

# 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: Hungriger Gull, Puzzler
- ★ Maschinensprachekurs
- ★ ST: GIA-Basic

## TI 99/4A

- ★ TI-Writer, Zwillings
- ★ Monopoly 2, Trill
- ★ Buchstabenmix

**Atari ST**  
Sonderseiten



# ATARI ST

## ASSEMBLER-PRAXIS AUF ATARI ST

ATARI 260ST, ATARI 520ST, ATARI 1040ST

### ASSEMBLER-PRAXIS AUF ATARI ST

Roland Lohr

...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 Systemkonzepte... systembezogene Erläuterung der 68000er Befehlsfunktionen.

#### Zeigt Anwendungen:

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

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

Anwender des Befehlsatzes in Musterprogrammen für: E/A-Routinen, Rekursionen, dez/bin Rechnerarten, Stackverwaltung, Adressverwaltung, Entscheidungen, Schleifenkonstrukte, Unterprogramme, numerierte Traps, Händeln von Interfacebausteinen, Texterkennung, Textverarbeitung, Test- und Dekodierung, memory dumps, Hoppy Tests/Funktionen, serielle RS232C 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 lesefreundlichem Druckbild, guter Bildokumentation und umfangreichen Listings von Musterprogrammen (auf Diskette beim Autor erhältlich). ca. 300 Seiten, Softcover, DM 59,-



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

## Weitere te-wi-Bücher



NEU

#### DAS C-BUCH

Herold, Jung  
Ein „C“ für die Industrie. Für schnelle C-Routinen. Über 100 Beispiele. Anspruchsvoll in Text. Bildmaterial, ca. 300 Seiten, Softcover, DM 79,-

UNIK

Tetris, Thonell US-Standardwerk der INEK Promotoren Votex. Eine anschauliche Übersicht und Einführung in die Anwendung. 160 Seiten, Softcover, DM 79,-



#### LOGO – Jeder kann programmieren (Daniel Witt)

Buch des Jahres in den USA. Best-renommiert von Pädagogen und deutscher Kultusministerien. Ein bildlicher Führer durch u.a. ATARI's LOGO. Von Papert's Schüler D. Witt. 384 Seiten, A4, DM 59,-



**M68000 FAMILIE, 2 Bd.**  
HIL/Neuzeit, ges. 968 Seiten  
Einige 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 Programmiersystem zur Simulation eigener Problemformulierungen und 29 kommentierte Modellbeispiele von Baumsterben, Heizungsbedarf, Nahrungsnetzen usw. Prospekt anfordern. Von Hartmut Büssel. 480 Seiten, Softcover, DM 59,-



**Mein ATARI Computer**  
Best-motiviertes Standardwerk deutscher ATARI User Groups. Kompatibel ATARI 100/500 System. Für plebeisches Lesetum. Von Tim Rawles. Cook, 500 Seiten, Softcover, DM 99,-

**Sprechende Ideen mit ATARI GRAPHIK**  
E-Schüler Lohrstoff in Geometrie und Farben. Lehrer eines amerikanischen Lehrstuhls mit ATARI Graphicsmöglichkeiten. Von Tim Rawles. 226 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

Höttingenstr. 34b  
8636 Ellingen  
Tel. (0 91 41) 61 74

## NEU FÜR C 64 zum Einführungspreis

**Super-Eprom-Brenner "BURNY"**  
Brennt bis 27512, 28-er Reihe, EPROMs,  
bis PROMs, PROMs, insgesamt 28 verschiedene  
Typen, inkl. Software auf Disk **DM 199,-**

**Super-Eprom-Karte "ESB"**  
Beitrag K, Autostart, Menüführung **DM 99,99**

**Flippy-Speicher "QUICK-DOO"**  
Laut 250 Blocks in ca. 3 Sek., Speed-DOO  
Kompatibel, inkl. Kopier-Disk **DM 149,-**  
Erstbau-Service **DM 25,-**

**"POWER-CARTRIDGE"**  
Das Super Modul **DM 199,-**

Alle Artikel inkl. aust. deutscher Anleitung,  
versand nutzige, Porto u. Verpackung.

## E EDV-Zubehör-Studio

Z Werbuckstraße 2-4  
S 4100 Durlauburg **DM 149,-**  
Telefon 02 03 / 40 21 96

## Brandheiße Krüllerpreise

### TI-99/4 A

CPIC 99 Peripherie mit 1-Datenträger  
Anschluß 220 - 10 Datenträger **199,-**  
Erstbau 224-Quadratmeter **199,-**  
Erstbau 224-Quadratmeter + Controller **279,-**  
Erstbau 224-Quadratmeter + Plotter **279,-**  
Mini-Memory (orig. TI) **169,-**  
Hochauflösende 224-Karte mit 256  
Größenklassen Superaktion - 01 Tag **179,-**  
- Controller - Speicher **179,-**  
Speichererweiterer **179,-**  
Fernbedienung (orig. TI) **89,-**  
Apparat, Original, Tisch, Original **919,-**  
Defektor, Dig. Tag, Car. Mem. Blasen Mem. **919,-**  
Munch Men, Invention **919,-**  
Buck Rogers, Chess, Stamina **919,-**  
• **Preisvergleich mit Hard- und Software**  
• **Währungs**

### Schneider

CPIC 99 mit Supermodule **899,-**  
CPIC 99 mit Partnermodul **1549,-**  
Speichererweiterer LX 99, anstandslos **979,-**  
Speichererweiterer FX 99, anstandslos **1299,-**

### Atari

512 K 179,- 130 K 209,- 1000 309,-  
512 K 1050 599,- 130 K 1050 729,-  
Speichererweiterer LX 99 + Speichererweiterer **1299,-**  
Speichererweiterer FX 99 + Speichererweiterer **1419,-**

### Commodore

Commodore AMIGA 1000 **3249,-**  
239-K-Erweiterer für AMIGA **299,-**  
Commodore 128 + Speicher VC 1811 **1299,-**  
Commodore C 128 G **1299,-**  
Partnermodul 1801 mit RGB-Modul **829,-**  
Partnermodul Commodore 1801 **579,-**  
Speichererweiterer LX 99 + Speichererweiterer **999,-**  
MAGS + Speichererweiterer **1899,-**  
Plotter - FX 99 **1899,-**  
Plotter - SX 99 **1799,-**  
Speichererweiterer LX 99, anstandslos **749,-**  
Speichererweiterer FX 99, anstandslos **899,-**  
Commodore-Drucker MP3 99 **399,-**  
Commodore Plot 4 200,- VC 1801 329,-  
Plot 4 + Flippy-Disk VC 1801 **599,-**  
Anstandslos Datenträger 0 21 G **999,-**  
• Kabel + Terminprogramm **299,-**

Versandkostenzuschüsse (Nennwert bis DM  
1000,-) werden: Volkswagen (DM 8,-), Nach-  
nahme DM 1,20/0,20, Amstel (DM 16,-), 20,-  
Lieferung nur gegen Vorkasse oder 100 %  
Anzahlung für Vorkasse. Positive Community  
angibt gegen Zahlung eines Preiszuschlags.

## CSV RIEGEL

Schulhofstr. 5  
7324 Reichenhausen  
Tel. (0 71 1) 5 28 89

Die nächste Ausgabe  
»Computer-Kontakt«  
erscheint am 24.11.1986

## Preisausschreiben

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 Phillips VG 8010 (MSX) mit Z 80-Prozessor (3,57 MHz), 48 K RAM, davon 32 K RAM 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	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Chip	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Homecomputer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Compute mit	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Happy Computer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
64er	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Run	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Computronic	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ST	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Ti-Revue	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Data-Welt	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

## 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, Inseratenverzeichnis, Impressum	122

### USER-CLUB SINCLAIR

ZX-Spectrum Maschinensprachkurs	10
Assemblierips für den Spectrum	13
Spielpokas	16
Jahrmärkt	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 B1-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 Spielkiste	65
Peters Assemblerdecke	66
Der hungrige Golf	68
Slow Motion	70
Kurs: 6502 Maschinensprache, Teil 2	72
Spiel zum Abtippen: Puzzle	74
Karteiverwaltung	77
Diac-Collector	79
Das MIDI-Disk-Programm für den 800XL	82
MicroMon	87

### ATARI ST

3D Flying Ace	91
Textomat ST	94
GIA-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
Buchstabennix	106
Schlüssel	110
Katalog für Maschinenspracheprogramme	115

## Topprogramm 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 C 64 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 C 64, 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 Einreichung 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 480.00**

Vorführung und Selbstabholung der Ware nur nach tel. Vereinbarung

● evtl. Leasing (Mietkauf)

Abdeckhaube für QL **DM 34.90**

Abdeckhaube für Spectrum **DM 19.90**

### MICRODRIVE CARTRIDGE STORAGE BOX

1984/86, 20 Cartridges

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

Cartridge-Box mit 20 Cartridges **DM 159.90**

Multiface One, neue Version **DM 159.00**

Multiform-Diskettenbox für 3"- und 3 1/4"-Disketten **DM 24.90**

Brandneu: Spectrum-Software

Der Stern des Druden (deutsches Adventure) **DM 64.20**

lieferbar ab ca. Mitte Oktober

### JEPOSOF

Kruppstraße 9, 4040 Heuss 21

Telefon: 0211 9781 54

Öffnungszeiten: Di - So 10:00 - 18:00 Uhr

Alle Preise in DM, inkl. MwSt. zzgl. Versandkosten

## 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 Abende 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

Vorkasse 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.)

Daten/Unterschrift .....

Dieses 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 1.00 DM in Briefmarken anfordern bei:

THOMAS WAGNER  
SOFTWAREVERSAND

T. WAGNER SOFTWAREVERSAND P.O. BOX 112243, D-6980 HÖRSBURG 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 gut 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 Computereclub  
Ralf Meers  
Riegstraße 17 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  
Dillenburgener 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 "Problemberatungsservice" enthalten ist, beträgt 10.- DM im Vierteljahr. Weitere Infos bei

A.U.C.C.  
Hilferweg 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  
6812 Hockenheim  
Tel. 0 6205 / 1 42 10

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 Bröstrup  
Tannenweg 18  
4544 Ludbergen

## 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. Pokojni 11 a 29  
62-200 Czystażowa  
Polen

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

Andreas Baro  
Brüdermannweg 44  
2000 Hamburg 61

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

Rainer Schmidt  
Köpingstr. 9  
6953 Guntersburg 1

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

Wolf Bartha  
Altenberndstraße 3  
4950 Deutschl

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

Bern Nowotka  
Am Fuchsbau 25  
61012 Pfungstadt

Ich suche Kontakt zu einer Atari-Club in Stuttgart oder Leoben.

Matthias Wunder  
Lehenbühlstraße 54  
7253 Renningen

## ★ RTS ★

T199/4A und MSX-Software

★ T199/4A Software ★	
Sommerspiele .....	19.- DM
Doppel-Pack I (Stumpfschmeißer)	20.- DM
Doppel-Pack II (807/Main)	25.- DM
Sammelbox (4 Programme)	25.- DM
Carthy (Adventure Disk)	25.- DM
★ MSX Software ★	
Nador .....	25.- DM
Quasid .....	29.- DM
Ritter .....	29.- DM
Coorn, die Mission .....	34.- DM
Gorbu .....	29.- DM

Infos bei: Roland Toonen Software, Pl. 31, 4178 Kevelaar 1

## DATENKASSETTEN

Deutsches Markenland alle Größen von C0-C50 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 520/280  
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 10 Stück  
3,25 Diskettenkopier auf Anfrage.

## DISKETTEN



Holechuh Tapes  
Kellertstr. 97, 01601 Beishem  
Tel. 0 62 51 / 6 26 60

## EXTENDED ROM

EXTENDED ROM V 3.5 16 BIT SPECTRUM 128 K, 64 Kbytes, 6400 & 3201 bytes.

- 28000-Programme über MSX
- 28000-Funktionen (Benutzerschnittstelle)
- Totaltest
- Programm zurück ins BASIC
- 40000-App-Memoriesortier
- Registerfunktion
- Benutzer
- Memorie von 0-Drive Befehle
- 28000er Schnittstelle
- 100% kompatibel zum Original MSX
- Spezialisierte Entwicklungssoftware für den MSX-Transfer 12800 Bytes
- ausführliche deutsche Anleitung
- MSX-Software 12800 Bytes

PREIS: 800 Speicher mit Schaltung nur 45,- (einsch. 20% MwSt.)  
Bestellungen & Befrag: Roland Toonen, Sillertstr. 31, 4168 Ravenswaaijen  
Tel.: 01187/93961 ab 18.30 Uhr

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

Michael Bösw  
Dielings 454  
4955 Steenweide 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.

Hermann Algner  
Markt 187  
A-3324 Eurenfeld

Ich suche einen User-Club für den QL.

Spamer  
Arnselweg 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, Spielröpske usw.. Der Beitrag beträgt im 1. Jahr 30.- DM (25.- DM Clubbeitrag und 5.- DM Aufnahmegebühr).

Club der 64er Fächer  
Georg Wenk jun.  
Waldhofweg 7  
8240 Brechingen

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)  
öffnen von 20:07 Uhr  
Tel. 04 51/3 16 42  
300 Raad, 8 Datenbits, 1 Stopbit, 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 Vorauskasse 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 einwerfen in den Verlag Rast-Thele, Postfach 1640, 7518 Bretten.

# Verlangen Sie mehr!

### Datenverarbeitung mit dem Sinclair QL



### Rainer Fischer Datenverarbeitung mit dem Sinclair QL

1986, 196 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 Datenverwaltungsprache von QL-ARCHIVE ein. Der Leser erhält dabei wichtige Hinweise für die Erstellung wohlstrukturierter Programme.

### HAJO LENCKE



### Hajo Lencke Assembler mit dem C-64

1985, 227 S., 22 Abb., 6 Tab., kart.,  
DM 35,-  
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.

### Cracker, Hacker, Datensammler



### Thomas Tai Cracker, Hacker, Datensammler

1985, 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 Schritt, die in einer A1 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 Spielfreunde 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

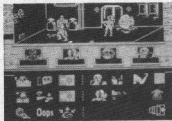
Rolf Koers

## 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 gegliückten 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 Syyik abstürzen. Dort führt Syyiks 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 raumtütiges 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 Labyrinth 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 Syyik gesteuert wird.

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

ausgestattet 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 Nachfolgeprogramm 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 Azgade- 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 Cassetteneleger 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 Kriegskräfte 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 Zeittunnele dar. Der Spieler muß sie nach und nach aufsuchen, um eine bestimmte Spielbene zu erreichen, wo er unterschiedliche Aufgaben zu lösen hat.

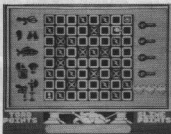
Die Spielfigur ist eine der in der Anleitung beschriebenen

Kampfkrieten, 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 unwerfend, 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 Kaorn



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 eingesandte Programme zugänglich macht, alle zwei Monate eine Mitgliederliste veröffentlicht (um Kontakte zwischen den Mitgliedern zu ermöglichen) und schließlich auch versuchen soll, Händlerarbeiten auszuhandeln.

David Tietze  
Nienburger Str. 37  
2000 Bremen

Der Sinclair User Club Sandbach ist im "Computerclub für Sinclair, Commodore und Schneider-Computer" in Reith 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.

Computersklub für Sinclair, Commodore und Schneider-Computer  
Postfach 1109  
8236 Reith im Winkel

Vor ein paar Monaten kaufte ich mir zu meinem Sinclair ZX Spectrum + einen Drucker (Seiko-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  
Neulandstraße 1  
031-8665 Jona

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

Sinclair ZX Spectrum			
Spectrum plus 48 KB	342,-	Spectrum 128 KB deutsch	450,-
Spectrum 48 KB	250,-	Doppelport-Joystick Interface	45,-
Saga 1-Tastatur	120,-	Kompaon-E-Druckerinterface	175,-
Saga 3-Tastatur	342,-	Multiface One (neue Ausf.)	193,-
Saga 2001-Tastatur	390,-	Currish Sprachsynthesizer	65,-
Discovery Disketten	ab 390,-	ZL Light 3 Druckerinterface	175,-
Beta-Disk Controller 5.03	320,-	Beta-Disk Kpl.-Syst. (1986)	ab 749,-
Beta Basic 3.0	93,-	Art Studio Mainprogramm	59,-

Sinclair QL		Drucker	
QL-englische Ausführung	490,-	Seiko-MS 15 Typenaddr.	699,-
QL-deutsche Ausführung	469,-	Seiko-SP 1000 (A/AS/I)	749,-
QL-Diskcontroller	ab 320,-	Seiko-SP 1300 AI	1799,-
QL-Disk-Kpl.-Syst. (1986)	ab 749,-	Citizen 125 D	749,-
QL-Centronics-Interface	ab 160,-	Centronics-QLP II Drucker	899,-
QL-Monitor	ab 240,-	Aufsatzaktor für GLP	99,-

QL-Speichererweiterung 256 KByte mit durchgeführtem Bus	330,-
QL-Speichererweiterung 512 KByte mit durchgeführtem Bus	420,-
QL-Giga-Soft-Mouse-Paket incl. leistungsstarker Software	220,-
CUB-Super-Farbmonitor für QL oder Spectrum 128	1099,-
QL-Super-Super-Diskcontroller + 612 KB + Centronics-Interface	899,-
QL-Exp.-Erweiterungssteckkarten (intellektuell an ROM-Port)	ab 19,-

Microdrive-Cartridges	4 99k. 32,-	12 99k. 99,-	20 99k. 190,-
3.5"-Disketten 5DD	1 99k. 7,90	10 99k. 79,-	50 99k. 390,-
3.5"-Disketten DDD	1 99k. 9,90	10 99k. 99,-	50 99k. 590,-
3.5"-Disketten 2D No Name	1 99k. 8,-	10 99k. 67,-	50 99k. 375,-

Lieber Kunde, diese Preise sind 5 Wochen alt, sie können sich daher inzwischen wesentlich geändert haben.

Alle Preise incl. MwSt. mit Versandkosten zum Selbstkostenpreis. Unsere Gesamtpreise enthalten Sie gegen 2,- DM in Bestellmarken Selbstabholung nur nach vorheriger telefonischer Terminabsprache. Versand nur per Nachnahme oder Vorauszahlung.

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?

## Michael Naujoks

LOAD ZX81 (dt.)	DM 39,-	Light Pen	DM 69,-
Vokaltrainer	DM 39,-	(dt./fr.) mit deutscher Anleitung	DM 69,-
The Art Studio	DM 59,-	3-Kanal Soundsynthesizer	DM 99,-
Beta Basic 3.0	DM 59,-	(dt./fr.) mit deutscher Anleitung und Softwareerweiterung	DM 99,-
Sachbuch		Speechsynthesizer	DM 99,-
Delikatent für Schachpartien mit Daten zum Nachspielen aller 72 Partien des Weltmeisters Kasparov gegen Komput.	DM 49,-	(dt./franco)	DM 99,-
The Colt	DM 58,-	Joystick-Interface	DM 29,-
Schachbild-GAD	DM 39,-	SpeoDrum	DM 149,-
Tasword II	DM 59,-	T-Stecker	DM 25,-
Tasword III	DM 29,-	Laser Basic	DM 49,-
Masterfile	DM 39,-	Laser Compiler	DM 36,-
Heoff C	DM 99,-	Laser Genius	DM 49,-

## Hardware Sinclair Software

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,-	Spittire 40	DM 39,-

Kostenlosen Katalog S/9 anfordern!

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

Hotline:  
(0622) 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 (Programmcounter) 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 Schleißenzähler verwendet wird.

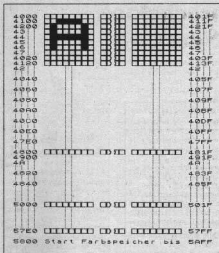
## Der Bildschirmspeicher

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 Farbattributen, wobei jeweils ein Byte die Farbinformation einer PRINT-Stelle (8x8 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 (SCREENS) 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 abgebildete 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 Schieb-Richtung, sondern auch durch das, was mit dem herausfallenden Bit passiert (im obigen Beispiel das 0. Bit) und was die freigeordnete Stelle ersetzt (im obigen Beispiel das 7. Bit). Dadurch wird der Unterschied zwischen Rotieren und Schieben deutlich. Bei erstem 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 S I H P NC	Dauer	Befehl mit Operanden	Flag's S I H P NC	Dauer	Befehl mit Operanden	Flag's S I H P NC	Dauer
ADC A,s	IX X IXI	4-19	INC IV		10	LDD	0 Y0	16
ADC HL,rr1	IX X IXI	15	INB	? ? ? ?	16	LDR	0 00	16&21
ADD A,(HL)	IX X IXI	7	INDB	? ? ? ?	16&21	LDI	0 Y0	16
ADD A,(IX+n)	IX X IXI	19	INL	? ? ? ?	16	LDIR	0 00	16&21
ADD A,(IY+n)	IX X IXI	19	INR	? ? ? ?	16&21	NES	IX I XIX	8
ADD A,n	IX X IXI	7	JP c1,nn		10	NOP		4
ADD A,r	IX X IXI	4	JP an		10	OR s	IX 0 IX0	4-19
ADD HL,rr1	X 0I	11	JP (HL)		4	OTDR	? ? ? ?	16&21
ADD IX,rr2	X 0I	15	JP (IX)		8	OTIR	? ? ? ?	16&21
ADD IY,rr2	X 0I	15	JP (IY)		8	OUT (C),r		12
AND s	IX I 100	4-19	JR c2,n		7/12	OUT (n),A		11
BIT b,(HL)	? I ? 0	12	JR a		12	OUTD	? ? ? ?	16
BIT b,(IX+n)	? I ? 0	20	LD rr1,(nn)		10	OUTI	? ? ? ?	16
BIT b,(IY+n)	? I ? 0	20	LD rr1,nn		10	POP rr4		10
BIT b,r	? I ? 0	8	LD r,n		7	POP IX		14
CALL c1,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	IX I XIX	4-19	LD (HL),n		10	PUSH IY		15
CPD	IY I YI	16	LD (HL),r		7	RES n,s		8-23
CPDR	IY I YI	16&21	LD r,(IX+n)		19	RET		10
CPJ	IY I YI	16	LD r,(IY+n)		19	RET c1		5/11
CPJR	IY I YI	16&21	LD (IX+n),n		19	RETI		14
CPL	I I	4	LD (IY+n),n		19	RETN		14
DAA	IX I I X	4	LD (IX+n),r		19	RL a	IX 0 IXI	8-23
DEC a	IX I XI	4-23	LD (IY+n),r		19	RLA	0 0X	4
DEC rr1		6	LD A,(na)		13	RLCA	0 0X	4
DEC IX		10	LD (nn),A		13	RLC r	IX 0 IXI	8
DEC IY		10	LD (nn),rr1		20	RLC (HL)	IX 0 IXI	15
DI		4	LD (nn),HL		16	RLC (IX+n)	IX 0 IXI	23
DJNZ n		8/13	LD (nn),IX		20	RLC (IY+n)	IX 0 IXI	23
EI		4	LD (nn),IY		20	RLD	IX 0 IXI	18
EX AF,AF'	IXIXIXIX	4	LD A,(BC)		7	RR a	IX 0 IXI	8-23
EX DE,HL		4	LD A,(DE)		7	RRA	0 0X	4
EX (SP),HL		19	LD A,I	IX 0 Y0	9	RRC a	IX 0 IXI	8-23
EX (SP),IX		23	LD I,A		9	RRCa	0 0X	4
EX (SP),IY		23	LD A,R	IX 0 Y0	9	RDD	IX 0 IXI	18
EXI		4	LD HL,(nn)		16	RST n		11
HALT		4	LD IX,nn		14	SBC A,s	IX I XIX	4-19
IN0		8	LD IX,(nn)		20	SBC HL,rr1	IX I XIX	15
INI		8	LD IY,an		14	SCF	0 0I	4
IN2		8	LD IY,(nn)		20	SET n,s		8-23
IN r,(C)	IX I 10	12	LD R,A		9	SLA a	IX 0 IXI	8-23
IN A,(n)		11	LD r,(HL)		7	SRA a	IX 0 IXI	8-23
INC a	IX I 10	4-23	LD SP,HL		6	SRL a	IX 0 IXI	8-23
INC rr1		6	LD SP,IX		10	SUB A,s	IX I XIX	4-19
INC IX		10	LD SP,IY		10	SDB s	IX 0 IX0	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

c1 = NZ, Z, NC, C, PO, PE, P, M

c2 = NZ, Z, NC, C

r' = Zweitregistersatz

### Kürzel bei den Flagbits

1 = Nach Befehlende auf 1

0 = Nach Befehlende 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

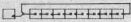
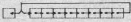
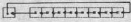
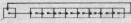
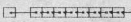
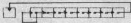
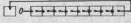
- 1. Wert bis 2. Wert möglich

& = 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 Bildschirmpunkadresse vereinfacht.

T. Bernoldo

C-Flag + Bearbeitetes Byte	Befehl	Genaue Bezeichnung
	RLC ... RLCA	Rotiere links zirkular
	RRC ... RRC A	Rotiere rechts zirkular
	RR ... RLA	Rotiere links
	RR ... RRA	Rotiere rechts
	S LA ...	Schiebe links arithm.
	S RA ...	Schiebe rechts arithm.
	S RL ...	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 Maschinencode-routinen 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 getreteten Startadresse und der programmrelativen Adresse (= Abstand zur Startadresse) der Daten gebildet wird. Sofern der Datenbereich klein ist, mag es auch genügen, am Programmumfang 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 un schön. 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 JP's/CALLs und für zeikritische 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 programmeneigenen 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 Register, 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 = $1681

```

```

ORG 38888 ;Startadr. beliebig

START LD HL, MEMBOT ;In 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 entsprechender Sprung.

UP1 LD A, 2 ;Strom 2 öffnen
CALL OPEN ;(Ausgabe auf den
LD A, "A" ;Bildschirm) und
RST $10 ;ein paar Buchstabe-
LD A, "B" ;n 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 ;PKEten" 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
HELPCB = 23728

```

```

ORG 40888 ; 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 UCALL
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
PUSH BC ;Aufruf der
PUSH DE ;BEEP-Routine
PUSH HL ;in ROM.
LD DE,100 ;DE bestimmt die
LD HL,200 ;Tonlänge,
CALL #03B5 ;HL die
POP HL ;Tonfrequenz.
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 Spielertips 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, 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, TAKEPAPER, 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=USR23296: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 "lgh590x" 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-codc=NG8INI94, Code-Nr. für alle Boxgegner:

Canadian Crusher	STC	10CIII010
Fling-Long-Chop	STC	1S6IIM0
Andra Pancheredov	STC	1MAIIIK10
Tribal Trouble	STC	104IIIS10
Frenchie	STC	1K8IIHQ10



Ravioli Mafiosi      STC    INBIII0I0  
 Antipodean Andy    STC    IR5IIII0I0  
 Peter Perfect        STC    IL9IIIKI0

**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""SCREENS: 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 Stamina

**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 &**  
**PIPE**                 das Schweißgerät.

**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. wegnehmen 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 Skintcode  
 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 "WRITETYPEN" 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  
 Daten 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  
49759,X  
50084,201  
50206,0  
53567,0

X=Anzahl der Leben  
X=Anzahl der Objekte sammeln  
unbegrenzte Zeit und nicht Werwolf  
unendlich Tage  
Unsterblichkeit

### Lazy Jones

56693,0

### Lunar Jetman

36966,244:36945,3  
37999,201  
36965,3  
36965,0  
36964,244:36965,3  
43092,X-1

Unsterblichkeit  
keine Feinde  
Unsterblichkeit  
unendlich Leben  
unendlich Leben  
(X max. 99) Stufe

### Manic Miner

35136,0

33799,X-1

Man kann auch nach dem Programmstart die Zahlen 6031769 eingeben. Es gibt ein System für die Wahl des Screens. Umwandlung der Nummer des gewünschten Screens minus 1 ins binäre Zahlensystem und die Tasten drücken, bei denen binär eine "1" steht. Dabei 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  
36301,201  
38004,0  
35874,255

Unsterblichkeit  
keine Feinde  
unendlich Leben

### Moon Alert

42249,24  
42585,2:52596,2  
37035,201  
39754,0  
39756,0  
42404,0  
42404,X  
42654,195

unbegrenzte Zeit  
Screen rollt schnell  
keine Feinde  
Unsterblichkeit  
Unsterblichkeit  
Unsterblichkeit  
X=153-99 Autos am Start

### Mr. Wimpy

33693,0  
33509,X  
33721,X  
43105,0  
33501,0

Unsterblichkeit  
X=Anzahl der Leben  
X=Anzahl der Schüsse  
unendlich viele Schüsse  
1. Bild wird weggelassen

### Mugsy

Auf die Frage: "How much You wish to spend on artillery and ammo?" Sofort 999999999 ein tippen = 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  
46457,0  
44416,X

Unsterblichkeit  
Unsterblichkeit  
X=Anzahl der Leben

### Pogo

44259,0:44260,33

### Push off

26862,167

### Raid over Moscow

10 CLEAR 65535:POKE 65535,195  
20 PRINT AT 9,6; "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 unendlich 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  
45520,255  
39393,0  
39702,30  
43575,0

1. Spieler wird unsterblich  
2. Spieler wird unsterblich  
keine Tiere und Krieger mehr  
keine Monster erscheinen  
unendlich Leben

### Sam Spade

25215,0  
26381,0

Unsterblichkeit  
unendlicher Sauerstoff

### Scuba Dive

55711,X  
55711,255:55779,255

X=Anzahl der Leben

### Sir Lanzelot

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, SONIO, AMAHA, KYZIA, AMIGA, RAMEX. 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=190: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 "tjh590x" 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

**Terrormollnos**

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**

27785,0

Unsterblichkeit

**Underwurdle**

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-ZMOI.3, 8-HRMEZ

**Worse Things happen**

35443,0

Dieter Thesen

**A View to a Kill**

30 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""SCREENS

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

**Swee's World**

POKE 37008,255

für 255 Leben

Daniel Scheidt

**Jupitersoft GdbR**

ZX 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 Weilboldshausen

(0 91 41) 22 49

**ZX-Spectrum - ZX-Spectrum**

DISCOVERY 180 DM 268,-

SAVIO-MONITOR (z. B. I. Discovery) DM 249,-

PANADONIC-Drucker DM 668,-

KEMPTON-G-Drucker- DM 148,-

interface

Telexler (8" Tonics) DM 118,-

8" Tonics Drucker- DM 88,-

interface

40 k-Erweiterung DM 49,-

Joystickinterface DM 29,-

GAMES PLAYER

-Joystickinterface mit Regelung der Spiel-

geschwindigkeit DM 39,-

8" Tonics

SPRACHSYNTHESIZER DM 79,-

3-KANAL-SOUND- DM 79,-

SYNTHESIZER

Liese (K) mit weiteren Angeboten anfordern oder direkt bestellen bei

U. KUNZ

August Hülken 5,

D-7620 Karlsruhe 81

Tel. Bestellannahme

07 21 / 48 28 79 (Sa 30 Uhr)

## Jahrmarkt (Spectrum 48K)

Falls Sie es leid sind, in immer neuen Versionen als Pacman durch ein Labyrinth dösen 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 Volkmann

## 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*XCOS m: LET y=my+
r*XSIN m
141 PLOT x,y: DRAW 0,-7: PLOT x
-7,y-6: 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 rXCOS m-1,
rXSIN 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 (dx/sy/sx)
214 LET ag=2*XCOS wXSIN w/sx
215 LET v=.8165
220 FOR t=(2*VXSIN w/ag)*n/ap T
O 2*VXSIN w/ag STEP (2*VXSIN (w)
/ag)/ap*5
225 PLOT dx+vXCOS wXt,dy+vXSIN
wXt-ag/2XtXt
226 PLOT dx-vXCOS wXt,dy+vXSIN
wXt-ag/2XtXt
227 NEXT t
231 FOR m=0 TO 7 STEP .2: PLOT
89+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: ": IAT 14,24: ": NEXT
m: REM ** LICHTERKETTE **
307 LET x1=195: LET x2=254: LET
y1=0: LET y2=68
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 210,39
,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
R 2: "CODE" AT 19,11: "EGH" AT 18,1
8: "DEC" AT 19,18: "EGH"
```

```
536 PRINT AT 21,0: INK 4: PAPER
21: "a" "a"
```

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

```
609 PRINT AT 13,27: BRIGHT 0: P
APER 6: INK 0: "SCHIE" AT 13,27+N
: BRIGHT 1: OVER 1: " "
```

```
620 POKE 23305,p: LET p=p+27: P
RINT USR 23300: NEXT n
```

```
5999 REM XX ANIMATION XX
```

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

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

```
6020 GO SUB 6000: RANDOMIZE USR
Z
```

```
6030 BEEP 2,40: PRINT "NUN EIN B
ISCHEN 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: F
OR 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 8,18: "JAHRMARKT" "/
" "BITTE 2 MINUTEN WARTEN BIS ALL
E 5 SEQUENZEN GEZEICHNET SIND !
": PAUSE 250
```

```
8005 DATA 33,0,119,17,0,64,1,0,2
7,237,176,201,0,1,3,7,15,31,63,1
27,128,192,224,240,248,252,254,2
55,0,0,60,0,60,0,60,0,0,127,113,
113,113,123,96,0,0,254,254,254,2
54,198,238,0,0,37,41,49,49,41,37
,0,0,221,81,221,221,68,93,0,0,22
0,16,220,220,80,220,0
```

