. Daher muß

ein Spielprogramm, das in der CK erscheinen soll, sich durch eine gute Grafik und hohe Spielgeschwindigkeit aus der Masse hervortun. Der wichtigste Punkt, der anscheinend von vielen übersehen wird, ist die Originalität der Spielidee. Wer uns die 124. Spaceinvader-, Frogger- oder Donkey-Kong-Version schickt, der hat nur geringe Chancen, die Auswahl erfolgreich zu überstehen.

Bei Anwenderprogrammen sind alte Hüte ebenfalls wenig gefragt. Textverarbeitung, Adressverwaltungen und Diskettenkataloge sind mittlerweile schon drangewesen, so daß für Programme mit dieser Thematik nur wenig Hoffnung besteht. Auch sehr spezielle Anwendungen, die nur für wenige Leser interessant sind, scheiden

aus. So mußte ich vor ca. einem Jahr ein Programm zur Futtermitteloptimierung bei Milchvieh zurückweisen, obwohl das Programm an sich recht gut gelungen ist. Populäre Anwendungen haben da wesentlich bessere Chancen, insbesondere dann, wenn das Listing nicht sehr lang ist. Für Anwenderprogramme ist die Wahrscheinlichkeit eines Abdrucks insofern größer, da uns Einsendungen von Utilities und Anwenderprogrammen weitaus seltener erreichen, als dies bei Spielen der Fall ist.

Wer also glaubt, daß sein Programm diesen Kriterien standhält, der sollte mit einer Einsendung nicht länger warten, denn auch hier gilt: Zeit ist Geld!

Euer TI-Spezialist
H. P. Schwaneck

TI-Software-Hitparade

Diesmal war das Echo schon bedeutend besser. Trotzdem können es noch ein paar mehr werden. Zu gewinnen gibt es wieder ein Softwarepaket.

Hier aber nun die neueste TI-Software-Hitparade:

1. TI Artist II	(-) A,AS
2. Bigfoot	(-) S,MO
3. Tennis	(1) S,AS
4. DM 1000	(4) A,AS
5. Jumpy	(5) S,AS
6. Disk-Katalog (CK)	(10) A,AS
7. Guardian	(-) S,AS
8. TI-Runner	(7) S,AS
9. Artillerie	(6) S,BA
10. Return t. P. Island	(9) S,MO

Der Gewinner aus dem letzten Heft:

Norbert Göbel, Bahnhofstraße 14, 3432 Großalmerode

Eure Tips müßt Ihr wieder an folgende Anschrift einschicken.

TI 99er Workshop Rheinland
Dept. Allgemein + Software
c/o Mike Heuser
Karl-Marx-Allee 18
5000 Köln 71

TI-99/4A in Perfektion

Wie in der letzten Ausgabe angekündigt, fand am 1.6.86 in der Wuppertaler Stadthalle der Bergische Computerflohmarkt der Buchhandlung Finke statt. Im Rahmen dieses Flohmarktes veranstaltete der TI 99er Workshop Rheinland zum dritten Mal in diesem Jahr seine Demo-Show "TI-99/4A in Perfektion". Hier nun, für alle die nicht dabei sein konnten, ein Bericht von dieser Veranstaltung.

Zu allererst möchten wir hier nochmals der Buchhandlung Finke in Wuppertal danken, daß sie uns einen geradezu riesig zu nennenden Stand kostenlos zur Verfügung gestellt hat.

Aufgrund der Abmessungen unseres Standes waren wir diesmal in der Lage, vier komplette Systeme mitzubringen. Schon beim Aufbau versammelten sich die ersten Schaulustigen um unseren Stand. Wie überall die Frage: "Was wollt ihr denn mit eurem Oldie hier?" Als dann alles aufgebaut war, kam der erste Hammer. Eines unserer Mitglieder hatte einen "neuen" TI gebaut, den TI-99/4A Omega! Es handelt sich hierbei um ein IBM-PC-Gehäuse mit erstaunlichem Inhalt: ein komplettes CPS-99 System, die total auseinandergenommene Hauptplatine des TI sowie alle Netzteile, der Sprach-Synthesizer und ein Modulexpander.



Return of the TI-99/4A

Torpedo Basic V. 3.0

Super-System-Tool: 32 Maschinenbefehle + 10 Utilities + Special Bonus:

- BRANCH
- CHASET
- CLEAN
- CLTEXT
- COPY
- CRESET
- DELETE
- END
- GET
- GETSEC
- GETSTR
- HIDE
- INTR
- MODE
- MOVE
- MOVEV
- OLDVDP
- OUTSTR
- PEEK
- PUTSEC
- QUIT
- SAVVDP
- SCROLL
- SEARCH
- SWAP
- TABLE
- TAKE
- TORPED
- USECOL
- WINDOW
- WRITE
- Bildschirmorientierter Cursor
- 40 Zeichen/Zeile, Moduswahl
- 32 frei definierbare Windows
- 10 internationale Zeichensätze
- Hardcopy für Epson + Sekosha
- Scrollen in alle Richtungen
- Basic-Zeilenblockweise löschen
- Umfangreiches Window-Handling
- VDP-CPU Speicherblocktausch
- Interruptgest. Bildschirmfarben
- Sprung zu Zeilennummer xy
- Disk-Sektoren lesen/schreiben
- PEEK und POKE im VDP-RAM
- und noch viel, viel mehr...
- **Spezielle Funktionen**
- C-EARSCREEN auf Tastendruck
- VDP-Block löschen/laden
- System-Direktzugriff möglich
- **Utilities:**
- Directory ausgeben, Sektoren editieren und kopieren, zwei Grafikzeichensätze, wirklich echte Kleinbuchstaben etc...
- **Bonus:**
- Handbuch in Deutsch + Englisch mit allen Systemadressen und Tips und Tricks und Demoprogrammen und, und, und...

Diskette DM 129.-

für Ext.-Basic und E/A-Modul!
(Speichererweiterung erforderlich)

Vorblätterung einer Handbuch-Kurzfassung in Deutsch, 104 S., begr. Auflage DM 15.90



PRICE ATTACK!!

TI-Multiplex (Modul)	DM 199.-
TI-Winter (Modul/Disk)	DM 199.-
TI-Logo II (Disk)	DM 199.-
Editor/Assembler (M+D)	DM 129.-
TI-Forth V. 4.5 (Disk)	DM 99.-
Tiny Logo (Cassette)	DM 59.-

*** BITMAC ***

A REVOLUTIONARY new 99/4A GRAPHICS PROGRAM from the USA: Switch single dots on or off in 16 colors, paint circles, lines etc., makes your TI compatible to trackballs, IBM PC, Apple Mac, IBM 370 etc. BITMAC offers boolean Disk input such like NASA enhanced photos and much more!!

Order the Disk only DM 129.-



- **CPS-99**
- **MEGARAM**
- **Peripheral Exp.**
- **Interfaces**
- **Disk-Controller**
- **Floppy Disks and**
- **other Hardware**
- **also TI-Consoles**

*** ACTUALLY ***
PRICES ON REQUEST

SPECIAL OFFER:

- Scott Adam's Airline Strategic game DM 99.-
- 99/4A Flyer Flight Simulator Modul für TI-Console DM 99.-
- GRAPHICS CODE Generator Disk DM 99.-
- Adventure Editor-Disk DM 199.-

MORE INFORMATION
send 2x50 Pf stamps



NEW:
Software
from the
USA

GET IT! New SCOTT ADAMS Adventures!!!

- (Specify CASSETTE or DISK if you order)
- Adventure Modul (necessary) DM 24.90
 - The Hulk DM 89.-
 - Claymorgue Castle DM 89.-
 - Bukaroo Banai DM 89.-
 - Spiderman DM 89.-
 - Adventure 4er-Set + Surprise DM 299.-

INFOCOM Supergames:

- HITCHHIKERS GUIDE TO T. GALAXY
- Planetfall, Witness, Zork I, Outposts, Enchanter s.o. a DM 79.90
- Standard Level (6 disks) only DM 299.-

- ZORK II, Zork III, Sorcerer and Infidels single order a DM 87.50
- Advanced Level (4 disks) only DM 279.-

- DEADLINE, Starcross and Suspended single order a DM 99.-
- Expert Level (3 disks) only DM 259.-

- TI-ARTIST (Disk) DM 95.-

SUPPORT:

- TI-Artist Companion #1: 5 Exports - Disk, single order a DM 129.-
- TI-Artist Companion #2: 2 Exports - Disk, single order a DM 99.90
- Artist's Extras Fonts, pictures, asc. for T. TI-ARTIST, Display Master (Utility) DM 79.90
- Windows, colors etc. for the TI-ARTIST.

CHARACTER SETS & GRAPHIC DESIGN (CSGD):

CSGD ermöglicht den kinderleichten Entwurf von unzähligen Schriftarten für Grußkarten, Briefköpfe etc. sowie Grafikschöner jeder Art. Die User-Disk enthält schon fertige Schriften, z.B. Block, Shelly, Fareast, IBM-type etc. (Angaben: Fonts/Pictures/Charactersets a Disk)

- CSGD I Font & sign creator DM 95.-
- CSGD II Bannermaker - 8/56 DM 99.-
- CSGD User Disk #1 7/1/3 DM 34.90
- CSGD User Disk #2 12/26/88 DM 99.90
- CSGD User Disk #3 12/27/127 DM 99.90
- Complete CSGD Set (5 Disks) DM 249.-



Rausch & Haub

Vertriebsgesellschaft dbR
Postfach 32 03 13
5300 BONN 3



02 28 / 43 10 59 ORDER ONLY
02 28 / 33 38 64 ORDER AND SERVICE

ORDER: DM 50.- Mindestwert + 4.50 (Vorkasse), 6.00 Nachnahme, Ausland 8.-/16.-. HARDWARE liefern wir ab DM 1000.-, verpackungs-, versandkostenfrei!

Mo.-Fr. 11.00 am. 10.400 pph. (11.00-18.00 Uhr)

Alle Schnittstellen sind nach außen geführt, so daß der gesamte Kabelsalat entfällt. Alles dachte natürlich, hier wäre ein IBM-PC mit von der Partie. Dementsprechend groß war das Erstaunen, als das Gerät eingeschaltet wurde und sich mit dem TI-Standard-Bildschirm meldete.

Eine weitere Sensation waren einige grafische Kabinettstückchen, die mit Hilfe des TI-Artist erstellt wurden. Der WDR Computerclub bietet in seiner KOMCOM-Mailbox verschiedene Bilder im ASCII-Format sowie ein Programm an, mit dem man diese Bilder auf dem TI sichtbar machen kann. Nun hatten sich einige User hingestellt, um diese Bilder Artist-kompatibel zu machen, damit auch Hardcopies möglich sind. Diese Hardcopies, auf einem Epson-Drucker erstellt, gingen weg wie warme Semmeln, zumal sie umsonst waren. Ebenfalls sehr gefragt waren verschiedene Exemplare der CK, die uns der Verlag freundlicherweise zur Verfügung gestellt hatte.

Nun aber zu einem sehr wichtigen Ereignis innerhalb dieser

Show. Da wir dieses Mal sehr früh über den Termin Bescheid wußten, hatten wir einige deutsche Clubs eingeladen, uns in Wuppertal zu besuchen. Die meisten davon kamen dann auch nach Wuppertal. So waren Leute vom TI-Club-Baunatal, TICOM e.V. Münster, vom TI-Club-Wiesbaden und auch Mitglieder der TI-Clubs aus Hannover und Mannheim da. Andere Clubs, die leider keine Zeit hatten, ließen uns die besten Grüße ausrichten, so z.B. die TI-99/4A User Group aus Bremen und der TI-99/4A User Club Berlin. Da nun endlich einmal eine ganze Reihe organisierter TI-User beisammen waren, gab es natürlich eine ganze Menge Gespräche über die Zukunft des TI. Als wichtigstes Ergebnis dieser Gespräche kann man die Tatsache nennen, daß alle anwesenden Club-Mitglieder sich dafür aussprachen, in nicht allzu ferner Zukunft ein nationales TI-Treffen mit Vorträgen, einer Computerbörse nur für TI und einige andere Attraktionen zu veranstalten. Als mögliche Veranstaltungsorte wurden Köln, Bremen oder Frankfurt genannt. Aber auch Berlin könnte ein möglicher

Veranstaltungsort sein. Darüber hinaus wurde über die Möglichkeit diskutiert, einen deutschen Dachverband aller TI-Clubs ins Leben zu rufen. Es gab zu diesem Thema allerdings sehr kontroverse Meinungen, so daß darüber das letzte Wort noch nicht gesprochen ist.

Bezüglich der TI-Veranstaltung haben wir auf jeden Fall schon einmal Verhandlungen mit der Stadt Köln aufgenommen, die dieser Aktion auch einigermaßen aufgeschlossen gegenübersteht. Als nächsten Schritt werden wir mit den verschiedenen Anbietern von Produkten für den TI Kontakt aufnehmen, um auch diese Leute zu einer Teilnahme zu bewegen. Wir sind auf jeden Fall guten Mutes, daß das Treffen noch in diesem Jahr stattfinden kann.

Doch nun wieder zu unserer Show. Der anfänglichen Überheblichkeit einiger User anderer Systeme folgte großes Erstaunen, als diverse Programme vorgeführt wurden. Sehr große Aufmerksamkeit erregten vor allem die Musik-Demos von Roman Majer und auch die Möglichkeiten des Extended Basic II Plus.

Sozusagen als Krönung wurde dann die TI-99er-Workshop-Rheinland-Mailbox auf dem TI gestartet. Aber wie meistens bei einer Premiere: Was bei der Generalprobe noch so wunderbar geklappt hatte, funktionierte zuerst mal überhaupt nicht. Aber nach einigen Problemen



ging dann doch die Post ab und sehr viele User wollten einen Blick in diese Box werfen.

Das Ende der Veranstaltung kam dann viel zu schnell, und man hatte gerade noch Zeit, sich voneinander zu verabschieden und eine gute Heimfahrt zu wünschen. Alles in allem kann man auch diese Veranstaltung als einen Erfolg nicht nur für uns, sondern für den TI insgesamt werten.

TI 99er Workshop Rheinland
Mike Heuser

TMS 9900 Assembler auf dem TI 99/4A

Beim heutigen Buchtest bin ich in der beneidenswerten Lage, Euch ein echtes Juwel unter der 99er Literatur vorzustellen. Der Autor, Simon Koppelman, den alten Hasen unter den TI-Besitzern sicherlich bekannt, hat unter dem Titel "TMS 9900 Assembler auf dem TI 99/4A" seine langjährigen Erfahrungen auf dem Gebiet der Assemblerprogrammierung auf 270 Seiten zu Papier gebracht. Dabei ist kein Lehrbuch zur Assemblerprogrammierung entstanden, sondern eher eine systematische Ausammlung von Hintergrundinformationen, die für den Assemblerprogrammierer unentbehrlich sein dürften. Selbst das sehr ausführliche Handbuch zum Editor/Assembler-Modul kann hier nicht mithalten. Informationen, die der erfahrende Assemblerprogrammierer zuvor nur durch Umwege, viele Versuche und langes Forschen erhalten konnte, sind hier fein säuberlich dokumentiert.

Das Buch beginnt mit einem Überblick über den Aufbau des Assemblercodes; es folgen Adressierungsarten, Befehlscode und die Befehlsformate des TMS 9900. Daran schließt sich eine Auflistung aller Befehle an. Zu jedem Befehl gibt es eine verständliche Funktionserklärung sowie ein oder mehrere Anwendungsbeispiele. Das nächste Kapitel behandelt die Steuerkommandos für den Assembler. Auch hier werden in verständlicher Weise z.T. schwierige Zusammenhänge erläutert.

Bis zu dieser Stelle ist das Buch nur eine gute Übersetzung des Editor/Assembler-Handbuches; die nun folgenden Kapitel machen jedoch den eigentlichen Wert des Buches aus. Nun geht der Autor daran, Themen anzusprechen, die lange nur fortgeschrittenen TI-Freaks bekannt waren. Beginnend mit dem Aufbau der CRU-Schnittstelle werden nun Dinge wie die Device-Service-Routinen (DSR) und deren Aufruf über PABS, der VDP-Prozessor und die Aufteilung

des VDP-RAMs in den unterschiedlichen Grafikmodi, GROMs und deren Funktion in Zusammenhang mit der Zwischensprache GPL sowie die Programmierung von Sound und Sprache erläutert.

Selbstverständlich hat der Autor auch die im RAM stationären Utility-Programme unter die Lupe genommen. Neben der Erklärung der wichtigsten Programme wurde der gesamte RAM-Bereich dieser Utilities disassembliert und sehr ausführlich dokumentiert. Der Anhang, der sich bei den meisten Büchern dieser Art auf ASCII-Tabellen beschränkt, kann hier



ebenfalls mit wertvollen Informationen aufwarten. Neben den verschiedensten Tips sind zusätzlich einige Programm listings enthalten, welche die Möglichkeiten des TI 99/4A demonstrieren. Wer sich die Arbeit des Abtippens ersparen möchte, der kann zum Preis von 20,- DM eine Diskette mit den Programmen dazukaufen.

Das vorliegende Buch ist kein Assemblerkurs, sondern eher ein Nachschlage- und Begleitwerk mit einer Vielzahl an nützlichen Informationen für den Assemblerprogrammierer. Es ist seinen Preis von 49,- DM wert und sollte eigentlich in keinem Bücherschrank fehlen.

H. P. Schwaneck

Das Buch kann über den Verlag bezogen werden. Bestellschein siehe Buchversand.

Der Zwillingenbruder des TI-Writers

Der TI-Writer ist trotz vieler ähnlicher Produkte das meistbenutzte Textverarbeitungsprogramm für den TI. Sicherlich würde es noch sehr viel mehr TI-Besitzer geben, die dieses Programm benutzen würden, wenn da nicht der Preis von satten 250 DM wäre, der für den Heimanwender sicherlich noch um einige Nummern zu hoch angesetzt ist. Mit diesen Überlegungen kämpften auch Vater und Sohn der Familie Brunsmann, bis der Entschluß gefaßt wurde, ein Programm zu schreiben, das dem Original in Leistung und Bedienung so nahe wie möglich kommen sollte. Vor wenigen Wochen wurde dieses Programm vollendet, und eines der ersten Exemplare wurde der CK zum Test überlassen.

Das Programm wurde vollständig in Assembler programmiert und läuft unter Ext. Basic mit der 32K-Speichererweiterung, einer Diskettenstation und einer Schnittstellenkarte. Nach Beenden des Ladevorganges, der etwas länger dauert als beim Writer, erscheint auf dem Bildschirm das vom TI-Writer bekannte Schreibfeld, auf dem 40 Zeichen pro Zeile untergebracht werden können. Auch das seitliche Scrollen bei Erreichen der rechten Bildschirmgrenze wurde verwirklicht, so daß man insgesamt 80 Zeichen pro Zeile unterbringen kann. Texte können wahlweise unter dem Festmodus oder unter der Word-Wrap-Funktion erstellt werden. Beim Festmodus muß der Anwender selbst darauf achten, daß die Zeilenlänge 80 Zeichen nicht überschreitet, während unter der Word-Wrap-Option das Wort, das nicht mehr in die aktuelle Zeile paßt, automatisch mit in die nächste Zeile transportiert wird, ohne daß der Schreiber sich um das Zeilenende kümmern muß. Durch Betätigen der FCTN-9-Taste erreicht man den Kommando-Modus, aus dessen Menüliste sich Sonderfunktionen aufrufen lassen. Auch in diesem Punkt steht das hausgemachte Programm dem

TI-Writer in nur wenigen Dingen nach. Beispielsweise kann man zwar wie beim Writer Tabulatorpositionen setzen, der rechte und linke Bildschirmrand lassen sich jedoch nicht beeinflussen. Im Editier-Modus müssen die Absatzzeichen durch eine Funktionstaste erzeugt werden, während der Writer dies automatisch bei Betätigen der ENTER-Taste tut. Bis auf solche Kleinigkeiten ist der Editor aber genauso gut wie der des Writers.

Um Texte formatiert auszu-drucken, benötigt man einen separaten Programmteil, den sogenannten Formatter. Dieser Formatter druckt den Text – falls gewünscht – in Blocksatz, stellt Seitenlänge und Zeilenabstand ein, ermöglicht den Druck von Kopf- und Fußzeilen, numeriert die Seiten und sendet Steuercodes an den Drucker. Serienbriefe werden zwar nicht unterstützt, man kann sich aber mit der Eingabeoption behelfen, die den Druck an vorher bestimmten Stellen unterbricht. So kann ein 80 Zeichen langer Text direkt in das aktuelle Schriftstück eingegeben werden.

Dieses hausgemachte Textprogramm macht einen sehr brauchbaren Eindruck, kann sich mit teureren Programmen durchaus messen und ist für die Bedürfnisse eines Homecomputerbesitzers mehr als ausreichend. Das mitgelieferte Handbuch ist jedoch bei weitem nicht so ausführlich wie das Handbuch des Writers, reicht aber aus, um das Textprogramm in den Griff zu bekommen. Mit einem Preis von ca. 60 DM ist dieses Programm sehr preisgünstig und kann all denen, die noch kein gutes Textprogramm besitzen, empfohlen werden.

Bezugsquelle:

J. Brunsmann

Gigastraße 3

4400 Münster

H.-P. Schwaneck

**Computer-Kontakt
preisgünstige
Kleinanzeigen**



TI Special



Tuning: 128K RAM plus Centronics-Interface

Stabiles Metallgehäuse, Versorgung über Stecker-Netzteil, durchgeführter Expansionsbus 4K ROM gesockelt, 32K RAM direkt eingebündelt, ermöglicht TI-Writer, Ext. Basic etc. RAMdisk, Centronics Druckeranschluß, leicht verständliches Handbuch.

Best.-Nr. TI 1 400.- DM



Komfort: TI-Maus

Maus mit Anschlußkabel, 5-V-Stecker-Netzteil, Software auf 5.25" Disk.

Best.-Nr. TI 2 296.- DM



Lehrreich: TI Assembler Kurs II

Eine Einführung in die Assemblerprogrammierung mit dem TI 99/4A.

300 Seiten inkl. Diskette.

Best.-Nr. TI 6 80.- DM



Lang erwartet: TI-Assembler-Kurs III

Die logische Fortsetzung zum TI-Assembler-Kurs II. Bietet noch mehr Einblick in das Innenleben Ihres TI.

Best.-Nr. TI 4 79.90 DM



Zeitsparend: Turbo-Assembler

Doppelt so schnell wie Editor/Assembler, weniger Platzbedarf auf Disk, schnelleres Laden von Disk.

Anforderungen: Konsole + Ext. Basic, Cassettenrecorder, 32K RAM Erweiterung

Best.-Nr. TI 5 99.- DM

Nützlich: TMS 9900 Assembler

DAS Nachschlagewerk für TI-Assembler-Programmierer. Der Prozessor, seine Befehle, Beispiellists, Speicheraufbau, Tabellen, Index...

Anforderungen: Editor/Assembler, Extended Basic

Best.-Nr. TI 3 49.- DM



CK SPEZIALITÄTEN-BESTELLSCHIN



Anzahl	Best.Nr.	Artikel	Preis incl. MwSt.
	TI 1	128K RAM plus Centronics Interface	
	TI 2	TI-Maus	
	TI 3	TI 9900 Assembler	
	TI 4	COPYA	
	TI 5	Turbo-Assembler	
	TI 6	TI Assembler Kurs II	

Name des Bestellers

Ich wünsche folgende Bezahlung:

Anschrift

☐ Nachnahme (zuz. 5,70 DM Versandkosten)

PLZ/Ort

☐ Vorauskasse (keine Versandkosten)

Datum/Unterschrift

Bei Vorauskasse bitte Scheck beilegen oder auf Postscheckkonto Karlsruhe 43423-750 überweisen

Coupon ausschneiden, auf Postkarte kleben und einsenden an:
Verlag Rätz-Eberle/CK-Software, Postfach 1640, 7516 Bretten

Def-Maker

Benötigt wird eine Konsole, Extended-Basic, eine 32K-Speichererweiterung sowie ein Cassettenrecorder.

Wenn in einem Programm mehrere Funktionen berechnet oder dargestellt werden sollten, mußte bislang nach herkömmlicher Methode beim Wechsel der Funktion das Programm zunächst unterbrochen werden. Danach wurde die neue Funktion in eine bestimmte Zeile eingegeben und das Programm wiederholt gestartet. Das Unterprogramm "Def-Maker" macht ab sofort die Unterbrechung des Hauptprogramms vor Eingabe jeder neuen Funktion überflüssig. Man kann nun die Funktion über beliebige Eingabemethoden in das Programm eingeben, z.B.: INPUT, ACCEPT AT usw. Dem Unterprogramm muß nunmehr nur noch mitgeteilt werden, in welcher Zeile die Funktion später erscheinen soll. Sobald sich das Unterprogramm zurückmeldet, kann im Hauptprogramm mit der neuen Funktion gearbeitet werden.

Der Vorteil dieser Methode liegt auf der Hand:

1. Das Hauptprogramm muß beim Wechsel der Funktionen nicht unterbrochen werden.
2. Man spart sich das lästige Neuaufrollen von Untermenüs, soweit diese vorhanden sind.

Zu dem Unterprogramm, das man mit CALL DEF (DEF\$, Zeile, Fehlerflag) aufruft, ist folgendes zu sagen: DEF\$ enthält den String mit der Funktion, "Zeile" enthält die Nummer der präparierten Zeile und das Fehlerflag wird vom Programm selbst gesetzt. Doch dazu später mehr. Zunächst muß eine wichtige Bedingung erfüllt sein. Die Zeile, in die später die Funktion eingesetzt werden soll, muß nämlich folgendermaßen aussehen:

```
DEF F(X)=X :: REM
AAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA
```

```
AAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA
```

Im Klartext: In der Zeile muß eine Funktion "X" stehen, die zwar einen beliebigen Namen haben kann, der aber nur einen Buchstaben lang sein darf. Zudem muß hinter dieser Funktion ein REM mit genau 134 Zeichen stehen, da sonst die Länge der Zeile nicht stimmt. In diesem Fall stürzt der Computer ab. Da das Unterprogramm den Funktionsnamen nicht verändert, braucht man nur eine solche Zeile ins Programm einzubauen und kann dann die Funktion in die Berechnungen einbeziehen. Mehrere solcher Zeilen mit verschiedenen Funktionsnamen sind natürlich auch möglich.

Das Unterprogramm erkennt folgende mathematische Operationen innerhalb einer Funktion: COS, SIN, TAN, SQR, ABS, ATN, EXP, INT, LOG, SGN; den Ausdruck PI; die Rechenoperationen "+", "-", "*" und "/"; die beiden Klammerzeichen, den Punkt und schließlich alle Zahlen.

Im allgemeinen hat das Fehlerflag den Wert Null. Sollte jedoch im Unterprogramm ein Fehler auftreten, wird es je nach Fehlerart mit einem bestimmten Wert besetzt. Hierbei gibt es drei Möglichkeiten:

- Wert=1 Im String ist ein Ausdruck, den das Unterprogramm nicht kennt.
- Wert=-1 Der String ist zu lang und paßt im interpretierten Zustand nicht in die vorgesehene Zeile.
- Wert=-2 Die Zeile, in die der String eingesetzt werden soll, existiert nicht.

Sollte einer dieser vorgenannten Fehlerquellen im Unterprogramm auftreten, so springt dieses zum Hauptprogramm zurück. Dadurch wird im Hauptprogramm allerdings nichts verändert. Hier kann leicht durch Prüfen des Fehlerflags festgestellt werden, ob Fehler vorliegen und um welche es sich handelt. Diese können dann mit den entsprechenden Verweisen beantwortet werden.

Kai Ratherling

Def-Maker

```
100 ! DEF - MAKER ! COPYRIGHT BY ! KAI RATERING
110 !GRUNDKONFIGURATION : !TI 99/4A + !32K EXPANSION +
!EXTENDED BASIC +
120 !AUFRUF DES UNTER- !PROGRAMMS MIT : CALL DEF(DEF$,ZEILE,ERR
DR)
130 !ZEILE IN DIE "GEPOKED" !WERDEN SOLL MUSS !FOLGENDERMASSEN AUS
- !SEHEN :
140 DEF F(X)=X :: REM AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
150 !FUNKTIONSNAMEN FREI !WAHLBAR(NUR EIN BUCH- !STABE LANG!)
32000 SUB DEF(F$,ZE,F)
32010 F=0 :: G$="" :: L=LEN(F$):: Z=1 :: IF L=0 THEN SUBEXIT
32020 IF Z>L THEN M$=SEG$(F$,Z,1):: A=ASC(M$)
32030 H$="" :: IF M$="+" THEN H$=CHR$(193)ELSE IF M$="-" THEN H$=CHR$(194)ELSE I
F M$="*" THEN H$=CHR$(195)ELSE IF M$="/" THEN H$=CHR$(196)
32040 IF M$="^" THEN H$=CHR$(197)ELSE IF M$="(" THEN H$=CHR$(183)ELSE IF M$=")"
THEN H$=CHR$(182)
32050 IF H$<>"" THEN G$=G$&H$ ELSE 32070
32060 Z=Z+1 :: GOTO 32020
32070 IF (A>48 AND A<57)THEN 32150
32080 IF M$="X" THEN 32130
32090 M$=SEG$(F$,Z,2)
32100 H$="" :: IF M$="PI" THEN 32140 ELSE IF M$="CD" THEN H$=CHR$(205)ELSE IF M$
="SI" THEN H$=CHR$(210)ELSE IF M$="TA" THEN H$=CHR$(212)ELSE IF M$="SQ" THEN H$=
CHR$(211)
32110 IF M$="AB" THEN H$=CHR$(203)ELSE IF M$="AT" THEN H$=CHR$(204)ELSE IF M$="E
X" THEN H$=CHR$(206)ELSE IF M$="IN" THEN H$=CHR$(207)ELSE IF M$="LO" THEN H$=CHR
$(208)ELSE IF M$="SG" THEN H$=CHR$(209)
32120 IF H$="" THEN F=1 :: SUBEXIT ELSE Z=Z+3 :: G$=G$&H$ :: GOTO 32020
```

```

32130 G$=G$&"X" :: Z=Z+1 :: GOTO 32020
32140 G$=G$&CHR$(221):: Z=Z+2 :: GOTO 32020
32150 Q$="" :: XX$=CHR$(200):: S1=0
32160 IF Z>L THEN 32180 :: M$=SEG$(F$,Z,1):: A=ASC(M$)
32170 IF (A>=48 AND A<=57)OR A=46 THEN Q$=Q$&CHR$(A):: Z=Z+1 :: S1=S1+1 :: GOTO
32160 ELSE 32180
32180 L$=CHR$(S1):: G$=G$&XX$&L$&Q$ :: IF Z>L THEN 32190 ELSE 32030
32190 CALL PEEK(-31952,J,I,K,P):: J=J*256+I-65536 :: K=K*256+P-65536
32200 CALL PEEK(J,I,O):: IF I*256+O<>ZE THEN J=J+4 :: IF J>K THEN F=-2 :: SUBEXI
T ELSE 32200
32210 J=J+2 :: CALL PEEK(J,I,O):: J=I*256+O-65536-1
32220 D$=G$&CHR$(130)&CHR$(154)&" "
32230 Y=LEN(D$):: IF Y>138 THEN F=-1 :: SUBEXIT ELSE Z=139-(Y+1):: D$=D$&RPT$("@
",Z)&CHR$(0)
32240 FOR X=1 TO LEN(D$):: I=ASC(SEG$(D$,X,1)): CALL LOAD(J+X+6,I):: NEXT X ::
SUBEND

```

Buchstabenmix

Sinn dieses Spieles ist es, Worte in ein 15x15-Spielfeld zu legen, um dadurch Punkte zu erlangen. Es darf jeweils nur ein Wort (eine Richtung) an bestehende Buchstaben gelegt werden, woraus mehrere Worte entstehen können, die aber in jeder Richtung gelesen einen Sinn ergeben müssen. Jeder Buchstabe (es sind 119 vorhanden) hat eine gewisse Wertigkeit. Als Punkte darf man sich die Summe der Buchstaben der neu entstandenen Worte gutschreiben, unter Berücksichtigung der zahlreichen Bonusfelder, die den Wert erhöhen. Ist ein neu gelegter Buchstabe an der Entstehung mehrerer Worte beteiligt, wird er entspre-

chend oft gewertet. Können alle 8 Buchstaben angelegt werden, wird ein Bonus von 50 Punkten gutgeschrieben. Eine umfassendere Spielanleitung sowie die Buchstabenverteilung und Wertigkeiten sind im Erklärungsteil des Programmes vorhanden.

Nun zur Funktionsweise des Spieles. Das Programm stellt ein Spielbrett zur Verfügung, auf dem 1-4 Spieler (1 Spieler nur als Übung) ihre Worte legen. Das gelegte Wort wird auf seine formale Richtigkeit überprüft. (Durfte hier angelegt werden?) Über den Sinn des Wortes müssen allerdings die Mitspieler entscheiden. Wird

das Wort akzeptiert, werden die Punkte automatisch ausgerechnet und zum Punktestand addiert. Es wird dann solange gespielt, bis keiner mehr legen kann. Am Ende werden die übriggebliebenen Buchstabenwerte abgezogen.

Die gesamte Eingabe erfolgt ausschließlich über Joystick 1. Dabei gilt: Annahme der gewünschten Funktion durch den Feuerknopf. Nach Spielbeginn stehen folgende Funktionen zur Verfügung:

ENDE: Beendet das Spiel und geht in den Direktmodus über.

NEIN: Nimmt den letzten gelegten Buchstaben zurück. Sollen mehrere zurückgenommen werden, ist diese Funktion so oft wie nötig anzuwählen.

TAUSCHEN: Alle Buchstaben des Spielers werden in den "Topf" geworfen und dieselbe Anzahl neu gezogen.

LEGEN: Legt einen Buchstaben ins Spielfeld. Zuerst wird der Buchstabe aus den maximal acht des Spielers ausgewählt und dann mittels des blinkenden Cursors im Spielfeld platziert. Eine Vorüberprüfung findet statt.

O.K.: Nimmt den Zug an und überprüft die gelegten Buchstaben. Bei Richtigkeit wird die Punktzahl ausgerechnet und zum Punktestand addiert. Falls vorhanden, werden neue Buchstaben gezogen. Ein Nichtlegen wird auch akzeptiert.

Peter Ulbrich

```

100 OPTION BASE 1
110 DIM A$(119),A(30),LS(8,2),LAST(8),LASTPOS(8,2),LASTCHR(8),POI(4)
120 CALL CLEAR :: CALL SCREEN(4):: CALL DELSPRITE(ALL):: CALL MAGNIFY(1)
130 FOR I=2 TO 9 :: CALL COLOR(I,2,15):: NEXT I
140 CALL COLOR(10,2,7,11,2,10,12,2,8,13,2,5)
150 CALL CHAR(94,"",136,"181818FFFF181818")
160 CALL WRITE("BUCHSTABENMIX",10,7,"W")
170 CALL WRITE("TEXAS",10,12,"S")
180 CALL WRITE("INSTRUMENTS",2,16,"S")
190 CALL WRITE("XBASIC",10,19,"S")
200 CALL WRITE("(C)1986^PETER^ULBRICH",24,3,"W")
210 CALL CHARPAT(35,B$):: CALL CHAR(97,B$)
220 CALL CHAR(91,"00440038447C4444")
230 CALL CHAR(92,"0044007C4444447C")
240 CALL CHAR(93,"0044004444444438")
250 C$="FF818181818181FF"
260 FOR I=96 TO 128 STEP 8 :: CALL CHAR(I,C$):: NEXT I
270 CALL SETFIELD(A$(),A())
280 RO=153 :: CALL SPRITE(#1,136,2,RO,50)
290 DISPLAY AT(20,9):">> START"
300 DISPLAY AT(22,9):">> ANLEITUNG"
310 CALL COLOR(#1,2)

```



```

320 CALL JOYST(1,X,Y)
330 IF Y=4 THEN RO=MAX(153,RO-16)
340 IF Y=-4 THEN RO=MIN(169,RO+16)
350 CALL LOCATE(#1,RO,50)
360 CALL COLOR(#1,1):: CALL KEY(1,K,S)
370 IF K<>18 THEN 310
380 IF RO=153 THEN 590
390 CALL SCREEN(15)
400 DISPLAY AT(1,1)ERASE ALL:"SINN DES SPIELES IST ES,    MÖGLICHST VIELE WÄRTER
   IN    DAS 15X15 SPIELBRETT ZU    LEGEN."
410 DISPLAY AT(6,1):"MAN DARF NUR AN BESTEHENDE WÄRTER ANLEGEN, BEGINNEND MIT
   DEM STERN."
420 DISPLAY AT(10,1):"DIE WÄRTER MÜSSEN IN ALLEN RICHTUNGEN GELESEN EINEN SI
   NN ERGEBEN!"
430 DISPLAY AT(14,1):"DIE ROTEN FÄLDER ( 2W, 3W ) BEDEUTEN ERHÄHTEN WÖRTWERT, DI
   E BLÄUEN ( 2B, 3B ) ERHÄH-TEN BUCHSTABENWERT."
440 DISPLAY AT(19,1):"ES DARF NUR EIN WÖRT GELEGT WERDEN, ABER ES KÖNNEN    ME
   HRERE ENTSTEHEN, DIE ENT- SPRECHEND GEWERTET WERDEN."
450 CALL TÄSTER
460 DISPLAY AT(1,1)ERASE ALL:"ES STEHEN 119 BUCHSTABEN ZURVERTEILUNG, VON DENEN
   JEDER SPIELER (MAX.4) SOLANGE 8"
470 DISPLAY AT(4,1):"STÜCK ERHÄLT, BIS KEINE MEHRVORHANDEN SIND."
480 DISPLAY AT(7,1):"KÄNN EIN SPIELER NICHT ZIE- HEN, DARF ER ALLE SEINE    BUC
   HSTABEN WECHSELN UND    SETZT AUS."
490 DISPLAY AT(12,1):"SIEGER DES SPIELES IST NÄ- TJRLICH DER, DER DIE MEISTENPU
   NKTE GESÄMMELT HÄT.": : "VIEL SPASS..."
500 CALL TÄSTER
510 DISPLAY AT(1,1)ERASE ALL:"BUCHSTABENVERTEILUNG:": "=====
520 RESTORE 1940
530 FOR I=0 TO 14
540 READ B$,S,W1,C$,Z,W2
550 DISPLAY AT(4+I,1):USING B$&": ## ST # P    "&C$&": ## ST # P":S,W1,Z,W2
560 NEXT I
570 DISPLAY AT(21,1):"DER JOKER (BLANK) IST PLÄTZ-HÄLTER FÜR JEDEN BUCHSTABEN!"
580 CALL TÄSTER
590 CALL SCREEN(15):: DISPLAY AT(3,1)ERASE ALL:"WIEVIELE SPIELER WÖLLEN MIT-SPIE
   LEN?"
600 FOR I=1 TO 4 :: DISPLAY AT(5+2*I,3):I;"SPIELER" :: NEXT I
610 RO=97 :: CALL LOCATE(#1,97,17)
620 CALL COLOR(#1,2)
630 CALL JOYST(1,X,Y)
640 IF Y=4 THEN RO=MAX(49,RO-16)
650 IF Y=-4 THEN RO=MIN(97,RO+16)
660 CALL LOCATE(#1,RO,17)
670 CALL COLOR(#1,1)
680 CALL KEY(1,K,S)
690 IF K<>18 THEN 620
700 PL=(RO-1)/16-2
710 CALL CLEAR :: CALL SCREEN(4)
720 B$=RPT$(" ",15)
730 FOR I=1 TO 15 :: DISPLAY AT(I,1):B$ :: NEXT I
740 DATA 1,1,1,8,1,15,8,1,8,15,15,1,15,8,15,15
750 DATA 2,2,2,14,3,3,3,13,4,4,4,12,5,5,5,11,11,5,11,11,12,4,12,12,13,3,13,13,14
   ,2,14,14
760 DATA 1,4,1,12,3,7,3,9,4,1,4,8,4,15,7,3,7,7,7,9,7,13,8,4,8,12,9,3,9,7,9,9,1
   3
770 DATA 12,1,12,8,12,15,13,7,13,9,15,4,15,12
780 DATA 2,6,2,10,6,2,6,6,6,10,6,14,10,2,10,6,10,10,10,14,14,6,14,10
790 RESTORE 740
800 FOR I=1 TO 8 :: READ X,Y :: CALL HCHAR(X,Y+2,104):: NEXT I
810 FOR I=1 TO 16 :: READ X,Y :: CALL HCHAR(X,Y+2,112):: NEXT I
820 FOR I=1 TO 24 :: READ X,Y :: CALL HCHAR(X,Y+2,120)
830 NEXT I
840 FOR I=1 TO 12 :: READ X,Y :: CALL HCHAR(X,Y+2,128):: NEXT I
850 DISPLAY AT(8,8)SIZE(1):"X"
860 DISPLAY AT(1,18):CHR$(104);"=3*WÖRT"
870 DISPLAY AT(2,18):CHR$(112);"=2*WÖRT"
880 DISPLAY AT(3,18):CHR$(120);"=2*BUCHST"