```
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 Danost

## 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 c=1 TO 100
```

```
40 READ e: IF e<-2 THEN GO TO
75
```

```
50 READ f
```

```
60 PLOT a+e,b+f
```

```
70 NEXT c
```

```
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)?" i a$: IF
a$="j" OR a$="n" 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,
8,5,8,6,0,6,1,6,2,6,3,6,4,6,5,6,
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,
5,3,6,4,7,5,7,6,7,7,2,2,3,1,4,
0,5,0,6,0,7,0,-3
```

```
100 DATA 0,0,0,1,1,2,1,3,1,4,2,
5,3,5,4,5,5,5,6,5,2,1,3,8,4,8,5,
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
```

101 DATA 0,0,1,1,2,1,3,2,4,3,5,  
3,6,4,6,5,6,6,5,7,4,7,3,6,2,5,2,  
4,2,3,4,1,5,0,6,0,7,1,-3

102 DATA 0,0,1,1,2,2,3,3,4,3,5,  
4,5,5,6,6,6,7,6,8,5,9,4,10,3,10,  
2,9,2,8,3,7,3,6,3,5,3,4,3,3,1,  
3,0,3,-1,3,-2,4,0,-3

103 DATA 0,2,0,3,1,4,1,5,2,6,3,  
7,4,7,5,7,2,5,2,4,2,3,3,2,4,2,5,  
3,6,4,6,5,6,6,6,3,6,2,6,0,6,-1,6,  
-2,5,-2,4,-1,5,0,7,1,-3

104 DATA 0,0,1,1,2,2,3,3,3,4,4,  
5,4,6,4,7,4,8,3,9,2,10,1,9,8,8,0,  
7,1,6,1,5,1,4,2,3,3,2,3,1,3,0,4,  
3,5,4,6,4,7,3,7,2,7,1,7,0,-3

105 DATA 0,0,0,1,1,2,2,3,3,3,3,  
4,3,5,3,6,3,2,4,1,5,0,6,1,3,8,3,  
9,-3

106 DATA 1,1,2,2,3,3,3,4,4,5,5,  
6,5,5,4,5,3,5,2,5,1,5,8,5,-1,5,  
-2,4,-2,3,-1,3,0,4,1,6,2,7,3,5,  
0,5,9,-3

107 DATA 0,0,1,1,2,2,3,3,4,4,4,  
5,4,6,4,7,3,0,3,9,2,10,1,9,8,8,0,  
7,1,6,1,5,2,4,4,2,4,1,4,0,5,3,6,  
4,7,4,7,3,6,2,7,1,7,0,-3

108 DATA 0,0,1,1,2,2,3,3,4,4,5,  
5,6,6,6,7,6,8,6,9,5,10,4,10,3,9,  
2,8,2,7,2,6,2,5,2,4,3,2,4,1,5,0,  
-3

109 DATA 0,5,0,6,1,7,2,6,2,5,2,  
4,2,3,2,2,2,1,2,0,3,7,4,6,4,5,4,  
4,3,4,2,4,1,4,0,5,7,6,6,6,5,6,  
4,6,3,6,2,6,1,6,0,7,1,-3

110 DATA 0,3,0,4,0,5,1,6,2,7,3,  
6,3,5,3,4,3,3,3,2,3,1,3,0,4,7,5,  
7,6,6,6,5,6,4,6,3,6,2,6,1,6,0,7,  
1,-3

111 DATA 0,2,0,3,0,4,1,5,2,6,3,  
7,4,7,5,6,6,5,7,4,7,3,7,2,6,1,5,  
0,4,0,3,0,2,0,1,1,6,6,7,7,-3

112 DATA 0,3,1,4,2,5,3,6,3,5,3,  
4,3,3,3,2,3,1,3,0,3,-1,3,-2,4,5,  
5,6,6,5,6,4,6,3,6,2,6,1,6,0,7,1,  
-3

113 DATA 0,2,1,3,1,4,1,5,2,6,3,  
7,4,7,5,7,6,6,6,5,6,4,1,2,2,1,3,  
1,4,1,6,3,6,2,6,1,6,0,6,-1,6,-2,  
7,2,-3

114 DATA 0,3,1,4,1,5,2,6,3,7,4,  
6,4,5,4,4,4,3,4,2,4,1,4,0,5,6,5,  
5,6,5,7,6,-3

115 DATA 0,0,1,1,1,2,2,3,2,4,3,  
5,4,6,5,7,6,6,6,5,7,4,7,3,7,2,7,  
1,6,0,5,0,4,1,-3

116 DATA 0,0,1,1,1,2,2,3,3,4,3,  
5,4,6,4,7,4,8,4,9,4,10,4,5,4,4,4

,3,4,2,4,1,5,0,6,1,7,2,7,3,2,8,3,  
7,5,7,6,7,-3

117 DATA 0,0,0,1,1,2,1,3,2,4,2,  
5,2,6,2,7,2,2,2,1,3,0,4,0,5,0,6,  
1,6,2,6,3,6,4,6,5,6,6,6,7,1,-3

118 DATA 0,6,1,7,2,7,2,6,2,5,2,  
4,2,3,2,2,2,1,3,0,4,0,5,1,6,2,6,  
3,6,4,6,5,6,6,7,7,-3

119 DATA 0,6,1,7,2,6,2,5,2,4,2,  
3,2,2,2,1,3,0,4,1,4,2,4,3,4,4,4,  
5,4,6,4,0,5,0,6,1,6,2,6,3,6,4,6,  
5,6,6,6,7,7,6,-3

120 DATA 0,0,1,1,2,2,3,3,4,4,5,  
5,6,6,1,7,1,6,1,5,2,4,4,2,5,1,6,  
0,7,1,-3

121 DATA 0,1,0,2,1,3,1,4,2,5,2,  
6,2,7,2,3,2,2,3,1,4,1,5,1,6,2,6,  
3,6,4,6,5,6,6,6,7,6,1,6,0,6,-1,6,  
-2,5,-2,4,-1,5,0,7,1,-3

122 DATA 0,2,0,3,1,4,1,5,2,6,3,  
7,4,7,5,7,6,7,7,7,6,6,5,5,4,4,  
3,3,2,2,1,3,0,4,0,5,0,6,0,7,1,-3

## Robin of Sherwood

Eines muß direkt zu Anfang gesagt werden: Mit "Robin of Sherwood" erhält man ein Text-adventure der absoluten Spitzenklasse. Nicht nur die besondere Technik bei der Texteingabe, sondern auch die Grafik, die einen immer wieder überrascht, machen den Reiz dieses Spieles aus. Der Schwierigkeitsgrad ist selbstverständlich hoch, wie man es von einem Programm dieser Güte auch erwarten kann. D.h., man ist damit einige Wochen oder vielleicht sogar Monate beschäftigt.

Schon beim Anfangsbild muß eine harte Nuß geknackt werden. Robin befindet sich in ei-

nem Verlies im Schloß des "King of Sherwood", und ein Entkommen scheint fast unmöglich. Hat man dieses Problem einmal gelöst und gelingt es einem, aus dem Schloß zu fliehen, kommt man in das Reich von Robin Hood, den Sherwood Forest. Hier erstellt man sich am besten einen Plan, denn sonst verirrt man sich sehr schnell. Im Sherwood Forest befindet sich auch einer der Höhepunkte in punkto Grafik: Ein fließender Wasserfall, den man gesehen haben muß. Fazit: "Robin of Sherwood" gehört zu den besten Adventures, die es für den Spectrum gibt.

System: ZX Spectrum  
48/128 kByte  
Hersteller: Adv. International  
Preis: ca. 43.-DM  
Bezugsquelle: H. G. Dresser



Typische  
englische  
Landschaft

# 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 78-seitigen Handbuch und einer Cassette beim Spectrum und einem 120-seitigen 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 Kopierstil entfernen. Außerdem sollte im Loader ein RS 232-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 Dypvac. Da hätte man Besseres bieten können. Zu allem Überflüssig 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 Textbeginnt 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 will, 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 Word LENGTH) überflüssig. Sie muß entfernt werden. Auch der Editor hat einige kleine Fehler. Das Kommando P(t) zum Sichern einer Programmquelle sichert unabhängig von den angegebenen Zeilenummern 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 PRINTFILES 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. Q. ab 240,-  
 GL. dt. 128K 444,- / 512K 689,-  
 LA 1PB NEC mit CST-Cont. 777,-  
 Sandy, Erweiterung, Dubehör a. A.  
 unicorns soft  
 3500 Kassel  
 K. Königstr. 46A  
 T. 0561/77037

## 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 unbesetzten 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 Datentyp 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"

stehen zwei Zahlen. Die erste gibt die Anzahl der freien Sektoren insgesamt, die zweite die Anzahl der unbesetzten 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 Dateizeichnung 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 Datenfile mit CHR\$(0, 1, 2 oder 3) beginnt 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
01cd481711e5010e00616922ae5c22a
c5c22aa5cdd*c901cd*93014469736b6
57474653a20f21e9015e235623ed53
a85cdd*b101cd*bb01cd*93014e616d6
52020202020202054797020204c6e676
5202020416472205072676c6effcd*bb
013aae5cc61032ae5c200f11e5012aa
a5c2322aa5c"
```

```
70 LET A%="A%+*4fcd*c901af21e5
010600f4097e23234e2346235e235634
ca*390123c5f5cd*b101f1c1626bed42
65f2b2b2b2b2b2e52aa85c622a85c
606937ed52ed5bac5c1922ac5c606911
e501d50e07cd*c901dde13e20d721*d
601dd7e00fe0538023e045782a06004
f090603cd*b301e1cd*9d01dd6e03dd6
6047aa72"
```

```
80 LET A%="A%+*01fcb7c280906063
e20d710fb1803cd*9d01dd6e03dd6606
cd*9d013e0dd7c3c66007afe0420f5fe
03200210ec3e20d73e20d77c15200bd6
40d7cd*93012029f118da660073c241
8f06069cd*be01cd*930146726569652
053656b746f726563afed5ba05c37e
d52eb2aac5c19cd*9d01ebcd*9d013e0
dd7ed7b3"
```

```
90 LET A%="A%+*d5cc3761bc54ffe8
03000cbb93e14d73e01d779fe203006c
bf13e1318f179d7cd*930114001300ff
c1c9e37e23e3f4efcd0710f6c0819019
1c5d5e33e205fd701f0d8cd2a19c3301
a060a7e23cd*6e0110f9c9c00dd70620
3e2d710fb3e0dd7c9cd08173e018002
cd0000c348175052474e4d4243485242
5954441154"
```

## Musterausdruck

Diskette: M. Schramm

Name	Typ	Lnge	Adr	Prgln
cat.s	PRG	2236		2236
array1	CHR	003	X%(1)	
array2	NMB	505	A()	
druck.c	BYT	620	32768	
Druck	PRG	1482	10	1399
Daten	DAT	130		

Freie Sektoren: 607 676



```

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 PRINTUSR
";A,"bzw. LPRINTUSR ";A
250 STOP
9000 SAVE #1,"Cat" LINE 10
9010 VERIFY #1,"Cat"
9020 STOP
9500 SAVE #1,"Cat.c"CODE A,405
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! Jede ISO-ROM verfügt nun über:

- Reset ohne Programmverlust! • Deutsche Umschrift!
- Neuer Zeichenstift! • Verbesserter CAT für MDI!
- Verkürzte Syntax für Microdrive oder Beta-Disk!
- Erweiterter Editor mit Cursor/Tabis Programmzeilen!
- Variable Zeichenbreite: 20, 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! Unabhängig 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" ausgefüllt, so daß nicht jedes Programm 40 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, Ocus Disk, Beta-Disk und Cassette (für alle anderen Speichermedien!)

Fördern 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-Soundmoduls. Synthesizer mit 7 vorgegebenen Instrumenten. Natürlich für bequeme Noteneingabe und Korrektur. Jetzt 48,- 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 dem Spectrum.

- 3 Steckplätze für 27128-er EPROMs!

• Umschaltbar Sinclair/next, ROM, abwärtsfrei, 3-fach Schalter für die ext. ROMs und preiswerte Reset/NAV-Taste!

• Durchgehender 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 - Schmales Backup mit fast 48 K auf einmal!
  4. BETA-ADDRESS - Findet autom. Anfangsadressen von Codeblöcken!
  - 4 Topoprogramme auf 1 Diskette: nur 60,- DM
- Bitte Laufwerk und Controller angeben (z.B. 3,5"/2x80/4, 12!)

ISO-DAT1 - Endlich ein Dateiprogramm, das den Speicherplatz der Beta-Disk voll ausnutzt! Bis zu 5000 Einträge auf einer Disk, komfortabler Screeneditor, deutsche Umsätze, Programm und Anleitung in deutsch, sehr benutzerfreundlich!  
Jetzt 48,- DM

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

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

ISO-COPY, QUICKSAVE, QUICKSAVE II je 20,- DM/ASS: 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 vorstatten als eine Multiplikation). In der neuen Version wird einfach der Exponent der Zahl in Gleitkomma Darstellung 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 Gleitkomma Darstellung 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 `33D` dafür sorgen,

daß das oberste Stapелеlement gegebenenfalls in Gleitkomma Darstellung überführt wird. Nach jedem Kalkulatoraufruf zeigt HL auf das Exponentenbyte 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 Exponentenbyte den Exponenten plus 128 enthält. Man muß also mit  $128+3=131$  vergleichen, denn 4 wird als  $2^{1/2}$  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 Gleitkommaoperationen eingespart, und das bringt etliche Minuten an Laufzeitverringerung.

Michael Schramm

### Basic-Lader

```
25 DEF FN Z*(X) = "*" + ("0" AND X
<=9) + STR* X
245 PRINT "Bildberechnungszeit
";PEEK 30997;FN Z*(PEEK 30998
);FN Z*(PEEK 30999)";
370 SAVE N% CODE 31000,6144
0020 LET A*="211879110040010010
db0c976af676f22785c327a5cc4811c0
640c5cd791cc110f921bd7622685cfc
002c782c602c102c502c402c20230cda
22432bc76cda22432ba7621000022b87
6efe6c00230c541f420d763abc761f3
```

```
00b21bd7611d176010a00edb021ba767
e237efe0e130efe0310431e13104c30
f3d02301afe033005efe02301019efe30
3e40fe0e1043d30834efe50fc102c0023
021b763520d0efc102c0023021bb767
e230fcb4e2002c6ffed4bb076d444760
c200ded43b076ef7e00fc0"
0030 LET A*="A*"+023010060478fec0
300eed43b076ef0e10fc10230c37a75
210040111079cd36752170sc11b6760e
0376ed00113200d73761e3cc0737632
17791e3cc073763216793ab076321579
ed7b3d5cc37d1be5f570e6c00f0f0fc6
406778e6078d6778e63007876f79e6f6
0f0f0f056f792fe607070707f6c6326f
76cb6c6f1e1c92ab0763aba700600d21
b076d70002dd7001dd70000004cb7920
00cb23cb12cb1110f3ddcb0026ddcb01
16ddcb0216ed52993005dd3400100219
09cb39cb1ac1b10e07dc9"
0040 LET A=3E4: LET S=42110
```

### Beispiel 1

```
RST 020 ;Diese Routine verdoppelt das
DEFB 03D ;Top-Element auf dem Stapel
DEFB 030 ;durch inkrementieren des
INC (HL) ;Exponenten.
RET
```

### Beispiel 2

```
RST 020 ;Diese Routine prüft, ob der
DEFB 03D ;Betrag des Top-Elements
DEFB 002 ;kleiner als 4 ist.
DEFB 030 ;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 eingeflogen 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 anderen finstere Gestalten wollen die Rettung natürlich verhindern.

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

SCORE 3600 ← LIVES 7 TIME 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 Unsterblichkeitspoker. 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 Bosson

### Maxi-Fließkomma

```

1 REM Divisions-Algorithmus
2 REM *****
10 INPUT "Zähler: ";z,"Nenner
: 20 n
30 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
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+1)))↑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 Ausgaberoutine.

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#="dfef2c3e002008cd011
cc941ee6032925c21001022935c262
022955cdfef2c2044cd791cc4941e329
45ca72820f5cd941e32935cc180382bf
e193022dfef2c2024cd791cc4941e329
65ca72807ed7b3d5cc3761bf5cd941e3
2955cc1803804fe213802cf8a3e1bcd1
20f3e41cd120f3e08cd120f3e1bcd120
f3e4bcd120f3e965c0707260017cb14f
dcb5846280317cb14cd120f7ccdd12"
```

```
70 LET A#="dfef2c3e002008cd011
cc941ee6032925c21001022935c262
022955cdfef2c2044cd791cc4941e329
45ca72820f5cd941e32935cc180382bf
e193022dfef2c2024cd791cc4941e329
65ca72807ed7b3d5cc3761bf5cd941e3
2955cc1803804fe213802cf8a3e1bcd1
20f3e41cd120f3e08cd120f3e1bcd120
f3e4bcd120f3e965c0707260017cb14f
dcb5846280317cb14cd120f7ccdd12"
```

```
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#="dfef2c3e002008cd011
cc941ee6032925c21001022935c262
022955cdfef2c2044cd791cc4941e329
45ca72820f5cd941e32935cc180382bf
e193022dfef2c2024cd791cc4941e329
65ca72807ed7b3d5cc3761bf5cd941e3
2955cc1803804fe213802cf8a3e1bcd1
20f3e41cd120f3e08cd120f3e1bcd120
f3e4bcd120f3e965c0707260017cb14f
dcb5846280317cb14cd120f7ccdd12"
```

```
140 PRINT
150 PRINT "Die Maschinencoderroutine ist", "bereit.""Aufrufen durch PRINT USR 6E4,...(genaue Beschreibung in CK 11/85 und CK 6/86)""Das Basic-Programm kann durch NEW geloescht werden."
```

```
160 STOP
170 SAVE "LPRINT-Cop"CODE 6E4,305
```

## 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 Kampfdroide. Bewaffnet mit zwei 57er Magnum-Strahlern, technisch auf dem neuesten Stand, soll dieser Droide 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 Abwehrkrakete. 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 Krosch

# 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

Bestellung:

● Nachdruck mit 5,78 DM Versandkosten

● Vorauszahlung

Bitte bezeichnen Sie Vorposten mit Buchtitel, belegen

oder auf Postcheckkonto, Kontonr. 45122-764 überweisen.

Nr.	Titel, Stk., Autor	
N1	Bomb Jack	
N2	Goat'n Goblins	
N3	Alles über den ZX81	
N4	QL unter Kontrolle	
N5	Peep-Show	
N6	Super-Story	

Name des Bestellers

RZDF

Anschrift

Nachdruckcode

Coupons ausschneiden, auf Postkarte kleben und einsenden an:

Verlag Rätz-Eberle / CK-Software, Postfach 1640, 7518 Bretten.

# Der Colour Pen Graph EB50

Eine Art Schreibmaschinen-Plotter

Rein zufällig entdeckte ich in der Filiale der Warenhauskette divi in Randsdorf 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 Kugelstiften 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 Zeichengröße – 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 Displayfenster über der gespeicherten Zeile hin- und herbewegen, um gegebenenfalls noch Korrekturen vorzunehmen. Erst das Betätigen der Wagenrücklauf-taste bringt die Zeile auf 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 tagenfülligsten, daß der Pen Graph eher ein Plotter als ein Drucker ist. Denn jetzt genügen wenige Tastendrucke und die Eingabe von Zahlendaten 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 Schiebeshalter wird bewegt, und schon gehorcht der Pen Graph nur noch dem Rechner – die Tastatur ist abgeschaltet. Der EB50 verhält sich jetzt genauso wie ein Typendrucker, 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 Kugelstift. 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-, Größenbereich-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 Zeichnung der abgebildeten Zeichensatz-tabelle etwa drei Minuten benötigt wurden.

Noch ein paar Worte zur mechanischen Seite: Die Tastatur

### Ihr direkter Draht

zur Redaktion:  
Telefon 0 72 52 / 4 29 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/GEI sowie Atari ST. Nutzen Sie unseren Telefon- und Auftragservice zu den angegebenen Zeiten, damit auch Sie über die Neheiten informiert sind.

Spectrum		QL	
Allen Highway	31,90 DM	Assembler	99,90 DM
Estman	20,90 DM	Karte	92,90 DM
Bomb Jack	33,90 DM	Pracool	309,90 DM
Planeta	39,90 DM	C	399,90 DM
Speedcity	32,90 DM	Cartridge Doctor	99,90 DM
Planetagram	39,90 DM	Spook	47,90 DM
Erkline	39,90 DM	Flight Simulation	77,90 DM
Atari		Hardware	
Assembler	(ST) 179,90 DM	AMF Mouse	(Spec) 299,00 DM
C	(ST) 247,90 DM	GL Jergal	309,00 DM
Pascal	(ST) 327,90 DM	Spectrum 120 (dt)	549,00 DM
Knock Out	(Q) 49,90 DM	Discovery 1	609,00 DM
Knock Out	(Q) 54,90 DM	SpecDrem	145,00 DM
Mr. Robot	(Q) 99,90 DM	Multiscore One	109,00 DM
Mr. Robot	(Q) 39,90 DM	Spectrum Plus	249,00 DM

Preise für andere Systeme sind

Dreaser Soft- u. Hardware, im Rosenthal 6, D-5380 Bonn 1, 9-18 (21-18) 40-84.  
Mo. bis Fr. von 17.00-19.00 Uhr, Sa. von 14.00-19.00 Uhr oder Auftragsbestellung rund um die Uhr.

SPECTRUM - QL - ATARI - SPECTRUM - QL - ATARI - SPECTRUM -

## SPECTRUM REPARATURDIENST

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

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

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

entspricht der deutschen Norm und macht einen soliden Eindruck. Man spürt jedoch keinen sauberen Druckpunkt. Bei zu sanftem Schreiben geht schon mal ein Zeichen verloren. Der EB50 verarbeitet nur DIN-A4-Papier. Ein schmaleres Format wird nicht korrekt eingelesen, denn die Papierführung übernehmen am Rand der Walze angeordnete Stachelräder, die übrigens das Papier etwas perforieren, aber wohl für die hohe Positionierungsgenauigkeit unerlässlich sind. Man sollte nur einzelne Blätter verwenden. Endlosformulare sind zu trügerisch und würden die schnellen Walzenbewegungen (beim Schreiben eines jeden Zeichens!) behindern. Durchschläge sind nicht möglich.

Die Lebensdauer der kleinen Stifte ist höher, als ich anfangs befürchtet hatte. Nach Ausdruck mehrerer Seiten Text (in verschiedenen Farben) macht sich noch keine Qualitätsabnahme bemerkbar. Ein Satz Stifte (vier Stück) kostet etwa DM 12,-.

Nach so viel Worten des Lobes ein paar Wermutstropfen. Mich stört, daß nicht alle Funktionen des EB50 sowohl im manuellen als auch im Printer-Modus zur Verfügung stehen: Bei manuellem Betrieb muß man auf die Möglichkeiten der dreifachen Schriftgröße, des Überkopf-Druckens und des Plottens verzichten. Im Printer-Modus (Ansteuerung über Centronics-Schnittstelle) besteht keine Möglichkeit, zwischen den Druckstilen umzuschalten (z.B. Kursivdruck). Auch kommt man nicht an die fest angebaute Diagramm-Funktion heran. Diese lassen sich aber natürlich auch selbst errechnen und mit Hilfe der Plot-Commandos aufs Papier zaubern.

Mit diesen Einschränkungen kann man jedoch leben, besonders wenn man den konkurrenzlos geringen Preis daneben setzt: Der Pen Graph EB50 wird für sage und schreibe DM 299,- angeboten! Im Lieferumfang mit enthalten sind zwei Sätze Stifte, ein Netzgerät und eine ausführliche deutschsprachige Bedienungsanleitung.

Jürgen Peters

## Musterausdruck

Drucken in normaler, ~~normaler~~ und doppelter Schriftgröße  
Kursivdruck und automatisches Unterstreichen

Der EB50  
Der EB50  
Der EB50  
schreib in jeder Richtung  
schreib in jeder Richtung  
schreib in jeder Richtung  
schreib in jeder Richtung  
schreib in jeder Richtung  
schreib in jeder Richtung

	O	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
+		0	B	F		P	E	A	X	B	T					
+	+	PCU	1	A	C	a	q	z	W	E	L	O				
+	+	PCU	1	B	S	c	e	L	A	T						
+	+	+	3	C	S	o	z	W	E	L	O					
+	+	+	4	D	T	a	Z	X	B	T						
+	+	+	5	E	L	u	o	u	O							
+	+	+	6	F	V	T	V	O								
+	+	+	7	G	M	q	W	I	I	O	G					
+	+	+	8	H	X	A	K	S								
+	+	+	9	I	V	I	V									
+	+	+	+	J	Z	E	L	O								
+	+	+	+	K	k	i	n									
+	+	+	+	L	l	i										
+	+	+	+	M	m	i										
+	+	+	+	N	n	a	O	X								
+	+	+	+	O	o	O	S	E								

Der Zeichensatz des Colour Pen Graph EB50

## Der ISS Datenrekorder

Immer mehr Hersteller erkennen, daß viele User ein kostengünstiges und doch leistungsfähiges Speichermedium haben wollen, um so ihre Daten preiswert sichern zu können. Dabei werden längere Ladezeiten in Kauf genommen, wenn das Gerät präzise seine Aufgabe erfüllt. Hier soll nun wieder ein Datenrekorder vorgestellt werden, der zwar speziell für Sinclair-Systeme gedacht ist, aber natürlich auch mit jedem anderen Computer, der die entsprechenden Anschlüsse besitzt, betrieben werden kann. Genauer gesagt handelt es sich um den ISS Datenrekorder, den der Sinclair-Ex-Generallieferant Jürgen Schampich in der Bundesrepublik vertreibt. Im Vergleich zum Profi-Datenrekorder (CK 2/85) verfügt dieses Gerät direkt über eine große

Anzahl von Anschlußmöglichkeiten und braucht keinen Adapter.

Nachdem man den Rekorder ausgepackt hat, sieht man ein relativ kleines Gerät vor Augen, das ohne Batterien nur 455 g wiegt. Die wichtigsten Bedienelemente des Datenrekorders sind: ein Bandzählwerk mit Rückstellaste, eine Batterie/Aufnahme-Anzeige, ein Monitor-Schalter, eine Netzadapter-Anschlußbuchse (DC 6V), eine EAR (Output) Buchse, eine Fernbedienungsbuchse (REM), eine MIC (Input) Buchse, ein Lautstärkeregler und die üblichen Bedienungstasten inklusive einer Pausen-Taste. Die Stromversorgung erfolgt entweder über 4 Mignon-Zellen oder über einen Netzadapter mit einer Netz-

spannung von 6 Volt. Dem Gerät liegt auch eine deutsche Bedienungsanleitung bei, die neben der Funktion der einzelnen Bedienelemente auch noch ein paar grundsätzliche Dinge beschreibt.

Der ISS Datenrekorder besitzt neben einer automatischen Endabschaltung auch eine automatische Aufnahmekontrolle (ALC), durch die die Regelung der Lautstärke bei zu speichernden Programmen entfällt. Beim Laden von Programmen kann dieser Regler bis auf eine Lautstärke von "10" gestellt werden. Der eingebaute Lautsprecher (Piezo Typ) ist zwar nicht viel lauter als der normale Spectrum-Beeper, aber erreicht zum Anhören von Computer-Daten aus. Die Bandgeschwindigkeit beträgt 4,75 cm/sec.

Angenehm fiel auf, daß das dauernde Umstecken zwischen

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 Kassettenrekordern einige Funktionen und Annehmlichkeiten bietet, die den Umgang mit dem Computer erleichtern.

Preis: 129,- DM  
(ohne Netzteile)  
Bezugsquelle: COMPUTER  
ACCESSOIRES,  
Jägerweg 10, 8012 Ottobrunn  
Markus Pister

## 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 Buchstabens 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 Maschinencode 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.

Heinz Müller

# Back up Kopie

von Knight Lore + Underwurdle

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 einlegen.

### Underwurdle

- Anfertigen eines HEADERS: SAVE "under" CODE 16384,66100
- Erstellen eines Kopierers:
  - Eingabe des mc-Programms
  - Abspeichern des mc-Programms: SAVE "copy" CODE 65487,32
- 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
- Starten des Programms: RANDOMIZE USR 24740
- Einladen des Originalprogramms; Einladen des erstellten HEADERS
- Einlegen einer Leerkassette für die BACK UP KOPIE, Aufnahmetasten des Rekorders drücken, 0-Taste des Spectrums drücken
- Erstellte Kopie in den Computer laden und durch folgende Befehle zerlegen: SAVE "wise" SCREEN, SAVE "under" CODE 26600,35000
- Erstellen eines Vorprogramms:
 

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

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

### Knight Lore

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

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

Lars Köber und Matthias 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  
Rahnerstraße 52  
4060 Viersen 1  
☎ 0 21 62 / 2 29 64

**Reparatur-Schnelldienst/Erstattteile**  
Rufen Sie uns an!

## MC-Programm

10 FOR 1485487 TO 85518  
20 READ A  
30 POKE A 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



## ZX 81-Kopierschutzprogramm

Dieses Hilfsprogramm erlaubt, ZX 81-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 Basis-Teil löschen, denn der entscheidende Maschinencode befindet sich geschützt am oberen Ende des RAM-Bereichs. Man läßt jetzt das Programm, das kopiersicher aufgezeichnet werden soll. Durch RAND USR 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 INKEYS).

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$="215D7F220440F901003
EC5017606C5ED730240E8219F4001A30
0EDB0C9B50000000F9D41C7E8F00940
00076217D40222940210000221440213
C40010021FD363B0051C5D4C037123C
11BF6217B40C37B03CDE7023A3B4032B
17F2A14402273"
1020 LET A$=A$+"7FFDCB3BB6213C40
222940217040221440215D7F11ED1210
FE1B7AB320F9CD1E032109401633CDFB
7F215E7F1634CDF67F2A737F22144021
7D40222940427E7E10FC213C401621CD
FB7F217B40CD16033A817F323B40CF00
CD1E03231520F9C9"
1030 LET S=16870
1040 FOR A=16514 TO 16705
1050 LET X=16*CODE A$+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 K RAM erläutert allen Biologiefreaks und solchen, die es werden wollen, anhand einer Grafik die Feinstruktur

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

Volker Heron

```

10 REM (C) VOLKER HERREN 1986
15 CLS
20 PRINT "
30 PRINT
40 PRINT "DIE DNS (DESOXIRIBONU
CLEINSÄURE)"
50 PRINT "ENTHALTEN DIE ERBINF
ORMATIonen"
60 PRINT "EINER ZELLE."
70 PRINT "SIE BESTEHEN AUS EIN
ZELLEN NAU"
80 PRINT "TEILEN, DIE IN EINER
GENAU FEST-"
90 PRINT "GELEGTEN ANORDNUNG U
ORANNDEN"
100 PRINT "SEIN MUESSEN."
110 PRINT AT 20,0;"
120 IF INKEY$(">")="1" THEN GOTO 12
0
130 IF INKEY$="1" THEN GOTO 900
900 CLS
1000 PRINT "
1010 FOR A=2 TO 20
1020 PRINT AT A,2;"0"
1030 PRINT AT A,0;"0"
1040 NEXT A
1050 FOR B=2 TO 20 STEP 5
1060 PRINT AT B,3;"
1070 NEXT B
1100 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 6,13;"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)O=DESOXIR
IBOSE"
1270 PRINT AT 14,13;"(=ZUCK
ER)"
1280 PRINT AT 16,13;"6)O=SEUERER
EST OER"
1290 PRINT AT 17,13;" PHOSPHO
RIBOSE"
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 "
1510 PRINT
1520 PRINT "ES IST KLAR, DASS DI
E ZEICHNUNG"
1530 PRINT "NATUERLICH SEHR UNGE
NAU IST."
1540 PRINT "ES WURDE ABER DARAU
F 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 HÖPFE, 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 "DNE"
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 "Paizer" 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 Laufschrift eingabe 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
14 REM ENGELHARDT STR. 17
14 REM 8501 SCHWANSTETTEN
20 LET X=0
30 LET M$=""
40 LET Y=X
50 PLOT X,Y
60 LET A$=INKEY$
70 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$="N" THEN RUN
130 LET X=X+(A$="B" AND X(63))-
A$="5" 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
AF.DNE"
300 IF INKEY$="X" THEN GOTO 10
300 IF N=LEN B$-32 THEN LET N=1
310 GOTO 260
320 POKE 16389,65
330 GOTO 260

```

## 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 erstzunehmenden 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 Cursorstasten und ENTER. Nach kurzer Übung klappt die Bedienung einwandfrei.