```

```

890 DISPLAY AT(4,18):CHR$(128);"=3*BUCHST"
900 DISPLAY AT(7,18):"=====
910 DISPLAY AT(6,18):"FUNKTIONEN:"
920 DISPLAY AT(9,18):"ENDE !!!"
930 DISPLAY AT(10,18):"NEIN !!!"
940 DISPLAY AT(11,18):"TAUSCHEN"
950 DISPLAY AT(12,18):"LEGEN"
960 DISPLAY AT(13,18):"O.K."
970 N=119
980 FOR I=1 TO PL
990 FOR J=1 TO B
1000 CALL ZIEHEN(A$( ),N,B$)
1010 DISPLAY AT(17-4*(I>2),1-14*(INT(I/2)=I/2)+J-1):B$
1020 DISPLAY AT(18-4*(I>2),1-14*(INT(I/2)=I/2)+J-1):STR$(A(ASC(B$)-64))
1030 CALL SOUND(50,392,7)
1040 NEXT J
1050 DISPLAY AT(19-4*(I>2),1-14*(INT(I/2)=I/2)): "a";STR$(I);":^^^OP"
1060 NEXT I
1070 CALL CHAR(99,"000402FFFF020400",100,"FFFFFFFFFFFFFFFF")
1080 P=1
1090 CALL SPRITE(#2,99,5,145-32*(P>2),B-112*(P=2 OR P=4))
1100 CALL MENU(C)
1110 ON C GOTO 1780,1720,1570,1120,1450
1120 SL=8-112*(P=2 OR P=4):: SC,SH=SL+64
1130 RD=129-32*(P>2):: CALL LOCATE(#1,RD,SH):: CALL PATTERN(#1,100)
1140 CALL COLOR(#1,2)
1150 CALL JOYST(1,X,Y)
1160 IF X=-4 THEN SC=MAX(SL+8,SC-8)
1170 IF X=4 THEN SC=MIN(SH,SC+8)
1180 CALL LOCATE(#1,RD,SC)
1190 CALL COLOR(#1,1)
1200 CALL KEY(1,K,S):: IF K<>18 THEN 1140
1210 CALL POSITION(#1,X,Y):: GX=X/8+1 :: GY=Y/8+1 :: CALL GCHAR(GX,GY,CH)
1220 IF CH=32 THEN 1100
1230 CALL SPRITE(#5,136,2,X,Y)
1240 HX,CX=14*8+1 :: MY,CY=2*8+1 :: CALL LOCATE(#5,CX,CY)
1250 MX=HX-14*8 :: HY=MY+14*8
1260 CALL COLOR(#5,2,#1,1)
1270 CALL JOYST(1,X,Y)
1280 IF X=-4 THEN CY=MAX(MY,CY-8)
1290 IF X=4 THEN CY=MIN(HY,CY+8)
1300 IF Y=4 THEN CX=MAX(MX,CX-8)
1310 IF Y=-4 THEN CX=MIN(HX,CX+8)
1320 CALL LOCATE(#5,CX,CY)
1330 CALL COLOR(#1,2,#5,1)
1340 CALL KEY(1,K,S)
1350 IF K<>18 THEN 1260
1360 PX=CX/8+1 :: PY=CY/8+1 :: CALL CHECK(PX,PY,FLAG,ANZ,DIR,LASTPOS(,))
1370 IF NOT FLAG THEN 1260
1380 ANZ=ANZ+1
1390 CALL GCHAR(PX,PY,LASTCHR(ANZ)):: CALL HCHAR(PX,PY,CH)
1400 CALL COLOR(#1,1):: CALL VCHAR(GX,GY,32,2)
1410 LAST(ANZ)=CH
1420 LASTPOS(ANZ,1)=PX :: LASTPOS(ANZ,2)=PY
1430 LS(ANZ,1)=GX :: LS(ANZ,2)=GY
1440 GOTO 1090
1450 CALL CONTROL(ANZ,LASTPOS(,),LAST(,),LASTCHR(,),CFLAG,UFLAG,A(,),POI(,),P,DIR)
1460 IF NOT CFLAG THEN 1090
1470 DISPLAY AT(19-4*(P>2),B-14*(INT(P/2)=P/2)-LEN(STR$(POI(P))))BEEP SIZE(LEN(STR$(POI(P)))):STR$(POI(P))
1480 P=P+1 :: IF P>PL THEN P=1
1490 FOR I=1 TO ANZ
1500 CALL ZIEHEN(A$( ),N,B$)
1510 IF B$="" THEN 1550
1520 DISPLAY AT(LS(I,1),LS(I,2)-2)SIZE(1)BEEP:B$
1530 DISPLAY AT(LS(I,1)+1,LS(I,2)-2)SIZE(1):STR$(A(ASC(B$)-64))
1540 NEXT I

```



```

1550 ANZ=0
1560 GOTO 1090
1570 IF ANZ>0 THEN 1090
1580 BA=0
1590 WX=17-4*(P>2):: WY=-14*(INT(P/2)=P/2)
1600 FOR I=1 TO 8
1610 CALL GCHAR(WX,WY+I+2,BC)
1620 IF BC<>32 THEN BA=BA+1 :: N=N+1 :: A$(N)=CHR$(BC):: CALL VCHAR(WX,WY+I+2,32,2)
1630 CALL SOUND(50,220+20*I,6):: NEXT I
1640 FOR I=1 TO BA
1650 CALL ZIEHEN(A$(N),N,B$)
1660 CALL HCHAR(WX,WY+I+2,ASC(B$))
1670 CALL HCHAR(WX+1,WY+I+2,A(ASC(B$)-64)+48)
1680 CALL SOUND(50,392,7)
1690 NEXT I
1700 P=P+1 :: IF P>PL THEN P=1
1710 GOTO 1090
1720 IF ANZ=0 THEN 1090
1730 CALL HCHAR(LASTPOS(ANZ,1),LASTPOS(ANZ,2),LASTCHR(ANZ))
1740 CALL HCHAR(LS(ANZ,1),LS(ANZ,2),LAST(ANZ))
1750 CALL HCHAR(LS(ANZ,1)+1,LS(ANZ,2),A(LAST(ANZ)-64)+48)
1760 ANZ=ANZ-1
1770 GOTO 1090
1780 CALL DELSPRITE(ALL)
1790 FOR I=1 TO PL :: FOR J=1 TO 8
1800 R=18-4*(I>2):: S=2+J-14*(INT(I/2)=I/2):: CALL GCHAR(R,S,C):: IF C=32 THEN 1820
1810 POI(I)=POI(I)-C+48 :: CALL SOUND(100,124,5,-8,9):: CALL VCHAR(R-1,S,32,2)
1820 NEXT J :: NEXT I
1830 CALL CLEAR
1840 FOR I=1 TO PL
1850 DISPLAY USING "SPIELER NR. #: ### PUNKT"&CHR$(-69*(POI(I)<>1)):I,POI(I)
1860 DISPLAY
1870 NEXT I
1880 DISPLAY RPT$("*",28)
1890 DISPLAY
1900 DISPLAY "BUCHSTABENMIX..."
1910 DISPLAY :: DISPLAY
1920 END
1930 SUB SETFIELD(A$(),A())
1940 DATA A,6,1,B,2,3,C,4,3,D,6,3,E,16,1,F,3,2,G,3,3,H,5,1,I,9,1,J,1,6
1950 DATA K,2,2,L,4,1,M,4,1,N,10,1,O,4,1,P,1,4,Q,1,9,R,7,1,S,8,1,T,5,1
1960 DATA U,6,1,V,1,6,W,2,3,X,1,8,Y,1,9,Z,2,4
1970 DATA [,1,5,\,1,5,1,1,5,^,2,0
1980 DATA EOD,0,0
1990 I,J=1
2000 RESTORE 1940
2010 READ B$,ANZ,WERT
2020 IF ANZ=0 THEN SUBEXIT
2030 A(J)=WERT :: J=J+1
2040 FOR K=1 TO ANZ
2050 A$(I)=B$ :: I=I+1
2060 NEXT K
2070 GOTO 2010
2080 SUBEND
2090 SUB WRITE(B$,Z,S,C$)
2100 TON=352
2110 IF C$="S" THEN 2170
2120 FOR I=0 TO LEN(B$)-1
2130 CALL HCHAR(Z+S+I,ASC(SEG$(B$,I+1,1)))
2140 CALL SOUND(50,TON,3,TON*2,4):: TON=TON+58
2150 NEXT I
2160 SUBEXIT
2170 FOR I=0 TO LEN(B$)-1
2180 CALL VCHAR(Z+I,S,ASC(SEG$(B$,I+1,1)))
2190 CALL SOUND(50,TON,3,TON*2,4):: TON=TON+58

```

```

2200 NEXT I
2210 SUBEND
2220 SUB TASTER
2230 DISPLAY AT(24,1): ">> TASTER DRICKEN"
2240 CALL KEY(1,K,S):: IF K<>18 THEN 2240
2250 SUBEND
2260 SUB ZIEHEN(A$( ),N,B$)
2270 IF N<1 THEN B$="" :: SUBEXIT
2280 RANDOMIZE
2290 R=INT(RND*N)+1
2300 B$=A$(R)
2310 A$(R)=A$(N)
2320 N=N-1
2330 SUBEND
2340 SUB MENU(C)
2350 CALL SPRITE(#3,99,14,7*8+4*8+1,18*8+1)
2360 C=4
2370 CALL COLOR(#3,1):: CALL JOYST(1,X,Y)
2380 IF Y=4 THEN C=MAX(1,C-1)
2390 IF Y=-4 THEN C=MIN(C+1,5)
2400 CALL COLOR(#3,14):: CALL LOCATE(#3,57+8*C,145)
2410 CALL KEY(1,K,S):: IF K<>18 THEN 2370
2420 SUBEND
2430 SUB CHECK(R,S,FLAG,ANZ,DIR, LASTPOS(,))
2440 CALL GCHAR(R,S,C)
2450 IF C>64 AND C<95 THEN FLAG=0 :: SUBEXIT
2460 IF ANZ>0 THEN 2500
2470 IF R>=2 THEN CALL GCHAR(R-1,S,M):: IF M>64 AND M<95 THEN DIR=-1 :: GOTO 2570
2480 IF R<15 THEN CALL GCHAR(R+1,S,M):: IF M>64 AND M<95 THEN DIR=-1 :: GOTO 2570
2490 DIR=0 :: GOTO 2570
2500 IF ANZ>1 THEN 2540
2510 IF R<>LASTPOS(1,1) AND S<>LASTPOS(1,2) THEN FLAG=0 :: SUBEXIT
2520 DIR=(LASTPOS(1,2)=S)
2530 GOTO 2570
2540 IF DIR=0 THEN 2560
2550 IF S=LASTPOS(ANZ,2) THEN 2570 ELSE FLAG=0 :: SUBEXIT
2560 IF R=LASTPOS(ANZ,1) THEN 2570 ELSE FLAG=0 :: SUBEXIT
2570 FLAG=-1
2580 SUBEND
2590 SUB CONTROL(ANZ, LASTPOS(, ), LAST( ), LASTCHR( ), CFLAG, UFLAG, A( ), POI( ), P, DIR)
2600 UMG, CFLAG=0
2610 IF ANZ=0 THEN CFLAG=-1 :: SUBEXIT
2620 IF UFLAG THEN 2650
2630 CALL GCHAR(8,10,C):: IF C=ASC("*") THEN SUBEXIT
2640 UMG=-1
2650 L=ANZ
2660 WORTW=1
2670 FOR I=1 TO L
2680 IF LASTCHR(I)=104 THEN WORTW=WORTW*3
2690 IF LASTCHR(I)=112 THEN WORTW=WORTW*2
2700 NEXT I
2710 SUM=0
2720 DX,JY=(DIR<>0):: DY,JX=(DIR=0):: Z=0
2730 AX,BX=LASTPOS(1,1):: AY,BY=LASTPOS(1,2)
2740 CALL GCHAR(AX,AY,C)
2750 IF C<64 OR C>94 THEN 2850
2760 F=1 :: FOR I=1 TO L
2770 IF LASTPOS(I,1)=AX AND LASTPOS(I,2)=AY THEN 2780 ELSE 2810
2780 IF LASTCHR(I)=120 THEN F=2
2790 IF LASTCHR(I)=128 THEN F=3
2800 GOTO 2820
2810 NEXT I :: UMG=-1
2820 SUM=SUM+A(C-64)*F
2830 AX=AX+DX :: AY=AY+DY
2840 IF AX>=1 AND AY>=3 AND AX<16 AND AY<18 THEN 2740

```



```

2850 Z=Z+1 :: IF Z=1 THEN MX=AX :: MY=AY :: DX=-DX :: DY=-DY :: AX=BX :: AY=BY :
: GOTO 2830
2860 FOR I=1 TO L :: IF LASTPOS(I,1)<MX OR LASTPOS(I,2)<MY OR LASTPOS(I,1)>AX OR
LASTPOS(I,2)>AY THEN CFLAG=0 :: SUBEXIT
2870 NEXT I
2880 SUM=SUM*WORTW :: CFLAG=UMG :: DX=JX :: DY=JY
2890 FOR I=1 TO L
2900 UMG,ZSUM,Z=0 :: F=1 :: WORTW=1
2910 AX,BX=LASTPOS(I,1)
2920 AY,BY=LASTPOS(I,2)
2930 AX=AX-DX :: AY=AY-DY
2940 IF AX<1 OR AX>=16 OR AY<3 OR AY>=18 THEN 2990
2950 CALL GCHAR(AX,AY,C)
2960 IF C<65 OR C>94 THEN 2990
2970 ZSUM=ZSUM+A(C-64):: UMG=-1
2980 GOTO 2930
2990 AX=BX :: AY=BY :: DX=-DX :: DY=-DY
3000 Z=Z+1 :: IF Z=1 THEN GOTO 2930
3010 IF UMG=0 THEN 3090
3020 CFLAG=-1 :: X=LASTCHR(I)
3030 CALL GCHAR(BX,BY,C):: ZS=A(C-64)
3040 IF X=104 THEN WORTW=WORTW*3
3050 IF X=112 THEN WORTW=WORTW*2
3060 IF X=120 THEN ZS=ZS*2
3070 IF X=128 THEN ZS=ZS*3
3080 SUM=SUM+(ZS+ZSUM)*WORTW
3090 NEXT I
3100 IF CFLAG THEN POI(P)=POI(P)+SUM-50*(ANZ=8):: UFLAG=-1
3110 SUBEND

```

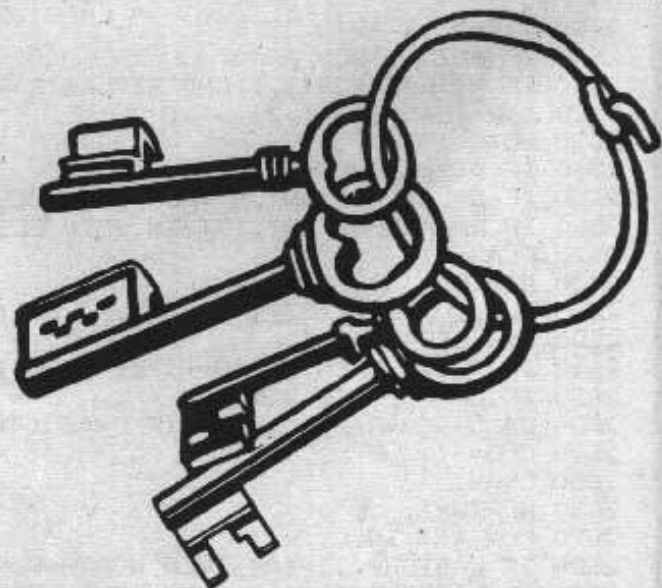
Schlüssel

Mit diesem Programm lassen sich beliebige Texte verschlüsseln und anschließend wieder entschlüsseln. Man benötigt für das Programm (3614 Bytes) den TI-99/4A, einen Cassettenrecorder oder ein Diskettensystem und wenn möglich einen Drucker (RS 232 u. Seikosha GP-250X).

Nach dem Start des Programmes erscheint ein kurzer erklärender Text auf dem Bildschirm. Kurze Zeit darauf kann man den Code eingeben, in

dem der Rechner verschlüsseln oder entschlüsseln soll. Nach einer weiteren kurzen Wartezeit kann man aus einer Wahlliste den Arbeitsbereich herausuchen (ver-, entschlüsseln, Code ändern). Danach wird der Text eingegeben. Er sollte nicht länger als zwei Bildschirmzeilen sein, da die Berechnung sonst zu lange dauert. Wer will, kann anschließend den ver- oder entschlüsselten Text ausdrucken.