Übersicht-  
lich in 2D



Anspiechend 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 nächstbesten 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;

**Mov.s** – 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 Knoor

## QL: Was nicht im Handbuch steht!

Wie schon beim ZX-Spectrum gibt es auch beim Sinclair QL einzelne 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	OB	Pufferoverflow
-6	NO	Kanal nicht eröffnet
-7	NF	Nicht gefunden
-8	EX	Existiert bereits
-9	IU	In Bearbeitung
-10	EF	Datierende
-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. Würden 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 number=ERNUM
170 SELECT ON number
180 ON number=17
190 IF ERLIN=330 THEN
200 PRINT#0;"Es muß eine Zahl
eingegeben werden"
210 RETRY
220 END IF
230 ON numbers=7
240 PRINT#0;"Laufwerk nicht gefunden"
250 PRINT#0;"Cartridge vergessen?"
260 ON number=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 *****
```

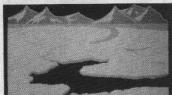
## Pengi

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 boshaften Snoobes 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 Snoobe, wird dieser vernichtet, was Punkte einbringt. Außerdem kann der Pinguin die Blöcke so stellen, daß ein Snoobe eingeschlossen und damit ungefährlich wird. Vier Pinguin-Leben stehen zur Verfügung; bei entsprechend hoher Punktzahl kann man aber ein Zusatzleben erhalten.

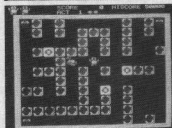
Auf dem Monitor sieht das Spiel recht gut aus. Nach einem Titelbild, das eine schöne Eislandschaft zeigt, erhält der Spieler eine kurze Erklärung. Auf Tastendruck wird das Spielfeld aufgebaut; die erste Runde kann beginnen. Die Anordnung der Eisblöcke ändert sich von Runde zu Runde, so daß das Spiel 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 Kaefer



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-Handbuchs 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 (BEEP 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$(127);" 1986"
1140 CSIZE 1,0:AT 24,17:PRINT"Drücke eine beliebige Taste"
1150 PAUSE
1160 END DEFine
1170 :
1180 REMark *****
1190 :
1200 DEFine PROCedure CHANGE
1210 neu>ton%(waagrecht)+inc
1220 IF neu>max(waagrecht) THEN neu=max(waagrecht)
1230 IF neu<min(waagrecht) THEN neu=min(waagrecht)
1240 SYMBOL waagrecht,neu
1250 ton%(waagrecht)=neu
1260 CURSOR 64*(waagrecht-1)+16,210
1270 PRINT ton%(waagrecht);" "
1280 END DEFine
1290 :
1300 REMark *****
1310 :
1320 DEFine PROCedure SYMBOL(xp,yp)
1330 LOCAL yalt,x,y
1340 x=64*(xp-1)+16
1350 yalt=180-(ton%(xp)-min(xp))/(max(xp)-min(xp))*135
1360 y=180-(yp-min(xp))/(max(xp)-min(xp))*135
1370 CURSOR x,yalt:PRINT" "
1380 CURSOR x,y:INK 4:PRINT"<==>":INK 7
1390 yalt=y
1400 END DEFine
1410 :
1420 REMark *****
1430 :
1440 DEFine PROCedure INIT
1450 DIM ton%(8),par$(8,10),min(8),max(8),start(8),inc0(8)
1460 RESTORE 1500
1470 FOR i=1 TO 8
1480 READ par$(i),min(i),max(i),start(i),inc0(i)
1490 END FOR i
1500 DATA " Deuer",-32768,32767,0,500
1510 DATA " Höhe 1",0,255,0,10
1520 DATA " Höhe 2",0,255,0,10
1530 DATA " Grad 1",-32768,32767,0,500
1540 DATA " Grad 2",-8,7,0,1
1550 DATA " Anzahl",0,15,0,1
1560 DATA " Verfreund",0,15,0,1
1570 DATA " Zufall",0,15,0,1
1580 FOR i=1 TO 8:ton%(i)=start(i)
1590 FOR i=1 TO 8:inc=0
1600 waagrecht=1:sprung=0
1610 END DEFine
1620 :
1630 REMark *****
1640 :
1650 DEFine PROCedure SCREEN
1660 CLS:CSIZE 0,0:
1670 INK 7:PRINT TO(23);"QL Sound-Demo von R.W. Gerling ";
CHR$(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" Andern 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=0 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=215          2390 ON a=REMAINDER
2310 inc=-1          2400 END SELECT
2320 CHANGE          2410 BEEP ton$(1),ton$(2),ton$(3),ton$(4),ton$(5),ton$(6),
2330 ON a=212          ton$(7),ton$(8)
2340 inc=inc0(waagrecht) 2420 END REPEAT main
2350 CHANGE          2430 CLS
2360 ON a=220          2440 REMark *****
2370 inc=-inc0(waagrecht) 2450 REMark *      Ende      *
2380 CHANGE          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 MDVLBOOT 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

```

Damach 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 Reiser

**Tom Rowley**  
**Sprühende Ideen mit Atari Grafik**



250 Seiten  
Sprühende Ideen ist ein Lehrbuch, das mit dem Gesamtgleichheit des ATARI in die Gestaltung der Objekte, in Farbgebung und die Entwicklung von Bildschirmarten einkehrt. Für den Leser genügt Kenntnisse der Programmiersprache Basic - auch wenn das Buch generell die Vorteile der Maschinensprache zeigt.

Bestellnummer TW 104

DM 45,-

**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-Spielereien mit dem Atari. Außerdem enthält es einige Tips und Programme zum Zerschneiden des Ataris.

Bestellnummer H 820

DM 29,80

**C. Lorenz**  
**Das große Spielebuch für Atari**



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

Bestellnummer H 821

DM 29,80

**Handbuch für Hacker**



In diesem Buch finden Sie endlich alle Antworten auf die Fragen, die sich jeder selbst und Bildung sowie beantwortbar kommt. Mit vielen Nummern und Datenbanken. Endlich wird hier erklärt, wie man mit Debugger arbeitet. Viele Tips und Tricks zum Mailboxbetrieb. Steigen Sie mit ein in die Unterwelt der DFPL.

Bestellnummer 9

DM 28,-

**Raeto West**  
**C 64/SX 64 - Computer Handbuch**



688 Seiten  
Insgesamt ein Jahr vorveröffentlicht der Autor Raeto West mit der Analyse und Dokumentation auf den C-64. Das Ergebnis seiner Arbeit: Das einzige einschlägigste (für Buch), das vielen fern Computer lernen 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 genährten.

Bestellnummer TW 102

DM 65,-

**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 geringen Grundkenntnissen in Basic, setzt aber keinerlei Assembler-Kenntnisse voraus.

Bestellnummer ID 202

DM 36,-

**Karl-Heinz Koch**  
**ATARI Spiele programmieren**



240 Seiten  
Das Buch führt Schritt für Schritt in die Programmierung in BASIC ein. Dabei werden schon mit den ersten einfachen Befehlen funktionierende Grafikbausteine erstellt. So werden die Befehle und ihre Wirkung nacheinander erlernbar gemacht. Auf Verständlichkeit wird besonders Wert gelegt, was für Bücher dieser Mäße weder keine Selbstverständlichkeit ist.

Bestellnummer SI 907

DM 32,-

**Chaos Computer Club**  
**Die Hackerbibel**



250 Seiten.  
Überall in den Medien ist seit einiger Zeit Rede von den Hackern. Und wenn man "Hacker" sagt, meint man in diesem unsere Länder zumeist Mitglieder des Chaos Computer Club Hamburg und seine Ableger. Sie sind über 130.000 DM Coup bei der Hamburger Sparkasse oder die diversen Aufträge in Tagesgeschäft, bei Frank Elstner und anderen; über sie ist viel berichtet worden. Hier erscheint ein Buch von Inman. Nicht nur über das "Was" der Hacker, sondern auch über das "Warum".

Bestellnummer H 800

DM 33,33

**Julian Reschke/ Andreas Wiethoff**  
**Das ATARI Profibuch**

296 Seiten.  
Das richtige Buch für Sie, wenn Sie einen Atari 400, 800, 800 XL, 800XL oder 1020X besitzen. Das Buch enthält eine vollständige Liste aller Systemadressen und -routinen mit Erläuterungen; detaillierte Informationen über Speicherbelegung und Ausnutzung des Betriebssystemes; einen Tabellenrat mit Zeichencodes, Adressen, Kommandos, Tonhöhen, Farben, Fehlermeldungen sowie einen alphabetischen Nachschlageregister.

Bestellnummer BY 108

DM 42,-

**Hal Hicksman**  
**Der ATARI als Musikbox**

104 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 Basisprogrammen Melodien, Akkorde und Rhythmen 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

**BUCH-BESTELLKARTE**

Bitte liefern Sie mir folgende Bücher:

Artikl	Bestell-Nr.	Titel	Preis/Pha./Mk./Stück

Name des Bestellers

Ich wünsche folgende Bezahlung:

- Nachnahme (+ 3,70 DM Porto + Versandkosten)
- Vorauskassa (keine Versandkosten)

Anschrift

Die Vorauskassa bitte Scheck belegen oder auf Postcheckkonto Karlsruhe 43423-750 überweisen.

PLZ/Ort

Titel

Bestellnummer

Coupon ausschneiden, auf Postkarte kleben und einsenden: Verlag Ritz-Eberle, Postfach 16 43, 7518 Brötten.



# 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 Einzelinstellungen in je-

den Schlag gelegt werden kann. Das vorliegende Programm bietet eine solche Schritt-für-Schritt-Simulation. Nachdem man den gewünschten Kurs angewählt 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-

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 Konstrukteur 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 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



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-





die Bewegung eines Springers über das Schachbrett, über Wortspele, 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

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 Kleck

## 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 Programteilen.

Der Kommando Prozessor 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 von C64 auf den C128 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 compilieren 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 64K des 128, gewählt worden. Sie ist auch weitgehend kompatibel zu den 1541/70/71-Laufwerken.

Der Kommando Prozessor 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 Programm Bibliotheken.

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 Tzsch

## 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 wählen.

"Shogun" besteht aus etwa 128 Bildschirmen, auf denen sicher 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 C64, 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: C64  
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 C64 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:

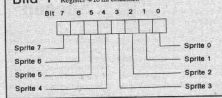
y-Position von Sprite 7.

Der C64 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).

**Bild 1** Register #16 im einzelnen



#### 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 Bildschirmcodes reduziert (siehe Tabelle der Bildschirmcodes im C64-Handbuch).

Bit 7: Enthält Bit 9 des Rasterregisters.

#### Register #18:

Enthält bei einem lesenden Zugriff 8 von 9 Bits der Punktezeit, 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 Punktbreite.

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 Grafikschirmen zu selektieren.

Bit 4-7: Mit diesen Bits wird die Basis des Video-RAMs festgelegt. Dadurch ist es möglich, den Bildschirmumfang 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 Light-pens 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 Standardbetriebsart 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 10dez
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 in 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 Zeichensatzinformation 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 das 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 abgefragt 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	0 \$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 (\*\$3792) 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 Betriebs-systemroutinen 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 ändern: 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 herzuschalten. Bis dahin: Spielen Sie ruhig ein wenig mit den Registern, denn "Probieren geht über Studieren!"

F. Lenczowski

## LOGO für Einsteiger

Von Dr. Dietrich Senfleben  
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-Zeilenditor und das Prinzip der Programmierspeicherung 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 Kleink

# 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 belanglos dickes Begleitbuch, dessen Umfang gut 290-300 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 sparen 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 Multi-colormodus, 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 Spätscreen oder das Scrollen der Bilder sind kein Problem mehr. Zweifellos positiv am Screen-Editor ist auch, daß nicht nur die Farben, sondern auch die für die Commodore-Grafik wichtigen Bitmapere bei allen Befehlen bedacht wurden. Fragt sich nur, wer bei der knappen Erläuterung mit diesem Feature etwas anfangen kann.

Zwiespaltiger präsentiert sich hingegen der Sprite-Editor des RP-Systems. Er enthält einiges an Standards, wie Directory, Editieren in Normal- oder Multicolormodus, Spiegeln und Kopieren 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 (woraan 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 Violschlüssel 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" gestaltet sich sehr kompliziert bis nicht unmöglich. An schwierigerer 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 Maschinencodier-

ten, die über SYS und entsprechende Mengen von Parametern aufgerufen werden. Mit dem freundlicherweise beigelegten Programmkopf läßt sich das Ganze auf einige Zweizeilen und einen Unterprogramm-aufruf verkürzen. Diese Befehle sind im großen und ganzen in drei Gruppen zu unterteilen: 1. Befehle zum Ausbew. 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-Screens, setzen oder steuern Sprites oder Lauf-schrift, lesen Joystickwerte oder synchronisieren gar Spritewebewegung 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 Spielgenerator, sondern schreibt seine eigenen Programme.

System: C:64  
Hersteller: Brillant Software  
Auel Klein

Centr. GLP II centr./ser. 508-  
STAR ML 10 auch 1. Conn. 848-  
CITIZEN 1200 an/ser. 749-  
SEIKOSM SP1000 A/V/C 749-  
unicom soft  
3500 Kasse  
H. K. 931 924 46A  
T. 0561/770367

# Mailbox Marke Eigenbau

In letzter Zeit explodiert die Datenfernübertragung (DFÜ) förmlich. Täglich kommen neue Mailboxes ("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 Pfeiltöne 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 Mailboxes 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 Mailboxes 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 Übertragungs-

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 Mailboxes 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 Mailboxes 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, sofort 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 Anweisungen von der Seite der Anrufer. Das Interessanteste und Aktuelle in einer Mailbox steht im sog. "Pinboard". Es ist eine Art Schwarzes Brett mit Verkauf- 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 Mailboxes 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 Mailboxes 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. 061 92/7960, 24h online (online = geöffnet). Parameter: 7/N/1.

Laus 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 C64 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-Eingangsmodul 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 Lötläthen konzipiert, sondern sind durch Schraubklammern herausgeführt. So lassen sich nicht nur die für den Laien schwierige Lötarbeiten einsparen, sondern auch schon einmal fixierte Drähte leicht wieder untklemmen.

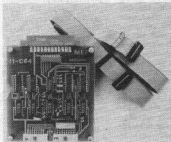
Dem Interface beigelegt ist eine knappe Erläuterung der Funktionsweise der einzelnen Module sowie die Erklärung der Anschlüsse. Für den Laien ist diese Anleitung leider erst nach mehrmaligem Durchlesen verständlich. Aber mit ein klein wenig technischem Verständnis nimmt man die Hürde leicht!

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

acht Ausgängen, ein Analog-Ausgangsmodul 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-Ausgangsmodul 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 PEEK-Befehl können die Relais einzeln angesteuert werden, wobei die Adresse durch einen sog. "Jumper" (steckbare Lötlücke) gewählt werden kann. Auch hier zeigt sich wieder die Durchsichtigkeit des Systems: Es müssen lediglich die Jumper umgesteckt und nicht etwa umständlicher Weise Lötlücken 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-Eingangsmodul 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-Ausgabemodul kann dann z.B. eine Lampe betätigt werden. Auch bei diesem Modul wird der Schaltzustand wieder durch LEDs angezeigt.

Die an das Analog-Eingangsmodul 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 Interna-

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

Das VIS bietet eine Umengung an Anwendungsmöglichkeiten 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 Interessierten 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

<b>S/S Modul</b> Hardcopy + Turbo Tape + Turbo Disk + Reset/Off	<b>30,00 DM</b>	<b>Super-Hardcopy-Modul</b> Druck bar/jedes Computer-Modul	<b>45,00 DM</b>
<b>Supersoftwepaket I</b> 12 Programme auf Disk, oder Kassette, Amerikanische und Holländische Sprache	<b>15,95 DM</b>	<b>S/S Modul</b> Turbo Disk + Hardcopy + Funktionen + O-Charger + Reset/Off	<b>30,00 DM</b>
<b>10 Disketten SS/DD</b> Saverette	<b>14,50 DM</b>	<b>S/1 Modul</b> Turbo Tape	<b>19,90 DM</b>
<b>S/4 Modul</b> Turbo Tape + Turbo Disk + 008 S.1 + Reset/Off	<b>23,00 DM</b>	<b>S/2 Modul</b> Turbo Disk + S1 und S2 ohne Gehäuse	<b>19,90 DM</b>
<b>Modul 2000</b> Turbo Tape + Turbo Disk + Memory 1024 + O-Charger per Seite Text (C.M. 11/85)	<b>45,00 DM</b>	<b>Profil Betriebssystem</b> Turbo Disk, LDR, D2 und 000 Funktionen, Funktionskatalog, im Diskette	<b>48,00 DM</b>
<b>Modul 3000</b> Hardcopy + Turbo Disk + Turbo Tape + Hardcopy + Floppy + O-Charger + Backup + Reset/Off	<b>65,00 DM</b>	<b>Benutzersysteme</b> umschaltbare 128	<b>30,00 DM</b>
<b>Modus S-128</b> Reparierbares Ersatzteil für die 1550-11 und ca. 3,7 mal so viele Module	<b>49,00 DM</b>	<b>umschaltbare 128</b> Dient für ein Programm, im Falle Modus auf 2048 weitere Betriebsmodi zu betreiben. Keine Kasse, sondern, weitere Platten auf 9,90 DM	<b>30,00 DM</b>
		<b>Leertplatte</b> 9,90 DM	
Alle Module im Gehäuse, mit Reset (außer von LDR) und O-Programme (außer Turbo-Hardcopy-Modul). 14 Stunden Service, Info gegen 1,50 DM. Kostenlose Zubehör- und Lernsoftware C-64 auf Lager, z.B. 8508			
<b>Klemmer &amp; Schulte Electronic</b>		Barnsdorferstr. 496 · 50330 Hirth-Effren Lieferzeit: Mo-Fr. 13.00-18.00 Uhr Sa. 13.00-13.00 Uhr, So 12.00-13.00 Uhr	

# Der Star Painter – Neu von Sybex

Der Star Painter ist kein gewöhnliches Grafikprogramm wie z.B. Koola Painter oder Blazing Paddles, die den Multi-color-Modus mit einer Auflösung von 320x200 Punkten benutzen. Der bedeutsamste Unterschied zu diesen Programmen ist die Größe der zusammenhängenden Grafik, die 640x344 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 256x168 Punkten. Hinzu kommen verschiedene Editiermöglichkeiten, eine praxisgerechte 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 Linien eilaufen, die das Formatieren der Zeichnung erleichtern. Meines Wissens nach ebenso einzigartig ist die Möglichkeit, die Zeichengeschwindigkeit zu variieren. Da mit einem C64 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 24x21 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 24x21 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 Grafik-speicher 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 kopiertes 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 Zeichensatz 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 640x344 Punkten stark verkleinert auf den Bildschirm bringen.

Einzige Voraussetzung für die Übernahme von Grafiken aus anderen Programmen ist eine Auflösung von 320x200 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-

table Geräte. Sollte Ihr Drucker in dieser Auswahl nicht enthalten sein, so können Sie Star Painter ohne viel Aufwand an 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 dem 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 der 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 Seibach

## Commodore 64





## Hardcopy-Routine für den C 64 mit MPS 801/803

Mit einer Hardcopy-Routine können Hires-Grafiken auch mit dem Drucker ausgegeben werden. Geben Sie dazu zunächst den vorliegenden Basic-Loader ein, speichern ihn sicherheitshalber einmal ab und starten Sie. Die ersten drei REM-Zeilen des Loaders dürfen beim Eingehen übrigens auf keinen Fall fehlen, da sie den Filenamen für die in den Loader integrierte Save-Routine enthalten. Falls Sie alles korrekt eingegeben haben, meldet sich der Loader nach kurzer Wartezeit, in der das Maschinenprogramm eingelesen wird, mit der Frage, ob das reine Maschinenprogramm auf Diskette oder auf Cassette gespeichert werden soll. Danach wird der Loader nicht mehr benötigt.

Bei Bedarf können Sie die Routine jetzt mit LOAD "HARDCOPY C64", 8, 1 la-

den. Da es sich um ein absolut geladenes Maschinenprogramm handelt, muß danach einmal NEW eingegeben werden. Mit SYS 49152, LFN, R, ADR wird die Hardcopy-Routine gestartet, wobei LFN die durch den vorhergehenden OPEN-Befehl festgelegte logische Filenummer ist. Der Parameter R bestimmt, ob es sich um eine normale Hardcopy handeln soll (R=0) oder um eine reverse (R>0), bei der alle Punkte invertiert werden.

ADR schließlich bestimmt die Anfangsadresse des Grafik-Bildschirmes. Der Hardcopy-Vorgang muß mit PRINT#4: LFN: CLOSE LFN abgeschlossen werden. Beispiel: OPEN 4,4: SYS 49152,4,1,8192: PRINT#4: CLOSE 4 ergibt eine reverse Hardcopy eines Grafik-Bildschirmes ab Adresse dez. 8192.

Diese Routine ist mit einer Länge von nur 174 Bytes außerordentlich kurz und läßt sich damit gut in eigene Programme integrieren.

Daniel Durstewitz

145 :

```

150 DATA32,155,103,32,201,255,32,155,103
134,2,32,253,174,32,130,173,32,247
155 DATA103,169,0,32,210,255,169,20,133,
254,169,7,133,6,169,40,133,253,169
160 DATA120,133,251,162,128,169,64,133,2
52,164,6,136,40,5,32,103,140,16,240
165 DATA165,254,200,2,162,128,32,196,140
,160,0,136,196,6,240,5,32,103,140
170 DATA16,246,165,20,24,105,64,133,20,1
44,2,230,21,230,21,130,166,2,240
175 DATA2,73,127,32,210,255,165,251,74,2
00,109,165,20,24,105,0,133,20,144
100 DATA2,230,21,198,253,200,172,169,13,
32,210,255,198,254,40,0,198,6,16
105 DATA155,32,196,140,200,146,169,13,32
,210,255,76,204,255,105,251,49,20
190 DATA240,4,130,5,252,170,70,252,96,16
5,20,56,232,64,133,20,176,2,190,21
195 DATA190,21,96
200 :
205 REM SAVE-ROUTINE
210 :
215 DATA162,0,160,1,32,106,255,169,12,16
2,95,160,0,32,109,255,169,0,133,251
220 DATA169,192,133,252,169,251,162,174,
160,192,76,221,245

```

## Zeilen-Inverter

Bei dieser Routine handelt es sich um ein Utility, welches ab Adresse 49152 (\$C000) im Speicher liegt und somit keinen Basic-Speicher verbraucht. Die Routine wird nach dem Einpoken mit SYS 49152,x (CR) gestartet. X ist eine Zahl zwischen 0-23, die eine Zeile angibt, die

das Programm dann sehr schnell invertiert. Dies kann z.B. zur Menüauswahl als Lesehilfe oder zum Herausstellen von Texten in Textverarbeitungsprogrammen verwendet werden. Wird die Routine mit derselben Zeilennummer zum zweiten Male aufgerufen, so ist die Zeile wieder normal.

Jörg Tottus

```

1 REM >>>>>>>>>>>>>>>>
2 REM ( ZEILEN INV. (
3 REM )-----)
4 REM ( (C) 1986 BY (
5 REM ) JOERG TUTTAS (
6 REM (<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
7 :
8 REM START MIT :
9 REM SYS 49152,ZEILENR.
10 FOR T=49152 TO 49201
11 READ A:G=S+A:POKE T,A
12 NEXT T
13 IF S<>6786THEN PRINT"FEHLER IN DATAS
?" :STOP
14 PRINT"OK !!":END
15 :
1000 DATA 32,253,174, 32,150,103,169, 4,
133,251,169, 0,133,250,224, 0,240
1001 DATA 14, 24,105, 40,176, 4,202, 76,
14,192,230,251, 76, 23,192,133,250
1002 DATA160, 0,177,250, 24,105,128,145,
250,200,192, 40,200,244, 96,160

```

```

10 REM *****
15 REM *
20 REM *      HARDCOPY C64      *
25 REM *
30 REM * EINGABEFORM:
35 REM *
40 REM * SYS 49152,LFN,R,ADR
45 REM *
50 REM * 1986 BY DANIEL DURSTEMITZ
55 REM * REICHENBERGER STR.26 *
60 REM * 4460 NORDHORN      *
65 REM * TEL.,05921/79200    *
70 REM *
75 REM *****
80 :
85 :
90 FOR T=49152 TO 49350:READ A:POKE T,A:P=P+A:
NEXT
95 IF P=2646100T0110
100 PRINT"FEHLER IN DEN DATAS! BITTE UE
BERPUEFEN"
105 PRINT"SIE DIE ZEILEN 150-220 NOCH EI
NMAL!" :END
110 PRINT"IBOLL DAS PROGRAMM AUF DISKETTE
TTE ODER"
115 INPUT" AUF KASSETTE ABGESPEICHERT W
ERDEN" :A$
120 IF A$="D" THEN POKE 49327,0:GOTO 135
125 IF A$="K" THEN POKE 49327,1:GOTO 135
130 GOTO 110
135 SYS49326:END
140 :

```

## 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 Datensette 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.

Stapel-Göckchenerei

## 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 FOR I=020TO85:READA:POKEI,A:5=5+A
110 NEXT:IFS(<)3126THENPRINT"DATA-ERROR
!!!":STOP
140 DATA162,0,160,0,232,142,32,288,173,1
,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
```

```
10 POKE53201,8:POKE53200,8:PRINT"1"
11 PRINT" "
20 PRINT" "
30 PRINT" 1"
48 PRINT" 1" TIME LOADE
R "
50 PRINT" 1"
52 GOSUB10000
55 PRINT" 1"
60 PRINT" "
65 PRINT" 1"
78 PRINT" 1" 1 UHRZEIT EINST
ELLEN "
80 PRINT" 1"
98 PRINT" 1" 2 PROGRAMM STAR
TEN "
91 PRINT" 1"
100 PRINT" "
105 IFV=1THENPRINT" " PROGRAMM
ARBEITET "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 OPEN1,8:PRINT" UHRZEIT (Z.B. 102056
) = " :INPUTW1,T1$:CLOSE1:GOTO11
200 PRINT" GEBEN SIE BITTE DIE UHRZEIT
EIN, BEI DER "
210 PRINT"DAS PROGRAMM GELADEN WERDEN SO
LL."
```

```
220 OPEN1,8:PRINT" UHRZEIT (Z.B. 102056)
= " :INPUTW1,LA$:CLOSE1:PRINT" "
230 OPEN1,8:PRINT"FILENAME (MAX.16 Z.) =
" :INPUTW1,FL$:CLOSE1:PRINT" "
240 L=LEN(FL$)
245 P=VAL(LA$)
246 T=LEN(LA$)
250 IFL>16THEN200
255 IFT<16THEN200
256 IFF<00000000RP>235959THEN200
260 PRINT" DISKETTE MIT DEM PROGRAMM "
"FL$":EINLEGEN."
270 PRINT" "
SPACE"
280 GETR$:IFR$(<)" *THEN200
290 V=1:PRINT" ":GOTO11
500 IFLA$=T1$THEN520
510 GOTO11
520 PRINT" ":POKE190,8
530 PRINT" "CHR$(34)+FL$+CHR$(34)+
",8"
540 PRINT" RUN"
560 POKE631,13:POKE632,13
580 POKE190,2
9999 END
10000 S$=LEFT$(T1$,2)
10001 M$=MID$(T1$,3,2)
10002 R$=RIGHT$(T1$,2)
10003 PRINT" UHRZEIT = "S$":
: "M$": "R$": "
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

ausgewirft 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 Rösch

```

100 POKE53200,0:POKE53201,0:POKE646,1:PR
INT*";POKE53202,0:"EASTERN FIGHT"
101 PRINT*";POKE53203,0:"(C) BY RADDISH-SO
FT*"
102 PRINT*";POKE53204,0:"BITTE WARTEN SI
E 4 MIN.":FOR T=1 TO 4000:NEXT
103 FOR X=032 TO 065:READ:POKE X,0:NEXT
104 DATA 120,169,51,133,1,169,0,133,95,13
3,90,133,80,169,200,133,96,169,240,133
105 DATA 70,169,224,133,91,32,151,163,169
,85,133,1,80,96
106 SYS302:POKE850,160:POKE56576,PEEK(5
576):AND(52):POKE53272,0:POKE5649,192
107 PRINT*";"
108 AD=30000:FOR I=1000 TO 1075:READA#,PS:C
=0:FOR N=1 TO LEN(A#):STEP 2
202 A=ASC(MID$(A#,N,1))-48:B=ASC(MID$(A#
, N+1, 1))-48:I=A#*7*(A)9:I#B#A#*7*(B)9
203 POKEAD, I6#A#B#C#C#PEEK(AD):AD=AD+1
204 NEXT:PRINT I#I#*":IF C(PS) THEN PRINT*ER
ROR":STOP
205 PRINT*OK":NEXT
206 GO TO 2000
1000 DATA A0640C13C0C88C10C0C80C16C0C88C
36C0C89C3EC0C89C390C8C3A0C80C3C0C, 4266
1001 DATA 0C3D0C89C38C0C20C78A900050205
F085F985FC85FE0D0C2E28E9A90180, 4206
1002 DATA C0020D0BF0027B8EA90001403A9750015
035060AD00002F00634CE67CA502F0034C, 3265
1003 DATA D075A90005FCA00000C93770E23C97B
F6262910F0034CD075AD000C2900F01F, 3909
1004 DATA DATA AD00DC2901F01FAD000C2902F01F4C
D075A90105024CD075A90265024CD0075, 3266
1005 DATA A90365024CD075A90489024CD0075A9
05502A502F013C901F01B0C902F070C9, 3416
1006 DATA 03F00AC904F0094CFD0754C09774CC0
704C00774C4A77A5F005C0C6F4C0977, 3800
1007 DATA A4F0C010F050AD0000C3C88038AD000
D069078000AD0002069078002D0A004, 3826
1008 DATA D069078000AD0002069078002D0690
207D00F0C3C089207D00F9C3C089207D, 4146
1009 DATA 8DFAC3C089207D00F0C310A5FC6904
05FCA900A95F04C8977A900005024C8977, 4189
1010 DATA A5FB005C6FB4C0977A4FCC010F050
AD000C932903B0AD000E900000AD4, 4365
1011 DATA 02D0E900002D0AD00E9000000400
AD0600E900000000932907D00F0C3C089, 4099

```

```

1012 DATA 387D0DF9C3C0899307D00FAC3C089930
7D00FBC310A5FC690485FCA90A05F04C, 4645
1013 DATA 0977A900005024C8977A5FB005C6FB
4C0977A4FCC008CF028909487D00F9C3C, 4401
1014 DATA B9407D00F0C3C0899307D00FAC3C0899
487D00FBC310A5FC690485FCA91485FB, 4012
1015 DATA C4C8977A90005024C8977A5FB005C6
FB4C0977A4FCC008CF028909547D00F9C3, 4209
1016 DATA C089547D00F0C3C0899547D00FAC3C08
99547D00FBC310A5FC690485FCA91405, 4797
1017 DATA FB4C8977A90005024C8977A5FB005
C6FB4C8977A4FCC008CF028909607D00F8, 4357
1018 DATA C3C0899607D00F9C3C0899607D00FAC3
C0899607D00FBC310A5FC690485FCA914, 4095
1019 DATA 05FB4C8977A90005024C8977A5FB05
F0024CE977A90005FEA5020F13C091F0, 4099
1020 DATA 22C90202FC90F040C90402E4CDE
77EAAD04DC2903F015C901F010C902F0, 3705
1021 DATA 294CE977AD04DC2903F0184CC977A9
0185FD4CE977A90205FD4CE977A90305, 3844
1022 DATA F04CE977A90405FD4CE977A90505FD
A5FD013C901F01E902F017C903F00A, 4297
1023 DATA C904F0094C09794CA4794CE1704C23
794C05794C7879AD0C02F006CE0C024C, 3368
1024 DATA A479A4F79A0810F051A00800C9C00030
A0000009000D000D00AD0A00690000A00, 3979
1025 DATA 08C0069000C00AD00E0000000E00000
D0097C7D00FCC3C0897C7D00F0C3C089, 4368
1026 DATA 7C7D00F0C3C0897C7D00F0C310A5FE
690405FEA9000002024CA479A00005FD, 4428
1027 DATA 4CA479AD0C02F006CE0C024CA479A4
FEC010F051AD0000C029039038AD00009, 4139
1028 DATA 0000000AD00AD000000000A00A00C00
E900000C00A00E00E900000E00896C70, 3768
1029 DATA 00FCC3C08996C7D00F0C3C08996C7D00
FCC3C08996C7D00F0C310A5FC690405FE, 5151
1030 DATA A909000C024CA479A00005FDCA479
AD0C02F006CE0C024CA479A4FCC00CF0, 4014
1031 DATA 2AB90C7D00F0C3C08996C7D00F0C3C08
996C7D00F0C3C08996C7D00F0C310A5FE, 5243
1032 DATA 690485FEA911000C024CA479A900005
FD4CA479AD0C02F006CE0C024CA479A4, 3020
1033 DATA FEC00CF02AB90C7D00F0C3C089967D
00F0C3C089967D00F0C3C089967D00FF, 5551
1034 DATA C310A5FE90405FEA911000C024CA4
79A900005FDCA479AD0C02F006CE0C02, 3933
1035 DATA 4CA479A4FCC008CF02AB9A947D00FCC3
C089A47D00F0C3C089A947D00F0C3C089, 5239
1036 DATA A47D00F0C310A5FE90405FEA91100
0C024CA479A900005FDAD0000002F0, 4235
1037 DATA 0CA502C9039010A5FD0C90390034CA5
7AA5FCC90C907EEBF0E2200937A20E79, 3016
1038 DATA 4CA57AA5FDC90390E5A502C90390003
4CA57AA5FDC90390E00002200937A20, 3978
1039 DATA EE79AC457AA9000000000000D0A0A9C7C
0002D00D0000A9C8000000000000A90E, 4172
1040 DATA 00AD0000D0A09010010010000000000
000000000A900000000000000000, 3863
1041 DATA 800F00A901002000002D00002D00A9
0002000000A9009926D00000F0A9009, 3750
1042 DATA 8023D00A903002D00A0004A907992AD0
0000F0A97FD0150001C000A100CF0C3, 4203
1043 DATA C00CF9C3C089FAC3C089FAC3C089FAC3
F0C3C089FAC3C089FAC3C089FAC308A9, 5774
1044 DATA 0000040A90F01004A9000005D0A9
000000D0A978000004001D0A9910004, 3420