Norbert Beckmann



Schlüssel

```

100 ! N. BECKMANN
105 ! WIKER STR. 3
110 ! 2340 KAPPELN
115 ! 29. DEZ. 1984
120 CALL CLEAR :: CALL SCREEN(16)
125 PRINT "SCHLUESSEL": : : :
130 PRINT "MIT DIESEM PROGRAMM LASSEN SICH BELIEBIGE TEXTE VER- UND ENTSCHLUE
SSELN"
135 PRINT : : : PRINT "DER TEXT SOLLTE NICHT MEHR ALS ZWEI ZEILEN LANG SEIN":
: : :
140 DIM B$(130),A$(130),Y$(150),Z$(150),A1$(130)
145 FOR C=1 TO 127

```

```

150 READ B$(C)
155 NEXT C
160 FOR C=1 TO 127
165 READ A$(C)
170 NEXT C
175 INPUT "KODE (1-50) ":C5 :: PRINT : : :
180 IF C5<1 OR C5>50 THEN 690
185 FOR C=1 TO 127
190 C6=C :: C7=C6+C5
195 IF C7>127 THEN 200 ELSE 220
200 IF XC=1 THEN 215
205 XC=1
210 CB=1
215 C7=C8 :: CB=CB+1
220 A1$(C7)=A$(C)
225 NEXT C
230 FOR C=1 TO 127
235 A$(C)=A1$(C)
240 NEXT C
245 DATA 1,2,3,4,5,6,7,8,9,0,.
250 DATA A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z
255 DATA AA,BB,CC,DD,EE,FF,GG,HH,II,KK,LL,MM,NN,OO,PP,RR,SS,TT,UU
260 DATA AB,AE,AH,AI,AU,EI,EU,IE,OH,OE,UH,UE,ER,ES,CH,TZ,ST,UN
265 DATA EN,EM,IN,UM,IM,CK,NE,ME,SO,US,IS,ZU,MO,NO,AN,AM
270 DATA SCH,UNG,ING,UND,SIE,WIR,IHR,VER,IEH,VOR,DEN,DAS,DER,IST,BIS,ZUM,NUN,DEM
275 DATA AEH,OEH,UEH,EIT,TER,TEN,TEL,EIS,TEM,LAS,MUS,BUS,RAM,ANG,ONG,NIE,TAS," "
280 DATA GF,7F,FF,6F,EF,5F,DF,4F,CF,3F,BF
285 DATA DA,5A,51,E1,61,6A,EB,5B,52,E2,EC,6C,63,6D,FD,7D,73,74,F4,E4,ED,E3,5E,GE
,65,7E
290 DATA G1,71,23,EA,FA,3A,4A,DB,5C,DC,GC,64,DD,3D,4D,75,3E,CE,C5
295 DATA 7A,53,DE,D1,41,GB,F1,6D,63,G2,7B,72,C1,5D,54,FB,FE,F5,64,7C,F2,6B,31,4B
300 DATA 62,CB,B5,BB,B2,BE,2D,22,D5,4E,FC,F3,6E,65,EE,E5,55,D2,42,C2,3B,D3,4C,43
,4D,CC,32,3C,C3,33,CD
305 DATA 34,C4,45,35,44,BA,BC,B3,BD,B4,2A,21,2C,B1,GA,CA
310 GOTO 320
315 CALL CLEAR
320 PRINT "(A) VERSCHLUESSELN": : : : PRINT "(B) ENTSCHLUESSELN": : : : PRINT "(C)
KODE AENDERN": : : : PRINT "(E) ENDE": :
325 CALL KEY(0,Z1,Z2)
330 IF Z2=0 THEN 325
335 IF Z1=65 THEN 355
340 PRINT : : :
345 IF Z1=67 THEN 175
350 IF Z1=66 THEN 565 ELSE 690
355 PRINT :: PRINT "EINGABE DES ZU VER- SCHLUESSELNDEN TEXTES" :: PRINT
: : : INPUT "":B1$
360 PRINT : :
365 B=LEN(B1$)
370 C1=1 :: FOR C=1 TO B :: C2=C
375 C$=SEG$(B1$,C,3)
380 FOR D=91 TO 125
385 IF C$=B$(D) THEN C2=C2+2
390 IF C$=B$(D) THEN 470
395 NEXT D
400 C$=SEG$(B1$,C,2)
405 FOR D=38 TO 90
410 IF C$=B$(D) THEN C2=C2+1
415 IF C$=B$(D) THEN 470
420 NEXT D
425 C$=SEG$(B1$,C,1)
430 FOR D=1 TO 37
435 IF C$=B$(D) THEN 470
440 NEXT D
445 IF H=1 THEN 460
450 IF C$=CHR$(32) THEN D=126
455 H=1 :: GOTO 470

```



```

460 IF C$=CHR$(32) THEN D=127
465 H=0
470 Z$(C1)=A$(D):: C1=C1+1 :: C=C2
475 NEXT C
480 FOR C=C1-1 TO 1 STEP -1
485 PRINT Z$(C);
490 NEXT C
495 PRINT : : : : : PRINT "DRUCKEN (J/N)"
500 CALL KEY(0,Z1,Z2)
505 IF Z2=0 THEN 500
510 IF Z1=74 THEN 515 ELSE 550
515 PRINT :: PRINT "DRUCKEN"
520 OPEN #1:"PIO.LF"
525 FOR C=C1-1 TO 1 STEP -1
530 PRINT #1:Z$(C);
535 NEXT C
540 PRINT #1
545 CLOSE #1
550 PRINT :: PRINT "DRUECKE ENTER" :: PRINT
555 CALL KEY(0,Z1,Z2)
560 IF Z2=0 THEN 555 ELSE 315
565 PRINT "EINGEBE DES ZU ENT-          SCHLUESSELNDEN TEXTES" :: PRINT : : : :
INPUT "":X$ :: PRINT : :
570 B=LEN(X$)
575 S1=1 :: FOR C=1 TO B STEP 2
580 C$=SEG$(X$,C,2)
585 FOR D=1 TO 127
590 IF C$=A$(D) THEN 600
595 NEXT D
600 Y$(S1)=B$(D):: S1=S1+1
605 NEXT C
610 FOR C=S1-1 TO 1 STEP -1
615 PRINT Y$(C);
620 NEXT C :: PRINT : : :
625 PRINT : : : : PRINT "DRUCKEN (J/N)"
630 CALL KEY(0,Z1,Z2)
635 IF Z2=0 THEN 630
640 IF Z1=74 THEN 645 ELSE 675
645 PRINT :: PRINT "DRUCKEN" :: OPEN #1:"PIO.LF"
650 FOR C=S1-1 TO 1 STEP -1
655 PRINT #1:Y$(C);
660 NEXT C
665 PRINT #1
670 CLOSE #1
675 PRINT :: PRINT "DRUECKE ENTER" :: PRINT
680 CALL KEY(0,Z1,Z2)
685 IF Z2=0 THEN 680 ELSE 315
690 DISPLAY AT(14,11)ERASE ALL:"E N D E"
695 END

```

Monopoly Teil 2 (Der erste Teil war in der CK 8/9 abgedruckt.)

```

1860 DISPLAY AT(18,5)SIZE(16):"GEFAENGNIS" :: DISPLAY AT(19,5)SIZE(16):"FREIKART
E" :: DISPLAY AT(20,5)SIZE(16):"ABGEGEBEN   FREI" :: FOR Z=1 TO 750 :: NEXT Z ::
GOTO 1940
1870 FOR L=990 TO 110 STEP -55 :: CALL SOUND(125,INT(L*1.6),0,INT(L*1.4),1,INT(L
*1.2),2):: NEXT L
1880 IF GEF(SP)>0 THEN 1950 ELSE CALL DELSPRITE(#SP):: CALL SPRITE(#SP,88,FA(SP)
,170-SP,26-SP):: GEF(SP)=5
1890 DISPLAY AT(11,5)SIZE(16):"IM GEFAENGNIS" :: DISPLAY AT(18,5)SIZE(16):"" ::
DISPLAY AT(19,5)SIZE(16):"FREIKAUFEN J N" :: DISPLAY AT(20,5)SIZE(16):""
1900 IF NAME$(SP)="COMPUTER" THEN IF GELD(SP)>0 THEN V=.5 ELSE V=.25 ELSE 1920
1910 IF RND<=V THEN SUBEXIT ELSE 1930

```

```

1920 CALL KEY(0,K,S):: IF K=78 THEN SUBEXIT ELSE IF K<>74 THEN 1920
1930 GELD(SP)=GELD(SP)-1000 :: GEF(SP)=0 :: FP=FP+1000 :: DISPLAY AT(7,13)SIZE(8)
:USING "#####";GELD(SP)
1940 CALL DELSPRITE(#SP):: CALL SPRITE(#SP,88,FA(SP),178-SP,18-SP):: SUBEXIT
1950 IF W1=W2 THEN GEF(SP)=-1 ELSE 1970
1960 DISPLAY AT(18,5)SIZE(13)BEEP:"EIN PASCH ICH" :: DISPLAY AT(19,5)SIZE(13):"B
IN ENTKOMMEN" :: FOR Z=1 TO 500 :: NEXT Z :: GOTO 1940
1970 SUBEND
1980 SUB BILD1(B())
1990 CALL SOUND(-1500,440,0)
2000 CALL CHAR(33,"80C0E0F0F8FCFEFF",64,"0103070F1F3F7FFF",35,"FFFFFFFFFFFFFFFFF
,36,"FF7F3F1F0F070301",37,"FFFEFCF8F0E0C000")
2010 CALL CHAR(38,"1F3F7FFFCF8F0F0F8FCFEFF3F1F0F0FF0F0F8FCFF7F3F1F0F0F1F3FFFFEE
CF8")
2020 CALL CHAR(42,"E73F3F3F3F3C3C3C",43,"F0F8FCFEFE7E3E1E",135,RPT$("3C",8),45,R
PT$("1E",7)&"1F")
2030 CALL CHAR(46,"FFFFFFFFCFBF0F0",47,"F0F0F8FCFFFFFFFF")
2040 CALL CHAR(58,RPT$("F0",8)&"0301"&RPT$("0",12)&"FCFE"&RPT$("F",24)&"7F3F")
2050 CALL CHAR(62,"0000000000000000",63,"0000FFFFFFFFFFFF",129,"F0F0F0F0F8FCFE7F
3F",130,"3F1F0F0703030303")
2060 CALL CHAR(131,RPT$("03",8)&RPT$("0F",4)&"1F3F7FFFCF8F0E0"&RPT$("C0",10)&"F
0F0")
2070 DISPLAY AT(4,1)ERASE ALL:" #!$%&'()*+,-./:;<=>?@A-Z[\\]^_`{|}~";CHR$(129);CHR$(132):" #
#>(<)" ;CHR$(135);"- (> /> (> #>";CHR$(130);CHR$(133
)
2075 DISPLAY AT(7,1):" : ";CHR$(131);CHR$(134):" ";RPT$("#",
24)
2080 DISPLAY AT(11,2):"PROGRAMMIERT VON": " GERHARD N E U M E S": " REITZSTRA
SSE 10": " 8750 ASCHAFFENBURG": " FERNRUF 06021 43627": " : " WEITER, TASTE
DRUECKEN"
2090 DATA 2,30000,30000,30000,30000,1,15000,293,1,15000,277,1,15000,293,1,15000,
329,1,15000,349,1,15000,329
2100 DATA 1,15000,349,1,15000,391,6,220,261,349,440,6,349,440,2,440,523,6,349,44
0,1,466,587,1,440,523
2110 DATA 6,349,440,1,15000,391,1,15000,349,4,174,220,293,349
2120 DATA 6,15000,293,2,15000,349,7,15000,293,1,15000,220
2130 DATA 2,30000,30000,30000,30000,1,15000,293,1,15000,277,1,15000,293,1,15000,
329,1,15000,349,1,15000,329
2140 DATA 1,15000,349,1,15000,391,6,220,261,349,440,6,349,440,2,440,523,6,349,44
0,1,466,587,1,440,523
2150 DATA 6,349,440,1,15000,391,1,15000,349,4,174,220,293,349
2160 DATA 6,15000,293,2,15000,349,7,15000,293,1,15000,440,4,15000,15000,15000,15
000,3,466,587,1,440,554
2170 DATA 1,466,587,1,440,554,1,466,587,1,440,554,6,466,587,1,587,698,1,466,587,
4,220,261,349,440
2180 DATA 6,440,523,1,466,587,1,440,523,6,349,440,1,15000,391,1,15000,349,4,174,
220,293,349,6,15000,293
2190 DATA 2,15000,349,8,15000,293,0
2200 RESTORE 2090
2210 AZ,STV=0
2220 FOR I=0 TO 2 :: CALL SOUND(-600,220,I):: NEXT I
2230 READ K :: IF K=0 THEN 2200
2240 READ B(1),B(2),B(3),B(4)
2250 FOR I=1 TO K
2260 FOR J=1 TO 4
2270 STV=STV+1 :: IF STV>AZ THEN READ AZ,M1,M2 :: STV=1
2280 FOR L=0 TO 2 :: CALL SOUND(-600,M1,L+1,M2,L,B(J),9):: NEXT L :: CALL KEY(0,
T,S):: IF S<>0 THEN CALL CLEAR :: SUBEXIT
2290 NEXT J :: NEXT I
2300 GOTO 2230
2310 SUBEND
2320 SUB BILD2
2330 DISPLAY AT(15,1)ERASE ALL:"ICH DEFINIERE MEINE ZEICHEN" :: CALL SOUND(-1500
,440,0)
2340 CALL CHAR(33,"FF808080808080FF",143,"818181818181FF",35,"FF010101010101FF
,36,"FF81818181818181",44,"000018243C2424")
2350 CALL CHAR(37,"01010101010101FF",38,"FF01010101010101",39,"FF80808080808080",
40,"80808080808080FF",46,"00001824242418")

```



```

2360 CALL CHAR(41,"8080808080808080",42,"00000000000000FF",43,"01010101010101",
45,"FF")
2370 CALL CHAR(47,"FF80B0B0B0B0B0B0FF",58,"FF80B6B6B6B6B680FF",59,"FF016161616101FF",
60,"FF016D6D6D6D01FF")
2380 CALL CHAR(61,"FF81B0B081818181",62,"FF81B0B081B0B081",63,"81B0B081818181FF",
64,"81B0B081B0B081FF")
2390 CALL CHAR(94,"FF01394545453901",95,"FF80B884B884B88",96,"FF808898A888888",9
7,"FF8098A48890BC8")
2400 CALL CHAR(98,"FF80BCA0B884B88",99,"FF01314911217901",100,"FF80848C94BE848",
101,"FF010919297D0901",102,"FF808C909C928C8")
2410 CALL CHAR(103,"FF01314171493101",104,"FF80BEA28488908",105,"FF8098A498A4988
",106,"FF01192519251901",107,"FF808888BE88888")
2420 CALL CHAR(108,"8080A0A0A0A0BC8",109,"0101192111093101",110,"80BCA0B8A0BC80F
F",111,"01192111093101FF",112,"80B8A4B8A4B880FF")
2430 CALL CHAR(113,"013D2139212101FF",114,"FF000818287C08",115,"FF80BCA0B8A0A08"
,116,"FF003824382824",117,"FF01010101012101")
2440 CALL CHAR(118,"8080B8A4B8A0A08",119,"0101392539292501",120,"80A4A8B0A8A480F
F",121,"003C2038203C00FF",122,"0135353D2D2D01FF")
2450 CALL CHAR(123,"80BE848890BE80FF",124,"02070E0C90A0C0F",125,"FF9999FFFF9999F
F",126,"002047FE472000FF",127,"FF8083B1BFB080FF")
2460 CALL CHAR(128,"FF01C181F11901FF",129,"FF8199A5A5A5A5A5",130,"A5A59999999981
FF",131,"FF000000003C424",132,"4E42423C000000FF")
2470 CALL CHAR(133,"FF000000007E4040",134,"7840407E000000FF",135,"80838484848483
80",136,"01C12101E121C101",137,"8087848484848780")
2480 CALL CHAR(138,"01E101018101E101",139,"FF81B0B0B0B0B0B0",140,"B0B0B0B0B0B081
FF",141,"FF80BFBFBFBFBFBFB0FF",142,"FF01FDFDFDFD01FF")
2490 DISPLAY AT(1,1):"stude'dedjd'bcbc'bg'-&";"v,w(%";CHR$(137);CHR$(138);"(<
%pq(%(%(%>0+":"xyz!#(%!#!#!#!#!#";CHR$(127);CHR$(128);"!#(%%"
2500 DISPLAY AT(4,1):"d^$";RPT$(" ",18);"$f^ WUR":"(%";CHR$(143);" SPIELER
";CHR$(143);"(% '---"
2505 DISPLAY AT(6,1):" _g$";RPT$(" ",18);"$f^ ) 0":"(%";CHR$(143);" KAPITAL
";CHR$(143);"(% (**"
2510 CALL HCHAR(4,31,70):: CALL HCHAR(5,31,38):: CALL HCHAR(6,31,43):: CALL HCHA
R(7,31,37)
2520 DISPLAY AT(8,1):"";CHR$(131);"&";RPT$(" ",18);"";CHR$(131);"&":"( ";CHR$(1
32);"% POSITION
(" ;CHR$(132);"% '---"
2525 DISPLAY AT(10,1):" _g$";RPT$(" ",18);"$fe ) 0":"(%";CHR$(143);RPT$(" ",18);C
HR$(143);"(% (**"
2530 CALL HCHAR(9,31,38):: CALL HCHAR(10,31,43):: CALL HCHAR(11,31,37)
2540 DISPLAY AT(12,1):"d^$";RPT$(" ",18);"$d^":"pq";CHR$(143);RPT$(" ",18);CHR$(
143);"pq"
2545 DISPLAY AT(14,1):" _c$";RPT$(" ",18);"";CHR$(133);"& '---":"(%";CHR$(143);RP
T$(" ",18);"(" ;CHR$(134);"% ) 0"
2550 DISPLAY AT(16,1):"aj$";RPT$(" ",18);"$h^ (**":"(%";CHR$(143);" KOMMENTAR
";CHR$(143);"(% "
2555 DISPLAY AT(18,1):" _^";CHR$(129);RPT$(" ",18);"$a^":"(%";CHR$(130);RPT$(" ",
18);CHR$(143);"ao"
2560 CALL HCHAR(14,31,38):: CALL HCHAR(15,31,43):: CALL HCHAR(16,31,37)
2570 DISPLAY AT(20,1):"aj$";RPT$(" ",18);"$1^":"(%";CHR$(143);RPT$(" ",18);CHR$(
143);"(% "
2575 DISPLAY AT(22,1):"'ÜÜ!#!#&!#!#!#!#!&!#kr^ )ÜÜaea^";CHR$(135);CHR$(136);
"a^d^d^c^cl.m"
2580 DISPLAY AT(24,1):"(*%(%(%(%%pqno(%(%(%β%"
2590 SUBEND
2600 SUB SPIELER(ZS,KR,SP,NAME$( ),B( ),FA( ))
2610 DISPLAY AT(15,1):"WIEVIELE SPIELER MAX 6 ? 1" :: ACCEPT AT(15,26)SIZE(-1)VA
LIDATE("123456")BEEP:SP :: Q=0
2620 FOR I=1 TO 6 :: NAME$(I)=" " :: NEXT I :: IF SP=1 THEN SP=2 :: Q=1
2630 FOR I=1 TO SP
2640 B(I)=INT(RND*SP)+1 :: IF I=1 THEN 2670
2650 FOR J=1 TO I-1 :: IF B(I)=B(J)THEN 2640
2660 NEXT J
2670 NEXT I :: CALL CLEAR
2680 FOR I=1 TO SP :: DISPLAY AT(6+I*2,1):"DEN";I;"NAMEN BITTE:" :: DISPLAY AT(2
3,1):"? "
2690 IF Q=1 THEN NAME$(B(I))="COMPUTER" :: Q=0 ELSE ACCEPT AT(23,3)SIZE(8)VALIDA
TE(UALPHA)BEEP:NAME$(B(I)):: IF NAME$(B(I))="" THEN 2690

```

```

2700 DISPLAY AT(6+I*2,20):NAME$(B(I)): NEXT I :: DISPLAY AT(23,1)BEEP:"" :: FOR
ZEIT=1 TO 750 :: NEXT ZEIT :: CALL CLEAR
2710 DATA " 2=SCHWARZ", " 5=DUNKELBLAU", " 7=DUNKELROT", 12=HELLGELB, 13=DUNKELGRUEN
, 16=WEISS
2720 RESTORE 2710 :: FOR I=1 TO 6 :: READ FARBE$(I): NEXT I
2730 FOR I=1 TO SP :: DISPLAY AT(5,1):NAME$(I);",": "WELCHE FARBE WILLST DU HAB
EN": "? 0"
2740 FOR J=1 TO 6 :: DISPLAY AT(11+J,1):FARBE$(J): NEXT J
2750 IF NAME$(I)<>"COMPUTER" THEN 2780
2760 FOR P=1 TO 6 :: IF FARBE$(P)<>" " THEN FA(I)=VAL(SEG$(FARBE$(P),1,2)): FARBE
$(P)=" " :: GOTO 2820
2770 NEXT P
2780 ACCEPT AT(9,3)SIZE(-2)VALIDATE(DIGIT)BEEP:F :: IF F=2 THEN IF FARBE$(1)=" "
THEN 2780 ELSE FA(1)=2 :: FARBE$(1)=" " :: GOTO 2820
2790 IF F=5 THEN IF FARBE$(2)=" " THEN 2780 ELSE FA(1)=5 :: FARBE$(2)=" " :: GOTO
2820 ELSE IF F=7 THEN IF FARBE$(3)=" " THEN 2780 ELSE FA(1)=7 :: FARBE$(3)=" " ::
GOTO 2820
2800 IF F=12 THEN IF FARBE$(4)=" " THEN 2780 ELSE FA(1)=12 :: FARBE$(4)=" " :: GOT
O 2820 ELSE IF F=13 THEN IF FARBE$(5)=" " THEN 2780 ELSE FA(1)=13 :: FARBE$(5)=" "
:: GOTO 2820
2810 IF F=16 THEN IF FARBE$(6)=" " THEN 2780 ELSE FA(1)=16 :: FARBE$(6)=" " :: GOT
O 2820 ELSE 2780
2820 NEXT I
2830 DISPLAY AT(5,1)ERASE ALL:"KREDITLIMIT ? 0" :: ACCEPT AT(5,15)SIZE(-6)VALIDA
TE(DIGIT)BEEP:KR :: KR=-KR
2840 DISPLAY AT(8,1):"KREDIT-ZINSFUSS MAX(5%)? 0" :: ACCEPT AT(8,26)SIZE(-3)VALI
DATE(DIGIT,".")BEEP:ZS :: IF ZS<0 OR ZS>5 THEN 2840
2850 DISPLAY AT(20,1)BEEP:"BEACHTET:" :: "SPIELEND MIT >E<" :: FOR ZEIT=1 TO 750
:: NEXT ZEIT
2860 SUBEND

```

Katalog für Maschinenspracheprogramme

Das folgende Maschinenprogramm für den TI-99/4A mit Extended Basic und 32K-Speichererweiterung erlaubt Ihnen, jederzeit festzustellen, welche Maschinenprogramme sich im Speicher befinden und ab welcher hexadezimalen Adresse sie beginnen. Außerdem können Sie Labels in die Namenstabelle eintragen.

Sie können das Programm auf zwei Wegen eingeben: Den Quellcode mit dem Editor/Assembler assemblieren oder den Basic-Loader mit den DATA-Zeilen starten. In diesem ist auch ein Prüfsummen-Generator enthalten. Da das Maschinenprogramm im Low Memory liegt, kann es von Basic-Programmen nicht versehentlich gelöscht werden. Selbst bei NEW bleibt es erhalten. Erst ein erneutes CALL INIT macht es für den Computer unzugänglich.

Ist nun das Programm geladen, so haben Sie zwei neue Befehle: CALL LINK ("CAT")

gibt einen Katalog der geladenen Maschinenspracheprogramme aus, der mindestens die beiden Einträge für dieses Maschinenprogramm enthält. Neben dem Programmnamen wird auch die hexadezimale Startadresse des jeweiligen Programms angezeigt, z.B.

```

>CALL LINK("CAT")
LABEL >3AA2
CAT >3A14

```

Das zweite Maschinen-Unterprogramm erlaubt Ihnen weitere Einträge in die Namenstabelle vorzunehmen: CALL LINK ("LABEL", "Name", Adresse). Der Name darf maximal sechs Buchstaben lang sein, andernfalls erfolgt eine Fehlermeldung. Mit diesem Unterprogramm können Sie z.B. eine versehentlich gelöschte Namenstabelle wieder herstellen oder auch Systemroutinen des TI-99 ausprobieren, die im ROM liegen. Ein Beispiel: Der Befehl "BYE", der einen Reset auslöst, darf nur im Direktmodus eingegeben werden. Wird

ein Reset ausgelöst, so springt der Computer über einen Vektor in der Adresse 0 auf Adresse 36. Geben Sie nun einmal ein:

```

>CALL LINK("LABEL",
"RESET", 36)
>CALL LINK("CAT")
RESET >0024

```

LABEL >3AA2
CAT >3A14

Nun können Sie auch in Programmen zum Titeldisplay zurückspringen: CALL LINK ("RESET"). Aber Vorsicht: Selbstverständlich werden dabei alle Programme und Daten gelöscht!

Martin Kotulla

Assembler-Listing

```

* *****
*          CATALOG & LABEL
* *****
*
* (C) Martin Kotulla 1985
*
* - CALL LINK("CAT") zeigt Katalog der Maschinenprogramme
* - CALL LINK("LABEL", Adr, Rm) übernimmt ein Label in die Namenstabelle
* *****
*
* ADDR 14000
* DEF CAT, LABEL
* IDT 'CAT&LBL'
*
* SCROLL EQU 10024
* NEXT EQU 10070
* OFI EQU 10298
* LSTBL EQU 12004
* NUMREF EQU 1200C
* STRREF EQU 12014
* RLINK EQU 12018
* KSCAN EQU 1201C
* VPM EQU 12024
* PAC EQU 12348
* STATUS EQU 1837C
* GPLUG EQU 183E0
*
* ZERO BYTE 100
* SIX BYTE 106
* EIGHT DATA 1000B
* SPACE DATA 12020
* SET DATA 12000
* UBRUS BSC 32
*
* HEXBUF BYTE 144,145,146,147,148
*          BYTE 149,150,151,152,153
*          BYTE 161,162,163,164,165,166
*
* NABUF DATA 12020,12020,12020 Buffer fuer einen geladenen Programmnamen
*          DATA 1202E,12020,12020
*
* Einiges SCROLL-Programme
* - Sprung in den GPL-Interpreter
* - Convert Floating Point to Integer
* - Start der Namenstabelle
* - Numerischen Parameter übernehmen
* - Stringparameter übernehmen
* - ROM-Routinen aufrufen
* - Tastatur abfragen
* - VDF-RAM mehrere Bytes schreiben
* - Fließkomma-Addi
* - GPL-Status-Byte
* - GPL-Workspace
*
* Null-Byte
* Sechs-Byte
* Acht-Wort
* Zwei Leerzeichen
* Taste gedrückt bei KSCAN
* Eigene Workspace-Register
* Hexadezimal-Zahlen plus Screen-Offset

```


***** * CALL LINK("CAT") * *****			HEXA	MOV R9,R2 ANDI R2,000F MOVB BHEXBUF(R2),R4+	Adresse kopieren Nibbles 1, 2 und 3 ausblenden Passende Hexziffer lesen und schreiben
CAT	LUP1 USRUS	Eigenen Workspace laden	SHOU	LI R0,745 LI R1,NAHBUF BLUP BVBUD BLUP BXLLNK DATA SCROLL INCT R7 CI R7,16383 JLT BUFADR JMP RETURN	VDP-Adresse der untersten Bildschirmzeile Zeiger auf Buffer des Programms 12 Bytes auf den Bildschirm zu schreiben Und in den VDP-RAM uebertrogen Bildschirm scrollen Zeiger auf naechsten Eintrag in der Tabelle Ende der Namenstabelle erreicht? Nein - weitermachen Rueckprung nach Basic
CATSCR	BLUP BXLLNK DATA SCROLL	Bildschirm scrollen	***** * CALL LINK("LABEL",NAME,ADRESSE) * *****		
START	MOV R0,STLOW,R7 BUFADR LI R8,NAHBUF LENGTH LI R9,6	Adresse der Namenstabelle holen Zeiger auf internen Bufferbereich 5 Zeichen pro Name lesen	***** * CALL LINK("LABEL",NAME,ADRESSE) * *****		
CHRGIT	BLUP BKSCAN MOV R0,STATUS,R5 CDC RSET,R5 JNE COPYLP	Testatur abfragen SPL-Status-Byte holen Wurde eine Taste gedrueckt? Nein - noraent weitermachen	***** * CALL LINK("LABEL",NAME,ADRESSE) * *****		
EXP2ND	BLUP BKSCAN MOV R0,STATUS,R5 CDC RSET,R5 JNE EXP2ND	Ja - Listing einholen Auf weiteren Tastendruck warten	***** * CALL LINK("LABEL",NAME,ADRESSE) * *****		
COPYLP	MOVB R7+,RA AI R8,0000 MOVB R6,R0+ DEC R9 JNE COPYLP	Ein Namensbyte in den Buffer schreiben ... um Bildschirm-Offset beruecksichtigen In den Buffer uebertrogen Laengenzuehler dekrementieren Fertig? - Nein, weitermachen	LABEL	LUP1 USRUS MOV R0,STLOW,R2 AI R2,-8	Eigenen Workspace laden Startadresse der Namenstabelle holen Zeiger auf naechsten freien Platz
DETADR	MOV R7,R9 LI R4,NAHBUF+B	Adresswort nach dem Namen lesen Zeiger auf Adressenspeicher in Buffer	CLEAR	MOV BSPACE,R2+ MOV BSPACE,R2+ AI R2,-7	Vorgegebenen Platz fuer den Eintrag loeschen ** Zeiger auf Byte vor dem Platz fuer den Eintrag
HEX1	MOV RY,R2 SRL R2,12 MOVB BHEXBUF(R2),R4+	Adresse kopieren Nibble 1 ins LSByte/Nibble 4 schieben Passende Hexziffer lesen und schreiben	GETSTR	MOVB BSLX,R2 CLB R0 LI R1,1 BLUP BSTRREF S BRIGHT,0LSTLOW	Byte fuer maximale Stringlaenge schreiben Einfache Stringvariable Erstes Parameter in CALL LINK String uebernehmen Zeiger auf Beginn der Namenstabelle berichtigen
HEX2	MOV R9,R2 ANDI R2,00F0 SRL R2,8 MOVB BHEXBUF(R2),R4+	Adresse kopieren Nibble 1 ausblenden Nibble 2 ins LSByte/Nibble 4 schieben Passende Hexziffer lesen und schreiben	GETNUM	INC R1 BLUP BNUMREF BLUP BXLLNK DATA CFI MOV BFAC,B7(R2)	Zeiger auf 2.Parameter in CALL LINK Zahl uebernehmen ... und in Integer umwandeln Nach dem Programmnamen eintragen
HEX3	MOV R9,R2 ANDI R2,00F0 SRL R2,4 MOVB BHEXBUF(R2),R4+	Adresse kopieren Nibbles 1 und 2 ausblenden Nibble 3 ins LSByte/Nibble 4 schieben Passende Hexziffer lesen und schreiben	RETURN	MOVB BZERO,0STATUS LUP1 BPLUS S BNEXT END	Basic-Rueckprung: SPL-Statusbyte Loeschen - GPL-Workspace laden - Sprung in den GPL-Interpreter

Basic-Lader

```

220 CALL CHAR(42,"AA55AA55AA55AA55",91,"7EB199A1A199B17E")
230 CALL CHAR(45,RPT$("0",8)&"FF")
240 CALL CHAR(128,"DFBF60C0919B9591918080B0C060BDFD")
250 CALL CHAR(130,"FBFD060301010149516151490306FDFB")
260 CALL CLEAR :: CALL HCHAR(2,4,42,26)
270 CALL VCHAR(3,4,42,5):: CALL VCHAR(3,29,42,5)
280 CALL MAGNIFY(3):: CALL SPRITE(1,128,2,28,54)
290 DISPLAY AT(5,9)SIZE(15):"CATALOG & LABEL"
300 CALL HCHAR(8,4,42,26)
310 DISPLAY AT(12,4):"C MARTIN KOTULLA 1985"
320 CALL HCHAR(13,6,45,22):: DISPLAY AT(17,7):"DATAs PRUEFEN? J"
330 ACCEPT AT(17,22)BEEP SIZE(-1)VALIDATE("JN"):JN$
340 IF JN$="N" THEN 390
350 FOR I=14800 TO 15088 STEP 2 :: READ A,B :: SUM=SUM+A+B :: NEXT I
360 DISPLAY AT(20,5)BEEP:"DIE DATA-ZEILEN SIND"
370 IF SUM(>)25091 THEN DISPLAY AT(22,10):"FEHLERHAFT!" :: END
380 DISPLAY AT(22,11):"KORREKT!"
390 RESTORE :: CALL PEEK(8198,ID):: IF ID(>)170 THEN CALL INIT
400 FOR I=14800 TO 15088 STEP 2 :: READ A,B :: CALL LOAD(I,A,B):: NEXT I
410 CALL PEEK(8196,A,B):: C=A*256+B-16
420 CALL LOAD(8196,INT(C/256),C-INT(C/256)*256)
430 CALL LOAD(C,76,65,66,69,76,32,58,162,67,65,84,32,32,32,58,20)
440 CALL CLEAR :: CALL DELSPRITE(ALL):: STOP
450 DATA 000,006,000,008,032,032,032,000,002,233,058,008,000,012,057,242
460 DATA 058,020,000,000,128,029,064,000,058,014,058,020,007,096,032,200
470 DATA 019,022,197,160,057,206,004,024,144,145,146,147,148,149,150,151
480 DATA 152,153,161,162,163,164,165,166,163,161,180,128,128,128,032,158
490 DATA 147,161,145,148,002,224,057,216,004,032,032,024,000,038,193,224
500 DATA 032,004,002,008,058,008,002,009,000,006,004,032,032,028,193,096
510 DATA 131,124,033,096,057,214,022,007,004,032,032,028,193,096,131,124
520 DATA 033,096,057,214,022,249,209,183,002,038,096,000,222,006,006,009
530 DATA 022,250,194,087,002,004,058,016,192,137,009,194,221,034,057,248
540 DATA 192,137,002,066,015,000,009,130,221,034,057,248,192,137,002,066
550 DATA 000,240,009,066,221,034,057,248,192,137,002,066,000,015,221,034
560 DATA 057,248,002,000,002,233,002,001,058,008,002,002,000,012,004,032
570 DATA 032,036,004,032,032,024,000,038,005,199,002,135,063,255,017,193
580 DATA 016,033,002,224,057,216,192,160,032,004,002,034,255,248,204,160
590 DATA 057,212,204,160,057,212,204,160,057,212,002,034,255,249,212,160
600 DATA 057,209,004,192,002,001,000,001,004,032,032,020,104,032,057,210
610 DATA 032,004,005,129,004,032,032,012,004,032,032,024,018,184,200,160
620 DATA 131,074,000,007,216,032,057,208,131,124,002,224,131,224,004,096,000,112
630 END

```



Sinclair

QL-ART – das Mal und Zeichenprogramm! Das neue Grafikprogramm für den QL! Komplett mit Font-Editor, Zoom, 2. Bild im Speicher, Hardcopy, Fill, Shapes, komfortable Menüführung, Kopieren von Bildteilen. Sehr schnell! Handbuch und Cartridge für nur 98.– DM. Auskunft ☎ 040/7019216 od. 040/7017430

UFO-Angriff für ZX81 (Orig.) nur 10.–, ☎ 05485/1980

ZX81 16K (Zusatztastatur!) inklusive Software + Lit. für 100.– bei R. Brandstätter, Frauentalstr. 53, 8880 Dillingen

Wer hat eine Lösung zur Behebung der Programm-Speicherungs- und Programm-Lade-Schwierigkeiten des ZX81? F.J. Becker, Beutligweg 5A, 7570 B-Baden, ☎ 07221/33291

- Vokabeln lernen besonders leicht!! ● Vokabeltrainer "Easy Learn"
 - 100 % Maschinen-Code
 - einmaliges Karteikastenprinzip
 - Speicher: 33 Tausend Zeichen
 - Lexikon-Modus
 - max. Suchzeit für eine Vokabel: 0,14 sek.
 - Tape oder Microdr./Opus-Disk
 - Modifikation auf Beta-Disk möglich
 - Preis: 19.80 DM + Porto
- Mit mehrseitiger Anleitung bei: M. Roland, Postfach 5026, 4902 Bad Salzungen, ☎ 18-21 h 05222/70505

SPECTRUM-PROGRAMMSAMMLUNG (1)

5 Anwendungen, 5 Spielprogramme. **TOOLBYTE** 100% MC, 5 KBytes: 26 außergewöhnl. starke u. effektive Ein-Tastenbefehle. **LINE-DELETE** 50 Bytes MC: Löscht Zeilen einzeln o. blockw. **COMPACTOR** 122 Bytes MC: Verkürzt alle Basic-Programme ca. 20%. **HELP** 200 Bytes MS: Verlangt laufende Progr. mit Anzeige d. lfd. Zeilen-Nr. **DRAW** Grafik 3,5 KBytes MC: 16 Befehle; 2 x Circle, Box, über 1000 versch. Schrifttypen, 3 Gummibänder, echt-Fill etc. **TECHNO** 3 KBytes: Erstellt techn. Zeichn. in 3 Ansichten – Perspektiven mit Demo. **U-BOOT-SIMULATION** 40 KBytes MC: 37 Befehle u. Kontr. + Kontrollraum, Kartenraum u. Periskop. **COSMAZOIGS** 8 KBytes MC: Sehr schn. Arcade. **SNAKE-ALIVE** 6 KBytes MC: 48 Ebenen. **QUAZIMODO** 9,5 KBytes MC: Aufwend. Grafik, 4 Screens. **BLOCKMAN** 8 KBytes: Spaß und Spannung. Alle 10 Programme in commercial quality aus Your Comp. 84/85 mit ausf. deutscher Dokumentation **DM 25.–**, an Josef Bauer, Jakobstr. 11, 6850 Homburg. Jede weitere Kopie der dtsh. Dokumentation DM 4.–.