```

```

1045 DATA D460AD080ED080C9089080A9018D
BE024CBF7A9000502A90185FDCE15C0, 3677
1046 DATA CE15C0AD15C0C95A90034077B18AD
15C0690AD15C0CE14C0AD14C0C95A90, 3566
1047 DATA 034C077B18AD14C0690AD14C0CE12
C0AD12C0C95A90034C077B18AD12C069, 3041
1048 DATA 0A0D12C0CE11C0AD11C0C95AD011AD
12C0C95A900A0AD14C0C95AD0034C227B, 3686
1049 DATA 4C9D7BAD15C0C95A906F6A9018D080E
A010B9B07D99F9C280D0F7A01BB9C17D, 4495
1050 DATA 9920C380D0F7A013B9D07D9948C380
D0F718AD0F8269598D2CC318ADC00269, 4152
1051 DATA 590D54C3AD0F82C0C0029010F01CA0
10B9F17D9970C380D0F74C927BA01FB9, 4209
1052 DATA 0A7E9970C380D0F74C927BA00D0947
1E9970C380D0F7A01C99A7E99C0C380, 4363
1053 DATA D0F7ACBF02A95188F0069953C04CA2
7BACC002A95188F0069967C04C00784C, 4138
1054 DATA 31EAA90005FBA9C405FCA206A90005
FD09CA05FEA000B1FB91FDC000F90EFC, 5192
1055 DATA E6FECAD0F020277CA00005FBA9CA05
FC91805F0A215A00001FB8502A002B1, 4460
1056 DATA FBA00091FBA502A000291FB18A5FB69
0305FBA5FC690005FCCAD0DC18A5FB69, 4546
1057 DATA 0185FBA5FC690005FCCFD05FDD0C7
60A90005FBA9CA05FCA206A000A0005, 4603
1058 DATA 02B1FB29CF018C9C0F009C900F00F
E6024C587C18A502690385024C587CE6, 3628
1059 DATA 02E602B1FB29CF023C930F00EC920
F01418A502690405024C817C18A50269, 3082
1060 DATA 0C85024C917C18A5026900502B1FB
290CF023C90CF00EC900F01418A50269, 2903
1061 DATA 1005024CA7C18A5026930005024CAA
7C18A502690200502B1FB29CF023C903, 2784
1062 DATA F00EC902F01418A502690405024C03
7C18A50269C005024C037C18A5026900, 3001
1063 DATA 0502A50291FBC90006E9FCCAD0040
4C337C4C317CAD01DC2910F0034C257D, 3536
1064 DATA EAA900008002A9600D11C0A95A0012
C00D14C00D15C000EE79A90180BF020D, 3701
1065 DATA C002A029A920994FC0000FA00EFA9
3599F8C290D0FA4C31EA101114151011, 3900
1066 DATA 161710111819101112131011181910
1116171011141510111213101118191A, 609
1067 DATA 1C1810111213101118191E201F21
10111213101118192224232510111213, 726
1068 DATA 29202D2C29202F2E29203130292020
2A2920313029202F2E29202D2C292020, 1334
1069 DATA 2A29203130343235332920262A2920
31303936393729202B2A292031303C3A, 1454
1070 DATA 3D302920262A35040905351A050914
3509131435150D35351309053500102, 861
1071 DATA 050E3535353510150E0A140535150E
043504051235030F0D10151405123535, 700
1072 DATA 35353510150E0014053504010D00914
35000102050E3513090505705170F0E, 632
1073 DATA 0E050E3504010D00914350001143504
051235030F0D1015140512350705170F, 562
1074 DATA 0E0E050E3513100103053514011314
053506150512350E05150513351310009, 579
1075 DATA 050C35150E050E1413080009050405
0E20, 243
2000 FORT=784*64T0910*64*PKET,0*NEXT
2010 FORT=0T062:READQ:POKET+784*64,Q:NEX
T
2020 FORT=36T052:READQ:POKET+785*64,Q:INE

```

```

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=40T057:READQ:POKET+791*64,Q:INE
XT
2090 FORT=0T059:READQ:POKET+792*64,Q:NEX
T
2100 FORT=12T057:READQ:POKET+793*64,Q:INE
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:INE
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:INE
XT
2180 FORT=0T056:READQ:POKET+801*64,Q:NEX
T
2190 FORT=39T062:READQ:POKET+802*64,Q:NEX
XT
2200 FORT=0T062:READQ:POKET+803*64,Q:NEX
T
2210 FORT=51T062:READQ:POKET+804*64,Q:NEX
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,,5,6
2250 DATA 5,05,,21,213,,21,221,,21,110,,
5,90,,1,90,,3,30,,1,221,,1,117
2260 DATA64,,00,,84,40,,149,160,,149,1
60,,20,120
2270 DATA 1,05,,3,07,,1,253,,1,117,,1,22
1,,5,93,,21,85,,21,85,,85,77,,85,195
2280 DATA 1,87,,1,92,,1,92,,1,00,,5,112,1
2,5,64,255,5,67,255,5,127,243,9,252,42
2290 DATA240,,40,,
2300 DATA,,,,,,,,64,,00,,00,,04,,04
,,05,,05,,21,64,,53,64,,5,64,,5,64
2310 DATA 1,00,,193,00,,241,92,,252,92,,
15,170,,3,234,,
2320 DATA 1,05,,3,07,,1,253,,1,117,,5,22
1,,5,05,,21,05,,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,,40,163,,160
2340 DATA,,,,,,,,64,,64,,84,,84,,
64,,64,,00,,00,,00,,00,,00,,00
2350 DATA,,00,,92,,160,,160,,
2360 DATA 1,05,,3,07,,1,253,,5,117,,5,22
1,,5,05,,5,117,,5,93,,1,92,,1,04,,3,84
2370 DATA,,84,,87,,1,223,,1,85,,1,117,,

```

```

1,117,,14,101,,2,190,,10,130
2300 DATA192,,128,,176,,160
2300 DATA1,85,,3,87,,1,253,,1,117,,1,22
1,,85,,1,85,,1,93,,1,85,,1,93,,1,81
2400 DATA,1,81,,5,112,,5,117,,5,69,,5,12
1,,5,122,,50,250,,62,142,,14,131
2410 DATA64,,64,,64,,80,,80,,80,,80
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 DATA1,93,,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,,60,,86,120,,90,120,,218,,
,,20,,192,,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,,62,120,,6
2,192,,50,120,,50,192,,50,120,,50,5
2500 DATA,10,85,,21,85,,95,85,,1,85,150,5
,,85,80,5,87,80,5,197,84,21,84,23,1,92
2510 DATA20,1,93,24,1,117,40,3,117,40,1,
213,0,1,125,,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,2,120,64,,84,,86,,90,120,,5
0,130,,2,,26,,90,,5,90,,21,80,,95,92
2540 DATA,85,192,,84,,80,,112,,
,,240,,63,12
2550 DATA,,243,
2560 DATA,,240,,252,,3,252,,3,232,,3,23
6,,3,160,,3,172,,3,168,,1,160,,5,161
2570 DATA,21,85,,85,85,,93,221,,87,85,,2
1,87,,6,140,,2,140,,1,140,,7,92,,21,245
2580 DATA,85,213,,1,87,117,1,85,5,1,92,,1
87,252,,85,255,,213,131,3,190,120,,2
2590 DATA120,,32,,160,,1,160,,5,120,,21
,,84,,92,,80,,112,,
2600 DATA,,64,,84,,85,,85,64,,53,8
0,,1,84,,255,86,,12,210,120,,10,120
3040 B=57344
3050 READZ:IFZE=-1THEN3300
3060 FORT=0T07:READD:POKEB+ZE#0+T,Q:NEXT
3070 GOT03050
3080 DATA27,0,0,4,6,21,21,102,89,20,0,0,
0,64,64,64,80,80,30,0,0,0,252,253,253
3090 DATA253,253,31,0,0,0,0,0,0,85,93,
0,1,1,5,22,86,85,35,85,101,85,149,85
3100 DATA86,102,85,36,140,64,101,85,153,
166,105,85,37,0,0,0,64,80,80,85,38,255
3110 DATA255,255,255,255,255,253,253,39,
255,255,255,255,255,255,2,40,255,255
3120 DATA255,255,255,255,0,41,255,255,
255,255,255,170,186,42,255,255
3130 DATA255,255,255,255,170,171,43,255,
255,255,255,255,0,126,44,253,253,253
3140 DATA255,255,255,255,255,45,10,42,85
,,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 DATA255,255,255,255,51,120,160,85,2
55,255,255,255,52,170,170,170,170
3180 DATA170,170,170,170,53,85,85,85,85,
85,85,85,85,54,0,0,85,255,255,255,255
3190 DATA255,55,0,0,1,85,85,21,0,0,50,0,
5,85,85,85,5,0,57,0,80,84,85,85,85,85
3200 DATA5,50,0,0,0,0,85,85,85,85,59,0,0,
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
,,106,86,86,106,86,86,106,94,65,101,102
3230 DATA102,106,86,86,86,95,106,101,
101,106,86,86,106,86,85,106,101,101,106
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 DATA101,37,37,37,37,37,37,37,102
,,80,80,80,80,80,80,80,80
3270 DATA103,37,37,37,9,2,0,0,104,80,0
6,80,96,120,0,0,0,105,85,85
3280 DATA85,85,170,0,0,0,106,85,85,85,85
,,85,150,40,0
3281 DATA1,160,160,160,160,160,160,160,
0,1,85,106,102,102,106,102,102,2
3282 DATA95,105,102,102,105,102,102,105,
3,85,106,101,101,101,101,106,4,85
3283 DATA105,102,102,102,102,102,105,5,8
5,106,101,101,106,101,101,106,6,85
3284 DATA106,101,101,106,101,101,7,8
5,106,101,101,101,102,102,106,8,85
3285 DATA102,102,102,106,102,102,102,9,8
5,89,89,89,89,89,89,10,85,106,86
3286 DATA86,86,102,102,102,106,11,85,102,102
,,102,105,106,102,102,12,05,101,101,101
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 DATA101,101,101,17,05,106,102,102,1
02,102,106,106,10,05,106,102,102,106
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 DATA23,95,102,102,102,102,106,106,1
02,24,85,102,102,102,89,102,102,102
3293 DATA25,85,102,102,102,102,89,89,89,2
6,85,106,86,89,89,101,101,106
3299 DATA-1
3300 PRINT"*****"
3310 PRINT" [E
[E
3320 PRINT"+++++###+++++*****"
+++++###+++++
3330 PRINT"((('&')#)*)#&(((((((&(((((((
('&')#)#+&*)
3340 PRINT"666,-./0123,66666666,6666666
6,-./0123,*)
3350 FORT=49632T049751:POKET,160:NEXT
3360 FORT=49752T049911:POKET,52:NEXT
3370 FORT=49912T050151:POKET,53:NEXT
3380 PRINT"*****799:*****
799: 799: "
3500 POKES3270,216:SYS30000:FORT=1T0500:
NEXT:POKE49169,96
9999 GOTO9999

```

# Der Star-Drucker NL-10 mit Commodore-Interface

Der NL 10 hebt sich schon durch sein gut gelungenes Design aus der großen Masse der Mitbewerber in dieser Preisklasse (Listenpreis: 1145,-DM) heraus. Der erste positive Eindruck nach dem Auspacken sollte sich im Verlauf des Tests bestätigen.

Geliefert wird der NL 10 (einschließlich eines Interfaces) wahlweise für den Commodore 64, den IBM PC oder für Computer mit Centronicschnittstelle (z. B. die CPCs von Schneider). Somit muß bei der Anschaffung eines neuen Systems oder dem Betrieb mehrerer unterschiedlicher Computertypen jeweils nur das Interface ausgewechselt werden. Der Test bezieht sich auf die Version mit dem Commodore-Interface, das den NL 10 zusätzlich mit dem Befehlssatz des Commodore MPS 803 ausstattet.

Das Interface wird in den dafür vorgesehenen Steckplatz an der Rückseite des Gehäuses eingeschoben und mit einer Schraube befestigt. Der Anschluß an den Commodore 64 kann nun über das gebräuchliche serielle Kabel erfolgen, wobei eine zweite Buchse am Interface den Anschluß weiterer Peripheriegeräte erlaubt. Neben dem Steckplatz für das Interface befindet sich acht Dip-Schalter, über die folgende Betriebsmodi fest eingestellt werden können: Auswahl eines der acht internationalen Zeichensätze, Commodore- oder ASCII-Modus, Papierlänge elf oder zwölf Zoll, Gerätemummer vier oder fünf, Anzeige des Papierendes und zusätzlicher Zeilenvorschub.

An der Vorderseite des Geräts gestattet ein Kontrollfeld aus fünf Bedienungselementen und sieben Leuchtdioden die direkte Wahl der vier Schriftarten

(MODE), eines Fettschriftmodus (BOLD) und des Blattvorschubs (TOF). Komplettiert wird es durch den ON LINE-Taster und den Taster für den Papiervorschub (PAPER FEED). Darüber hinaus stehen dem Anwender noch acht Sonderfunktionen zur Verfügung, die über Tastenkombination angewählt werden: Zwei verschieden lange Selbsttestmodi, die Einstellung des linken und rechten Blattendes, des Blattanfanges und des Papiertransports in  $\frac{1}{2}$  Zoll Schritten in beide Richtungen. Außerdem besteht die Möglichkeit, einen Hex-Dump auszuführen, der die korrekte Übertragung der ASCII-Codes zwischen Computer und Drucker überprüft. Die Power-LED dient auch der Anzeige von evtl. auftretenden Fehlern (blinkt z. B. bei fehlendem Papier).

Das Einlegen von Endlospapier oder Einzelblättern gestal-

tet sich problemlos, unterstützt durch einen Hebel an der linken Seite des Gerätes, der den Druck des Papiers auf die Walze kontrolliert. Selbst der Farbbandwechsel erfordert nur wenige Handgriffe. Zu bemängeln ist die Konstruktion der Staubschutzhaube, die während der Verarbeitung von Endlospapier nicht abgenommen werden kann, da sie nach hinten geschlossen ist und das Papier durch die Haube geführt wird.

Dies und die Tatsache, daß der NL 10 beim Abnehmen der leider fast völlig undurchsichtigen Haube sofort Off-Line geschaltet wird, behindern diffizilere Arbeiten wie z. B. das Bedrucken von Etiketten oder Briefumschlägen. Bei der Verwendung von Einzelblättern kann die Abdeckung des Papiereinzugs hinter der Staubschutzhaube aufgestellt und zur Führung der Blätter genutzt wer-

## Musterausdruck (kleine Auswahl)

```

##EJH IZ The Quick Brown Fox Jumps Over The Lazy Dog
##SKOURGUB The Quick Brown Fox Jumps Over The Lazy Dog
##KOURGUB The Quick Brown Fox Jumps Over The Lazy Dog
##ADURGUB The Quick Brown Fox Jumps Over The Lazy Dog
##DURGUB The Quick Brown Fox Jumps Over The Lazy Dog
##GUB The Quick Brown Fox Jumps Over The Lazy Dog

```

```

##$Äöüäöüß The Quick Brown Fox Jumps Over The Lazy Dog

```

```

##$Äöüäöüß The Quick Brown Fox Jumps Over The Lazy Dog

```

```

STAR NL-10C VER 1.1

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```

```

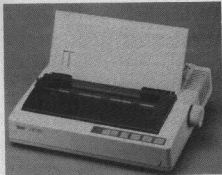
! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [ \ ] ^ _ ` a b c d e f g h i j k l m n o p q r s t u v w x y z { | } ~

```

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ 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 [ \ ] ^ _ ` 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 { | } ~

```



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 Befehlsatz 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 NLQ-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 Blattänder, horizontale und vertikale Tabulatoren und der Papiervorschub in beide Richtungen (z. B. in 21tel 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 Escapesequenz 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 Proportionalabschriftzeichen oder in NLQ-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 Basis-Beispielprogramme erleichtern auch dem Einsteiger die Einarbeitung in den umfangreichen Befehlsatz.

Mit der Arbeitsgeschwindigkeit von 120 Zeichen pro Sekunde (30 Zeichen pro Sekunde im NLQ-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 C64 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/FS.  
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 Microcomputer-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 Fehlerfunktionen vorbeugen.

Die Bereiche Hardware und Software werden getrennt behandelt. Eine ausführliche Fehlersuchtafel schließt den Hardwareteil ab. Im Softwareteil werden die Standardfehlermeldungen 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 Ausstöbern 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 Witt

## Green Beret

Daß wir auch bei indizierten Spielen (also Spielen, die die Gewalt verherrlichen) keinen Index liberorum 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 vorrücken: die Brücke, den Hafen, die Raketenstation und last not least das Gefangenenlager. 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

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



schußgeräte. Mit diesen geht das Abmürksen 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 spitzennmäßig gestaltet und erinnert leicht an einen Destruction-Mix. Es sieht so aus, als würden die Möglichkeiten des Sound Interface Device (SID 6581) im C:64 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 Nimmerwiederssehen 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 Schach

*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.*

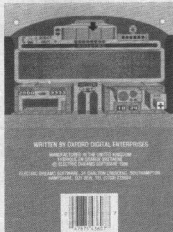
Verlag Rätz-Eberle,  
c/o Stefan Selbach,  
Postf. 1640, 7518 Bretten l.

## 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 Spieles "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 Feinsens ausgerüstet.

Ziel des Spieles ist es, mit T.S. 1 zur Titanic hinabzutauken, sich im Labyrinth der Räume der Titanic zurechtzufinden und den Auslöser der acht, vor kurzer Zeit angebrachten Luftschlägen, deren genaue Lage jedoch nicht bekannt ist, aufzustoßern, 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 überst ungünstig auf die Zahlungsmoral der Kapitalgeber auswirkt. Ohne deren Zuwendungen wäre das Unterfangen binnen kurzer Zeit zum Scheitern verurteilt.



**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 Bordtressen 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 Wolk

## 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 Topaganten 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 über 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.

Jah wird man aus diesem Naturschauspiel auf den schlüpfriegen Steg über einen moarstigen 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 Liebes 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 dieltantisch



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 Hühlig 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 Spritenaufbau eingegangen. Die Basic-Befehle sind alphabetisch aufgelistet und mit kurzen Beispielen und Hinweisen versehen (inklusive den Systemvariablen ST und TI bzw. TIS). 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



## 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-Buche, ein Einfüh-

runghandbuch 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 Buchlein 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, zurückgespalt, gestartet, gelüftet und geläutet, so wird einem vom Computer eine Fremdsprache (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 Entspannungsgrafik, während einem aus dem Recorder eine angenehme Frauenstimme entgegenstrahlt: "Erinnere Dich an eine Zeit, in der Dir das Lernen besonders Spaß gemacht hat."

Auch wenn letzteres schwerfallen sollte, so sind die nachfolgenden Lektionen wirklich nicht schlecht aufgezeigt. Unterhalt 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 ähnlichen Unertäglichkeiten.

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 "Texthandela" (ein pantomimisches Spiel), "Multiple Choice" (Beidungszuordnungsspiel 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  
Pasargartenstraße 4  
8000 München 83

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

Axel Klück

## Paint Boutique

Mal- und Zeichenprogramm 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, Groß- 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 Lappenbefehl,
- 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 Zusatzdiskette

kannte für DM 29,- bezogen werden. Laut Data Becker werden weitere Disks in Kürze folgen.

"Paint Boutique" vereint 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: C64  
 Hersteller: Data Becker  
 Preis: 69,- DM (Disk)  
 Frank Weik

## Commodore 128 Intern

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

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 Weik



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 gegenseitig konträre Meinung zum Ausdruck bringen und zur Diskussion stellen.

Zur Pro-Seite nimmt Thomas Tai aus Braunschweig 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 Pankto 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 Computerlai 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 eindrucksvoll 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 Computereiniger eine wertvolle Hilfe ist, stellen sich nach kurzer Zeit Probleme, die mit der Benutzeroberfläche nicht mehr zu lösen sind (Verkettungen, 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 steht, vor allem dann, wenn wegen jedem "Pip-fax" 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 zurück greifen 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 austatten (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 durch aus 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 in 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 richtungweisend 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 Elementen

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, Side-Car) 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 Soundunterlegung 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-PROGRAMME

SINCLAIR

Fant (nicht veröffentlicht), Pyramide (7/84), Superhit (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), Spielfeld mit Demo (3/85), Dats (3/85), Uhr (7/85), Roulette  
Best-Nr. S 10

Catalog (2/85), Software (2/85), Filroutine (2/85), Computer figures (2/85), Ku Bernd (2/85), Sterngrafik (2/85), Manic Train (3/85), Semo (4/85)  
Best-Nr. S 11

Schöffner (5/85), Spectrum-Incoystem (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), Münzlotto (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-Fischfütterung (9/85), Roby Doby (11/84), "Musk-Editor (4/85), HELPF 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 Panic (7/85) nur mit Encl. Funktions-Pfader (5/85), Blackcode (9/85), Jewel Jäger (5/85), Zeilen-Assembler (7/85), Joystick-Controller (9/85), Holztafel-Scrolling (5/85), "Converter (DOS II in DOS I) (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), Mines-Pat (1/85), Nova-Madaga (1/85), Parachute Jumper (5/84), Permanente Kleinbuchstaben (10/84), Passwörtergenerator (12/84), Screen Utilities (11/84), Burglar Time (12/84), Cowboy (7/84), Desert Flight (9/84), Fassadenkletterer (11/84), Hangman (nicht veröffentlicht), Mines-Pat (1/85), Nova-Madaga (1/85), Parachute Jumper (5/84), Permanente Kleinbuchstaben (10/84), Passwörtergenerator (12/84), Screen Utilities (11/84)  
Best-Nr. B 1 Diskette, B 1a Kassette

Alpha Lock (2/85), Cube (3/85), Epsilon II (4/85), Jünger (4/85), Macrocode (4/85), Merge-Filer (3/85), Motor ON (2/85), Pooyan (2/85), Progload (3/85), Rotation (3/85), Vokabel (2/85), Achtung: Macropede, Merge-Filer und Progload nur auf Diskette.  
Best-Nr. B 10 Diskette, Best-Nr. B 10a Kassette

Balade pour Adeline (nicht veröffentlicht), Flugsimulator (6/85), "Super Disk-Katalog (6/7/85), Plot (6/85), Plut (6/85), Säulendiagramm (5/85), Calculator (7/85), Tester (4/85), Würfel-Duell (7/85). \* Nur auf Diskette  
Best-Nr. B 11 Diskette, Best-Nr. B 11a Kassette

"Goto Jump (10/85), Cavern (9/85), Crazy Man (11/85), Devil Guest (10/85), Elitel-Star (11/85), High Res. Grafik mit Demo (11/85), Sound-Editor (10/85), "Spite-Utilities (11/85), "Sector 0 (9/85). \* Nur auf Diskette  
Best-Nr. B 12 Diskette, Best-Nr. B 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), Plättummengenanzer und Indikator (11/84), Grafik-Erweiterung (12/84), Bekliste (12/84), Photonx (1/85), Nüßbugging (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), Pugna (4/85), Delele (4/85), Merge (2/85), Find (2/85), Screen-Designer (2/85), Scr.Des.Obl C200 (2/85), Data-Gen (2/85), Rahmenstaben (3/85), Auto-Sortier 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 Grafikerweiterung Teil I und II sowie die Basic-Erweiterung EX3 V.1.8 kann auf Diskette / Kassette inklusive Anleitung bezogen werden (Siehe auch CK-Hefte 10/84, 12/84 und 5/85)  
Best-Nr. DS65 Diskette DM 13,-, Best-Nr. KS65 Kassette DM 10,-

## FREIE AUSWAHL

### 3er Pack

### 5er Pack

3 Kassetten nach Ihrer Wahl nur **40,-** 5 Kassetten nach Ihrer Wahl nur **60,-**  
3 Disketten nach Ihrer Wahl nur **50,-** 5 Disketten nach Ihrer Wahl nur **75,-**

# M M S E R V I C E

Paint (9/85), Window (9/85), 3-D Plotter (10/85), Superlist (10/85), Carpetide (10/85), Nitro Joe (11/85)  
**Best.-Nr. 5 30**

Hörschutz (1/86), The Servant (1/86), Snake-Smasher (3/86), Axi (3/86), List/LIST (3/86), Programmieren löschen (3/86), Spectrum-Gütekopie (3/86), On Error Goto (3/86)  
**Best.-Nr. 5 40**

**neu** Ass. Taps Uhr (7/86), GUP-Druckerroutine (7/86), Datenverschlüsselung (7/86), Pro Dos (7/86), Apfelmännchen (7/86), Teile (7/86), Dishmaster (5/86), Speachce (5/86)  
**Best.-Nr. 5 50**

**Kassette  
16,-**

DI Designer 64K (10/85), Joypoint 64K (10/85), Musicreator 64K (11/85), Chettedoktor 64K (1/86), Unprotector V 1.0 16K (1/86), Key Maker 16K (1/86)

**Best.-Nr. A 12 Diskette, Best.-Nr. A 13a Kassette**

Cherry Harry (3/86), Mission X auf dem Atlant (5/86), \*Basic-Erweiterung (5/86), Mini-Bildart (10/85), Zeichen-Zauberer (3/86), Sound-Demo (3/86) \*Nur auf Diskette

**Best.-Nr. A 13 Diskette, Best.-Nr. A 13a Kassette**

**neu** Revolver Kid (1/86), \*Tys-Dos (7/86), Text im Großdeuter (7/86), Rollenball (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,-**

Berböbe (3/86), Büchchen (5/86), \*Copy 5 (1/86), Hardcopy für Seikooha CP 100 (1/86), Lander (3/86), Moonrider (5/86), Polargrafik (3/86), T-Point (3/86), 40-Zeichen-Mode (3/86), Turtle Jumper (1/86), \* Nur auf Diskette

**Best.-Nr. II 13 Diskette, Best.-Nr. II 13a Kassette**

II-Assemblerdiskette 1 Für die II-Assemblertheoks 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. II 14 Diskette**

II-Assemblerdiskette 2 Diese Diskette enthält die Quellcode-Dateien für die Assemblerprogramme Adressdatei (4/86) und Disk-Utilities (8/86)

**Best.-Nr. II 15 Diskette**

**neu** \*Willy der Wurm (7/86), Seikooha GP 50 Hardcopy (7/86), Iason (7/86), \*Adressdatei (5/86), Soundeditor plus (9/86), Screen Editor (7/86), \*Assembler-Gratik-Generator (7/86), The Plague (9/86), \*Disk Utilities (9/86), \* Nur auf Diskette

**Best.-Nr. II 16 Diskette, Best.-Nr. II 16a Kassette**

**Diskette  
20,-**

**Kassette  
16,-**

Baloon (7/85), Rescue in the Stone-Age (6/85), Softfile (7/85), Sprite Editor (9/85), Data Loader (6/85), Rahmenroutine (7/85)  
**Best.-Nr. C 11 Diskette DM 20,-, Best.-Nr. C 11a Kassette DM 16,-**

Roadish One (10/85), 3-D-Hubschrauber Simulator (11/85), Head to Head Bathion (1/86)

**Best.-Nr. C 12 Diskette DM 13,-, Best.-Nr. C 12a Kassette DM 10,-**

Formel One (3/86), Directory in Farbe (3/86), Roadish in the Jungle (5/86)

**Best.-Nr. C 13 Diskette DM 13,-, Best.-Nr. C 13a Kassette DM 10,-**

Silflieger (7/86), Jumping Jive (9/86), Eastern Flight (11/85), Hardcopy C64 (11/86), Time Loader (nur für Diskette) (11/86)

**Best.-Nr. C 14 Diskette DM 13,-, Best.-Nr. C 14a Kassette DM 10,-**

**Sonder-  
für alle  
Prozessor-  
arten und  
Systemen**

**neu**

**SOFT-  
WARE-  
BESTELL-  
SCHEIN**

Coupe ausschneiden und einwickeln bei  
 Verlag Falck/Markt CD-Software  
 Postfach 1040, 7518 Eppingen



Anzahl	Bestell-Nr.	Bericht-Paket	Preis	Name des Bestellers
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____

Straße \_\_\_\_\_ PLZ/Ort \_\_\_\_\_  
 Telefon \_\_\_\_\_ Datum/Unterschrift \_\_\_\_\_  
 Ich erkläre folgende Bezahlung:  
 Nachnahme (zusätzlich DM 5,70 Porto- und Versandkosten)  
 Vorauskassa (ohne Versandkosten)  
 Bitte Schein beilegen oder auf Postkassentonto Karlsruhe 4042-756 überweisen.

# ATARI®

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 Maschinensprache-Monitor, 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 "Golf" und meinen in Assembler geschriebenen "Puzzler".

## Satte 320 KByte für den Atari 800 XL

Der ST hat es gezoigt: Mehrere 100 KBytes bis zu einem Megabyte Speicher für den häuslichen Computer sind heute kein Traum mehr. Moderne Speicherechips 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

Als "ernsthafte Anwendung" (was immer das sein mag) finden Sie ein Kartelkartenprogramm 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

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. Wenn sie einbauen lassen will, muß nochmal 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  
Gneisenaustraße 29  
4330 Mülheim

Peter Fintel

## 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 Tip 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 Schmo  
Heidtschackweg 7 Data-Light  
2120 Lüneburg 5481 Burg



# Neues für die Atari-Spielekiste

Eigentlich konnte man kaum noch Besseres erwarten. Schon die erste Ausgabe von "Spy vs. Spy" überzeuget durch das ausgefeilte Spielprinzip und durch farbenfrohe Bilder. Der Nachfolger dieses Spieles mit dem Untertitel "The Island Cap" 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 schlechte Neuaufgabe 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 Raketenstücken zu manchem

Hinterhalt kommen wird. Der smarte Spion erledigt so etwas mit seinem Trapatlor (neuestes Modell FSS 85), mit dem Gruben ausgehoben und Kokosnuss-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 gleichzeitigem 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 Schlusszenen reichlich zum Ausdruck kommt. Gesamturteil: Sehr empfehlenswert.

## Knockout

Videospiele rund um den Bocksport 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 Modellaautos 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 Mondoauto (!) reichen. Wenn 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 Ramschutz 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 Fissel



1. No Laufwerk f. ATARI ST 499.-  
Disk 3,5" 20 ab 55.-  
Grünmonitor f. ATARI ab 228.-  
unisolat soll  
3500 Casses  
u. Konj. str. 46A  
T. 0561/770367

Atari 800 XL/800 XL/130XE-User!  
Lichtstift mit Programmieren und  
deutscher Anleitung mit 49 u. DM.  
Wensand gegen Sachschaden  
me. Info gratis Firma Schilbauer,  
Postfach 1717A, 8452 Sulzbach, 8  
09661/6592 bis 21.00 Uhr.

## PETER'S ASSEMBLERECKE

### Display-List Interrupts

FM-Grafik, Scrolling und Page-Flipping sind alles Techniken, mit denen Atari-Computer vor einigen Jahren Meilensteine gesetzt haben. Doch das I-Pünktchen 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 Assemblerprogrammiern vorbehalten. Das ist natürlich ein Grund für die Assemblercke, sich dieses Thema 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 Assemblercken; 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 Farbregister 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 DLI's 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 NMEN (\$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 Register 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.

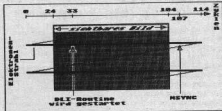
- Vor dem Verlassen der DLI-Routine müssen die ursprünglichen Inhalte der Register wiederhergestellt werden.

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 DLI's eine eigene Farbe zugewiesen. Die Farber sind in der Tabelle FARBTAB (ab \$603) zusammengefaßt und können z. B. von Basic geändert werden - eine praktische Methode, wenn man einzelne Zeilen hervorheben will. Im Basic Listing 3 ist ein kleines Demo enthalten, an dem Sie sehen was man alles damit anstellen kann.