Verkaufe ZX-Spectrum 48K mit Plus-Tastatur inkl. Demo + 10 Original-Cassetten, VB 200.– DM. ☎ 02164/7610

S.A.V. is coming ... Cassette 25.– DM, Cartridge 30.– DM, inkl. Porto. By Software, Stefan Swiergiel, Südring 11a, 3440 Eschwege

KLEINANZEIGEN

●●●● Endlich ist sie da! ●●●●

Die 64-Zeichen-Karte für Ihren Spectrum. Dieser Hardwarezusatz bringt 64x32 Zeichen auf den Schirm. Kompatibel zu nahezu allen Anwendungsprogrammen (Pascal, Devpac etc.). Gratisinfo bei Ingo Kaun, Jan-Wellen-Str. 35, 5630 Remscheid

●●●● ZX-SPECTRUM ●●●●

Suche Computerprogramme für Sinclair ZX Spectrum 48. Tauschangebot an Gerold Amann, Oberledweg 9, 7967 Bad Waldsee, ☎ 07524/1301

Kaufe alte und beschädigte Spectrums 16 + 48K. DATATEXT, Lorbeerweg 5, 4230 Wesel 1, ☎ 0281/64048

Einmalige Gelegenheit für Sinclair-Freaks: 1 ZX-Interface 1 + 1 ZX-Microdrive, fab. neu, orig. verpackt, mit Garantie zus. 200.– DM. Schröder, Kolpingstr. 26, 4937 Lage, ☎ 05232/80554

USER-CONTACT-SERVICE-SCIUC

Kennen Sie ihn noch nicht? Info gegen 2x50 Pf. anfordern! Spectrum Computer Info User Club R. Frank, Carl-Spitzweg 17, 7920 Heidenheim

ZX Spectrum 80KB, ISO-ROM, + Tast., Recorder, GP 50 S, Digital-Tracer, viel Software und Literatur. Preis: VS. ☎ 07331/60165

Suche defekten Sinclair QL oder Spectrum. ☎ 06587/7007 ab 18.00 Uhr

Zeichentrick mit dem Spectrum: 200 Byte machen aus Screens den Film und 150 Byte zeigen ihn. Preis: 10.– DM. Peter Christmann, Gutenbergstr. 16, 7340 Geislingen

Spectrum Speichererweiterung 48K 40.– DM, ZX81 + Aufsatztastatur + 16K RAM eingebaut + div. Software + Monitorausgang 100.– DM. G. Pfeiffer, ☎ 0941/95290

Verkaufe: PROCEED 1 von Logitek – Floppy- und Druckerinterface 150.– DM; LIGHTPEN + deut. Anleitung 35.– DM; KEMPSTON-Joystickinterface 35.– DM; Cassetten-Video-Interface am User-Port (aus c't Heft 2/1986), Josef Listner, Wellersweilerstr. 184, 6680 Neunkirchen, ☎ 06821/12090

●●● SINCLAIR-Freaks! ●●●

Ein User-Club informiert: Seit 2 Jahren besteht der Spectrum-User-Club Wuppertal. Unsere monatlich erscheinende Clubzeitschrift enthält neben Erfahrungsberichten über Hard- und Software auch Tips, Tricks, die Mitgliederbörse und vieles mehr. Außerdem kann jedes Mitglied eine monatlich erscheinende Cassette mit den besten von Club-Mitgliedern entwickelten Programmen beziehen. Wer mehr über diese und weitere Leistungen des Clubs wissen will, kann gegen Rückporto mehr Informationen anfordern unter der Clubanschrift Rolf Knorre, Postfach 20 01 02, 5600 Wuppertal 2. ● **Angebote von Händlern, die Club-Rabatte einräumen, sind erwünscht!**

Verkaufe orig. Spectrum-Software. 28 Top Progr. je 5.– DM. Außerdem andere gute Progr. ab 4.– DM. RAMBO, COM-MANDO je 12.– DM usw.! Suche strategische Wargames, zahle gut! Liste an oder anfordern für –50 DM Rückporto bei W. Schwans, Chamissostr. 12, 2120 Lüneburg, ☎ 04131/59753. Suche Resetknopf!

Verkaufe Kempston Pro Joyst.-Interface (3 Anschl.) ● **Competition-Pro** 3000, unben. zus. 70.– DM, Sperber, 8800 Bamberg, Sankt-Getreu-Str. 9a, ☎ 0951/58028

Spectrum 80K in dk'tronics Gehäuse mit Interf. 1 + Microdr. + 24 Cartridges + ZX Printer + Kempst. / programmierb. Joystick I/F + Datenrec. + Bücher "ZX Masch. Code" + ROM Buch + 280 Buch + mehr als 50 Kassetten. Preis VS, ☎ 05222/70505 ab 18 Uhr

●● **Achtung Spectrum-Freaks!** ●● ZX-Spectrum 48K, CUB-Farbmonit. 452 (TTL- und Spec.-Eingang, Spec.-Interface), Interface 1, Sprint Digi-Recorder, ZXLprint III mit Parallel- und Serialkabel, Orig.-Software u.a.: Beta-Basic, Masterfile mit MF-Print, Flight, Statistik, günstigst (auch einzeln) abzugeben! P. Enders, ☎ 09131/85-3972, 16115 oder 992711

Spectrum-Software. Verkaufe Original Text-, Grafik-, Drucker-, Spreadsheet-Utility-Programme (je 1 x). Liste anfordern! G. Schmidt, Espenstr. 43, 6230 Frankfurt 80

ZX Spectrum: Verkaufe PROCEED 1 – Floppy- und Druckerinterface – 150.–, LIGHTPEN mit deut. Anleitung = 35.–, KEMPSTON-Joystickinterface = 35.–, CASSETTEN-Video-Interface am User-Port (aus c't Heft 2/1986) = 85.–, J. Listner, Wellersweilerstr. 184, 6680 Neunkirchen, ☎ 06821/12090

Spectrum defekt????

Kaufe Deinen defekten Specc! 40.– DM, Plus 50 inkl. allem Zubehör! ☎ 07321/63879, öfters probieren!

●● **Einkommen-/Lohnsteuer 1986** ●● Schon jetzt die Steuer 1986 berechnen und beeinflussen, da Testauswertung möglich. Update-Service 1987 ff. Spectrum: Cassette 50.–, Microdrive 55.– DM. Info –80 DM. Auch C64/Apple II. Dipl. Finanzwirt U. Olufs, Bachstr. 70, 5216 Niederkassel 2, ☎ 02208/48156

Suche Programme für den Spectrum 128. Thomas Bernhart, Hansestr. 110, 4407 Emsdetten, ☎ 02572/2909

●●● Wegen Systemwechsel ●●●

Verkaufe Spectrum Plus (48K) + IF1 + Microdrive + Multiface one + Datenrecorder + Monitor (Philips TP200) + Joystick + Joystick-IF + ZX-Drucker + 5 Rollen Metallpapier + jede Menge Software u. Lit. + RS-232-Kabel für VB 900.–, ☎ 02134/55050 (ab 18 Uhr)

●●● Utilities zu Tasword Two ●●●

Tasword Two Ergänzung: viele n. Fkt. Tasdata: sortieren beliebiger Daten. Tasformat: neues Format für Texte. Taslist: suchen von Textteilen. Alles MC, je 20.– DM. 3 Pgm 50.– DM oder Info von Johann Weinzierl jun., Eichstätt 29, 8301 Hohenhann, ☎ 08784/289

Hallo Spectrum-User!

Suche ZX-Spectrum Tauschpartner. Listen an: Uwe Gernersheim, Grüner Winkel 11, 4053 Jüchen 3. Antworte 100%ig

● **Hilfsprogramme zu Tasword Two** ● **TASWORD ERGÄNZUNG:** viele zus. Funkt. TASDATA: universelles Sortieren! TASFORMAT: Umformatieren des Textes. Alles MC, je 20.– DM, alle 3 zusammen DM 50.–. Scheck/bar/NN oder –80 DM für Info an: Johann Weinzierl jun., Eichstätt 29, 8301 Hohenhann, ☎ 08784/289

●●● An alle Spectrum-User ●●●

●●● **Werfen Sie Ihr ROM weg!** ●●● Von mir bekommen Sie ein besseres. NMI-Save, eigener Zeichensatz und viele neue Routinen für nur 50.– DM. Info anfordern! S. Hartmann, Am Rain 6, 3503 Lohfelden 1

Suche für Spectrum zuverlässige Software-Tauschpartner (Cass.). Zuschriften und Listen an: Claudia Meurers, Postf. 1233, 4053 Jüchen 2 ● **Antwort 100%ig**

ZX-48: Verkaufe Printer und Original-Programme, suche Astrologieprogramme und Hardcopy für GP-100 Centronics-Version-V-Interface, tausche auch Original-Programme. ☎ 0511/583120

●●● ACHTUNG ●●●

Programmverwaltung löst Ihre Probleme bei einer unübersichtlichen Software-Sammlung (mit 84 Zeichen pro Zeile und Quicksaveroutine). Info gegen Rückporto oder Programm direkt gegen 20.– DM (Schein, Scheck). Jörgsoff, Postf. 1006 10, 4040 Neuss 1

ZX Spectrum Datenbank für Schachpartien. M-Code. Es spielt und speichert über 250 Partien. Info –80 DM bei C. Spark, Hofgasse 23a, 6117 Schaaheim-Mosbach

●●● **ZX-Drucker-Metall-Papier!** ●●● Ab 3 Rollen à 7.– DM + Versandkosten 6.90 DM per Nachnahme bzw. 3.50 DM bei Vorkasse oder Verrechnungsscheck. EST-Müller, Flügeldamm 13, 3000 Hannover 21

Suche Software für den ZX 81 16 K. Marcus Thier, Hartgerstr. 7, 3300 Braunschweig

●●● Für Sinclair QL ●●●

Epromkarten mit integrierter Software. Karten mit 16 oder 32 K Eproms und bis zu acht Steckplätzen. Englische QLS zu absoluten Tiefpreisen! Infos bei: Armin Leng, Rheinfelderstr. 30B, 4047 Dormagen 1

● **Buchungsprogramm SP-DV** ●

Buchungen nach versch. Schlüsseln, Kassenbuch, Kontogegenbuch, Jahresbericht. Sehr komfortabel und extr. flexibel, inkl. Adressenverwaltung zum Lastschriftdruck, Etiketten usw. nur 69.– DM. J. Weinzierl, Eichstätt 29, 8301 Hohenhann, ☎ 08784/289

Verkaufe ZX-Spectrum 48K, Interface I, Microdrive + Software für 250.– DM. H. Haas, Jahnstr. 5, 6330 Wetzlar 15. ☎ 06441/27362

● **Verkaufe wegen Systemwechsel:** ● ZX-Spectrum 48K + Aufsatztastatur + ISO-ROM + Interface 1 + Microdrive + 16 Cartridges + 2 Currah-Slots + zahlreiche Software + viel Literatur + Kempston-Centronics-Interface-E für nur 800.–, ☎ 06131/82583

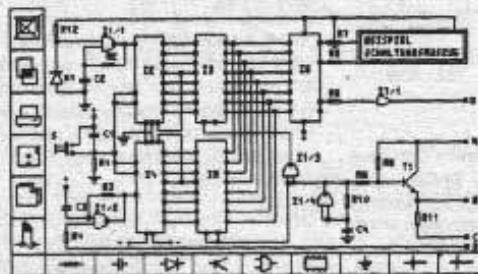
●●● Hallo Spectrum-Freaks! ●●●

Ich suche für den ZX Spectrum ein Joystickinterface mit zwei Ports. Andreas Töpfer, ☎ 07144/12373

Atari • Atari • Atari • Atari

Elektronische Schaltungen konstruieren mit dem ATARI

Bildschirmorientiertes Arbeiten mit dem Joystick.
Widerstand, Kondensator, Diode, Transistor, Gatter, Inverter, ICs usw.
- Drucker-Hardcopy
- Laden und Saven auf Diskette
- Bauteillebibliothek führen usw.



**ATARI
XL/XE 64K
40.- DM**

inkl. Versand
Bestellung:
Schein/Scheck,
Info kostenlos!
Nachnahme
+ 6.- DM

Jürgen Dörr
Einsteinstr. 6
6520 Worms 26
☎ 06241/34140

Spectrum: GP-100A mit Kempston-Centronics-Interf., voll grafikfähig! 80 Zeichen/Zelle, für 300.- DM! Sound Generator, erweiterbar zur PIO für 45.- DM. T. Steiner, Nr. 220, 2812 Hoyersheim, ☎ 04251/3849 (ab 18 Uhr)

MC-Programme debuggen - kein Problem mit dem Z80-Monitor (ZX-SP 48K) 20.- DM. Grafik auf dem Seikosha GP 100 mit LPRINT3- oder Wafadr.-Centronics-IF (Anpassungen für andere IFs auf Anfrage) 30.- DM. Info geg. 80 Pf. bei Jürgen Ohmdorf, Euelsbruchstr. 29, 5905 Freudenberg

Verkaufe ZX Spectrum, Interface 1, Microdrive, ZX-Drucker, Bücher, Original-Software (z.B. Decathlon, Superstest, Nightshade, Boxing, Beta-Basic usw.) ☎ 07148/6148 (ab 17 h).

Lotto-Statistik 6 aus 49: Spectrum 48K. Alle Ziehungen seit 1955! Auswertung in 10 sec.! Mit Gewähr! Bis zu 5 Bank- u. beliebig viele Auswahlzahlen können untersucht werden. Cass. 45.- DM, Scheck od. + NN.: G. Blatt, Wiesenstr. 10, 5509 Schillingen. Auch für CPC 464, 664 und 6128.

Verkaufe: • ZX DATAFORM II •
Ein selbstgeschriebenes Dateiprogramm der Spitzenklasse mit Anleitung und Cassette für 12.- DM. Eric Böhnisch, In den Beeten 80, 7121 Ingersheim 1, ☎ 07142/51661

Drucker GP 50 S zu verkaufen, Preis: 200.- DM VB. C. Fiehnherm, Heukamp 7, 4837 Verl, ☎ 05246/2750

●●● Suche für ZX Spectrum ●●●
●●● und QL ●●●

Selbstgeschriebene Programme zu kaufen (Anwendersoftware u. Spiele). Schickt erst mal eine Liste (wenn's geht) mit Beschreibung an: DATATEXT, Loorbeerweg 5, 4230 Wessel

Verk. Spectrum 48K für 110.- DM, Interface 1 für 40.- DM., beides defekt u. Micro-Hefte je 1.50 DM. ☎ 0621/663284 nach 14 h

Neuheit: Drosseln Sie den Takt Ihres Spectrums per Software! Anwendungsgebiet: Debugging, Bremsen von Spielen etc. Gratisinfo oder Bestellung (Cass. und ausf. Anleitung = 25.- DM) bei Ingo Kaun, Jan-Wellem-Str. 35, 5630 Remscheid

Spectrum 80 K mit Interface 1 und Microdrive und Software: 480.- DM. ☎ 06131/38232

●●● Eprom-Programmierer ●●●
Das komplette Superding. Programmiert 64, 128, 256 u.a., direkt aufstecken. Inklusive Software für nur 69.95 DM bei T. Krüger, Julius-Brecht-Str. 13-15, 2400 Lübeck. Info gratis!

Seikosha GP 100 A Mark II für VB 250.- DM, **ZX LPrint III** Interface für VB 150.- DM, **zusammen für VB 350.- DM** zu verkaufen. Wolfgang Leven, 4250 Bottrop 2, ☎ 02045/83074 ab 18 Uhr

●●● 15 Spitzenprogramme ●●●
für den ZX Spectrum 48 K. Utilities, Anwendungen und Spiele auf einer Cassette für nur 35.- DM. Per Nachnahme zu bestellen bei: Oliver Vornholt, Schuter 5, 4416 Everswinkel 1, ☎ 02582/259

Top?i Opus-Floppy 439.- DM, GP 50 S + Zubeh. 179.- DM, GP 50 S def. 49.- DM, LPRINT III + Kabel 99.- DM, Kempst. Centr. E + Kabel 99.- DM, SPRINT-Recorder 99.- DM, Spectr., nur 169.- DM etc. Software ... ☎ 06383/7490

Suche für ZX 81 serielles Interface RS 232 von Memotech! ☎ 09252/6323

Suche progr. Joystickinterface, dt. Anleitung, Beta-Basic 3.0, Tasword II, Masterfile. Angebote an J. Witkowski, Ernst-Mehlich-Str. 8, 4600 Dortmund 1

Verkaufe Spectrum 48 K Joystick mit Interface, 5 Bücher, 13 Original-Software Progr., VB 350.- DM; Spectrum Plus 48 K, 350.- DM. J. Borchard, Dahlienweg 18, 4010 Hilden, ☎ 02103/61193 ab 17 Uhr

ZX 81 mit Profitastatur, Repeat, Tonbox, 64 K, Highresol. Graphics, Fachliteratur für Basic und Masch.-Code; alles zusammen VB 250.- DM. Hans Adler, Schweinhöfe 27, 4300 Essen 13, ☎ 0201/550583

ULA oder kaputter Spectrum gesucht. Günter Varga, Hönigst 111, A-8301 Lasznlitzhoehe. Meldet Euch!

Sinclair QL, RGB-Monitor (auch für andere Computer geeignet), Basic-Compiler, Bücher, 25 Cartridges mit Software, komplett 1000.- DM, einzeln VH. ☎ 06721/12421

48 K-Spectrum, IF1, MDrive, Masterfile. Tasword, Leonardo-Grafik, MCTest-Tool, Schachbuch, Masterchess, Spechess, The Turk, Superchess 3.5, ein paar kleinere Programme, Kabel, Handbücher + Literatur: 444.- DM. König, Rüschausweg 67, 4400 Münster

●●● Achtung Spectrum-User! ●●●
Tausche Software für Spectrum 16-48 K. Suche User, die mit Discovery 720 K Erfahrung haben. Hajo Kamphausen, Postf. 1133, 4053 Jüchen 2, ☎ 02164/7610 nach 19 Uhr

● Neue Tasword 2 - Druckeroutine ●
Druckt einen Text mit jedem Zeichensatz (8x4 bis 8x8-Matrix) aus. Programm + 20 Zeichensätze: 15.- DM. ☎ 05545/1592

Verkaufe: 48K Spectrum in Org. Verp. + Discovery (neu) für 600.- DM. Epromer: Unicorn 256 + Softw. 120.-, diverse Bücher bitte anfragen! Suche: Discovery-User zwecks Erfahrungsaustausch! ☎ 07321/63879

Wußten

Sie eigentlich schon, daß Sie die Fähigkeiten Ihres Spectrums nur zu einem Bruchteil ausnutzen? NEIN??? Gerade deswegen haben wir vom Spectrum-Profi-Club ein **aktuelles Sonderheft** entwickelt, das die **besten Artikel** aus unseren sämtlichen, bisher erschienenen Clubmagazinen enthält. Der Inhalt wurde optimal zusammengestellt, so daß möglichst viele Anwendungsbereiche aufgenommen werden konnten. Von Interfacebauplänen über Listings für ernsthafte Anwendungen bis zum Poke, der den Spectrum in die ewigen, technischen Jagdgründe versetzt, ist wirklich fast alles vertreten. Dieses selten- und inhaltsstarke "Spectrum-Profi-Sonderheft" kann gegen 10.- DM Selbstkostenherstellung bei uns angefordert werden: (bar, Scheck oder per NN, dann jedoch gegen 15.- DM) Spectrum-Profi-Club, Lärchenstr. 2, 8091 Maitenbeth. Als kleine Geschmackssprobe einige Ausschnitte aus dem einmaligen Inhalt: ● Bauplan einer Centronics- und Kempstonschnittstelle mit passender Software ● Auflistung von ROM-Routinen, Systempokes und Spielepokes ● des weiteren jede Menge Basic- und MC-Listings ● Tips und Tricks für jede Lebenslage ● Computerkurse ● Erfahrungsberichte ● Tests und so weiter...

Tasword 2 mit Grafik

Programm Tasword-Grafik ermöglicht Schrift und Grafik gemischt in einer Zeile, ohne jedoch zusätzlichen Speicherplatz zu belegen. INFO mit frankiertem Rückumschlag anfordern: U. Trutter, Rosenstr. 1d, 7590 Achern oder ☎ 07841/25809

●●● Achtung Lehrer!! ●●●

ZX Spectrum Arbeitserleichterung durch Notenprogramm: verwaltet Noten-/Namenslisten. Max. 10 Klassen zu max. 40 Schüler mit max. 50 Einzelnoten/Schüler; erstellt Notentabelle; druckt Formulare; erstellt Leistungskurven; deutscher Zeichensatz; schnell durch MC; in der Praxis erprobt.... INFO mit frankiertem Rückumschlag anfordern: U. Trutter, Rosenstr. 1d, 7590 Achern oder ☎ 07841/25809

Billige Interfaces 1 + Drives gesucht! Kaufe auch defekte IF 1. ☎ 07321/63879, öfters probieren!

ZX Sinclair Spectrum Videointerface für alle monochr. Monitoren, hardwaremäßige inverse Darstellung (regelbar), Reset-Taste nur an den User-Port anstecken, Preis 70.- DM inkl. P+V. ☎ 07146/5580 ab 18 Uhr.

●●● VERKAUFE KOMPLETT ●●●
Spectrum 48K + Kempston-Maus + Kempston Centronics-IF + Plotter-Schreibemasch. Silverreed (4 Farb) + Joystick + IF + Software: Art Studio, Tasword 2, Planets, Laser Basic, Winter Games, 3D Chess + Datenrecorder, alles 3 Mon. alt, orig. Verp. NP 1600.- DM für 1200.- DM. ☎ 06021/69274

● Kaufe ● Verkaufe ● Tausche ●
Programme für Spectrum und Atari (nur Cassette). Listen an/bel: Marcus Czybulka, im Mellsig 10, 6000 Frankfurt, ☎ 069/520549

Ich programmiere alle EPROM-Typen für Spectrum, QL, ATARI ST und alle anderen Computer. Anfragen unter ☎ 0911/406896 ab 18 Uhr, D. Appel

Der SM AC wendet sich an alle Spectrum-User, die ernsthaft mit ihrem Gerät arbeiten. Kosten! Info + Heft: D. Kompaß, Waldstr. 70, 5200 Siegburg

Suche CK-Hefte bis Oktober 1985 und GP 250 X Drucker (Seikosha). ☎ 040/7966337, Dierr Reuter, Jochenhofwinkel 5, 2104 Hamburg 92

ZX Spectrum Video Interface, anpassungsfähig (regelbar), inverse Darstellung über Hardware Resetaste. Nur an den Userport stecken. Preis 70.- DM, inkl. Porto + Verpackung. ☎ 07146/5580 ab 18 Uhr

●●● Sinclair Spectrum ●●●

Drucker Seikosha GP 50 S + 2 Rollen Papier + 2000 Einzelblatt Mehrfarb + 2 Farbbänder 145.- DM. CUB-Color Farbmonitor mit Sinclairstecker 698.- DM. Beides jeweils mit Softwarepaket. ☎ 0221/5905074

Super-Angebot

Currah µ-Speech, Interface 1, Microdrive, 24 Cartridges, für nur 400.- DM zu verkaufen (zusammen, auch einzeln)! ☎ 06226/3206 (ab 18 Uhr)

Spectrum 48: Schnelles SAVE und LOAD bis zu 6000 Baud, nur 20.- DM; 2-Passav./Disav., MDV/Disk komp., 100 Zeilen/sec., Trace, alle inoff. Bef., nur 40.- DM. Bei M. Stramm, Rüttscher Str. 155/1513, 5100 Aachen

●●● ZX Spectrum ●●●

Verkaufe meine Originale: Scrabble, The Chess Player, Chess, OCP Schach, Trans-Express, Strip-Poker, Survival, Ghostbusters, Chequered-Flag, Horracegoes Skiing, Pimania, The Hobbit je 25.- DM, nur 1 x vorhanden. H. D. Gansert, Waschgasse 13, 6714 Weisenheim/Sd.

★ Business-Software ★ ★ Commodore - Sinclair ★ ★ QL-Software & Zubehör ★

PC-FIBU	ab 199.-
PC-SM Business Paket	598.-
C 64 FIBU	ab 98.-
C 64 Kunden-/Lief. Buchh.	99.-
Vizawrite-Vizastar-Multiplan	
SP-Tasword III	69.50
SP-Masterfile/MF-Print	69.-
SP-Statistik II	49.-
SP-Pascal/C-Compiler	je 98.-
Kempston-E-Interface	149.-

Katalog DM 3.-
(Bitte Rechner angeben)

Fa. Lucker/CK
R.-Wagner-Straße 71
6239 Kriftel

Atari

***** SCHWEIZ *****

Suche Atari-ST-Besitzer! Bitte sendet mir eine Postkarte mit Eurer Adresse. Gegenleistung: Liste der Atari-Besitzer, Infoaustausch, Postkarte an: Stephan Athanas, Haslistr. 16, CH-5630 Muri

*** NEU *** NEU ***

Atari ST-Freaks! Public Domain + Contactclub Gründung. Näheres unter ☎ 07321/63879 oder R. Frank, Carl-Spitzweg-Str. 17, 7920 Heidenheim 5, Rückporto 80 Pf.

Super Lotto-Programm, geschrieben auf Atari 520 ST+. Mit VEW-Systemen, eigene Ziehung, Gewinn-Tabellen, kurz gesagt, ein Programm mit vielen Möglichkeiten. 50.- DM oder 40.- sFr, ein-senden an D. Steiger, Wittlingerstr. 164, CH-4058 Basel

Telefonuhr für 800 XL/130 XE

Laufende Anzeige von Gebühren und Sprechzeit während des Gesprächs/Mitprotokollierung in Rechnungsdatei möglich/Abrechnung der Gesamtkosten für einen selbst zu best. Zeitraum/ auf Disk 19.80 DM + Porto bei: M. Roland, Postfach 5026, 4902 Bad Salzungen, ☎ 18-21 h 052 22 / 7 05 05

Hallo Briefmarkensammler! Atari-Freaks!!! Mit diesem Super-Programm verkaufen/tauschen Sie Ihre Briefmarken garantiert. 15.- DM an Luciano Lichlus, Dollbrunnerstr. 10, 7530 Pforzheim-Wülm. P.S. Die ersten 10 Einsender erhalten 50 Briefm. gratis.

Atari: Gute Spiele, garantiert keine Raubkop., sehr günstig zu verk. Liste gegen 80 Pf. Rückporto bei Gernot Ebert, Austraße 15, 8587 Creussen. Suche Tips zu Chimera! ☎ 092 70 / 13 62

Suche preiswerte 1050-Floppy. H. G. Schmittlein, ☎ 024 35 / 17 06

Verkaufe Atari: 1027-Drucker gegen Gebot. ☎ 0228/364538, Andreas verlangen!

***** STAUBKILLER*****

Abdeckhauben für Atari 600/800 XL/1050, C 64, 1541, Klarsicht 14.95 DM. Kunstleder 19.95 DM, VK oder NN. Bei M. Kühn, Ulmenstr. 16, 4100 Duisburg 17, ☎ 021 36 / 3 19 77, nach 17.00 Uhr. Bei Vorkasse keine Gebühr!

Atari 800XL mit 1050 (nur kpl.) mit Büchern 550.- DM zu verk. Zubehör auf Anfrage zu verk. ☎ 07321/63879, öfters probieren!

** Atari 400/800/XL/XE **

Utility-Disk, randvoll mit Utilities, Grafik & Sound in Basic und MC, mit Beschreibung! Spiele-Disk, ca. 30 Spiele! Jede Disk (auch auf Cass.) für nur 20.- DM. Schein(e) an: R. Evertz, Venloer Str. 76, 5042 Pulheim 1. Bitte Datenträger (D/C) angeben!

800XL ●● 924 Programme ●● wegen Systemwechsel zu Schleuderpreisen zu verkaufen. Nur Disk. Gratisliste anfordern! Karl-Heinz Hanemann, Casa Graziella, CH-8518 Gorduno, Schweiz

Verkaufe Orig.: Disk/Cass./Modul. ☎ 092 51/5329 - Stefan Bayer, Vrchowstr. 3, 8660 Münchenberg

Atari XL-XE: Verkaufe Software ab 50 Pf. Liste gleich bei Michael Schäfer, Paßstraße 47, 4250 Bottrop anfordern. Verkaufe auch Listings ab 0.50 DM.

*** Atari 800 XL D/C ***

Tausche und verkaufe Supersoftware-Liste an/bel Alexander Spoden, Max-Planck-Str. 5, 8660 Münchenberg. ☎ 092 51/80260, 19.00-20.00 Uhr. Verkaufe Superdisk (DD), randvoll mit Spielen für 20.- DM / 3 Disks für 50.- DM. Schein bitte beilegen.

Verkaufe ATARI Diskdrive 810 inkl. DOS 2.5. 1a Zustand: 500.- DM (+ Porto). Drucker SEIKOSHA GP-250X, Centronics Interf. (für ATARI + Commodore). Endlospapier, graphikfähig, 4 Schriftgrößen: 550.- DM (+ Porto). Anfragen: Marco J. Meyer, Kreuzmattweg 15, CH-4144 Arlesheim/Schweiz

*** Hallo Freaks! ***

Verkaufe Atari 800 XL, Recorder, 2 Spielbücher, Software (zum Bsp. Bruce Lee one on one soccer etc.). Preis VS, ☎ 062 02 / 34 30

ATARI 260 ST: 1 Megabyte Aufrüstung, steckbare Platine ohne Lötarbeiten, Epromsteckkarte und User-Port-Karte. Für 800 XL: Oldrunnertarte mit 800/800 XL OS schaltbar. Für 400/800: OS Rev. B und es laufen alle Programme. Happy 1050 und Highchip, Basic XL/XE, Action! ☎ 02 02 / 40 28 14

Verkaufe ATARI 800 XL + 1010, Programme, diverse Bücher, Stick, VB 500.- DM. ☎ 02129/50105

Atari-XL/XE-Software! Verkaufe wegen Systemwechsel meine kompletten Programme zu 30-70% des NP. Beispiels: Print Shop, Typesetter, Ultima 3, Arcade Mach. Nur Originale, keine Raubkopien! Liste anfordern bei: M. J. Meyer, Kreuzmattweg 15, CH-4144 Arlesheim/Schweiz ● ATARI!

Atari 800 XL: Suche Programme aller Art (Disk). R. Möller, Klefnerweg 16, 6946 Gornheimertal

Verkaufe Computerzeitungen, drucke ATARI-Listings. Listen gegen 2.50 DM in Briefmarken von H. W. Hermann, Mohrweg 21, 5628 Heiligenhaus

● Lohn- und Einkommensteuer ●

Super Jahresausgleich mit Analyse, jährl. Aktualisierung ohne Neukauf, Cass. 60.- DM, Diskette 70.- DM für Atari. Info gegen Rückporto von H. Iichen, Niederfelder Str. 44, 8072 München, ☎ 084 59 / 16 69

ATARI 400, 800, XL, XE

Interf. für Fischertechn. computing; Schaltinterf. 3-8 Kan.; Schaltuhr; I/O-Karte GE/8A; Alarmanlage; Sondert. 12 Tasten; A/O-Wandler 8 Bit - alles im Geh. inkl. Software - Info bei K. Huber, Kreuzstr. 6, 3563 Dautphetal 5

Suche Tauschpartner für Software aller Art! Schickt eure Listen an: F.-J. Kolbeck, Waaterlaage 11, 2842 Kroege

The Magic Card für Floppy 1050. Backup jeder Disk durch HWA und 7.0 184 K pro Seite, bis 18fache Geschwindigkeit beim Lesen und Schreiben, Happy- und Axlon-Kompatibel, nachrüstbar mit Track/Density Display, Controller und Happy-Archiver. Suche Hardware. Christian Wende, ☎ 02 02 / 40 28 14

*** Logo-Software präsentiert *** SuperUTLs 1 + 2 mit 80 Zeichen, Progr.-Schützer, ... Außerdem von LOGO-Software: Programm-Lister, Mathe-Star, ... Spottpreise! Info anfordern: CH, Kanja, Mainzer Straße 11, 6117 Schaathelm, ☎ 060 73 / 8 09 81

VOGEL Computer- bücher

Neue Bücher zum Schneider CPC

Wagenknecht, Fred
Experimentelle 2D- und 3D-Animation
Der PC in Grafik, Werbung und Design
180 Seiten, 48 Abb.
35.- DM
ISBN 3-8023-0877-8

Hier wird dem Leser das notwendige Know-how zur grafischen Anwendungsprogrammierung vermittelt. Das Buch ist im weitesten Sinne eine Anleitung für spannendes, schöpferisches Experimentieren mit konkreten Aufgaben. Angesprochen sind alle, die sich beruflich mit zeichnerischen, gestalterischen und werblichen Dingen befassen. Die Programme sind auf dem Schneider CPC konzipiert und dennoch auf jedem Rechner mit hochauflösender Grafik einsetzbar. Die entsprechenden Hinweise zur Programmanpassung liefert dieses Buch.

Aschoff, Martin
Hardware-Erweiterungen für Z80-Rechner
Selbstgeplant und selbstgebaut
236 Seiten, 120 Abb., 38.- DM, ISBN 3-8023-0849-2

Das Buch beginnt mit einer Einführung in die Hardware von Z80-Rechnern am Beispiel des Schneider CPC 464 und verdeutlicht den grundlegenden Aufbau von Computern. Es beschreibt die Verfahren der Datenein- und -ausgabe und stellt Standardschaltungen zur Steuerung von elektrischen Verbrauchern zu vielen Selbstbauprojekten vor.

Weitere starke Titel:

Baumgart, Harald
Höhere Mathematik auf dem CPC 464
Programme mit Erklärungen des Lösungsweges und Beispielen
192 Seiten, 33.- DM, ISBN 3-8023-0856-5

Vine, Jeremy
Start in die Künstliche Intelligenz mit dem Schneider CPC 464
Eine Einführung in BASIC mit Dialogprogrammen
104 Seiten, 41 Abb., 23.- DM, ISBN 3-8023-0863-8

Gregory, Jim
Superspiele und Utilities für CPC 464 und 664
Eine Fundgrube für BASIC- und Spielefreaks
224 Seiten, 25 Abb.
33.- DM
ISBN 3-8023-0870-0

Sie erhalten bei Ihrem Buch- und Computerfachhändler kostenlos das neue Verzeichnis „VOGEL-Computerbücher“ mit ca. 100 Titeln.

VOGEL-BUCHVERLAG
WÜRZBURG

VOGEL - Computerbücher
mehr wissen, mehr leisten

Neu: Die externe 80-Zeichen-Karte ist da!

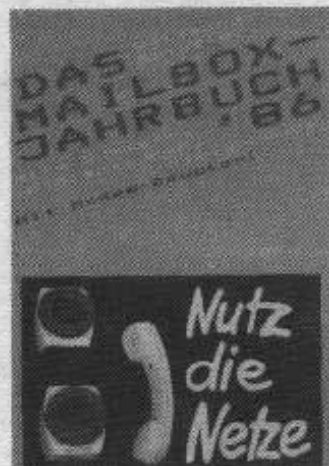
Verlag Rätz-Eberle GdH
Postfach 16 40
7518 Bretten



Das Mailbox-Jahrbuch '86

Von Wolfgang Spindler
Eichborn Verlag
130 Seiten, 10.- DM
ISBN 3-8218-1043-2

Der Eichborn Verlag schwimmt mit dem zweiten Mailbox-Jahrbuch voll auf der



im Moment äußerst stark wogenden DFÜ-Welle mit. Wolfgang Spindler schildert hier in Kurzporträts den Aufbau, Inhalt, den Betreiber und andere wichtige Daten von über hundert der interessantesten Mailboxen im In- und Ausland. Neben einer Einführung über die Entstehungsgeschichte der Mailboxen übersetzt er das Fachchinesisch der DFÜler in allgemeinverständliche Sprache. Für den Vollblut-Hacker wird es erst im nächsten Abschnitt interessant. Hier werden Mailboxen nach Postleitzahlen alphabetisch und nach den verwendeten Rechnertypen vorgestellt.

Wer des Hackens via Telefonleitung oder seiner enorm gestiegenen Telefonrechnung überdrüssig ist, hat hier jedoch die Möglichkeit, einiges über Datex-P (z.B. wie man damit umgeht und was es kostet) zu erfahren. Enthalten ist auch ein

kompletter Bauplan für ein 300-1200/75-Baud-Modem, nebst Stückliste und Schaltplan, das man aus tausend Gründen (von denen mir im Moment keiner einfallen will) nicht ans Postnetz anschließen darf! Den Anschluß des Buches bildet ein umfangreiches Stichwortverzeichnis.

Alles in allem ein Buch, das für den geradezu lächerlich erscheinenden Preis von 10.- DM eine Fülle von Informationen rund um die Szene bietet. Prädikat "Wertvoll!"

Schick in Schale

Schick in Schale geschmissen hat sich der "neue" alte C64, der jetzt mit neuem Gehäuse und einem zusätzlichen, von Disk ladbaren, Betriebssystem ausgeliefert wird. Erfreulich ist, daß der Preis ungefähr gleich bleiben wird. Der Preis für das GEOS-Betriebssystem, das auch einzeln erhältlich sein soll, war trotz mehrmaliger Anfragen bei Commodore nicht zu erhalten.

Das GEOS-Betriebssystem verleiht dem C64 eine dem Amiga ähnliche Benutzeroberfläche mit Pull-Down-Menüs und Softwareanwahl per Joystick oder Maus. Ferner werden ein anspruchsvolles Text- und ein vielseitiges Grafikprogramm namens GEOWrite und GEOPaint gleich mitgeliefert.

Frank Wölk



C64 im neuen Gewand

SpeedDos kompatibel

Ab sofort werden alle C64/C128-Druckerinterfaces vom Typ 92000/G in einer Version ausgeliefert, die vollständig kompatibel ist mit High-Speed Betriebssystemen wie SpeedDOS und Geos. Ältere Interfaces dieses Typs können für 50.- DM in die neue Version umgerüstet werden.

Info:
Wiesemann Mikrocomputertechnik
Winchenbachstraße 3-5
5600 Wuppertal 2

Traum-Interface!

Wiesemann Computertechnik, vielen Anwendern als Lieferant hochwertiger Interfaces bekannt, startet eine große Umfrageaktion zum Thema Interface: Gesucht werden Vorschläge zum "Trauminterface". Jeder interessierte Leser ist aufgefordert, eine möglichst genaue Beschreibung seines "Trauminterfaces" an die Firma Wiesemann einzusenden. Die Beschreibung sollte enthalten, welche Funktionen man sich wünscht, wieviel Pufferkapazität vorhanden sein müßte und wie das optimale Interface sonst noch aussehen könnte.

Als besonderen Anreiz wird ein Preisnachlaß von 50.- DM für jeden ausgeschrieben, der eine ernsthafte Darstellung ein-sendet.

Wiesemann Mikrocomputertechnik
Winchenbachstraße 3-5
5600 Wuppertal 2

Inserentenverzeichnis

albs-Altronic	S. 120
Belkenheid	S. 30
C & M Meyer	S. 32
CSV Riegert	S. 3
Dörr	S. 118
Dreeser	S. 30
EZS	S. 3
Holschuh	S. 6
Hüthig-Verlag	S. 7
Individual Software	S. 25
Jäckel&Klintworth	S. 9
Jeposoft	S. 5
Jupitersoft	S. 3/19
Kunz	S. 19
Klemmer&Schulte	S. 47
Lücker	S. 118
Merkens	S. 6
Naujoks	S. 9
Padercomp	S. 99
Rausch&Haub	S. 101
Schleißbauer	S. 65/121
tewi-Verlag	S. 2
Toonen	S. 6
T. S. Datensysteme	S. 124
Unicom	S. 23/45/65
Vogel-Verlag	S. 119
Wagner	S. 6

Impressum

Verlag Rätz-Eberle GdbR
Pforzheimer Straße 43
7518 Bretten
Tel. 07252/42948

Verantwortlich für Text und Anzeigen:
Thomas Eberle, Dipl.-Wirt.-Ing. (FH)

Computer-Kontakt erscheint zweimonatlich am letzten Montag des Vormonats und kostet pro Heft 5,50 DM. Anzeigenschluß ist am 5. des Vormonats. Zur Zeit gilt die Anzeigenpreisliste Nr. 3.