Listing 2 zeigt Ihnen die inneren Vorgänge dieses schon relativ komplizierten Interrupts. Beispielsweise, 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 vorbeigegangen, 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 Rücksetzen des Farbregisters nötig war. Ganz einfach, weil die VBI-Routine im ROM des



Timing bei Farbwechsel

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 dabei 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 einzigen Farbgeregister geändert werden soll. Aber wenn man mehr als 3 Hardware-Register überschreiben will, muß schon mit jedem Maschinenzklus 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 Farbgeregister 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 Farbgenze auch noch hin und her und führt zu einer Verunstaltung der Grafik. Aus dieser Klemme hilft uns das Register WSYNC (5D40A, 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 Farbgeregistern vorbereitet wird:

```
PHA
TXA
PHA
TXA
```

```
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. Denkbare 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 Maschinenzklus rechnen muß. Ich erinnere mich noch bestens an einen DLI in meinem letztem Spiel, der vier Players versetzt, zwei Farben ändert und die Bildschirmbreite modifizieren sollte. Im wahrsten Sinne des Wortes - das kostete Schweiß!

Peter Fissal

## 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:PDKE 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 * Vielfarbeeft in BR.0
120 REM * BASIC-Loader
130 REM * P. FINZEL 1986
140 REM *****
200 MPGM=1536:FARBTAB=1540
220 GOSUB 500
230 X=USR(MPGM)
300 FOR I=0 TO 23
310 X=PEEK(FARBTAB+I)
320 POKE FARBTAB+1,14
330 FOR T=0 TO 50:NEXT T
340 POKE FARBTAB+1,X
350 NEXT I
360 GOTO 300
500 REM * Vielfarbeeft 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 $D018 Hardware-Reg. fuer Farbe
WSYNC EQU $D40A Warten auf Zeilenende
NMIEN EQU $D40E NMI-Freigabe

VDSLST EQU $0200 Vektor fuer DLI
VVVLKI EQU $0222 VBI-Vektor (immediate)
BDLSTL EQU $0230 Zeiger auf D-List
BYSVBV EQU $E45F VBI-Routine im ROM

HILFIZP EQU $D4 Hilfsregister in Zero-P.
*
* in ATNAS Monitor mit 8 0601 starten
*
ORB $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 HILFIZP f. indirekte
STY HILFIZP+1 Adr.-Art
LDY #2 DLI-Bit in
LDA (HILFIZP),Y dritter Leerzeile
ORA #80 setzen
STA (HILFIZP),Y
INY DLI-Bit in der
LDA (HILFIZP),Y LMS-Anweisung
ORA #80 der D-List setzen
STA (HILFIZP),Y
INY LMS-Adresse
INY ueberspringen
INTSET INY Schleife f.
LDA (HILFIZP),Y DLI-Bit in restl.
ORA #80 Display-List
STA (HILFIZP),Y
CPY #27 schon alle Zeilen?
BNE INTSET nein-->
*
* DLI und VBI aktivieren
*
LDA #0 Interrupts aus
STA NMIEN (VBI & DLI)
LDA $DLIPGM: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 NMIEH       freigeben
RTS

```

\*  
\* DLI-Routine Vielfarben-Effekt in GR.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
TXN            Zeiger weiter-
                schalten
BTX ZEIGER
PLA            X-Reg. vom Stack
TAX
PLA            Akku besorgen
RTI            Schluss!

```

\*  
\* VBI-Routine Vielfarbeneffekt

```

VBIPGM LDA @0      Zeiger auf Anfang
        STA ZEIGER  der Farbtablelle
        JMP SYSVAV  VBI weiter

```

```

11 POSITION 0,22:7 M0;"5C";5C:POSITION
   N 15,22:7 M0;"L";L:POSITION 6,23:7 M0
;"LEVEL ";M1
12 POKE 756,POK:POK+4:IF POK:PAGE+
  4 THEN POK+PAGE
13 POKE 77,0:5=PEEK(6323):FOR M=1 TO IX
:NEXT M
14 IF 5=1 THEN XR=-1:VR=0:CHR=163
15 IF 5=7 THEN XR=1:VR=0:CHR=161
16 IF 5=13 OR 5=9 OR 5=8 THEN XR=0:VR=
  1:CHR=164
17 IF 5=14 OR 5=10 OR 5=6 THEN XR=0:VR
  -1:CHR=165
18 SOUND 0,50,0,12:COLOR N:PLOT X,Y:N=
  X+XR:Y=Y+VR:LOCATE X,Y,Z:COLOR CHR:PL0
  19 IF Z=7 THEN 39
20 IF Z<32 THEN 32
21 GOTO 12
22 IF BI=0 THEN BI=1
23 POKE 559,0:GOSUB 26:10M BI GOSUB 65,
  60,73,77,81,85,92,98
24 GOSUB 25:POKE 559,34:CHR=164:FRU=M1
  +4
25 RETURN
26 COLOR 8:PLOT 0,0:DRAMTO 19,0:DRAMTO
  19,21:DRAMTO 0,21:DRAMTO 0,0
27 F=INT(CHR(0)*3)+10
28 X=9:Y=10:N=32:COLOR 164:PLOT X,Y:TX
  =TIM:RETURN
29 A=INT(CHR(0)*8)+1:1=0=INT(CHR(0)*9)
+1:LOCATE A,B,C:IF C<32 THEN 29
30 COLOR F:PLOT A,0:RETURN
31 REM [ ]
32 COLOR 82:PLOT X,Y:GOSUB 17
33 FOR M=120 TO 150:SOUND 0,M,10,10:50
  HND 0,M,1,10,10:POKE 756,POK:POK+4
  :IF POK:PAGE+4 THEN POK+PAGE
34 NEXT M:GOSUB 37
35 L=L-1:IF L<0 THEN 40
36 N=32:NR=0:VR=0:CHR=164:TI=TIM:GOTO
  58
37 FOR M=0 TO 3:SOUND M,0,0,0:NEXT M:R
  ETURN
38 REM [ ]
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=50:50:TI=TI-5:N=130
41 FRU=FRU-1:IF FRU<1 THEN M1=M1+1:BI=
  BI+1:FRU=M1+4:NR=0:VR=0:SC=50+200:GOSU
  B 43:GOTO 10
42 GOSUB 29:GOTO 11
43 FOR M=15 TO 0 STEP -2:FOR M1=00 TO
  00 STEP -1:SOUND 0,M1,10,M:SOUND 1,M-
  10,10,10: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 0=50 TO 100 STEP 2
  :SOUND 0,0,10,M:NEXT 0:BIET M1=L+1
46 SOUND 0,0,0,0:POSITION 17,22:7 M0:L
  47 RETURN
48 FOR M=15 TO 0 STEP -1:FOR M1=20 TO
  35 STEP 2:SOUND 0,0,50,M1,10,10,M:NE
  XT M1:NEXT M:SOUND 0,0,0,0
49 POSITION 5,10:7 M0;"GAME OVER"
50 FOR M=0 TO 1000:NEXT M:GOTO 8
51 GRAPHICS 17:POK:PAGE:POKE 708,150
52 POSITION 1,5:7 M0;"[ ]"
53 POSITION 1,2:7 M0;"Wolfgang heencot
  h"
54 POSITION 0,9:7 M0;"((( (( ( ( ( ( (
  ((
  ((
55 POSITION 0,10:7 M0;"( ( ( ( (
  ((
  ((
56 POSITION 0,11:7 M0;"( ( ( ( (
  ((
  ((
57 POSITION 0,12:7 M0;"( ( ( ( (
  ((
  ((
58 POSITION 0,13:7 M0;"((( (( ( ( (
  ((
  ((
59 POSITION 5,17:7 M0;"SCORE:";5C
59 POSITION 4,20:7 M0;"[ ]"
59 POSITION 1,22:7 M0;"[ ]"
60 POKE 756,POK:POK+4:IF POK:PAGE+
  4 THEN POK+PAGE
61 FOR M=1 TO 20:NEXT M

```

## 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 Heencoth

```

1 PAGE=PEEK(106)-12:CH5=PAGE*256
2 POKE 559,0
3 FOR I=120 TO 471:POKE CH5+I,PEEN(572
  44+I):POKE CH5+1024+I,PEEK(57544+I):N
  E XT I
4 READ C:IF C=-1 THEN 6
5 FOR I=0 TO 7:READ A:POKE CH5+CH0+I,0
  :NEXT I:GOTO 4
6 READ C:IF C=-1 THEN 8
7 FOR I=0 TO 7:READ A:POKE CH5+CH0+I+1
  024,0:NEXT I:GOTO 6
8 GOSUB 51
9 BI=L:NR=0:VR=0:CHR=164:TI=50:N=32:T
  I=TIM:SC=0:1=4:M0="I:FRU=M1+4
10 GRAPHICS 17:POKE 756,PAGE:POK:PAGE:
  POKE 711,36:GOSUB 22

```

```

62 IF PEEK(53279) < 6 AND PEEK(6443)=1 T
63 EN 63
64 RETURN
65 REM *****
66 POKE 709,64
67 RETURN
68 REM *****
69 POKE 709,136:POKE 708,12
70 COLOR 39:PLOT 5,9:DRAMTO 5,5:DRAMTO
8,5:PLOT 11,5:DRAMTO 14,5:DRAMTO 14,9
79 PLOT 6,14:DRAMTO 13,14
71 RETURN
72 REM *****
73 POKE 708,164:POKE 707,244
74 COLOR 41:PLOT 1,1:DRAMTO 6,6:PLOT 1
8,1:DRAMTO 11,6:PLOT 1,20:DRAMTO 6,15:
PLOT 18,20:DRAMTO 13,15
75 RETURN
76 REM *****
77 POKE 709,182:POKE 708,180
78 COLOR 45:PLOT 4,4:DRAMTO 9,9:PLOT 1
0,9:DRAMTO 16,3:PLOT 3,17:DRAMTO 9,11:
PLOT 18,11:DRAMTO 15,16
79 RETURN
80 REM *****
81 POKE 709,186:POKE 708,20
82 COLOR 48:PLOT 7,1:DRAMTO 2,8:PLOT 1
2,1:DRAMTO 17,8:PLOT 2,12:DRAMTO 7,17:
PLOT 17,12:DRAMTO 12,17
83 RETURN
84 REM *****
85 POKE 709,70:POKE 708,230
86 COLOR 47:PLOT 3,15:DRAMTO 3,19:DRAM
TO 16,18:DRAMTO 16,15
87 PLOT 5,6:DRAMTO 1,3:DRAMTO 16,1:DRAM
TO 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 *****
92 POKE 709,212:POKE 708,118
93 COLOR 60:PLOT 8,3:DRAMTO 3,3:DRAMTO
3,6:PLOT 13,3:DRAMTO 16,3:DRAMTO 16,6
:PLOT 3,15:DRAMTO 3,18:DRAMTO 6,18
94 PLOT 13,18:DRAMTO 16,18:DRAMTO 16,1
5:PLOT 6,7:DRAMTO 8,9:PLOT 13,7:DRAMTO
11,9
95 PLOT 6,14:DRAMTO 8,12:PLOT 13,14:DR
AMTO 11,12
96 RETURN
97 REM *****
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:DRAMTO 3,3:DRAMTO 3,5:PLOT
1,3,16:DRAMTO 3,18:DRAMTO 5,18
101 PLOT 14,18:DRAMTO 18,18:DRAMTO 16,
15:PLOT 14,5:DRAMTO 16,1:DRAMTO 16,5
102 RETURN
103 DATA 1,96,215,255,190,192,112,80,1
80
104 DATA 2,17,68,17,68,8,127,187,127
105 DATA 3,6,235,255,125,3,14,18,5
106 DATA 4,28,62,42,62,28,28,28,8
107 DATA 5,28,62,62,62,62,28,16,48
108 DATA 6,255,129,129,153,153,129,129
,255
109 DATA 7,182,153,165,98,98,165,153,1
82
110 DATA 8,56,92,188,12,27,54,62,24
111 DATA 9,68,66,151,165,165,153,66,60
112 DATA 10,4,8,8,28,28,62,62,28
113 DATA 11,8,4,7,8,14,28,128
114 DATA 12,8,12,16,38,118,116,124,48
115 DATA 13,8,28,28,127,28,127,28,28
116 DATA 14,54,121,121,54,54,79,79,54
117 DATA 15,8,28,34,73,28,73,34,28
118 DATA 28,68,98,153,255,255,153,98,6
8
119 DATA 29,68,66,129,129,129,129,66,6
8
120 DATA 30,8,68,66,66,66,68,8,-1
121 DATA 1,96,288,254,255,247,112,80,4
8

```

```

122 DATA 2,68,17,68,17,8,127,187,127
123 DATA 3,6,11,127,255,233,14,18,28
124 DATA 4,28,62,42,62,28,28,28,8
125 DATA 5,28,62,62,62,28,4,6
126 DATA 6,255,195,165,153,153,165,195
,255
127 DATA 7,182,153,129,98,98,129,153,1
82
128 DATA 8,56,92,188,12,27,54,62,24
129 DATA 9,68,66,151,165,165,153,66,60
130 DATA 10,4,8,8,28,28,62,62,28
131 DATA 11,8,4,7,8,14,28,128
132 DATA 12,8,12,16,38,118,116,124,48
133 DATA 13,8,28,28,127,28,127,28,28
134 DATA 14,54,79,79,54,54,121,121,54
135 DATA 15,8,28,42,85,42,85,42,28
136 DATA 28,68,182,221,129,129,251,182
,68
137 DATA 29,8,8,24,38,36,24,8,8
138 DATA 30,8,38,36,36,36,38,8,-1
139 REM *****
140 REM *
141 REM * MOLFGBRG HEENSOTH *
142 REM * (C) 1985 *
143 REM * 2818 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 Programmabläufe rascher durchschaut.

Nach dem Drücken der Reset-Taste ist Slow Motion ausgeschaltet, läßt sich aber mit X=USR (1536) wieder starten. Frank Heindrich

```

29000 REM *****
29018 REM *SLOW-MOTION*
29028 REM *****
29030 DIM HENS(3):ZELLE=1536
29040 READ HENS:IF HENS="END" THEN M=H
58(15360):GOTO 29100
29050 HI=ASC(HEX$(1,1))-48:IF HI>9 THE
M=HI-7
29060 LO=ASC(HEX$(2,2))-48:IF LO>9 THE
M=LO-7
29070 MERT=HI*16+LO
29080 POKE ZELLE,MERT
29090 ZELLE=ZELLE+1:GOTO 29060
29100 DATA 87,87,82,86,88,88,20,5C,E4,
6E
29118 DATA 68,83,88,8D,1F,DD,AD,1F,8D,
C9
29120 DATA 86,78,8B,C9,82,F8,14,C9,84,
F8
29130 DATA 18,4C,62,E4,82,30,80,48,88,
88
29140 DATA F8,CA,8D,F8,4C,62,E4,82,2D,
88
29150 DATA C9,82,F8,EC,CE,25,86,4C,22,
86
29160 DATA 8D,25,86,C9,68,F8,DF,EE,25,
86
29170 DATA 4C,22,86,END
29180 ? "4":FOR I=29080 TO 29180 STEP
10:I:NEXT I:"POKE 842,121."I:POSITI
ON 8,8:POKE 842,13:STOP

```

# 1 Ohrenschmaus: Soundmaschine

Verstärker, 10 Lautsprecher, Schallkopf, 80 bis 2000 Hz, auch von eigenen Programmen steuerbar, Frequenz über Tastatur oder Joystick, 360 Grad um 2 Dimensionen drehbar, ausführliches Handbuch.

ATARI 400 - 130XE, 16/80K

Best.-Nr. AT 1

29.80 DM

# 2 Bargeldlos: Slotmaschine

Spezialfilm zum Spieltastentrieb, von Konigsee-Beispiel dringend zu empfehlen.

ATARI 800 XL/1040K/160XL

Best.-Nr. AT 2

Diskette 29,- DM

# 3 Vielfalt: Atari Power Superbuch

Benutzungs-, Lese-, Tips & Tricks - 77 Seiten DIN-A4, nicht nur für Anfänger, erhältlich.

Best.-Nr. AT 3

20,- DM

# 4 Nachschlag: Die Hexenküche

Ausführlich für Disks/Ausgaben und Profis durchgearbeitet, Tips & Tricks, Kaffee, Danks etc., Maschinenreparatur-Programme als Leitfaden, immer fertig. Abbei geben so leicht wie (und Sie auch!)

Best.-Nr. AT 4

29.80 DM

# 5 Zuschlag: Disk zu Hexenküche

Dabei kann man viel Zeit sparen.

Best.-Nr. AT 5

19.80 DM

# 6 Fix: ATMAS-II Macro Assembler

3K Code für 4 Sekunden zum Start, Erweiterung von Diskassembler, Post-Processor, schnell zu beiden Plattformen, integrierter Monitor, 50-seitiges Handbuch und Disk mit Eingabebefehl.

ATARI 400 - 130XE

Best.-Nr. AT 6

Diskette 49,- DM

# 7 Geistesparend: ATMAS Toolbox

Rechnerwörter-DIO-Makro-Conversion-For-3038, Sammlung von noch dazu mehr Auf Disketten mit Anleitung dazu.

ATARI 400 - 130XE, 16/80K

Best.-Nr. AT 7

19.80 DM

# 8 Praktisch: Monitor XL

Verknüpft Basic-Programme mit Menüs, Formeln, Eingaben, Ausgaben, Listen, Single-Step, Disketten/Queltext, Drucken-Anzeige, drehbare Tastenmechanik, auch für Rote und DGs, über 1000-Speicherplatz, mehr als 1000er-Ausprägung und Disk.

ATARI 800 XL/16K/160XL/130XE

Best.-Nr. AT 8

19.80 DM

# 9 Brandneu: Design Master

Lösung über Fritze-Technik, Auflösung 300 x 192, Füllkreis-Markierungen ein/auswählbar, 2 Seiten gleichzeitig, über 120.000 Punkte im Drehtast, über 100 verschiedene Schriften, Handcopy für Text mit Maus, Tracker (als Nachteil), Ausdrucken verschiedener Größen möglich, ausführliche Anleitung.

ATARI 800 XL/16K/160XL/130XE

Best.-Nr. AT 9

Diskette 19.80 DM

# take 9 !!



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 Bestellung:

Anschrift \_\_\_\_\_  Nachnahme (Zutlg. DM 5,70 Versandkosten)

F.L./Ort \_\_\_\_\_  Vorkasse (ohne Versandkosten)

Datum/Umschrieb \_\_\_\_\_ Bei Vorkasse bitte schnell belegen oder überweisen auf Postcheckkonto Kontonr. 43423-755

Coupons ausschneiden, auf Postkarte kleben und versenden an:  
Verlag Rötz-Eberle/CX-Software, Postfach 1640, 7518 Brehlen

# 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) anbietet. 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 USER-Befehl aufrufen 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 Programmierers 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 Programmentwicklung auf Maschinenebene.

Verwendbar für unsere Zwecke ist das Editor/Assembler-Modul von Atari, der MAC05 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 eingeben werden. So ist es z.B. nicht möglich, zwei Befehle in eine Zeile zu schreiben. Sehen wir uns dazu in Bild I 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 in endgültigen Programmen. 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 Beschreibung 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 "w", 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:

- 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.
- Eine andere Möglichkeit wäre es, den Inhalt einer Speicherzelle in den Akku zu schufen (mehr oder weniger ein PEEK). Dazu dient die "absolute" Adressierung. Der Operand wird hier als direkte Adresse aufgeführt. 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 Adressierungsarten, 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 Variable 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.

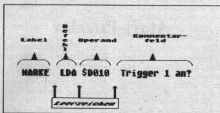
In der Schreibweise unterscheidet sich diese Adressierungsart durch das an den Operanden angehängte Index-Register. Nicht alle Befehle lassen allerdings die Wahl zwischen X und Y frei; bei manchen ist das fest vorgeschrieben. Bei LDA sind beide Möglichkeiten zulässig.

#### 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

Format	Label	Flag	N, Z
adr. - Art	Syntax	opcode	Länge
Immediate	LDA #dat	07	2
Zero-Page	LDA Zadr	05	2
Zero-Pg., X	LDA Zadr, X	05	2
Absolute	LDA adr	06	3
Absolute, X	LDA adr, X	06	3
Absolute, Y	LDA adr, Y	06	3
(Indir.), Y	LDA (Zadr), Y	01	2
(Indir.), X	LDA (Zadr), X	01	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 indiziert, 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 48K. Ich hoffe, daß es trotz der vielen Daten ebenfalls seine Fans findet. Um es unseren Lesern so bequem wie möglich zu machen, habe ich den "Puzzler" mit einer Zeilen-Prüfsomme 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 Binärlad-Funktion 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 Farbweider-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 löst dann das angezeigte Bild. "Puzzler" kann sowohl Bilder von der Maltaf (bzw. dem Koolapad) als auch in 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 6x10 Felder aufgeteilt, die der Computer munter durcheinanderschiebt. 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 Thoenes

## Puzzler

```

10 REM *****
20 REM * Puzzler (c) by TTh 86
30 REM *****
40 REM * ein Maschinensprache-Spiel
50 REM * von Thomas Thoenes
60 REM * Road 17 + 8961 Sulzberg
70 REM * Tel. (08376)/1664 ab 19.00
80 REM *****
90 REM * Vier ATARI ab 64K und Disk
99 REM *****
100 DIM A$(22):I%=1000
101 PRINT CHR$(123):"Puzzler:"
102 PRINT "Bitte DOS II Disk"
103 PRINT "einlegen und RETURN"
104 PRINT "druecken.":INPUT A$
105 PRINT "Bitte etwas Geduld!":?

```

```

189 OPEN #1,8,"D:TEST.COM"
110 READ A$,S$IF A$="M" THEN END
120 S=0:FOR N=1 TO 21 STEP 2
130 B$=200
140 S=S+A$PUT #1,A
150 NEXT N:?' '
160 IF S<>N THEN ?"FEHLER IN 'Z$!":
  ?":END
170 Z$=Z$+10
180 GOTO 110
200 A=(ASC(A$(N))-48)-(ASC(A$(N+1))-68)*7
  +16*(ASC(A$(N+1))-48)-(ASC(A$(N+1))-68)
  *7
210 RETURN
1800 DATA FFFF004F0400900000406,1289
1900 DATA 80C86A978D1086A9918D,889
18020 DATA 1286A9930806A8C2611A7F,630
1830 DATA 80E01A8A990A85220A454,778
1840 DATA A9080CF82201F44A9980,1081
1850 DATA 1486280947201E434C9952,772
1860 DATA A218A9C9D428328564A9,1180
1870 DATA 839A4283A94394483A945,931
1880 DATA 9D4583A9849D4A8328564,982
1890 DATA 18834CA9C68A218A559F9,1824
1900 DATA 4483A559D4503A980940,952
1110 DATA 83A91E93A983A987934283,805
1120 DATA 285E6418834CA9C284143,850
1130 DATA 60C8220A14380C4822841,943
1140 DATA 4380C8220A14380C482284,1874
1150 DATA 18A9C9D428328564A950,1822
1160 DATA 83CA53983CEA98802A86,1283
1170 DATA 8D2A68031C86288A267B,1579
1180 DATA 442893A9A9885A0AD1F08,1138
1190 DATA C98308628F64C28F6428,1158
1200 DATA EE43AD7882C98F8C26A7B,1579
1210 DATA 82E88F98F9C980886288E,1178
1220 DATA 414CC448C98F78886288D4,1861
1230 DATA 4CC448C98F788F7418086,1393
1240 DATA 288D42CC448C98D8828C,1889
1250 DATA 98424CC4488D2886C98F78,1213
1260 DATA 7285F43A288A988892286,825
1270 DATA 8D2886A5C8883C88288C8,1593
1280 DATA A98083888A88718882881,748
1290 DATA C32A91C8888C88F88F48,1748
1300 DATA EE886A8D8886C98D88E518,1183
1310 DATA ASC8672883C88CC8886,1456
1320 DATA CCE2286A8D2286C92888C,1337
1330 DATA E8818D8D418A83C698485,1496
1340 DATA C8ASC8A98883CCE2886A,1476
1350 DATA 1C86E88D445A8A988D4A,1258
1360 DATA 4586C9D445E88E1C868,1249
1370 DATA AD286A988F87828F43A9,1141
1380 DATA 888D288A83A8C8E98485,1147
1390 DATA C8ASC8E9888C8A98882,1492
1400 DATA 86A5C88C8A8C8E98C8A9,1589
1410 DATA 8D888A88818882881C86A,873
1420 DATA 91C888C888888F48888,1558
1430 DATA 86A8888C98288E18A8C8,1225
1440 DATA 692883C8A8C8A9888CCE,1578
1450 DATA 228A8D2286A9288C9288C8,1351
1460 DATA 18D887C2886A81C8F813,1415
1470 DATA 4286C88A445A8A988D4A,1258
1480 DATA 458688D445E88E1C868,1249
1490 DATA A3248A98C8F78A988D2,1128
1500 DATA 86288F4318A8C8D62888C,1875
1510 DATA ASC8E9888C8A288488A8C,1463
1520 DATA E92885C8A8CCE9888D8A,1716
1530 DATA 8881C891C8A98891C88C8,1641
1540 DATA 8A88F318A8C8692888C8A,1493
1550 DATA C6A98885C8A8E8818D2C,1697
1560 DATA 8886E2286A8D2286C9288,1387
1570 DATA 8718A8C8E98885CCE2886,1387
1580 DATA 18A81C286788A88A8445A,1818
1590 DATA A988D44A45E1C8688D4A,1192
1600 DATA 45A1C86888A8D1C868A8D,835
1610 DATA 248AC988D8834C1D43A98,795
1620 DATA 8D2286288F4318A8C8E988,888
1630 DATA 85CE38A8C8E92888C8A8C,1745
1640 DATA E9885C8A2883A8C8E928,1493
1650 DATA 85C8A8CCE9888D8A8881,1628
1660 DATA CF91C8A98891CF8C888A8,1688
1670 DATA F338A8C8E92888C8A8CCE,1778
1680 DATA 8885C8A8E8818D82C8E886,1394
1690 DATA EE2286A8D2286C928888A,1178
1700 DATA F442E943A8CCE98A8CCE,1175
1710 DATA 248A8D1C8638E98A8A8A,1871
1720 DATA 45A88A988D445A81C868,1812
1730 DATA 7D44A5A8D18638E98A8D1C,186A
1740 DATA 86A8A8C88283CFAD318285,1822
1750 DATA D8A8831CF898F887C8A,1498
1760 DATA F8834C8A3E98191CF8C8,1423
1770 DATA C788A88E888A4218A987,1335
1780 DATA 9D4283A9889D8839D4983,868
1790 DATA 285641883CA894C8A88A,963
1800 DATA 682883A3A983888F82A98A,1885
1810 DATA 888182A988851488888A8,994
1820 DATA 8886888882E8888628843,842
1830 DATA C98388F8A988D888D2A98,1581
1840 DATA 888182688A8514A84148F,1563
1850 DATA 868A88A98881E88A81F8,973
1860 DATA A9818D828628C43A81888,1813
1870 DATA D8F8A9888D8286A88286,1851
1880 DATA F888A81E86A9818D1E86A,889
1890 DATA 83EE1F86A8A9818A828A,927
1900 DATA 6932883A8D842186932A8,788
1910 DATA 83A88D41869328883A98,831
1920 DATA 421869328883A8E414CC,821
1930 DATA 43A288F43E8A48D445D,1482
1940 DATA 68A5888286E8883C88F3A98F,1578
1950 DATA 2889A4A9818D128A9888D,982
1960 DATA 88A8A281147A9828D1A8A9,645
1970 DATA 888A928A44A44C8D4468A,1826
1980 DATA 88A98D94283A9459D4A83,866
1990 DATA A9458A58A9A149D88A9A,1857
2000 DATA 889D4983285A8E42818A8D,888
2010 DATA 4883C98188834C4A48A888,1837
2020 DATA 4283A81A8A8D88A488A83,883
2030 DATA E88D8A988A83988D4883,1289
2040 DATA A9888D4983A288285E4A8,998
2050 DATA A2888E2286A55888C8A59,1191
2060 DATA 8D888881CF9336A8E38,1598
2070 DATA C88A88F318A8C8A9288C8,1538
2080 DATA 988E6A88E228A8D2286C,1276
2090 DATA 2888E6A8E2888E228A85,1155
2100 DATA 85CF4589588A8888D3A6,1488
2110 DATA 91CF88C88A8D8F318A8C8,1829
2120 DATA A92888CF9882E6A88E2286,1347
2130 DATA A8D2286C928D8E8A8A9818D,1283
2140 DATA F88A281E8685A8A1F8A85,1139
2150 DATA 8528A8A9A98388F3A9A785,1583
2160 DATA F428E6A8D2832E8C4A8E45A5,1477
2170 DATA A5F38D4483A5F48D458A9,1141
2180 DATA 898A4283A98C8D4883A98,785
2190 DATA 8D4A93C88288288A8A1F08,1137
2200 DATA C9A88F92818A8A8C88A88,1872
2210 DATA 88E988112818A8A92888,958
2220 DATA 1486A28D947201E434C8A8,785
2230 DATA 4C88A8E8F88F88F31F8,929
2240 DATA 7F91F38A99891F368A43,1649
2250 DATA 28282828282828282828,352
2260 DATA 28282828282828282828,352
2270 DATA 28282828282828282828,352
2280 DATA 3A988888888888888888,249
2290 DATA 8A988888888888888888,11212,154
2300 DATA 1415161718191A30313233,359
2310 DATA 3435363738394041424344,662
2320 DATA 444448494A4B4C4D4E4F48,825
2330 DATA 512535455565788818283,594
2340 DATA 8A88888788888888888888,958
2350 DATA 8F8111213141516171819,228
2360 DATA 1A38313233343536373839,551
2370 DATA 4142434445464748494A4B,778
2380 DATA 4C4D4E4F58595A5B5C5D5E,916
2390 DATA 57182828284269744E48A5,916
2400 DATA E3A65286765626566883,1858
2410 DATA 6A52864A56E28A6E16A85,964
2420 DATA 6E28A4A57398282828A269,888
2430 DATA 6C86A65732861A8288A61,934
2440 DATA 7286A93A82886C16A65A6,1888
2450 DATA 2873A6A8C3A9D1D1D28,886
2460 DATA 28282828282828282828,352
2470 DATA 2828283132333435363738,516
2480 DATA 3938313233343536373839,582
2490 DATA 3831323132333435363738,582
2500 DATA 39383132333435363738,567
2510 DATA 39383132333435363738,582
2520 DATA 39383132333435363738,582
2530 DATA 3831323132333435363738,567
2540 DATA 39383132333435363738,582
2550 DATA 3831323132333435363738,567
2560 DATA 39383132333435363738,582
2570 DATA 3831323132333435363738,567
2580 DATA 39383132333435363738,582
2590 DATA 3831323132333435363738,567
2600 DATA 2853A652868616265A6E28,895
2610 DATA 657328676573A68616A66,1871
2620 DATA 7428656A988888888888,1168

```

```

2638 DATA 65E446F47287A75286CA,9257
2640 DATA 657365E219B131D28228,769
2642 DATA 28496872652858756E874,986
2644 DATA 653A282828282828282828,447
2646 DATA 28282828282828282828,743
2648 DATA A9808080808080808080,1868
2649 DATA 318285C8A94A688391C8A8,1298
2649 DATA 8A2808D9C49C1FF88791,1434
2718 DATA C8E8C8C2F47A988557AD,1391
2720 DATA 28496874F9A5C891C8A8D6,1379
2728 DATA 86D9F849C8E891C8C8E8A,1252
2748 DATA 809849C891C8E8D9898D,1733
2750 DATA 18862898478968D1868A,793
2768 DATA 85A9828484A4A128A8F8C,937
2778 DATA 287C48A838248A987285C,975
2788 DATA E428A847A98E8C828C8A,1311
2798 DATA 80C8E28988D8C8E28A84,1531
2808 DATA C891C8C8A5C891C8C8A5C,2833
2818 DATA 91C688882A9F891C8A89E,1722
2828 DATA 8088E28A978D81C82A8C8,1829
2838 DATA 8ED4848A88A868D8A8D,1887
2848 DATA 19D8E8A86A88A8C898D,1885
2858 DATA 80C8E28988A868A8A28A8,1288
2868 DATA 8C9A283C8E47D848285E4,1178
2878 DATA A268A983A4283A9629A4,1142
2888 DATA 83A9A59D483A8D168A9A4,988
2898 DATA 83A9899A4883283A56A8A2,1819
2908 DATA 88A9889D4282A9889D4883,827
2918 DATA 98A983A97288A56A8E887,1187
2928 DATA 48AC8E8A97791C8A387A8,1349
2938 DATA 91C8A862A8E4A8874C8E4,1568
2948 DATA AD128C81989F1EAD1F8C8,1258
2958 DATA 85D817ADFC82C92D8818A9,1382
2968 DATA 1C8D8C2A9898D8688D12,913
2978 DATA 8A4C8F4A8E1886A8E8A8D,1137
2988 DATA 8E188E888D82E28E8C8A8D,1861
2998 DATA 8C86CD188A8F1C8A988D18,854
3008 DATA 86A8084D8AC8E8B1E81C86,1065
3018 DATA 8191C8C881C8A9891C8C4C,1438
3028 DATA 62E4A8C8E8A8E868808A8,1178
3038 DATA 91C8E8C88D898A991C8A898,1183
3048 DATA 88C886A4C8E488A978D8E4,1888
3058 DATA A21828A8CA98A218D942,1814
3068 DATA 83A9F89D488A9A4C9C8B3,1019
3078 DATA A98A9A8832856A4188D4C,858
3088 DATA A94CA218A985D4283A9A5,1861
3098 DATA 90A483A9A59D480C48D7A9,1272
3108 DATA 83A91A9D488A9A89A983,826
3118 DATA 2856E418834C4A9A28D8,1856
3128 DATA 45A5C928F818A98A8D1A8A,984
3138 DATA A98855A4A88855A81A89,1146
3148 DATA 8928A4A4A98855A4A8C,784
3158 DATA 2852A9A988D18A8A9885,858
3168 DATA 5A9A8855A814A98928A8,949
3178 DATA 4A49878D1F8D8A97D8C9A5,1247
3188 DATA F88798D8F8A5C8E488A9F,1292
3198 DATA D8C95F8F8A81469A8A31A,1428
3208 DATA D8FC4CC48AC1F44A82D1E8,1382
3218 DATA 8D45A9D4345E88A8F88A9,1299
3228 DATA 28D8F1A92E9D434588A89,1344
3238 DATA 45A5E89D4345C88D8A8F,1528
3248 DATA A928E89D4345E81D8F8A,1592
3258 DATA 99A94345A8E8A58A68A8,1272
3268 DATA AE4945533A4788A4A8A,1288
3278 DATA 485B134C6A8A8788A88A8,398
3288 DATA 87C28282828282828282,272
3298 DATA FF7D8A8A8A8A8A8A8A8A,1622
3308 DATA 8A8A8A8A8A8A8A8A8A8A,1288
3318 DATA 8A282828282828282828,458
3328 DATA 2828282828A98A28F8D8,1879
3338 DATA 49D3A8A8F8A8A8A8A8E8,1979
3348 DATA A8E5A8F2288A282828A8C,1227
3358 DATA 28282828282828282828,352
3368 DATA 282828A998A8A8A8A8A,1217
3378 DATA 8A8A8A8A8A8A8A8A8A8A,1518
3388 DATA 982828282828282827928,638
3398 DATA 28282828282828282828,352
3408 DATA F4E8E8E81F3A8A8F4E1F5,2454
3418 DATA F3E8E8A98D1C1C1C1C1C,1262
3428 DATA 1C1C1C1C1C1C1C1C1C1C,388
3438 DATA 1C1C1C1C1C1C1C8A4725,527
3448 DATA 65638A6568873A658862,937
3458 DATA 697474688F34E1F2F88,1636
3468 DATA 756888A659A8E88575A5,971
3478 DATA 7388A2A96A4886A9A6A,964
3488 DATA 756C16A56E8E888F8F5,1132
3498 DATA FAFAC8E72888A63888A62,1677
3508 DATA 7988F4E8E8E1F388F4E1,2266
3518 DATA F9F3E8E4881C1C1C1C1C,1451
3528 DATA 1C1C1C1C1C1C1C1C1C1C,388
3538 DATA 1C1C1C1C1C1C1C1C1C1C,388
3548 DATA 1C1C1C1C1C1C1C1C1C1C,388
3558 DATA 1C1C1C1C88A76A62A5E88,625
3568 DATA 73A9D4A4C8F48A88A45A6,1288
3578 DATA 886E1A8A56A88A6A57388,843
3588 DATA 78757A8A6C657388A56A8,1113
3598 DATA 5888A617388A63A888A6,919
3608 DATA 61A465E888736F8A888A6,519
3618 DATA 6A6572886472756368A5,1854
3628 DATA 6E8873A9588F2E8F4F5F2,1633
3638 DATA E8E875A88873A96A888A6,987
3648 DATA 617388A9668A6C747376,1885
3658 DATA 65727A656A686A697388,1876
3668 DATA 6A6572886469736A657474,1875
3678 DATA 6588A16E7A757368A56E,1878
3688 DATA 881C1C1C1C1C1C1C1C1C,288
3698 DATA 1C1C1C1C1C1C1C1C1C1C,388
3708 DATA 28282828282828282828,352
3718 DATA 28282898888888888888,251
3728 DATA 88888888888888888888,8
3738 DATA 88888888888888888888,8
3748 DATA 88888888888888888888,8
3758 DATA 88888888888818181818,16
3768 DATA 81818181818181818181,11
3778 DATA 8181818186163687475A,647
3788 DATA 8788657388A9737A886A5,858
3798 DATA 68A6C87288D848C48E75,1216
3808 DATA 68A6C57288888888886175,647
3818 DATA 66A65747267A65A68888,977
3828 DATA 8882A97474A588F34E1F,1498
3838 DATA F886A67275A56368A5E41,1158
3848 DATA 88818181818181818181,18
3858 DATA 81818181818181818181,23
3868 DATA 81818888888888888888,172
3878 DATA 83888888888888888888,138
3888 DATA 7A68F8A61738874617573,1897
3898 DATA 456A4A88677275A57374,1892
3908 DATA 88A6A56E88725737A88A,857
3918 DATA 63728877A56C7A88D88E1,897
3928 DATA FAE1F2E98867A8862A5,1479
3938 DATA 7373A57288A1A7388C32E,1838
3948 DATA 2E282E2E2E2E2E888888,361
3958 DATA 88888888888888888888,143
3968 DATA 88888888888888888888,143
3978 DATA 88888888888888888888,143
3988 DATA 88888888888888888888,442
3998 DATA 428288A8E48E888D8A888,1177
4008 DATA 89D5A21828A8C28A89A9,1388
4018 DATA 8783F4A8A85F4286A8A2,1852
4028 DATA 88A88881F329738C4C8,1285
4038 DATA 45E9289D7848E8C8E8A3D8,1686
4048 DATA F8A98A88888888888888,1762
4058 DATA 4728E8A4C88A88E1F8C8,1164
4068 DATA 8A88F98A4A3A2E2A8A9,1139
4078 DATA 8888288A58888A958195,1155
4088 DATA CCA28888A85D8A445F88,1429
4098 DATA 8A6F28A4A8A6F8E88C88,1427
4108 DATA 816818A8388888888A8C,1383
4118 DATA 698888CCE288A8288A8,1178
4128 DATA 8A8D8A888888288A8A8C,1138
4138 DATA 88888888A8C8A8A88C4C,1548
4148 DATA 85A8A8C8888A8A88888,1288
4158 DATA 8A8288A88A888F1C891C,1289
4168 DATA C888A8F858A8C888888,1519
4178 DATA C8A8C8A88888888888,1718
4188 DATA E1A8A8A88888888888,1437
4198 DATA 68A8C9F48C82A8A9F88D1A,1267
4208 DATA 82A9F88E2882A88C8E28A,942
4218 DATA A988A82878888288A8C4,1274
4228 DATA 18F78A8A98888F2A9888,1287
4238 DATA 88D288A8888A8286858A,871
4248 DATA 805A8A9878D1F8A8A8D,1299
4258 DATA 2882A94882982A9818D1A,889
4268 DATA C8A8C8A88888888888,1718
4278 DATA E288A8E288A89F88A8C,1828
4288 DATA 83A8A9888288A8A8C8,1875
4298 DATA F83828F9F8A84A8A8E88F,1352
4308 DATA 382889A8D4C8A8E88D7F8,1226
4318 DATA 28948E48E88887F88C8F,1328
4328 DATA F83828F9F8A84A8A8E88F,1352
4338 DATA 805A8A9878D1F8A8A8D,1299
4348 DATA 80528A42C2E4E2388E488A,849
4358 DATA A98F8C8A8A8E4A8A88F8A,1281
4368 DATA 88A8D7882A8A8888882E,1558

```

```

4370 DATA 4A8A4E4B035F4F806D2,1056
4380 DATA F809A9A187D5F4C354E,1112
4390 DATA A908D07D2EE4EBA0D5A8,1242
4400 DATA F803CE58A6E806B07F4F,1196
4410 DATA F80D0586A9F80D356A69,1067
4420 DATA 000D01D24D5806D08D2F,1210
4430 DATA 86A9A184D5A606D1D2EE,1150
4440 DATA 506A25486D085A980D54,964
4450 DATA 86A4C86B03F4F8D02D2F,1106
4460 DATA 83AD5806D083D2CE5806C,1122
4470 DATA 5406E4C86CE5286A90103,1015
4480 DATA 1A8248182A2A2A2A600000,414
4490 DATA 00000004E8F4F8000000,542
4500 DATA 000000000000004252757,386
4510 DATA 4949424242425757494942,796
4520 DATA 00494961615757494949,886
4530 DATA 6161575749004C4C6767,886
4540 DATA 574C4C4C4C6767574C00,347
4550 DATA 42004200420042004200,396
4560 DATA 0042004200490049004900,351
4570 DATA 49004900490049004900,446
4580 DATA 005100510051005100,446
4590 DATA 51005100570000000000,249
4600 DATA 00000000000000000000,0
4610 DATA 00000000000000000000,0
4620 DATA 000000282A2C00F90000,141
4630 DATA 282A2D00F900F0000000,157
4640 DATA 00000000000000000000,0
4650 DATA 00002040204020004004,22
4660 DATA 00200040002200000000,0
4670 DATA 00000000000000000000,0
4680 DATA 003C0000000000003C00,120
4690 DATA 3C003C003C0032000003,200
4700 DATA 00350000003C00000000,113
4710 DATA 00000000000000440044,136
4720 DATA 00440040033C04F35000,727
4730 DATA 000035003C0000000044,101
4740 DATA 00400000100000000000,65
4750 DATA 32003200320032003200,295
4760 DATA 3235003C000000000000,163
4770 DATA 0000200000200020000,130
4780 DATA 00000032003500000000,103
4790 DATA 000000440040000000,132
4800 DATA 0000000020002000200,120
4810 DATA 20002000200020000020,200
4820 DATA 002000000000320000,95
4830 DATA 0000000020002000200,100
4840 DATA 0020002000200000200,100
4850 DATA 32000000000035000000,103
4860 DATA 000000320032003200320,200
4870 DATA 320032003200000032003,253
4880 DATA 00000000003C00000001,114
4890 DATA 00000000350020002000,100
4900 DATA 00350000001000000100,55
4910 DATA 0001010000010100101,7
4920 DATA 00010100002010100020,10
4930 DATA 01000010100001010001,12
4940 DATA 01010000201010001010,10
4950 DATA 0101020010103020C00,202
4960 DATA 075101010401010105010,200
4970 DATA 01062010107020101001,31
4980 DATA 0101010101020010103,15
4990 DATA 02010104010101010101,19
5000 DATA 062010107020101020401,34
5010 DATA 0009401000405100005,46
5020 DATA 01000C0610000601000E,54
5030 DATA 0701000F000100000001,26
5040 DATA 00010101010101020101,11
5050 DATA 01000101010001010100,11
5060 DATA 01010001010100010100,22
5070 DATA 000101010001010200101,9
5080 DATA 00001010000101050001,10
5090 DATA 01060001010000010100,19
5100 DATA 00010900000100000100,33
5110 DATA 0000010C000001000000,20
5120 DATA 000700010FF000000000,461
5130 DATA 0002200020002000A207,620
5140 DATA 43009852F0034C252C00,1479
5150 DATA F2A26204143CA0F0A200,1396
5160 DATA 0002200020002000A207,1434
5170 DATA A200204143CA0000F0A5,1301
5180 DATA 5805C8A59F00C2000520,1307
5190 DATA 5E00519F5206C900F0220,1113
5200 DATA 4143A200A91C0B200520C,1056
5210 DATA 0004A0000C9000E0E0E0,1036
5220 DATA 86A00186C9FF00E14C0351,1411
5230 DATA 20414305C8AC2006A5C91,1227
5240 DATA C0200352C00006A00006C9,957
5250 DATA 0004E0CE186A00186C9F,1293
5260 DATA 00E2A0C0351D02041430900,1291
5270 DATA 862900D5E8A40006297F,763
5280 DATA 000006C900F006A900001,905
5290 DATA 0660204143000186204143,570
5300 DATA 0000066018A5CA4F5005C,1156
5310 DATA 6500A9005000E220AEE2,1361
5320 DATA 00022006C90C003A2C5F52,1100
5330 DATA 38A5C8A5CCE91E35A9000,1615
5340 DATA 05CCA901002206A02206C9,1102
5350 DATA C1F083ACT0525000A9C920,1417
5360 DATA 05C8A5CCE91E35A9000,1615
5370 DATA 2206E2006A02006C920F,1000
5380 DATA 0140A06060203C40207451,706
5390 DATA 4C0A020C30040C3400001,704
5400 DATA 001AC7C900FF00E210200,1262
5410 DATA 40000000000000000000,64
5420 DATA 0,0

```

32767 REM

## Atari-Karteiverwaltung

Als Speichermedium benötigt man hier einen Cassette-Recorder oder noch besser ein Diskettenlaufwerk. Besitzt ein Anwender auch einen Drucker, so kann er sich die Kartei teilweise ausdrucken lassen. Die Bedienung des Programms 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 **"J"** 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 dem 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 Darstellungsblatt 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 Wiern

## Karteiverwaltung

```

1 REM VERWALTUNG
2 REM VON MORN KARL/WIEDERGLEINKESTR.
32/4407 BIEZACH
3 REM *****
180 DIM AS(20000),BS(256),DTAS(25),TIS
(118),CS(258),DMS(25):TI=10:DMS="*****
"
120 OPEN RL,4,0,"K":GOTO 500
150 REM TITEL *****
280 GRAPHICS 17:POKE 87,0:SETCOLOR 0,4
,10
250 PRINT " GEBEN SIE DIE:";POSITION
0,1:PRINT " GEMEINSCHEN TITEL:";POSITI
00,2:PRINT " E I N "
300 TIS="":TIS(118)=TIS(1):TIS(2)=TIS
(1)
350 FOR T=10 TO 100 STEP 10:POSITION 5
,3:PRINT T/10;" ",TITEL "
480 POSITION 5,7:PRINT ".....":PO
SITION 4,7:INPUT DTAS
450 TIS(1,T+9)=DTAS:NEXT T:POKE 87,0
470 REM MENUE *****
580 GRAPHICS 0:POKE 82,1:POKE 16,64:PO
KE $3774,64:N=1
680 SETCOLOR 2,6,10:SETCOLOR 4,6,0:SET
COLOR 1,12,4
650 PRINT CHR$(125):PRINT " *****
*****"
780 PRINT "      M E N U E M
"
710 PRINT " *****"
*****":POSITION 2,5
715 PRINT "*****":DMS(1),CLEN(DMS
)):POSITION 3,8
720 PRINT " 1.SPEICHERN,LADEN"
730 PRINT " 2.TITEL NEU ANLEGEN"
740 PRINT " 3.KARTEI NEU ANLEGEN"
750 PRINT " 4.KARTEI SUCHE"
760 PRINT " 5.KARTEI AENDERN"
770 PRINT " 6.KARTEI EINFUEGEN"
775 PRINT " 7.KARTEI SORTIEREN"
780 PRINT " 8.KARTEI LESEN"
785 PRINT " 9.KARTEI AUSSPRUEHEN"
790 POSITION 12,21:PRINT "*****"
"
880 GET RL,T
810 IF T=49 THEN 1000
815 IF T=50 THEN 200
820 IF T=51 THEN 40="":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
990 REM LADEN /SPEICHERN
1000 PRINT CHR$(125):DMS="
1010 PRINT "*****"
*****"
1020 POSITION 9,6:PRINT " [ ] .LADEN VON
DISKETTE":POSITION 9,8:PRINT " [ ] .SPE
ICHERN VON DISKETTE"
1025 POSITION 9,10:PRINT " [ ] .LADEN VO
N KASSETTE":POSITION 9,12:PRINT " [ ] .S
PEICHERN VON KASSETTE"
1030 POSITION 13,20:PRINT "*****"
"
1035 GET RL,T
1040 IF T=49 THEN DMS="0":GOTO 1100
1050 IF T=50 THEN DMS="1":GOTO 1500
1052 IF T=51 THEN DMS="2":GOTO 1100
1054 IF T=52 THEN DMS="3":GOTO 1500
1060 DMS="*****":GOTO 500
1100 PRINT CHR$(125):POKE 752,1
1110 PRINT "*****"
*****"

```

```

1120 POSITION 3,8:PRINT "D I S K E T T
E / K A S S E T T E":PRINT "
E I N L E G E N ! ! "
1130 POSITION 13,22:PRINT "*****"
*****"
1135 GET RL,T
1132 DTAS="":POSITION 13,14:PRINT "DAT
EINAMEN":PRINT "
"*****":INPUT
T DTAS:AS="":TIS="
1134 DMS(CLEN(DMS)+1)=DTAS:DMS(CLEN(DMS)
+1)=",DAT"
1135 TRAP 1210
1140 OPEN H2,4,0,DMS
1142 INPUT H2,TIS
1150 FOR I=1 TO 99000
1152 BS="
1160 TRAP 1210:INPUT H2,BS:AS(CLEN(AS)+
1)=BS
1170 NEXT I
1210 FEHLER=PEEK(1195)
1220 IF FEHLER<136 THEN POSITION 13,1
7:PRINT "4FEHLER NR.":FEHLER:CLOSE H2:
FOR I=0 TO 1000:NEXT I:GOTO 1000
1230 CLOSE H2:GOTO 500
1500 PRINT CHR$(125):POKE 752,1
1510 POSITION 5,2:PRINT "*****"
*****"
1520 POSITION 5,8:PRINT "D I S K E T T
E / K A S S E T T E":PRINT "
E I N L E G E N ! ! "
1530 POSITION 13,22:PRINT "*****"
*****"
1535 GET RL,T
1532 DTAS="":CS="":POSITION 13,14:PRIN
T "DATEINAMEN":PRINT "
"*****":INPUT DTAS
1534 DMS(CLEN(DMS)+1)=DTAS:DMS(CLEN(DMS)
+1)=",DAT"
1536 OPEN H2,8,1,DMS
1538 INPUT H2,TIS
1540 FOR I=1 TO LEN(AS) STEP 250
1542 PRINT H2:AS(I,I+249)
1544 NEXT I
1550 CLOSE H2:GOTO 500
2800 PRINT CHR$(125):GOTO 200
3800 GOSUB 9000
3910 REM EINKADRE *****
3920 DTAS="":BS="
3930 FOR S=1 TO 21 STEP 2:POSITION 12,
5:INPUT DTAS
3940 GOSUB 3100
3950 NEXT S
3955 IF T=42 THEN RETURN
3958 TRAP 6000
3960 AS(LEN(AS)+1)=BS
3965 PRINT :PRINT "*****"
*****"
3970 GET RL,T
3975 IF T=78 THEN 500
3980 GOTO 3000
3100 FOR Z=1 TO 25:IF LEN(DTAS)>25 THEN
H DTAS(CLEN(DTAS)+1)=":NEXT Z
3110 BS(CLEN(BS)+1)=DTAS
3120 RETURN
3990 REM SUCHE *****
4000 PRINT CHR$(125):SETCOLOR 2,6,10:5
ETCOLOR 4,6,0
4010 PRINT :PRINT "*****"
*****"
4020 POSITION 12,21:PRINT "*****"
*****"
4030 GET RL,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 I":INPUT X
4110 GOTO 8000
4200 Z=0:POKE 752,0:POSITION 10,15:PRI
NT "MORT EINGEBEN":POSITION 9,17:IMPU
T DTAS
4210 Z=Z+1
4230 TRAP 500:IF DTAS=AS(Z,Z):LEN(DTAS)
-1 THEN H=INT(Z/250)+1:GOTO 8000
4240 GOTO 4210
4990 REM AENDERN *****

```

```

5000 GOSUB 3020
5010 Z=CN-1)*250+1:ZZ=Z+250
5020 AS(C,Z)=S:GOTO 8100
5030 REM SPEICHERUEBERLAUF
4000 PRINT CHR$(125):POSITION 10,9:PR
NT "SPEICHERUEBERLAUF":PRINT "44
DATEN ABGESPEICHERT"
6010 FOR I=0 TO 1000:NEXT I
5020 GOTO 500
5030 REM SORTIEREN *****
7000 POSITION 6,21:PRINT " B I T T E
W A R T E !"
7100 FOR I=1 TO LEN(AS) STEP 250
7110 BS="" :CS="" :TRAP 7150
7120 BS=AS(I,I+249):CS=AS(I+250,I+499)
7130 IF BS<CS THEN AS(I,I+249)=CS:AS(I
+250,I+499)=BS:POP :GOTO 7100
7150 NEXT I
7160 GOTO 500
7300 REM KARTEI LESEN *****
8000 GOSUB 3000
8010 REM AUSGABE *****
8020 XX=CN-1)*250+1
8030 TRAP 500
8040 FOR S=1 TO 19 STEP 2
8050 POSITION 13,5:PRINT AS(XX,XX+24)
8055 XX=XX+25
8060 NEXT S
8070 POSITION 13,21:PRINT AS(XX,XX+24)
8080 POSITION 0,0:PRINT " "
1000:PRINT " " :N: " "
8100 POKE 752,1:POSITION 0,21:PRINT "
" :S: " " :C: " " :C: " "
";
8110 GET M1,T
8120 IF T=0 THEN X=X-1:IF X=0 THEN X=
LEN(AS)/250
8125 IF T=0 THEN 0000
8130 IF T=62 THEN X=X+1:IF X=LEN(AS)/2
5041 THEN X=1
8135 IF T=62 THEN 0000
8140 IF T=42 THEN POKE 752,0:GOTO 5000
8150 GOTO 500
8370 REM EIN-AUSGABE MASKE *****
9000 PRINT CHR$(125):SETCOLOR 4,0,8:SE
TCOLOR 2,0,10
9010 TRAP 500:FOR S=1 TO 21 STEP 2:P0S
ITION 1,5:PRINT TAB(1,II+9):" :II:
TI+10:NEXT S:TI=10
9020 RETURN
10000 PRINT CHR$(125)
10010 PRINT " " :A: " " :S: " " :R: " "
" :X: " " :POSITION 8,20:PRINT " HI
" :O: " " :C: " " :C: " " :TRAP 10200
10020 POSITION 10,10:PRINT "WELCHE SEI
TE " :I:INPUT X
10025 IF X=0 THEN GOTO 500
10030 XX=CN-1)*250+1
10040 OPEN #2,0,0,"P"
10050 FOR I=1 TO 10
10055 TRAP 10110
10065 CS=AS(XX,XX+24)
10060 PRINT #2;" "
"; " :AS(XX,XX+24)
"; " :TAB(1,II+9)
10070 XX=XX+25:II=II+10
10080 NEXT I:II=10:X=0
10085 CLOSE #2
10090 LPRINT :LPRINT :LPRINT
10100 GOTO 10000
10110 LPRINT #2:PRINT "444 L E E D
E S E I T E " :FOR Z=0 TO 400:NEXT Z
:GOTO 10000
10200 POSITION 10,15:PRINT "FEHLER NR.
" :I:PRINT (195):CLOSE #2: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 eingelese, 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 gefüllt 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 Filenamens COLL.DAT.

Stefan Funk

```

10 REM *** Disk-Collector
15 REM *** erstellt mit GBJ-LBT-J
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-Dateinamen (mit Wild-
90 REM *** cards) sind beim Suchen
95 REM *** sehr hilfreich
100 DIM Z(164),Y(64),DISK(20),Z(16)
110 J=110-999:DISK="D:AUTORUN.SYS " :Y="ADD"
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(103)+256*PEEK(104) THEN 940

```

```

232 IF LEN(X)<32 THEN 970
234 IF LEN(Y)<32 THEN 980
236 @09UB 700
238 @09UB 800
240 IF I<Y THEN 990
250 NEXT I:POKE 559,34
252 TRAP 240
254 READ X#
256 @0TO 940
260 ? CHR$(125):? CHR$(253):? "Diskette fuer *;DISK*? *ni
nlegen und *START* drucken..."
270 IF PEEK(53279)<6 THEN 270
300 ? CHR$(125):? CHR$(253):? "Schreibe File...":?
310 TRAP 900:RESTORE :OPEN #1,0,0,DISK#
320 FOR I=1 TO ANZHL
330 READ X#,Y#:@09UB 710
340 FOR J=1 TO 16
342 IF I(J)>255 THEN 400
344 PUT #1,I(J):NEXT J
350 NEXT I
400 ? CHR$(125):? CHR$(253):? "Arbeit beendet...":? DISK#;
"-File erstellt!!"
410 CLOSE #1
420 END
700 I=0
702 FOR Q=1 TO 32:IF X#(Q,0)<"A" OR X#(Q,0)"0" THEN 920
704 IF X#(Q,0)"0" THEN I#(Q,0)"A"
706 NEXT Q
710 FOR K=1 TO 16
720 I(K)=I#*(ASC(I#(K+2-1,K+2-1))-65)+(ASC(I#(K+2,K+2))-65
):I=I+I(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,0)<"A" OR Y#(Q,0)"P" THEN 930
804 NEXT Q
810 RETURN
900 ? "Fehlermeldung...":PEEK(195)
910 ? CHR$(253):POKE 559,34:END
920 ? "1.Datenkette"/Q"/Zeichen (I;I#(Q,0)) falsch":?
'in Zeile...":LIST I+999:@0TO 910
930 ? "2.Datenkette"/Q"/Zeichen (Y;Y#(Q,0)) falsch":?
'in Zeile...":LIST I+999:@0TO 910
940 ? "Zeile I+999" fehlt !!!:@0TO 910
950 ? "Zu wenig Daten !!!:@0TO 910
960 ? "Zu viele Daten !!!:@0TO 910
970 ? "1.Datenlaenge falsch in Zeile...":LIST I+999:@0TO 9
10
980 ? "2.Datenlaenge falsch in Zeile...":LIST I+999:@0TO 9
10
990 ? "Zeichenfolge falsch in Zeile...":LIST I+999:@0TO 91
0
1000 DATA PPPPAKIXLFXJAAJFEKJABJFIEXJDA,HIA
1001 DATA IFIFKAAKJAAJJAABJINAPKXFJFIFC,HF
1002 DATA M#ICKJAFIFIAKCKATIBKAAAJJAJMNA,III
1003 DATA PLNKDEICMAPKJXJHJNDACKJLLINDACK,INA
1004 DATA KJPKINBACJAJINMFACKJCCJINCPACK,HCN
1005 DATA AKKAAJINAPMNAKCANKECAJILEDA,IMN
1006 DATA AGCAKLECAIDLEKJANKATKCECAFND,HKJ
1007 DATA KJAINCPACKJAJINAAKCKJINABACKJ,FDD
1008 DATA MAINAKNEKJGBINDAKCKJINDBACKJCD,HFK
1009 DATA INCPACKJDAINBACJAJINMFACKJPPIN,HCN
1010 DATA PNACKMNPANJHGAADENAAKIKNPRACN,HKE
1011 DATA PPPADPKNDKICCBMNDKINMBSLNFKI,LDN
1012 DATA EILNKEITEIKJPPINMFACKJEAJNAGNEBA,ILE
1013 DATA OI0I0I1INOCENJKIABPHFKLBJOJLA,THO
1014 DATA BKEELBJPFLBNDMLCBLBALDHAHHANA,8NB
1015 DATA ECEKJBAACBACNAHAFACHPACBAC,EXD
1016 DATA BAACBAACBAACBACBAHAFACCEBCKE,SNB
1017 DATA KJCCINBINAACKJCCINAAKCKJAJINAAKCKJ,8BN
1018 DATA KJINBACBCEAEIKJCCINBINAACKJCCINAA,FLA
1019 DATA MEXJHINAAACKJAJINABACCEAEIKJCC,FOA
1020 DATA INBINAACKJCCINAAKCKJAJINAAKCKJIN,8BP
1021 DATA ABACIEAAAAAARAAAAAARAAAAAARAK,ALP
1022 DATA AKAAACEKJDCCLAAAAAARAAACDCPCMC,8BN
1023 DATA CFCDECPCAAARAKAKAAAAAARAAAAA,APL
1024 DATA AAAAAAARAAAAAARAAAAAARABGPHAJ,BLL
1025 DATA HCGJGHEIHEIADAJBACJAJACBHFGBL,EOB
1026 DATA HJGHPGHEKNCBKHJHDHIBAAAAAARAA,DIJ
1027 DATA AAAAAAARAAAAAARAAAAAARAAAAA,ARJ
1028 DATA ABGDACBFBGHEBFBEAASCHJADDEBFGB,FCF
1029 DATA GBDAACBFBGHEBAAAAAARAAAAAARAA,CEM
1030 DATA AAAAAAARAAAAAARAAAAAARAAACB,8BN
1031 DATA CGBFBNHFBFBHEBFPCAAACJBNHFAHE,FBJ
1032 DATA SNBFBGHEFFFAAAABGDBDAARAD,8BNKJ
1033 DATA ADACEGJHCFGBGHEBFHCHJABFBJGBSF,DAL
1034 DATA HCAACEGJHGBLAGNBGBBFGBLBCDND,8KJ
1035 DATA ADACEGJHCFGBGHEBFHCHJABBFHGBH,FBG
1036 DATA BFCBFBGDAAAAAAARAAAAAARADLBD,800
1037 DATA ADACEGJHCFGBGHEBFHCHJAGNSPFBH,FBE
1038 DATA GDEBFBGAAAAAARAAAAAARADLBDND,CD1
1039 DATA ADACEGJHGBLAGNBFBGDAABGHEBFGB,8BN
1040 DATA SJHNEBFBGAAAAAARAAAAAARADLBDND,CLD
1041 DATA ADADACBFBGHEBGBGNAABHFBGDEBF,FFJ
1042 DATA BAAAAAARAAAAAARAAAAAARADLBDND,BAK
1043 DATA ADACBFBGHEBFBGHEBFHCHJAGNSPFB,FFM
1044 DATA GBNDHGBGDAANBQACECPDAAAAACCB,8FB
1045 DATA HEHFBACBHFGBLHEBGPDAABGJHE,8FB
1046 DATA AAEGBNDHEBFBGHEBFHCHJAGNSPFB,FXA
1047 DATA B0AABAAKCBKJAJNNECADENFBDEEIJ,FBG
1048 DATA EIEIKCBKJAJNNECADENFBDEEIJN,FKL
1049 DATA EFABJINEXAKJAJNLAJEMFDEKCB,FBG
1050 DATA KJAFJNECADJAJNNECADJAJNNEFADJ,FBG
1051 DATA BEJNEIADJAJNNECADENFBDEKCBKJAJ,8FB
1052 DATA JNECADJAJNNEIADJNEIADENFBDEELK,FBH
1053 DATA ADDBGNDMAACECPDAAKCBFBGHEBF,8FB
1054 DATA BHFGBGHEBFBGDAABHFBGHEBFBGDL,CKX
1055 DATA DNBDAADLCOBFBGDBDPPMNAACEB,8FB
1056 DATA HEBFBGHEBMEKAAABHFBGDBHCAAA,FDJ
1057 DATA HBFHCFBFBGDLCKDNBAADLCOBFBG,8FB
1058 DATA BDAACBFBGHEBGBGNAABHFBGHEBF,8FB
1059 DATA BNAADHFBGJFBDAABFBGHEBFBGDF,FCI
1060 DATA BDAHCFBHEHFCDAABHFBGHEBF,8FB
1061 DATA GDFBFBGDAABHFBGHEBFBGHEBFBG,8FB
1062 DATA AAKNEELBNAKNEELBNAKJAJJAAAB,8BQ
1063 DATA MNBKBPJINFBGJINFBGHEBKAJAIN,8BN

```



```

1130 DATA DAAKJLJLINDACKJCCINCPACCA00KKEN,FDC
1131 DATA JZKLIFJCI0IAIEI8KJPPINPACCA0AALB,IME
1132 DATA IAEJIAJBJAJIEICAINLAKXGKJLBIJAEJ, HML
1133 DATA IAJZBIAIKJLPACANJOPAPCAHSLAJ, IKJ
1134 DATA IAHMEICNAACKAAE8KJLAKJAAJBIJAI, HWH
1135 DATA BNAKKEICIEIEMDJLAGEICIAJFJ0I0I, BJA
1136 DATA BALAHMICALAGBIJAEJFJ0GADJ0CA, FDA
1137 DATA EMIELAKJAEKCEIKAKLCAPIKKAC0KLEI, H8N
1138 DATA CADOKKSI0AKFIEMJANBANDENELLKJAA, HOH
1139 DATA INDAACKJLLINDACKJCBINCPACCAANLE, FEH
1140 DATA HEIENAADENHOLKCAHLN8DKJNP0K0NK, ILJ
1141 DATA BAPHKJKEINHBACKJADINDACKJLLINDG, HEB
1142 DATA ACKJCBINCPACCAFKLE0BIB8IBKAPPL, HNG
1143 DATA IAINEBL8MB0IB0CABPLEKJAKFICKJ, I0D
1144 DATA CAJFIDKCKFAALLBIAJBI0IIBAPINDEB, H8D
1145 DATA L8KNEBL8PANKFIABIGIAJFIKFI1802, HK0
1146 DATA AAIJF0KFI0BICJAI0FICKFJ0BJAAJFID, 88D
1147 DATA M0MANKJPPINPACCKPACJANNAJPKJ, KJF
1148 DATA EBL8NALEEMJLAA0AAAKKFIENJABNA, 8NE
1149 DATA 8CKJMEINDACKJLJINDACKJCBINCPAC, FPJ
1150 DATA ENJL0KJNDINDACKJLJINDACKJCBIN, HEK
1151 DATA CPACCAANLEIEMANKJ0IINDACKJLJ, 80D
1152 DATA INDBACENJ0KJIEIKJ0INDACKJLJIN, HHI
1153 DATA 8BACAINLAKJCPAINEJ0NAPFBIEMEF, HHA
1154 DATA L88IETAKAKKIFIC8IEIKIJJIAJAKAK, F8M
1155 DATA KIKFIEKKMKIKAKAKKKLMAAGJAA, 8L*
1156 DATA ABKJAAJNA880I0IEM0AP0IKCAF, HJ9
1157 DATA LEKJAAJFICKFIFIFJ0I0I8MID8I88, J8H
1158 DATA IEMJF8I8I8I8FKK0KAAALBIC0BIAKI, I8I
1159 DATA NAFJ08I88I8MKNAPNEFLKJLBI8I8, K8H
1160 DATA ACKJLJINDACKJCBINHBACKJCCINCPAC, 8FA
1161 DATA KJABIFIAKJCAJFJCBAGPLEKCKAKAAL, 8NA
1162 DATA AASJ8IADIMINAJNAPFKFIABIEJAKFI, HJ2
1163 DATA IAKFI86JAAJFJ08AANAC0EMAL8KFI, IXP
1164 DATA HJABNADEML8CASP8LX8KINDACKJ, 8KA
1165 DATA LLINDACKJ8I8HBACKJCBINCPACCA0K, 8TB
1166 DATA L0KN0PKJNA88AFKAAALJPAKJ0J0PK, JNL
1167 DATA HINAALNAPFKJAAI8PKK8ENFLCJ88I8, IMH
1168 DATA ECL8CAEML8KNEK8HEI8PAHCAFKLE0, I8D
1169 DATA I88I8KAPPL8IAIMEDL8K8I88I8K8AA, J8H
1170 DATA KCAALNDPKJNAK8AGCADD8ENL8L8CJ, ICJ
1171 DATA 80NAFKAAIEML8CBIC8I8ANF0I8A8PA, IIA
1172 DATA AF8INA8MND08L8K8I8A88I88I8I8I, I8N
1173 DATA BAFJCEEL8K8KAKAKK8K8ABILNA88, 8EB
1174 DATA J8IC0I8I8KACANAPFKFIC8I8JCAJFICK, IMH
1175 DATA I86JAAJFJ8ME8L8K8K8I88I88I8I8I, I8N
1176 DATA PA8EKJPPINPACCKPACJANNAJPKJAE, JAM
1177 DATA L0KNEK8FJEP80KFIABIGIAJFIKFI, IAG
1178 DATA I86JAAJFJ8ME8L8K8K8I88I88I8I8I, IAK
1179 DATA EMPL8C8JL8CENDL8C8I8NAI8PAHMAAL, HJL
1180 DATA P888E8D8L8NPK8KJ8A8F8I8DA8NA, I8A
1181 DATA P88KJAAI8N8L8KJAI8ME8L8I88I8I8I, 8LH
1182 DATA KJCAJFIDEM8PLEKJ8I8I88I8L8JFLIND, HMF
1183 DATA ACKJLLINDACKJPEI8HBACKJCCINCPAC, 8K8
1184 DATA CAI8N8AJED8F8J8I8K8NA8FK8I88I8D, H8D
1185 DATA JLEEDKCKCOCKJLEEDK8E8F8EN8C8DE8, EJB
1186 DATA FEJL8M8D8H88F8J888I88I88I88I88I, F8D
1187 DATA A88888J8E888888888888888888, 88I
1188 DATA A888888888888888888888888888, 88D
1189 DATA A888888888888888888888888888, 88E
1190 DATA A888888888888888888888888888, 88F
1191 DATA A888888888888888888888888888, 88G
1192 DATA A888888888888888888888888888, 88H
1193 DATA A888888888888888888888888888, 88I
1194 DATA A888888888888888888888888888, 88J
1195 DATA A888888888888888888888888888, 88K
1196 DATA A888888888888888888888888888, 88L
1197 DATA A888888888888888888888888888, 88M
1198 DATA A888888888888888888888888888, 88N
1199 DATA A888888888888888888888888888, 88O
1200 DATA A888888888888888888888888888, 88P
1201 DATA A888888888888888888888888888, 88Q
1202 DATA A888888888888888888888888888, 88R
1203 DATA A888888888888888888888888888, 88S
1204 DATA A888888888888888888888888888, 88T
1205 DATA A888888888888888888888888888, 88U
1206 DATA A888888888888888888888888888, 88V
1207 DATA A888888888888888888888888888, 88W
1208 DATA A888888888888888888888888888, 88X
1209 DATA A888888888888888888888888888, 88Y
1210 DATA A888888888888888888888888888, 88Z
1211 DATA A888888888888888888888888888, 89A
1212 DATA A888888888888888888888888888, 89B
1213 DATA A888888888888888888888888888, 89C

```