Eine Haftung für die Richtigkeit der Veröffentlichungen kann trotz sorgfältiger Prüfung durch die Redaktion nicht übernommen werden.

Die gewerbliche Nutzung, insbesondere der Schaltpläne und Programme, ist nur mit schriftlicher Genehmigung des Herausgebers zulässig.

Grafik + Satz: Druckerei Sprenger,
7143 Vaihingen/Enz

Druck: Verlags- und Industriedrucke
GmbH + Co KG
7730 Villingen-Schwenningen

Titelbild: Rainer Grinda

Vertrieb: Verlagsunion
Friedrich-Bergius-Str. 20, 6200 Wiesbaden



Compy - Shop Ohg



Inhaber: J. Jähn und Peter Bee, 4330 Mülheim Ruhr – Gneisenastraße 29, Telefon 02 08 / 49 71 69

Der Atari Spezialist

**Wir liefern die folgende Soft- und Hardware
für Ihren Atari 600 – 800 XL und 130 XE:**

Die Compy-Shop Speichererweiterungen:

Atari 600 XL von 16K RAM auf 64K RAM – ohne RAMdisk	98.– DM
Atari 600 XL von 16K RAM auf 192K RAM – mit 128K RAMdisk	198.– DM
Atari 800 XL von 64K RAM auf 320K RAM – mit 256K RAMdisk	248.– DM
Atari 130 XE von 128K RAM auf 192K RAM – mit 128K RAMdisk	98.– DM

Alle Speichererweiterungen von 64K sind kompatibel zum 130 XE und werden mit einem DOS für die RAMdisk geliefert!

Die Oldrunnerkarten:

Oldrunnerkarte – für XL/XE-Computer – Speicher bei \$C000 frei	98.– DM
Oldrunnerkarte mit 80-Zeichenkarte – für XL/XE-Computer	148.– DM
Oldrunner mit 4K-BIBOMON – für XL/XE-Computer	178.– DM
Oldrunner mit 80-Zeichen u. 4K-BIBOMON – für XL/XE-Computer	198.– DM

Der 16K-Bibomon:

Maschinenmonitor mit eingebautem DOS, Druckerschnittstelle, Hardcopy und Hardwareuhr	448.– DM
--	----------

•• Siehe Testbericht Computer Kontakt ••

Programmiersprachen:

kyan Pascal, für 1 Laufwerk, liefert echten Objectcode	298.– DM
•• Testbericht CK 6/7 '86 ••	
Basic XL, von OSS, eines der schnellsten Basic	228.– DM
Basic XE, von OSS, speziell für den 130 XE	248.– DM
Action!, von OSS, superschnelle Compilersprache	248.– DM
Mac/65, von OSS, Makroassembler	298.– DM
Atmas II, Makroassembler	49.– DM
Atmas II, Toolbox zum Makroassembler	19.80 DM

Software:

Design Master, Das Mini-CAD-Programm	19.80 DM
Die Soundmaschine, Super!	29.80 DM
Monitor XL, Masch.-Monitor für's Basic	19.80 DM
Pyramidos, deutsches Action-Adventure	39.– DM

Mike's Slotmaschine, die beste Slotmaschine!	29.– DM
--	---------

Bücher:

Die Hexenküche, Tips und Tricks	29.80 DM
Hexenküche, Diskette zum Buch	19.80 DM
Das Atari Power Buch	29.– DM
Der Programmieren, Spielebeschreibungen	29.– DM
Das Atari Profi Buch, mit Schaltplänen des 800 XL	59.– DM

Drucker:

Atari 1029, mit Grafik-Druckprogramm!	498.– DM
•• Druckerprogramm für 1029 einzeln ••	39.– DM
Okidata Microline 182, anschlussfertig	1178.– DM
Star NL 10, anschlussfertig	1336.– DM

und vieles mehr! Fordern Sie unsere Preisliste an!



Pascal Software Atari XL/XE

**kyan pascal für die
XL/XE Serie DM 298.–**

kyan pascal ist ein mit DOS 2.5 arbeitender Compiler für die Atari XL/XE Serie. Es umfaßt den vollen Jensen-Wirth Standard und eignet sich für Anfänger genauso wie für den fortgeschrittenen Programmierer. Es ist blitzschnell und hat folgende Eigenschaften:

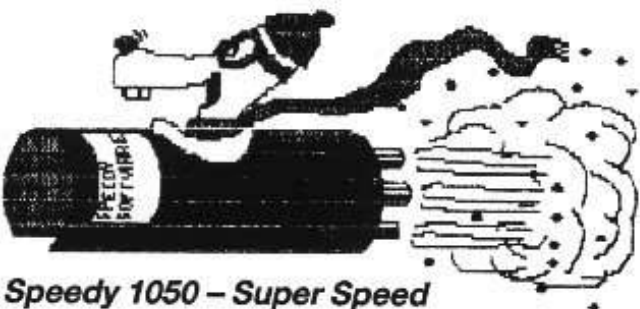
- 6502 Maschinencode-Compiler erlaubt das Einbinden von Assemblersource
- Bildschirmeditor, Stringbefehle, Atari-Grafik und Soundunterstützung
- Source Code Linking, Chaining und Random-Files
- 13-stellige Floatingpoint Genauigkeit, mit Tutorial/Referenzunterlagen

kyan pascal wird auf einer ungeschützten Single Density Diskette geliefert und benötigt nur 48 k Speicherplatz und ein Laufwerk.

kyan pascal für Ihren Atari Computer bei Ihrem Fachhändler oder direkt bei uns.

Speedy 1050

(c) 1986 Compy-Shop



Speedy 1050 – Super Speed

SPEEDY 1050 N, normale Ausführung 198.– DM

SPEEDY 1050 NE, Entwicklungssystem, mit allen technischen Unterlagen 248.– DM

SPEEDY 1050 T, mit digitaler Trackanzeige 298.– DM

Die Vorteile der SPEEDY 1050 auf einen Blick

- Drei Speicherformate: 88 KByte – Single Density
127 KByte – Medium Density
176 KByte – Double Density
- Drei Arbeitsgeschwindigkeiten:
Normale 1050-Geschwindigkeit
normale SPEEDY-1050-Geschwindigkeit
SPEEDY 1050 – SUPER SPEED
- SPEEDY 1050 ist voll programmierbar
- Formatiert jedes der drei Formate in 9 Sekunden
- Leichter Einbau, von jedem Laien ausführbar (SPEEDY N)
- Benutzerfreundliche Software
- Arbeitet mit dem 16 K BIBOMON zusammen
- Kann kopiergeschützte Software kopieren
- Digitale Trackanzeige (SPEEDY T)
- Akustischer Fehlermelder (SPEEDY T)

SPEEDY 1050 TE, Entwicklungssystem, mit allen technischen Unterlagen 348.– DM

EINBAU einer SPEEDY 1050 NE/TE in Ihr Laufwerk, z.zg. zum Kaufpreis 50.– DM

Fordern Sie unseren ausführlichen Prospekt an!

**Ihre Bestellungen richten Sie bitte an den Compy-Shop.
Händleranfragen bitte an die Firma Witt Datentechnik. Telefon 0 52 01 / 40 06**

Das sind Preise!!!

DISCOVERY 180
ein Laufwerk
180K Speicherplatz
DM 399,-

DISCOVERY 1430
Zwei Laufwerke
1.4M Speicherplatz
DM 1199,-

DISCOVERY 360
Zwei Laufwerke
360K Speicherplatz
DM 599,-

DISCOVERY PLUS 18
Einbauset mit 120-
sat-Laufwerk, 180K
DM 199,-

DISCOVERY 720
ein Laufwerk
720K Speicherplatz
DM 799,-

DISCOVERY PLUS 72
Einbauset mit 120-
sat-Laufwerk, 720K
DM 399,-

**BIS ZU 1,4 MBYTE
FÜR JEDEN SPECTRUM**

Die neuen Discovery-Systeme mit Centronics-Drucker-Interface - Joystick-Interface (Kempston-Typ) - Video-Monitor-Ausgang - stabilisiertem Netzteil (versorgt auch Ihren Spectrum) - Befehlsatz voll Microdrive kompatibel - belegt keines RAM im Spectrum - deutsches Handbuch - 1/2 Jahr voll. Garantie

UNGLAUBLICHE PREISE

(eulange Vorkasse) für Spectrum

- Computer-Hits 10 Teil 1 ... 19,90
- Computer-Hits 10 Teil 2 ... 22,90
- Mounty on the Run ... 14,90
- They sold a Million ... 22,90
- Winter Games ... 19,90
- Surf Champ (inkl. Surfboard) ... 14,90
- Zorro (Originalspiel) ... 17,90
- William Wobbler (70phie) ... 15,90
- Fighting Warrior ... 19,90
- Ring-Pong ... 19,90
- Night-Shadow (von Ultimate) ... 29,90

SUPER TRACKBALL

Extrem schwere Ausführung. Sofort anschließbar an jeden C-64, VC 20, Atari, Schneider Über Trackball Interface (49,90) auch an jeden Spectrum **DM 69,90**

SAMANTHA FOX Strip-Poker

Bei diesem neuen und wirklich aufregenden 7 Card Stud Poker müssen Sie schon Karten behalten, Ihre Spielblüten kann Samantha auch für Schneider 48K/128K und für Schneider (Disk) 29,90 für Schneider (Disk) 39,90 für C-64 (Disk) 39,90

Drucker-Interface
Kempston Typ E-Interface mit Software im ROM, d.h. es wird kein Speicherplatz belegt. Routinen für alle gängigen Drucker-typen bereits implementiert. COPY- und Vierfachcopy möglich.
Phantastischer Preis ... 179,90

MULTIFACE ONE

Endlich ist das Problem der Sicherungskopien gelöst. Mit Multiface können Sie jedes Programm, das Sie in Ihren Spectrum speichern, auf Band, Microdrive-Cartridge oder Floppydisk abspielen. Daneben ist eingebaut Joystick-Interface, Composit Video-Interface, 8KB RAM, den Sie auch Register) auf jedes am Spectrum anschließbare Medium. Natürlich haben Sie 6 Monate Garantie

Spectrum + 333,-

Spectrum 48K 248,-

Utilities für den Spectrum

HISOFT-Pascal mit deutscher Anleitung ... 99,90
HISOFT-Basic-Compiler COLT mit deutscher Anleitung ... 59,90
HISOFT-C-Compiler, die Programmiersprache der Zukunft ... 99,90
HISOFT-Dewpac Assembler und Disassembler im Paket ... 59,50
Beta Basic 3.0 (deutsche Version) auch für Discovery ... 59,90

Kempston PRO
Das ist ein universelles Joystick-Interface für den Spectrum. Es gestattet eine vollständige Kontrolle über alle Joysticks, die an den Spectrum angeschlossen sind. Auch mit ROM 300 tot

Brandneue Spectrum Software

Combat Lynx	29,90	Schulze	29,90
Flyer 77	19,90	Green Beret	29,90
The Great Fire of London	29,90	Heavy on the Magic	29,90
Three Works in Progress	29,90	ACT Studio	29,90
Yabba Dabba Do	29,90	Verdon	29,90
Spill 40	29,90	Barrie at the Plains	29,90
Supersuits	29,90	Turbo Sprint	29,90
Critical Mass	29,90	Superpolic II	29,90
Death Walk	29,90	Edman	29,90
Pod Pod	29,90	Conan Capers	29,90
Fighting Warrior	29,90	Commando (Originalspiel)	29,90
Rambo (Superdisk)	29,90	Caution	29,90
Confusion	29,90	The Young Line	29,90
Snake Eye	29,90	Lower Computer	29,90
Back to School	29,90	1488 Basic	29,90
Stargate	29,90	The Art of Kung Fu	29,90
Aladdin	29,90	Phong Rang	29,90
Falcon (Disk II)	29,90	Basketball	29,90
Super Bow	29,90	Tai Chi	29,90
Satanus	29,90	Blade Runner	29,90
2112 AN	29,90	The Rocky Horror Show	29,90
Gungfink	29,90	Juggernaut	29,90
Twister	29,90	Monie	29,90
Pyramide	29,90	Horus Focus	29,90
		Mentor	29,90

DISKETTES 3 1/2"

Markenqualität mit Garantie
auch doppelseitig verwendbar. 6,99

QL-Match Point (3.25-Termin) 89,90
QL-Cross (Schachprogramm: neue Version) 89,90
Spock
Zapper
APL (Keyboard) 79,90
B.J. in 3D Land 49,90
Superchance (Compiler) 49,90
A Word 49,90
Key Down 49,90
Assembler 79,90
Stack Control 49,90
Superform - Reviewer V1.3 119,90
Motoric Short
Integrated A/Ca (Gatesoft) 49,90
Macro Assembler (Schneider) 299,90
Slog Astrologer
G Draw
Compendium (Disk) 199,90
BCL
ICF Compendium
HISOFT Mon-OL
HISOFT Superbasic
English Trainer
Lands of Harco
Sensible
QL-Sprite Generator
Daggers
Knight Flight
Area Radar Controller
West
QL-Jabber
Quadrado
QL-Ren
Bridge-Player II
Star Guard & Galactic Invader 499,90
Night Nurse
Kempston Doctor
Kempston Diskinterface (Standard Bus)
Laufwerk 3 1/2" kompakt mit Netron (ROM) 399,90

SINCLAIR QL

QL-Match Point (3.25-Termin) 89,90
QL-Cross (Schachprogramm: neue Version) 89,90
Spock
Zapper
APL (Keyboard) 79,90
B.J. in 3D Land 49,90
Superchance (Compiler) 49,90
A Word 49,90
Key Down 49,90
Assembler 79,90
Stack Control 49,90
Superform - Reviewer V1.3 119,90
Motoric Short
Integrated A/Ca (Gatesoft) 49,90
Macro Assembler (Schneider) 299,90
Slog Astrologer
G Draw
Compendium (Disk) 199,90
BCL
ICF Compendium
HISOFT Mon-OL
HISOFT Superbasic
English Trainer
Lands of Harco
Sensible
QL-Sprite Generator
Daggers
Knight Flight
Area Radar Controller
West
QL-Jabber
Quadrado
QL-Ren
Bridge-Player II
Star Guard & Galactic Invader 499,90
Night Nurse
Kempston Doctor
Kempston Diskinterface (Standard Bus)
Laufwerk 3 1/2" kompakt mit Netron (ROM) 399,90

Schneider Software

Law on Magic (IC)	35,90	Gradator (IC)	35,90
Law on Magic (D)	45,90	Gradator (D)	45,90
Amphawa (IC)	35,90	Room Ten (IC)	29,90
Amphawa (D)	35,90	Room Ten (D)	35,90
Bridge Player III (IC)	35,90	Winter Games (IC)	35,90
Bridge Player III (D)	35,90	Winter Games (D)	35,90
Trulle Walle (IC)	9,90	Green Beret (IC)	35,90
Trulle Walle (D)	9,90	Green Beret (D)	35,90
Killspede (IC)	9,90	Toad Runner (IC)	35,90
Killspede (D)	9,90	Toad Runner (D)	35,90
Nuclear Heat (IC)	9,90	Movie (IC)	35,90
Nuclear Heat (D)	9,90	Movie (D)	35,90
Sam Fox Strip Poker (IC)	29,90	Eden Blues (IC)	35,90
Sam Fox Strip Poker (D)	29,90	Eden Blues (D)	35,90
Equinox (IC)	35,90	Batman (IC)	45,90
Equinox (D)	35,90	Batman (D)	45,90
X (IC)	45,90	Shogun (IC)	35,90
X (D)	45,90	Shogun (D)	35,90
Yie Ar Kung Fu (IC)	35,90	Task Commander (IC)	35,90
Yie Ar Kung Fu (D)	35,90	Task Commander (D)	35,90
Bored of the Rings (IC)	45,90	Juggernaut (IC)	35,90
Bored of the Rings (D)	45,90	Juggernaut (D)	35,90
The Rocky Horror Show (IC)	35,90	Zoids (IC)	35,90
The Rocky Horror Show (D)	35,90	Zoids (D)	35,90
The Way of the Tiger (IC)	35,90	Blade Runner (IC)	35,90
The Way of the Tiger (D)	35,90	Blade Runner (D)	35,90

Competition Pro 5000 35,90
Joystick

C-16 Software

Gundling	19,90	ACE Series Trilogy	35,90
Bonjo	19,90	The Snake	25,90
Pogo Pogo	19,90	Knappe	25,90
Flight Path 131	19,90	Bedlam	25,90
Kung Fu Kid	19,90	P.T. Star Events	25,90
Atwoods	19,90	Logan	14,90
Mutter	19,90	Swich	25,90
Watch for the Sky	14,90	Law Tennis	29,90
Sipery Sid		Alfred	

C-64 Software

The Second City (IC)	35,90	Empire (IC)	35,90
The Second City (D)	45,90	Superbow (IC)	29,90
Macadam Bumper (IC)	35,90	Sam Fox Strip Poker (IC)	29,90
Macadam Bumper (D)	45,90	Sam Fox Strip Poker (D)	29,90
Leader Board (IC)	35,90	Ball Blazer (IC)	35,90
Leader Board (D)	45,90	The Planets (IC)	35,90
Knight Games (IC)	35,90	J (IC)	29,90
Knight Games (D)	45,90	Wai Fie Trading Co (IC)	35,90
Speech (IC)	35,90	War Play (IC)	29,90
Speech (D)	45,90	Pyramids (IC)	29,90
Touchdown Football (IC)	35,90	Automats (IC)	19,90
Touchdown Football (D)	45,90	Combat Lynx (IC)	35,90
Graphic Adventure		Spunkin' (IC)	19,90
Creator (D)	69,90	Critical Mass (IC)	35,90
The Way of the Tiger (IC)	35,90	Enigma Force (IC)	35,90
The Way of the Tiger (D)	45,90	A.C.E. (IC)	35,90
Nexus (IC)	35,90	Weekend (IC)	29,90
Ghosts'n Goblins (IC)	35,90	Laser Basic (IC)	35,90
Space Invasion (IC)	35,90	Laser Basic (D)	45,90
Law of the West (IC)	35,90	Laser Compiler (IC)	35,90
Bounce (IC)	35,90	Laser Compiler (D)	35,90
Biggins (IC)	35,90	Everyone's a Wally (IC)	29,90
Caution II (IC)	35,90	Herbert's Dummy	
Saboteur (IC)	29,90	Run (IC)	29,90
The Cornet Game (IC)	35,90	Blade Runner (IC)	29,90
Green Beret (IC)	35,90	Rocky Horror Show (IC)	29,90
Green Beret (D)	45,90	Space Shuttle (IC)	29,90
Rupert Ice Castle (IC)	29,90		

Alle Preise enthalten die gesetzliche Mehrwertsteuer. Versand per Nachnahme (zuzüglich 5,90) oder Vorkasse mit Scheck (zuzüglich 2,50). Informationen können angefordert werden. Händleranfragen erwünscht.