```

1114 DATA ACEB3H0GLANCDF06N05F080E8P3CA,EP
1115 DATA CEBB0E0F080D080E080A080H080E080H,FKL
1116 DATA 0F080F080F080A080A080E080H080L080P,ELA
1117 DATA 08080F080E080P3CA080E080E080F080D080E,0A3
1118 DATA HKA080H080E080A080F080H080E080F080H080E080,FKX
1119 DATA C080L080J080P080H080J080K080P080F080H080A080,C0N
1120 DATA KJLL080A080JPK080H080A080J080A080H080A080,KHC
1121 DATA CC080P3CA080KAA080I080P080K080N080P3CA080P,IAK
1122 DATA L080CAJ080L080E080F080C080K080L080E080A080C080J080L080E080J080P,08N
1123 DATA 080P3CA080F080E080H080A080E080K080I080J080E080H080A080,C0H
1124 DATA C080J080E080H080L080C080I080L080J080P080E080K080I,08N
1125 DATA K080L080J080A080J080P080I080P080K080C080K080P080J080A080,08L
1126 DATA C080M080K080A080L080J080P080A080I080I080P080I080E080H,08C
1127 DATA 080K080I080A080E080H080A080J080K080A080A080H080J080D080K080I,08P
1128 DATA 080P080K080C080K080K080J080I080C080L080K080A080H080J080D080K,08J
1129 DATA N080I080I080P080I080I080E080K080I080P080H080E080I080P080,08F
1130 DATA 080J080I080I080K080K080K080A080L080J080K080N080A080B080A,08H
1131 DATA K080I080M080A080I080M080C080B080I080M080G080I080E080D080E,08C
1132 DATA K080A080I080F080A080J080I080I080P080A080J080I080B080J,08C
1133 DATA 080B080C080E080K080A080K080A080J080A080C080J080A080C080J,08I
1205 DATA ET080K080F080J080E080A080E080F080E080K080A080H080L080J080,K0F
1206 DATA A080I080K080L080B080K080J080L080J080K080A080I080K080L080B,08J
1207 DATA K080I080I080F080A080J080I080F080K080A080I080C080J080A080I080,08E
1208 DATA C080L080E080A080L080K080A080I080F080A080J080I080F080K080A080I080,08B
1209 DATA 080K080J080I080D080L080E080A080B080K080A080I080F080A080J080A,08L
1210 DATA 080F080K080A080I080K080F080I080D080A080I080D080L080E080,08D
1211 DATA 080P080P080I080K080L080H080H080A080H080E080C080K080E080B080I,08F
1212 DATA 080L080H080H080A080E080C080L080K080A080H080A080B080A080E080,08G
1213 DATA 080L080K080H080H080A080E080C080L080K080A080B080E080L080K080H080A,08E
1214 DATA 080H080A080E080C080N080E080K080B080K080H080H080H080A080E080C080L080,08D
1215 DATA 080H080A080E080C080A080A080C080A080C080A080C080A080C080A,08K
1216 DATA 080C080B080A080H080E080C080K080E080B080L080K080H080H080A080E080C,08F
1217 DATA 080L080K080E080M080L080K080H080H080A080H080E080C080L080K080A080C080L080,08M
1218 DATA 080E080N080L080K080H080H080A080H080E080C080L080K080B080L080K080H080A080,08L
1219 DATA 080A080E080C080L080K080E080P080L080K080H080H080A080E080C080L080K080A080E,08J
1220 DATA 080L080K080E080B080L080H080H080A080E080C080L080K080A080C080A080A,08A
1221 DATA 080A080A080A080A080A080A080A080A080A080A080A080A080A080A080A,08A
1222 DATA 080A080A080A080A080A080A080A080A080A080A080A080A080A080A080A,08A
1223 DATA 080A080A080A080A080A080A080A080A080A080A080A080A080A080A080A,08A
1224 DATA 080H080A080E080C080L080K080E080B080L080K080H080H080A080H080E080C080,08E
1225 DATA 080F080L080H080H080A080E080C080L080K080H080H080E080F080L080H080A080,08D
1226 DATA 080H080A080H080A080E080C080L080K080B080A080H080E080C080A080A080A,08C
1227 DATA 080A080A080A080A080A080A080A080A080A080A080A080A080A080A080A,08C
1228 DATA 080E080B080L080K080H080H080A080H080A080E080C080L080K080E080B080L080,08C
1229 DATA 080P080A080C080B080A080C080A080K080I080D080D080D080D080D080D080,08M

```

## 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 zeitkritische Teil – wie Senden und Empfangen, Schreiben und Lesen – in Maschinensprache erstellt. Darüber hinaus werden die Assembler-Programmeile 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 Dateneingang 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 Zeitaktwerte 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 Initialisierungs-Bild (für die Zeitspanne, in der die Maschinendaten eingepflegt 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 die se 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 zu nächst im Kopf die Zeil "KEYBOARD SEND:BE REIT" 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 "FS" verwendet werden, erscheint diese Kopfzeile natürlich nur dann, wenn das Keyboard diese Statusbytes sendet. Zum Zeitpunkt der Programm-erstellung stand mir nur das Yamaha-Keyboard PS6100 zu Verfügung, und das sendet die se 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 "verbraht" ist.

Das nächste wichtige Feature heißt SAVE und beinhaltet das Sichern 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 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 aktive 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 aufgerufen 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 Maschinenendaten eingepackt 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. Den Zeilen 7000-7090 werden die Maschinenendaten für das MIDI-Kernprogramm eingelesen und der Zähler für die Anzeige während der Initialisierung aktiviert.

Ab 7100 folgen dann die Daten-Werte des Maschinenprogramms.

Es muß wohl nicht besonders betont werden, daß das Übertragen der Maschinenendaten 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 Leerzeile ihre Bedeutung. Wie schon im ersten Teil erwähnt, hat hier derjenige die wenigste Arbeit, der über den MAC/85 verfügt und mit einem Hilfsprogramm den Object-Code in Datenwerte 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 (einmal 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 Assemblerteil 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 Zeitaktwerten 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 Zeitaktwerte 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

- PLA1 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 aufgerufen Routine für die Zeitakt-Steuerng 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 Zeitaktwerte, 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 Maschinprogramm 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 I 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 Casseiten-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 Rechner Tasten (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 Miescher

## 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 HEKASOUND 1985 *
19 REM *
20 REM *****
21 REM
22 REM
23 REM -----
24 REM *   HAUPTPROGRAMM *
25 REM *
26 DIM DATS(17),FILES(15),FS(15),T15(
0),TAS(2),AAS(4),BAS(4),ES(4):ES="
140 BA=2400:REM BUFFER-ANFANG
180 GRAPHICS 18:POKE 710,232:POKE 712,
2:POKE 700,10
190 POSITION 2,0:?: NO;"GIBT-GIBSY VERB
20"
210 POSITION 0,1:?: NO;"
22"
230 POSITION 1,2:?: NO;"FOR ATARI 800B
185"
240 POSITION 3,5:?: NO;"MASCHINENDATEN"
250 POSITION 3,6:?: NO;"READY BL"
260 POSITION 1,9:?: NO;"DOCKE-REKORD"
27"
280 POSITION 0,10:?: NO;"
29"
300 POSITION 2,11:?: NO;"C HEKASOUND K
1"
31"
320 REM FUER "1985" EINGEBEN:INVERS UN
0 CTRL 1=0,9=9,8=8,5=0
330 GOSUB 6000
340 GOSUB 6000
350 GRAPHICS 18:POKE 710,232:POKE 712,
2:POKE 700,10
360 POSITION 3,0:?: NO;"STAR MIDIDISK"
370 POSITION 0,1:?: NO;"
38"
390 POSITION 2,3:?: NO;" REC.: TASTE 0

```

```

130 POSITION 2,4:?" M6;" PLAY: TASTE ☐
"
150 POSITION 2,6:?" M6;" DIR.: TASTE ☐
"
160 POSITION 2,8:?" M6;" SAVE: TASTE ☐
"
170 POSITION 2,9:?" M6;" LOAD: TASTE ☐
"
190 GOTO 410
400 POKE 752,1:POSITION 1,22:?"
DRUCKE HEAD-START
410 POKE 764,255
420 TASTE=PEEK(764)
430 IF TASTE=33 THEN 300
440 IF TASTE=40 THEN 1000:REM B
450 IF TASTE=10 THEN 1000: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 710,0:POKE 712,226
:POKE 709,0
720 POKE 752,1:POKE 764,255
730 POSITION 0,0:?" MID$F5,VERS.0
" C H R K E A S O U N D 1 9 0 5
740 POSITION 1,5:?"
"
750 POSITION 1,22:?"
"
790 RETURN
800 POSITION 10,11:?"
"
810 POSITION 1,22:?" MET RETURN
"
820 CLOSE H1:OPEN H1,4,0,"X":?"
830 POSITION 10,10:?" FILENAME 0:?" :IM
PUT FILE:CLOSE H1:TRAP 090
840 POSITION 1,22:?"
"
850 IF LEN(FILE$)>0 THEN POSITION 10,2
0:?" FILENAME ZU LANG:?"GOTO 400
860 IIS=FILES
870 FS="D:":FILES=LEN(FILE$)+1:?" .DAT"
880 FS=LEN(FILE$)+1:FILES=RETURN
890 POSITION 9,20:?" EIN-NEHMT VORNA:
" :IIS="":GOTO 400
1000 REM -----
1001 REM * RECORD/PLAY *
1002 REM -----
1010 GOSUB 700
1020 POSITION 14,2:?" R E C O R D ":TI$
="NEU"
1030 POSITION 0,4:?" ADRES CMD ☐
AT1 DATZ KOMMENTAR "
1040 POSITION 5,22:?" DATENSPEICHER:
24000 - 30000 "
1050 X=USR(0)10=0
1060 GOTO 300
1100 GOSUB 700
1110 POSITION 11,2:?" P L A Y : " :TI$
1120 POSITION 0,4:?" ADRES CMD ☐
AT1 DATZ KOMMENTAR "
1130 B=PEEK(254)+PEEK(255)*256-80
1140 IF B<=1 AND B>=0 THEN POSITION 10,
20:?" MASCHINENRECORD:GOTO 400
1150 POSITION 7,22:?" STOP: DRUCKE L
EERTASTE "
1160 X=USR(0)+4
1170 GOTO 400
2000 REM -----
2001 REM * DIRECTORY *
2002 REM -----
2010 GOSUB 700
2040 POSITION 11,3:?" D I R E C T O R
Y":?
2050 POKE 02,11:?"
2100 CLOSE H1:OPEN H1,6,0,"D:N,M"
2110 TRAP 2140
2120 INPUT H1,DATS:?" DATS
2130 GOTO 2120
2140 ?
2150 CLOSE H1:POKE 02,0: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 1,10:?"
"
3060 GOSUB 000
3100 LN=PEEK(254)+PEEK(255)*256-80
3120 IF LN<=0 THEN POSITION 10,20:?"
" :IIS="":GOTO 400
3130 CLOSE H1:OPEN H1,0,0,FS
3140 PUT H1,255:PUT H1,255
3150 PUT H1,192:PUT H1,93
3160 PUT H1,PEEK(254):PUT H1,PEEK(255)
3170 POSITION 12,12:?"Wird gespeichert
"
3180 X=USR(0),1,00,LN0
3190 CLOSE H1
3200 IF H1<120 THEN POSITION 10,20:?"
" :X="":GOTO 400
3220 GOTO 100
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 1,10:?"
"
4060 GOSUB 000
4100 CLOSE H1:OPEN H1,4,0,FS
4110 GET H1,H1:GET H1,R2
4120 IF H1<255 OR H2<255 THEN CLOSE
H1:POSITION 10,20:?" KEIN BINAER-FILE
" :IIS="":GOTO 400
4130 GET H1,AL:GET H1,AH:ADDR=AL+AH*25
6
4140 GET H1,AL:GET H1,AH:EADR=AL+AH*25
6
4150 LAE=EADR+ADDR
4160 POSITION 14,12:?" Wird geladen"
4170 X=USR(0),0,ADDR,LAE:IO=1
4180 CLOSE H1
4190 IF H1<120 THEN POSITION 10,20:?"
" :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:5=S+D:NEXT B
6030 IF S<411 THEN ?"I/O-DATEN-FEHL
ER":STOP
6040 MD=1526
6050 RETURN
6060 DATA 104,162,40,104,160,7,104,240
,2,160,11,152,157,86,3,184
6070 DATA 157,63,3,104,157,60,3,104,15
7,71,3,104,157,72,5,32,86,220
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:N=20400
7030 FOR A=20400 TO 21921:READ D:POKE
A,D:5=S+D
7035 WS=4:MC=005:STRS(M):0A05=ES:0A05(N
5-LEN(005)+1,WS)=005
7040 C=C-1
7045 POSITION 12,6:?" M6;" :?"A05(1,WS)
7050 NEXT A
7060 IF S<135905 THEN ?"DATEN-FEHLER
IM ZEILE 7320":STOP
7070 RETURN
7100 DATA 104,76,25,00,104,76,106,81,0
,232,0,0,37,20,0,0,1,176,64
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  
 80,3,76,75,81,32,4,87,52,201  
 7130 DATA 80,144,238,160,8,145,254,201  
 128,144,10,166,254,142,21,80  
 7140 DATA 166,255,145,142,2,80,72,32,122,  
 82,184,230,254,200,2,230,255  
 7150 DATA 166,255,224,144,144,3,76,75,  
 81,201,120,144,176,160,8,177  
 7160 DATA 22,230,222,145,254,177,222,  
 230,222,200,145,254,165,254  
 7170 DATA 24,105,2,133,254,165,255,185  
 0,133,255,201,140,144,3,76  
 7180 DATA 75,81,76,39,80,120,163,217,1  
 41,22,2,169,80,141,22,2,159  
 7190 DATA 3,141,0,209,169,146,141,0,20  
 9,169,0,133,200,133,210,169  
 7200 DATA 80,133,207,133,211,169,0,133  
 220,133,222,169,89,133,221  
 7210 DATA 133,221,169,0,141,3,80,141,1  
 0,80,141,11,80,141,35,80,141  
 7220 DATA 11,80,80,96,165,200,197,210,  
 240,0,160,0,177,210,230,210  
 7230 DATA 56,96,24,96,120,72,130,72,15  
 2,72,32,30,81,173,2,209,74,144  
 7240 DATA 46,160,0,173,2,209,145,206,1  
 6,29,141,0,80,201,240,240,24  
 7250 DATA 201,254,240,20,140,8,80,173,  
 9,80,145,220,100,173,10,160  
 7260 DATA 220,230,220,230,220,230,200,  
 104,160,104,170,104,64,104,160  
 7270 DATA 104,170,104,70,45,152,163,2,  
 26,20,240,20,44,11,80,16,33  
 7280 DATA 230,3,80,200,3,220,10,80,169  
 0,141,11,80,240,18,44,11,80  
 7290 DATA 40,13,230,9,80,200,3,210,10,  
 80,163,230,141,11,80,96,160  
 7300 DATA 0,163,255,145,254,152,200,14  
 5,254,200,145,254,120,163,3  
 7310 DATA 141,0,209,169,48,141,22,2,16  
 9,132,141,22,2,80,30,32,133  
 7320 DATA 83,32,214,81,173,252,2,201,3  
 3,200,3,76,23,82,160,2,173,10  
 7330 DATA 80,209,254,144,237,136,173,9  
 80,209,254,144,237,136,177  
 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  
 82,165,254,24,185,3,133,254  
 7360 DATA 165,255,185,0,133,255,201,14  
 0,144,3,76,23,82,160,0,177,254  
 7370 DATA 48,179,32,3,82,32,122,82,230  
 254,200,240,230,255,165,250  
 7380 DATA 224,48,144,237,23,82,120  
 216,162,82,160,16,163,7,32  
 7390 DATA 52,220,169,3,141,0,209,169,1  
 0,141,0,209,169,192,133,254  
 7400 DATA 169,93,133,255,169,0,141,9,0  
 0,141,10,80,141,15,80,141,12  
 7410 DATA 80,80,96,72,172,2,209,74,74,  
 144,249,184,141,3,209,96,120  
 7420 DATA 32,30,81,76,96,220,163,252,3  
 2,3,82,169,255,32,3,82,169,255  
 7430 DATA 32,3,82,169,255,32,3,82,76,8  
 8,1,173,0,80,201,240,240,14  
 7440 DATA 201,254,240,20,197,20,240,1,  
 96,182,20,76,84,82,169,0,141  
 7450 DATA 3,80,162,16,76,84,82,169,0,1  
 41,0,80,162,16,189,133,85,141  
 7460 DATA 185,82,232,187,132,85,141,10  
 6,82,169,0,141,11,80,160,8,189  
 7470 DATA 230,84,174,13,80,187,154,156  
 238,11,80,200,192,20,200,233  
 7480 DATA 96,174,15,80,157,17,80,240,1  
 0,201,120,144,22,169,9,141,15  
 7490 DATA 80,96,201,120,144,251,173,17  
 80,41,127,74,74,74,74,141,16  
 7500 DATA 80,230,15,80,172,16,80,185,1  
 54,85,205,15,80,172,225,168  
 7510 DATA 173,17,80,201,144,200,10,173  
 17,80,201,0,200,3,141,16,80  
 7520 DATA 169,2,141,12,80,173,127,83,2

81,159,200,13,173,126,83,201  
 7530 DATA 120,144,6,32,132,83,76,227,8  
 2,173,126,83,24,105,40,141,126  
 7540 DATA 83,144,3,230,127,83,37,171,0  
 3,165,213,32,100,83,165,213  
 7550 DATA 32,79,83,183,215,32,79,83,16  
 9,18,141,32,80,160,8,189,17  
 7560 DATA 80,142,20,80,201,144,200,10,  
 152,72,32,192,83,104,160,76  
 7570 DATA 25,83,32,99,83,238,12,80,230  
 12,80,230,12,80,220,12,80,230  
 7580 DATA 12,80,170,28,80,232,224,2,20  
 8,6,200,12,80,206,12,80,136  
 7590 DATA 16,201,163,0,141,15,80,163,2  
 7,141,12,80,173,18,10,170  
 7600 DATA 187,132,85,141,82,83,232,189  
 132,85,141,83,85,160,8,185  
 7610 DATA 230,84,170,73,64,240,7,130,3  
 2,122,83,200,200,241,76,139  
 7620 DATA 82,72,74,74,74,74,74,74,74,74,  
 184,41,15,201,10,144,5,24,85  
 7630 DATA 23,200,3,24,182,16,174,12,80  
 157,40,157,230,12,80,160,170  
 7640 DATA 0,133,77,162,48,142,120,83,1  
 69,157,141,127,83,162,0,169  
 7650 DATA 0,157,48,157,232,200,250,157,  
 48,158,232,200,250,157,48,159  
 7660 DATA 232,224,255,200,248,96,152,7  
 2,32,60,218,172,21,80,133,212  
 7670 DATA 172,22,80,132,213,32,170,217  
 104,160,96,162,6,169,100,96  
 7680 DATA 232,12,202,240,7,205,10,00,2  
 48,2,170,243,141,24,80,173,18  
 7690 DATA 80,56,237,24,80,18,10,224,0,  
 240,6,24,105,40,202,200,250  
 7700 DATA 160,169,4,141,24,80,105,250,  
 83,32,122,83,200,200,24,80,200  
 7710 DATA 244,96,35,0,8,35,41,51,0,1  
 6,0,9,36,41,51,0,37,0,0,0  
 7720 DATA 38,0,0,30,41,51,0,39,0,0,0  
 39,41,51,0,33,0,0,44,0,0  
 7730 DATA 0,40,0,0,59,0,0,0,99,105,1  
 0,5,100,0,0,100,105,115,0  
 7740 DATA 101,0,0,102,0,0,102,105,  
 115,0,102,0,0,101,105,115  
 7750 DATA 0,97,0,0,98,0,0,0,104,0,0,  
 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 17,183,17,0,0,183,105,115,17  
 97,17,0,0,98,17,0,0,104,17  
 7780 DATA 0,0,99,18,0,0,99,105,115,10,  
 100,10,0,100,105,115,18,101  
 7790 DATA 18,0,0,102,18,0,0,102,105,11  
 5,10,105,18,0,0,103,105,115  
 7800 DATA 18,97,10,0,0,90,10,0,104,1  
 0,0,99,19,0,0,99,105,115,13  
 7810 DATA 100,87,0,0,100,105,115,13,10  
 1,13,0,0,102,13,0,0,102,105  
 7820 DATA 115,15,103,19,0,0,102,105,11  
 5,13,37,13,0,0,90,13,0,0,104  
 7830 DATA 17,0,0,99,20,0,0,46,47,52,57  
 0,47,30,38,64,46,47,52,17,0  
 7840 DATA 47,46,84,48,47,44,57,13,48,5  
 8,37,51,51,64,33,47,46,52,50  
 7850 DATA 47,44,44,37,50,64,48,50,47,3  
 9,14,35,40,33,46,39,17,64,33  
 7860 DATA 40,43,46,40,14,40,50,37,51,5  
 1,64,48,41,52,15,40,0,34,37  
 7870 DATA 40,26,64,51,57,51,52,37,45,1  
 1,41,46,38,47,64,0,20,24,0,11  
 7880 DATA 0,52,41,45,41,46,39,0,15,0,3  
 5,44,47,35,43,64,43,37,57  
 7890 DATA 34,47,33,50,30,6,51,37,46,36  
 37,34,37,50,37,41,52,64,0,8  
 7900 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0  
 0,0,0,0,64,230,84,247,84,250  
 7910 DATA 84,10,85,21,85,13,85,45,85,5  
 6,85,68,85,98,05,111,85,2,0  
 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 keiner 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 so wie 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 auflisten. 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.

### 1. (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 #51C    unmittelbar
LDA 502FC   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 ($700)  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
```

Arzo Weizel

```
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 1686:READ
A:POKE I,&NEXT I
20 DATA 162,48,169,1,157,66,2,167,68,1
57,68,3,169,6,157,67
30 DATA 3,169,8,157,74,3,169,128,157,7
5,3,2,86,228,162,48
40 DATA 184,184,157,67,3,184,157,68,3,
184,157,73,3,184,157,72
50 DATA 3,169,11,157,68,3,32,86,228,16
2,48,169,12,157,66,4
60 DATA 32,86,228,96,67,58,155
70 GRAMPNICS 0:SETCOLOR 2,0,0
80 ? "MicroMon / Basic-Loader"
90 ? "(C) 1986 by Freak"
100 DIM A$(16):RESTORE 1000
110 ? "H-load-Routine":H:=04036
120 FOR I=0 TO 15:READ A$,B$:C=0
130 FOR J=0 TO 7:J=J+1
E:=ASC(A$(0,B)):F:=ASC(A$(0,1)+1)
150 G=16#(E-48)-(C-57)*7+(F-48-(F)57)*
7)
```

```

168 POKE H,G:H=H1:C=C+G:WEXT J
170 IF C=0 THEN ?"Datenfehler in Zei
171 ""1000":END
180 ? "+");I-1:" ""NEXT I
190 ? "Monitor":H=74089E
200 FOR I=0 TO 511:READ A5,B,C=0
210 FOR I=0 TO 7:READ I1
220 A=C-(A&C,D,D):J=-(A&C,A5,B,I1)
230 C=10*(C-48)-(E)57*73+I+48-(F)57*7
240 POKE H,G:H=H1+C=C+G:WEXT J
250 IF C=0 THEN ?"Datenfehler in Zei
251 ""102":END
260 ? "+");511-I:" ""NEXT I
270 ? "Load-Routine saved ... "
280 A=USR(1536,68486,154)
290 ? "Monitor saved ..."
300 POKE 784,12:A=USR(1536,74089E,409E
310):POKE 784,255
310 ? "4es ist geschafft !!! "
320 END
1000 DATA 000200077907409C,325
1001 DATA 074370704370902,059
1002 DATA 102037000101070E,1045
1003 DATA 80FC92021003070E,934
1004 DATA 428307099440030E,534
1005 DATA 0779045020704090,640
1006 DATA 030939074003208E,650
1007 DATA E40210070770428E,940
1008 DATA 0700790440307099,983
1009 DATA 4503097704403079,937
1010 DATA FF7047032052402E,39E
1011 DATA 20030C790420320E,57E
1012 DATA E40790E50E00E70E,1072
1013 DATA 0707950F00E00E20,1072
1014 DATA 7905000705000100,800
1015 DATA 00009707C7000000,844
1016 DATA 09FF097707407907,350
1017 DATA 070309012012000E,1078
1018 DATA 0703000120160000,747
1019 DATA 0000000000000000,0
1020 DATA 425240004F524187,468
1021 DATA 2F3F3F0A3F3F0E,370
1022 DATA 2F3F3F0A3F52410A,429
1023 DATA 41514C063F3F0E,429
1024 DATA 504859004F524100,468
1025 DATA 41534C063F3F0E,429
1026 DATA 2F3F3F0A3F524181,426
1027 DATA 41534C013F3F0E,424
1028 DATA 42594C063F524100,465
1029 DATA 2F3F3F0A3F3F0E,370
1030 DATA 2F3F3F0A3F52410E,429
1031 DATA 41534C043F3F0E,427
1032 DATA 434C43004F524181,449
1033 DATA 2F3F3F0A3F3F0E,370
1034 DATA 2F3F3F0A4F524181,427
1035 DATA 41534C023F3F0E,425
1036 DATA 4053201414E4007,458
1037 DATA 2F3F3F0A3F3F0E,370
1038 DATA 4249540414E4400A,442
1039 DATA 524F4C043F3F0E,440
1040 DATA 50535000414E400E,440
1041 DATA 524F4C043F3F0E,440
1042 DATA 4249540414E4401,435
1043 DATA 2F3F3F0A3F3F0E,370
1044 DATA 424D490414E4400A,444
1045 DATA 2F3F3F0A3F3F0E,370
1046 DATA 2F3F3F0A14E4400E,411
1047 DATA 524F4C053F3F0E,441
1048 DATA 53454300414E4401,443
1049 DATA 2F3F3F0A3F3F0E,370
1050 DATA 2F3F3F0A414E4402,411
1051 DATA 524F4C043F3F0E,438
1052 DATA 5244490454F5207,406
1053 DATA 2F3F3F0A3F3F0E,370
1054 DATA 2F3F3F0A54F5204,432
1055 DATA 4C5252043F3F0E,444
1056 DATA 5048410045F5200,457
1057 DATA 4C5252043F3F0E,450
1058 DATA 40493001454F5201,463
1059 DATA 4C5252013F3F0E,441
1060 DATA 5244490454F5207,406
1061 DATA 2F3F3F0A3F3F0E,370
1062 DATA 2F3F3F0A54F520E,434
1063 DATA 4C5252053F3F0E,445
1064 DATA 434C4300454F5203,435
1065 DATA 2F3F3F0A3F3F0E,370
1066 DATA 2F3F3F0A454F5202,431
1067 DATA 4C532023F3F0E,450
1068 DATA 53454300414E4401,443
1069 DATA 2F3F3F0A3F3F0E,370
1070 DATA 2F3F3F0A3F3F0E,370
1071 DATA 2F3F3F0A3F3F0E,370
1072 DATA 2F3F3F0A3F3F0E,370
1073 DATA 2F3F3F0A3F3F0E,370
1074 DATA 2F3F3F0A3F3F0E,370
1075 DATA 2F3F3F0A3F3F0E,370
1076 DATA 2F3F3F0A3F3F0E,370
1077 DATA 2F3F3F0A3F3F0E,370
1078 DATA 2F3F3F0A3F3F0E,370
1079 DATA 2F3F3F0A3F3F0E,370
1080 DATA 2F3F3F0A3F3F0E,370
1081 DATA 2F3F3F0A3F3F0E,370
1082 DATA 2F3F3F0A3F3F0E,370
1083 DATA 2F3F3F0A3F3F0E,370
1084 DATA 2F3F3F0A3F3F0E,370
1085 DATA 2F3F3F0A3F3F0E,370
1086 DATA 2F3F3F0A3F3F0E,370
1087 DATA 2F3F3F0A3F3F0E,370
1088 DATA 2F3F3F0A3F3F0E,370
1089 DATA 2F3F3F0A3F3F0E,370
1090 DATA 2F3F3F0A3F3F0E,370
1091 DATA 2F3F3F0A3F3F0E,370
1092 DATA 2F3F3F0A3F3F0E,370
1093 DATA 2F3F3F0A3F3F0E,370
1094 DATA 2F3F3F0A3F3F0E,370
1095 DATA 2F3F3F0A3F3F0E,370
1096 DATA 2F3F3F0A3F3F0E,370
1097 DATA 2F3F3F0A3F3F0E,370
1098 DATA 2F3F3F0A3F3F0E,370
1099 DATA 2F3F3F0A3F3F0E,370
1100 DATA 2F3F3F0A3F3F0E,370
1101 DATA 2F3F3F0A3F3F0E,370
1102 DATA 2F3F3F0A3F3F0E,370
1103 DATA 2F3F3F0A3F3F0E,370
1104 DATA 2F3F3F0A3F3F0E,370
1105 DATA 2F3F3F0A3F3F0E,370
1106 DATA 2F3F3F0A3F3F0E,370
1107 DATA 2F3F3F0A3F3F0E,370
1108 DATA 2F3F3F0A3F3F0E,370
1109 DATA 2F3F3F0A3F3F0E,370
1110 DATA 2F3F3F0A3F3F0E,370
1111 DATA 2F3F3F0A3F3F0E,370
1112 DATA 2F3F3F0A3F3F0E,370
1113 DATA 2F3F3F0A3F3F0E,370
1114 DATA 2F3F3F0A3F3F0E,370
1115 DATA 2F3F3F0A3F3F0E,370
1116 DATA 2F3F3F0A3F3F0E,370
1117 DATA 2F3F3F0A3F3F0E,370
1118 DATA 2F3F3F0A3F3F0E,370
1119 DATA 2F3F3F0A3F3F0E,370
1120 DATA 2F3F3F0A3F3F0E,370
1121 DATA 2F3F3F0A3F3F0E,370
1122 DATA 2F3F3F0A3F3F0E,370
1123 DATA 2F3F3F0A3F3F0E,370
1124 DATA 2F3F3F0A3F3F0E,370
1125 DATA 2F3F3F0A3F3F0E,370
1126 DATA 2F3F3F0A3F3F0E,370
1127 DATA 2F3F3F0A3F3F0E,370
1128 DATA 2F3F3F0A3F3F0E,370
1129 DATA 2F3F3F0A3F3F0E,370
1130 DATA 2F3F3F0A3F3F0E,370
1131 DATA 2F3F3F0A3F3F0E,370
1132 DATA 2F3F3F0A3F3F0E,370
1133 DATA 2F3F3F0A3F3F0E,370
1134 DATA 2F3F3F0A3F3F0E,370
1135 DATA 2F3F3F0A3F3F0E,370
1136 DATA 2F3F3F0A3F3F0E,370
1137 DATA 2F3F3F0A3F3F0E,370
1138 DATA 2F3F3F0A3F3F0E,370
1139 DATA 2F3F3F0A3F3F0E,370
1140 DATA 2F3F3F0A3F3F0E,370
1141 DATA 2F3F3F0A3F3F0E,370
1142 DATA 2F3F3F0A3F3F0E,370
1143 DATA 2F3F3F0A3F3F0E,370
1144 DATA 2F3F3F0A3F3F0E,370
1145 DATA 2F3F3F0A3F3F0E,370
1146 DATA 2F3F3F0A3F3F0E,370
1147 DATA 2F3F3F0A3F3F0E,370
1148 DATA 2F3F3F0A3F3F0E,370
1149 DATA 2F3F3F0A3F3F0E,370
1150 DATA 2F3F3F0A3F3F0E,370
1151 DATA 2F3F3F0A3F3F0E,370
1152 DATA 2F3F3F0A3F3F0E,370
1153 DATA 2F3F3F0A3F3F0E,370
1154 DATA 2F3F3F0A3F3F0E,370
1155 DATA 2F3F3F0A3F3F0E,370
1156 DATA 2F3F3F0A3F3F0E,370
1157 DATA 2F3F3F0A3F3F0E,370
1158 DATA 2F3F3F0A3F3F0E,370
1159 DATA 2F3F3F0A3F3F0E,370
1160 DATA 2F3F3F0A3F3F0E,370
1161 DATA 2F3F3F0A3F3F0E,370
1162 DATA 2F3F3F0A3F3F0E,370
1163 DATA 2F3F3F0A3F3F0E,370
1164 DATA 2F3F3F0A3F3F0E,370
1165 DATA 2F3F3F0A3F3F0E,370
1166 DATA 2F3F3F0A3F3F0E,370
1167 DATA 2F3F3F0A3F3F0E,370
1168 DATA 2F3F3F0A3F3F0E,370
1169 DATA 2F3F3F0A3F3F0E,370
1170 DATA 2F3F3F0A3F3F0E,370
1171 DATA 2F3F3F0A3F3F0E,370
1172 DATA 2F3F3F0A3F3F0E,370
1173 DATA 2F3F3F0A3F3F0E,370
1174 DATA 2F3F3F0A3F3F0E,370
1175 DATA 2F3F3F0A3F3F0E,370
1176 DATA 2F3F3F0A3F3F0E,370
1177 DATA 2F3F3F0A3F3F0E,370
1178 DATA 2F3F3F0A3F3F0E,370
1179 DATA 2F3F3F0A3F3F0E,370
1180 DATA 2F3F3F0A3F3F0E,370
1181 DATA 2F3F3F0A3F3F0E,370
1182 DATA 2F3F3F0A3F3F0E,370
1183 DATA 2F3F3F0A3F3F0E,370
1184 DATA 2F3F3F0A3F3F0E,370
1185 DATA 2F3F3F0A3F3F0E,370
1186 DATA 2F3F3F0A3F3F0E,370
1187 DATA 2F3F3F0A3F3F0E,370
1188 DATA 2F3F3F0A3F3F0E,370
1189 DATA 2F3F3F0A3F3F0E,370
1190 DATA 2F3F3F0A3F3F0E,370
1191 DATA 2F3F3F0A3F3F0E,370
1192 DATA 2F3F3F0A3F3F0E,370
1193 DATA 2F3F3F0A3F3F0E,370
1194 DATA 2F3F3F0A3F3F0E,370
1195 DATA 2F3F3F0A3F3F0E,370
1196 DATA 2F3F3F0A3F3F0E,370
1197 DATA 2F3F3F0A3F3F0E,370
1198 DATA 2F3F3F0A3F3F0E,370
1199 DATA 2F3F3F0A3F3F0E,370
1200 DATA 2F3F3F0A3F3F0E,370
1201 DATA 2F3F3F0A3F3F0E,370
1202 DATA 2F3F3F0A3F3F0E,370
1203 DATA 2F3F3F0A3F3F0E,370
1204 DATA 2F3F3F0A3F3F0E,370
1205 DATA 2F3F3F0A3F3F0E,370
1206 DATA 2F3F3F0A3F3F0E,370
1207 DATA 2F3F3F0A3F3F0E,370
1208 DATA 2F3F3F0A3F3F0E,370
1209 DATA 2F3F3F0A3F3F0E,370
1210 DATA 2F3F3F0A3F3F0E,370
1211 DATA 2F3F3F0A3F3F0E,370
1212 DATA 2F3F3F0A3F3F0E,370
1213 DATA 2F3F3F0A3F3F0E,370
1214 DATA 2F3F3F0A3F3F0E,370
1215 DATA 2F3F3F0A3F3F0E,370
1216 DATA 2F3F3F0A3F3F0E,370
1217 DATA 2F3F3F0A3F3F0E,370
1218 DATA 2F3F3F0A3F3F0E,370
1219 DATA 2F3F3F0A3F3F0E,370
1220 DATA 2F3F3F0A3F3F0E,370
1221 DATA 2F3F3F0A3F3F0E,370
1222 DATA 2F3F3F0A3F3F0E,370
1223 DATA 2F3F3F0A3F3F0E,370
1224 DATA 2F3F3F0A3F3F0E,370
1225 DATA 2F3F3F0A3F3F0E,370
1226 DATA 2F3F3F0A3F3F0E,370
1227 DATA 2F3F3F0A3F3F0E,370
1228 DATA 2F3F3F0A3F3F0E,370
1229 DATA 2F3F3F0A3F3F0E,370
1230 DATA 2F3F3F0A3F3F0E,370
1231 DATA 2F3F3F0A3F3F0E,370
1232 DATA 2F3F3F0A3F3F0E,370
1233 DATA 2F3F3F0A3F3F0E,370
1234 DATA 2F3F3F0A3F3F0E,370
1235 DATA 2F3F3F0A3F3F0E,370
1236 DATA 2F3F3F0A3F3F0E,370
1237 DATA 2F3F3F0A3F3F0E,370
1238 DATA 2F3F3F0A3F3F0E,370
1239 DATA 2F3F3F0A3F3F0E,370
1240 DATA 2F3F3F0A3F3F0E,370
1241 DATA 2F3F3F0A3F3F0E,370
1242 DATA 2F3F3F0A3F3F0E,370
1243 DATA 2F3F3F0A3F3F0E,370
1244 DATA 2F3F3F0A3F3F0E,370
1245 DATA 2F3F3F0A3F3F0E,370
1246 DATA 2F3F3F0A3F3F0E,370
1247 DATA 2F3F3F0A3F3F0E,370
1248 DATA 2F3F3F0A3F3F0E,370
1249 DATA 2F3F3F0A3F3F0E,370
1250 DATA 2F3F3F0A3F3F0E,370
1251 DATA 2F3F3F0A3F3F0E,370
1252 DATA 2F3F3F0A3F3F0E,370
1253 DATA 2F3F3F0A3F3F0E,370
1254 DATA 2F3F3F0A3F3F0E,370
1255 DATA 2F3F3F0A3F3F0E,370
1256 DATA 2F3F3F0A3F3F0E,370
1257 DATA 2F3F3F0A3F3F0E,370
1258 DATA 2F3F3F0A3F3F0E,370
1259 DATA 2F3F3F0A3F3F0E,370
1260 DATA 2F3F3F0A3F3F0E,370
1261 DATA 2F3F3F0A3F3F0E,370
1262 DATA 2F3F3F0A3F3F0E,370
1263 DATA 2F3F3F0A3F3F0E,370
1264 DATA 2F3F3F0A3F3F0E,370
1265 DATA 2F3F3F0A3F3F0E,370
1266 DATA 2F3F3F0A3F3F0E,370
1267 DATA 2F3F3F0A3F3F0E,370
1268 DATA 2F3F3F0A3F3F0E,370
1269 DATA 2F3F3F0A3F3F0E,370
1270 DATA 2F3F3F0A3F3F0E,370
1271 DATA 2F3F3F0A3F3F0E,370
1272 DATA 2F3F3F0A3F3F0E,370
1273 DATA 2F3F3F0A3F3F0E,370
1274 DATA 2F3F3F0A3F3F0E,370
1275 DATA 2F3F3F0A3F3F0E,370
1276 DATA 2F3F3F0A3F3F0E,370
1277 DATA 2F3F3F0A3F3F0E,370
1278 DATA 2F3F3F0A3F3F0E,370
1279 DATA 2F3F3F0A3F3F0E,370
1280 DATA 2F3F3F0A3F3F0E,370
1281 DATA 2F3F3F0A3F3F0E,370
1282 DATA 2F3F3F0A3F3F0E,370
1283 DATA 2F3F3F0A3F3F0E,370
1284 DATA 2F3F3F0A3F3F0E,370
1285 DATA 2F3F3F0A3F3F0E,370
1286 DATA 2F3F3F0A3F3F0E,370
1287 DATA 2F3F3F0A3F3F0E,370
1288 DATA 2F3F3F0A3F3F0E,370
1289 DATA 2F3F3F0A3F3F0E,370
1290 DATA 2F3F3F0A3F3F0E,370
1291 DATA 2F3F3F0A3F3F0E,370
1292 DATA 2F3F3F0A3F3F0E,370
1293 DATA 2F3F3F0A3F3F0E,370
1294 DATA 2F3F3F0A3F3F0E,370
1295 DATA 2F3F3F0A3F3F0E,370
1296 DATA 2F3F3F0A3F3F0E,370
1297 DATA 2F3F3F0A3F3F0E,370
1298 DATA 2F3F3F0A3F3F0E,370
1299 DATA 2F3F3F0A3F3F0E,370
1300 DATA 2F3F3F0A3F3F0E,370

```



1240	D6TA	A42800A046C8A5E5, 1238	1334	D6TA	8A480B3A92800F2, 1165	1468	D6TA	836606A80001A014, 593
1241	D6TA	A57F45D0A61B0A11, 1095	1335	D6TA	8A6E8E8799E7E8, 1218	1469	D6TA	A43189C80B15A491, 1612
1242	D6TA	8E7F04D4058A8A5E, 1823	1336	D6TA	Z8A76D24277A85, 742	1470	D6TA	A46179D35F660A80, 563
1243	D6TA	A673467806A2A000A, 1033	1337	D6TA	729A2900D7C679, 1069	1471	D6TA	B88A739A186A48, 1181
1244	D6TA	A8D98F6A7280AF260, 937	1338	D6TA	10A4E3E0A80F5A2, 1255	1472	D6TA	94A8F5A28E021A04, 573
1245	D6TA	A8E3E11CD8F16879, 1324	1339	D6TA	002B10A4A014A485, 710	1473	D6TA	E87F903288680A5, 1871
1246	D6TA	A8E67E726F486F6, 884	1340	D6TA	80AD15A4A5B168A0, 761	1474	D6TA	A105855A005186392, 435
1247	D6TA	702A412728151339, 477	1341	D6TA	85A7C9230BCAD3E, 1063	1475	D6TA	85049002E60A014, 711
1248	D6TA	Z428617284E7255, 622	1342	D6TA	A7C924D005A98085, 517	1476	D6TA	A48380A015A48507, 881
1249	D6TA	A16870A2800A4A39, 922	1343	D6TA	87E8008587C78480, 1021	1477	D6TA	A587C5851871A86, 448
1250	D6TA	D0A72800F26A0A6E, 1184	1344	D6TA	A2A08A7C9280095, 734	1478	D6TA	C5843831A6507385, 755
1251	D6TA	E0A048F1649E4F48, 1082	1345	D6TA	A9A48576600A867, 482	1479	D6TA	85585853E858A, 1842
1252	D6TA	702A44F2C39B00F9, 1316	1346	D6TA	C32A0B13A088A793, 1141	1480	D6TA	85A48002C4980A5, 688
1253	D6TA	C0B10C9F5A538508, 1832	1347	D6TA	ZC8900A0A8D9A7958, 1030	1481	D6TA	C908F085A37F8506, 1809
1254	D6TA	A559581A65A4CA08, 974	1348	D6TA	8085A70E50876A00, 796	1482	D6TA	86058A9C7330E14, 720
1255	D6TA	A4A280A58918658A, 490	1349	D6TA	85A7C92A0B13A088, 1071	1483	D6TA	8C4C0A00A59A3180, 787
1256	D6TA	8500780E2E81E3E, 766	1350	D6TA	A7C92C900CA09367, 1109	1484	D6TA	A98285806A08A52, 632
1257	D6TA	Z800F80082B10018, 951	1351	D6TA	C95700859A68807, 018	1485	D6TA	E58785878A81A023, 838
1258	D6TA	63289379A78A82E, 1088	1352	D6TA	A885A7C92A0B13A088, 1052	1486	D6TA	805858802C687A65, 693
1259	D6TA	80F3584728383A18, 900	1353	D6TA	87E8008587C78480, 1021	1487	D6TA	87C908F085A37F85, 1810
1260	D6TA	Z028282828282828, 272	1354	D6TA	87E8008587C78480, 1021	1488	D6TA	8688588C7832383, 855
1261	D6TA	Z028282828282828, 256	1355	D6TA	Z80085A90A8576A0, 650	1489	D6TA	A4C80A8C8A643FF18, 568
1262	D6TA	Z028282828282828, 256	1356	D6TA	A885A7C92A0B13A088, 1128	1490	D6TA	59A100A11300A92, 503
1263	D6TA	Z028282828282828, 448	1357	D6TA	86A7C92A0B13A088, 1071	1491	D6TA	8586860A9800B02, 745
1264	D6TA	8028F906Z80A779, 733	1358	D6TA	C958088CA088A793, 1189	1492	D6TA	A31A0D028A29028, 861
1265	D6TA	A1673100A870A, 1878	1359	D6TA	Z90089A7870A760, 568	1493	D6TA	D0A22800A480B30, 823
1266	D6TA	E0A0F088E38A100A, 1212	1360	D6TA	A885A7C92A0B13A088, 1128	1494	D6TA	A98280F280A8E8B, 1351
1267	D6TA	F3A4C7F790A0A0A, 1158	1361	D6TA	86A7C92A0B13A088, 1071	1495	D6TA	A8FC0E311E8F8E39, 1123
1268	D6TA	E0A0D0CC4780C40, 1488	1362	D6TA	86A7C92A0B13A088, 1114	1496	D6TA	1F08F50A7F80F0C2, 1287
1269	D6TA	80C9A72826A8A4C7, 934	1363	D6TA	C32C900CA08A793, 1145	1497	D6TA	A394A8C682A3288, 1008
1270	D6TA	A749443B2E475148, 635	1364	D6TA	598085A988A5876A, 715	1498	D6TA	Z8A28A650A0A8A29, 323
1271	D6TA	A4C886817A84A8818, 707	1365	D6TA	A885A7C92A0B13A088, 1114	1499	D6TA	702808F260A97F80, 1236
1272	D6TA	A82E8A8130A43A0A, 867	1366	D6TA	86A7C92A0B13A088, 1071	1500	D6TA	F0C2280A74C90A8, 874
1273	D6TA	A9A2800D78A7918, 985	1367	D6TA	A7C9270005A98085, 935	1501	D6TA	112D424153A43A98, 683
1274	D6TA	A4E3E80A409F8A280, 1233	1368	D6TA	A7C9270005A98085, 935	1502	D6TA	32Z4A076E657A46, 725
1275	D6TA	2013A04A01A4A4018, 748	1369	D6TA	8C0A8A8A7C92A0B13A088, 1128	1503	D6TA	732898088000000, 424
1276	D6TA	A00A150A8017A048, 946	1370	D6TA	A918130A760A85A7, 879	1504	D6TA	808008000000000, 1184
1277	D6TA	Z8C367E0FFD08168, 1216	1371	D6TA	C92A0B13A088A793, 1141	1505	D6TA	8000828E70E8FF80, 1257
1278	D6TA	Z8D78A428A28A6E828, 718	1372	D6TA	ZC890A808A7C958, 1032	1506	D6TA	81680A140A858000, 760
1279	D6TA	F3A7E70FF90816828, 1216	1373	D6TA	8085A7C92A0B13A088, 1071	1507	D6TA	15A0485018A2E2800, 578
1280	D6TA	3A452828A78E02E9, 828	1374	D6TA	85A7C92A0B13A088, 1071	1508	D6TA	F2A97F85868A000A, 1123
1281	D6TA	A780FF9081680A018, 1146	1375	D6TA	A7C92C900CA09367, 1109	1509	D6TA	81280A448A928080, 766
1282	D6TA	A4A8380A0817A040, 1833	1376	D6TA	C958088CA088A793, 1189	1510	D6TA	F2A97F85868A000A, 1123
1283	D6TA	3C48280085A28A67, 988	1377	D6TA	80A7F85878A28E7, 815	1511	D6TA	A98280F280A8E8B, 1351
1284	D6TA	60EAC80A02808087C, 1173	1378	D6TA	80A7F85878A28E7, 815	1512	D6TA	80F2A4C3285A522E, 1808
1285	D6TA	A739180A4E809A08, 1273	1379	D6TA	80A4A70A885886A8, 988	1513	D6TA	80F2A4C3285A522E, 1808
1286	D6TA	F5A2800210A4E0FF, 1186	1380	D6TA	80818A049E17A880, 988	1514	D6TA	80F2A4C3285A522E, 1808
1287	D6TA	808160A081A4A4800, 795	1381	D6TA	C8C80A80F458E06, 1188	1515	D6TA	D8E3170F2808F2E0, 1423
1288	D6TA	80F150A9581808087, 857	1382	D6TA	A59A18E9A058490, 583	1516	D6TA	8080802C061A9FF80, 1886
1289	D6TA	81A78013A4898A7, 1121	1383	D6TA	82E9A5A585C76A08, 914	1517	D6TA	F82A8FCE2337FF8, 1077
1290	D6TA	813A6478228180A, 780	1384	D6TA	E8A98285880A8289A, 810	1518	D6TA	F7C721F80F280A7, 1160
1291	D6TA	E8FF78081689A40A, 1181	1385	D6TA	A9E8FF8A016828E7, 1158	1519	D6TA	A9778FFC926A8A44, 1448
1292	D6TA	1A4A0800918086A8, 761	1386	D6TA	A7C9270005A98085, 935	1520	D6TA	Z8E4A732A8A4C32E, 496
1293	D6TA	E8680802E581C8C8, 1071	1387	D6TA	A7C9270005A98085, 935	1521	D6TA	Z8E4A732A8A4C32E, 496
1294	D6TA	C3C810D802A932828, 1094	1388	D6TA	28E6A6A586C90280, 1014	1522	D6TA	7A4378282828000A, 811
1295	D6TA	80F2A4C3285A522E, 1808	1389	D6TA	81680A085A0691A0, 573	1523	D6TA	1708A4A04A8C78A8, 832
1296	D6TA	A481280F40A4A2828, 774	1390	D6TA	A58780A88E78A080, 976	1524	D6TA	13A525A04A8A7C0A0, 1887
1297	D6TA	80F2A4C3285A522E, 1808	1391	D6TA	51A880F7C880852A8, 1138	1525	D6TA	Z8EA3AEECA8C8B0A, 1357
1298	D6TA	Z81A9F9F90448A3A9, 808	1392	D6TA	Z806AC5A96C9FF98, 1045	1526	D6TA	8DAD80A8D8A0A208, 1248
1299	D6TA	A780FF9081680A018, 1146	1393	D6TA	81680A085A0691A0, 573	1527	D6TA	807C8A7918A0A8E, 1273
1300	D6TA	A4A8380A0817A040, 1833	1394	D6TA	8085A7C92A0B13A088, 1071	1528	D6TA	8A0F93A280A2180A, 837
1301	D6TA	3C48280085A28A67, 988	1395	D6TA	8085A7C92A0B13A088, 1071	1529	D6TA	8085A7C92A0B13A088, 1071
1302	D6TA	60EAC80A02808087C, 1173	1396	D6TA	80F2A4C3285A522E, 1808	1530	D6TA	8085A7C92A0B13A088, 1071
1303	D6TA	A739180A4E809A08, 1273	1397	D6TA	80F2A4C3285A522E, 1808	1531	D6TA	8085A7C92A0B13A088, 1071
1304	D6TA	F5A2800210A4E0FF, 1186	1398	D6TA	80F2A4C3285A522E, 1808	1532	D6TA	8085A7C92A0B13A088, 1071
1305	D6TA	808160A081A4A4800, 795	1399	D6TA	80F2A4C3285A522E, 1808	1533	D6TA	8085A7C92A0B13A088, 1071
1306	D6TA	80F150A9581808087, 857	1400	D6TA	80F2A4C3285A522E, 1808	1534	D6TA	8085A7C92A0B13A088, 1071
1307	D6TA	81A78013A4898A7, 1121	1401	D6TA	80F2A4C3285A522E, 1808	1535	D6TA	8085A7C92A0B13A088, 1071
1308	D6TA	813A6478228180A, 780	1402	D6TA	80F2A4C3285A522E, 1808	1536	D6TA	8085A7C92A0B13A088, 1071
1309	D6TA	E8FF78081689A40A, 1181	1403	D6TA	80F2A4C3285A522E, 1808	1537	D6TA	8085A7C92A0B13A088, 1071
1310	D6TA	1A4A0800918086A8, 761	1404	D6TA	80F2A4C3285A522E, 1808	1538	D6TA	8085A7C92A0B13A088, 1071
1311	D6TA	E8680802E581C8C8, 1071	1405	D6TA	80F2A4C3285A522E, 1808	1539	D6TA	8085A7C92A0B13A088, 1071
1312	D6TA	C3C810D802A932828, 1094	1406	D6TA	80F2A4C3285A522E, 1808	1540	D6TA	8085A7C92A0B13A088, 1071
1313	D6TA	80F2A4C3285A522E, 1808	1407	D6TA	80F2A4C3285A522E, 1808	1541	D6TA	8085A7C92A0B13A088, 1071
1314	D6TA	A481280F40A4A2828, 774	1408	D6TA	80F2A4C3285A522E, 1808	1542	D6TA	8085A7C92A0B13A088, 1071
1315	D6TA	80F2A4C3285A522E, 1808	1409	D6TA	80F2A4C3285A522E, 1808	1543	D6TA	8085A7C92A0B13A088, 1071
1316	D6TA	Z81A9F9F90448A3A9, 808	1410	D6TA	80F2A4C3285A522E, 1808	1544	D6TA	8085A7C92A0B13A088, 1071
1317	D6TA	A780FF9081680A018, 1146	1411	D6TA	80F2A4C3285A522E, 1808	1545	D6TA	8085A7C92A0B13A088, 1071
1318	D6TA	A4A8380A0817A040, 1833	1412	D6TA	80F2A4C3285A522E, 1808	1546	D6TA	8085A7C92A0B13A088, 1071
1319	D6TA	3C48280085A28A67, 988	1413	D6TA	80F2A4C3285A522E, 1808	1547	D6TA	8085A7C92A0B13A088, 1071
1320	D6TA	60EAC80A02808087C, 1173	1414	D6TA	80F2A4C3285A522E, 1808	1548	D6TA	8085A7C92A0B13A088, 1071
1321	D6TA	A739180A4E809A08, 1273	1415	D6TA	80F2A4C3285A522E, 1808	1549	D6TA	8085A7C92A0B13A088, 1071
1322	D6TA	F5A2800210A4E0FF, 1186	1416	D6TA	80F2A4C3285A522E, 1808	1550	D6TA	8085A7C92A0B13A088, 1071
1323	D6TA	808160A081A4A4800, 795	1417	D6TA	80F2A4C3285A522E, 1808	1551	D6TA	8085A7C92A0B13A088, 1071
1324	D6TA	80F150A9581808087, 857	1418	D6TA	80F2A4C3285A522E, 1808	1552	D6TA	8085A7C92A0B13A088, 1071
1325	D6TA	81A78013A4898A7, 1121	1419	D6TA	80F2A4C3285A522E, 1808	1553	D6TA	8085A7C92A0B13A088, 1071
1326	D6TA	813A6478228180A, 780	1420	D6TA	80F2A4C3285A522E, 1808	1554	D6TA	8085A7C92A0B13A088, 1071
1327	D6TA	E8FF78081689A40A, 1181	1421	D6TA	80F2A4C3285A522E, 1808	1555	D6TA	8085A7C92A0B13A088, 1071
1328	D6TA	1A4A0800918086A8, 761	1422	D6TA	80F2A4C3285A522E, 1808	1556	D6TA	8085A7C92A0B13A088, 1071
1329	D6TA	E8680802E581C8C8, 1071	1423	D6TA	80F2A4C3285A522E, 1808	1557	D6TA	8085A7C92A0B13A088, 1071
1330	D6TA	C3C810D802A932828, 1094	1424	D6TA	80F2A4C3285A522E, 1808	1558	D6TA	8085A7C92A0B13A088, 1071
1331	D6TA	80F2A4C3285A522E, 1808	1425	D6TA	80F2A4C3285A522E, 1808	1559	D6TA	8085A7C92A0B13A088, 1071
1332	D6TA	A481280F40A4A2828, 774	1426	D6TA	80F2A4C3285A522E, 1808	1560	D6TA	8085A7C92A0B

```

1492 DATA 15A4850520000920,902
1493 DATA EG00A9237800F200,1260
1494 DATA 77C094A001F3277F,1267
1495 DATA 2000F26A0001F3C9,1343
1496 DATA 7F90EA6982000F2,1283
1497 DATA 2020A76020E3A7E0,994
1498 DATA FF0001600D1A0005,1050
1499 DATA 000015A4850510A9FF,916
1500 DATA 0506000004812004,682
1501 DATA 04072D7000F2A000,908
1502 DATA 96A0051B2000F26A,974
1503 DATA 00400100C9700002,983
1504 DATA 07202000F20A00C0,1123
1505 DATA C02000EA99902000,1192
1506 DATA F2A0001807000500,701
1507 DATA 9002E00107FF00FC,1194
1508 DATA 02A0FC02C9FF00F9,1174
1509 DATA C721F0062020A709,1067
1510 DATA FF00FC0250000000,746
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
- ROM Killer
- ROM Marker & -Entmarker
- COLOR-Maker
- HEX, BIN, DECZ -Wandlung
- Block-Zeilen-Delete
- Variablen-Lister...

\* **Tolle Hilfsprogramme wie**

- Zeichensatzeditor
- Farb-Zeicheneditor
- TT-Titelmachine

\* **nützliche MS-Programme wie**

- Zeichensatz LOAD/SAVE/COPY
- 120-Farben-Effekt
- Bilder-Laderoutine

und vieles mehr! nur net 29,-



Bitte bestellen Sie per Nachnahme oder Scheck bei:

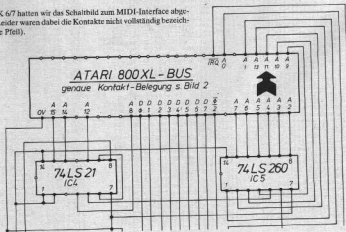
Thomas Tausend, Pflad 17

Tel.: (06376) / 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

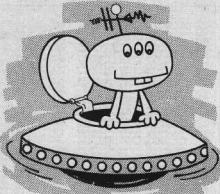
Achtung! Besitzer eines STs mit nur 512 K müssen vor dem Abtippen den Grafik-Puffer abschalten. Das Spiel läuft nur mit einem SW-Monitor.

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 hinwegrollen. Ihr Ziel ist es jetzt, so lange wie möglich den gegnerischen Angriffen, die pausenlos stattfinden, standzuhalten. Sie können Ihr Fadenkreuz über eine auf Sie zulaufenden Gegner steuern und durch Knopfdruck versuchen, ihn zu treffen. Doch passen Sie auf. Wenn ein Gegner so weit an Sie heran kommt, daß er abdreht, so schließt 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 Rübner



```

1 *****
2 *
3 * 3-D FLYING ACE
4 *
5 * (C) BY RADDISH-GOET GERMANY
6 *
7 * Written by Christian Rübner
8 * Tel. 02363 / 59880
9 *
10 *****
11 dis pos(0)
12 restore for i = 1 to 85
13 read po i:restat
14 restore 1000 for i = 1 to 84
15
16 restore c = 0 to 6:dirmax = 6
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

2688	data 000,000,000,014,976,070,200,000	3620	data 121,000,000,000,000,000,013,900
2689	data 000,001,100,012,185,000,000	3640	data 007,000,000,000,000,000,000,000
2629	data 011,040,000,000,013,100,100	2628	data 001,000,000,013,074,805,252
2630	data 000,000,000,000,000,000,000	2628	data 000,000,019,070,000,000,100
2640	data 000,000,017,170,013,100,000	3670	data 166,835,252,000,000,014,100
2650	data 000,000,017,024,183,000	3600	data 000,000,013,170,000,000,000
2660	data 000,030,000,252,000,000,000	3670	data 000,000,013,070,070,000,000
2670	data 000,000,000,014,000,000,000	3710	data 143,801,252,000,000,000,000
2680	data 000,000,000,000,000,000,000	3720	data 013,000,000,012,000,000,013
2690	data 072,072,252,000,000,000,014	3740	data 001,000,000,000,000,000,000
2700	data 000,000,014,070,070,000	3740	data 013,000,000,000,000,000,010
2710	data 000,000,170,000,252,000,000	3760	data 200,000,000,013,210,000,000
2720	data 020,192,000,000,014,000,000	3760	data 000,000,014,141,000,000,000
2730	data 252,000,000,000,021,00,000,000	3770	data 120,012,000,000,120,100,000
2740	data 014,072,000,000,014,976,003,121	3780	data 000,000,070,240,000,000,000
2750	data 004,000,000,014,976,003,121	3790	data 010,012,120,000,001,000,000
2760	data 000,000,013,222,012,121,000	3800	data 011,078,183,000,000,000,000
2770	data 000,000,000,013,222,102,000	3810	data 252,000,000,000,000,000,013,072
2780	data 000,000,000,000,000,000,000	3820	data 001,252,000,001,000,000,012
2790	data 000,011,222,000,123,000,000	3830	data 070,000,000,000,000,000,000
2800	data 014,004,002,180,000,000,010	3840	data 120,120,000,000,014,004,090
2810	data 200,000,000,013,190,100,000	3850	data 000,000,070,004,004,000,032
2820	data 000,000,000,000,000,000,000	3860	data 170,120,000,000,014,004,109
2830	data 000,014,004,012,121,000,000	3870	data 000,000,000,000,000,000,000
2840	data 000,000,000,013,170,100,000,000	3870	data 000,000,000,000,000,000,014
2850	data 190,012,100,000,000,019,220	3890	data 000,109,000,000,000,004,004
2860	data 000,000,000,000,000,000,000	3900	data 000,000,170,120,000,000,014
2870	data 000,014,121,000,000,014,004	3910	data 000,000,000,000,000,000,000
2880	data 012,121,000,000,000,000,014	3920	data 000,000,011,040,000,000,014
2890	data 004,109,000,000,146,030,250	3930	data 000,000,252,000,000,012,170
2900	data 000,000,000,000,000,000,000	3940	data 000,000,014,092,030,252,000
2910	data 000,030,252,000,000,000,100	3950	data 000,012,252,000,000,010,150,000
2920	data 000,000,000,000,000,000,000	3960	data 000,013,100,001,252,000,100
2930	data 000,019,152,000,000,004,000	3970	data 000,000,013,220,070,000,000
2940	data 000,000,000,000,000,000,000	3980	data 044,040,000,000,000,010,000
2950	data 170,040,000,000,017,070,070	3990	data 120,120,000,000,000,010,000
2960	data 000,000,000,000,000,000,000	4010	data 000,000,000,000,014,000,032
2970	data 000,004,004,000,019,150,000,000	4020	data 170,170,000,000,000,010,109
2980	data 000,000,000,000,000,000,010	4030	data 000,000,000,000,000,000,000
2990	data 300,000,000,013,100,070,240	4040	data 010,000,170,120,000,000,013
3000	data 000,000,000,000,000,000,000	4050	data 002,100,000,000,000,004,004
3010	data 000,014,000,000,012,000,010	4060	data 000,000,170,120,000,000,013
3020	data 000,000,000,000,000,000,000	4070	data 002,200,000,000,000,000,000
3030	data 000,170,000,000,000,000,010	4080	data 000,000,011,040,000,000,013
3040	data 000,000,014,100,000,000,000	4090	data 166,835,252,000,000,012,170
3050	data 000,000,000,000,000,000,000	4100	data 000,000,013,170,000,252,174
3060	data 092,030,252,000,000,019,132	4110	data 000,000,000,000,000,010,190,000
3070	data 000,000,004,070,000,000,000	4120	data 020,252,000,000,013,174
3080	data 017,010,019,252,000,000,000	4130	data 000,013,210,000,000,252,000
3090	data 000,000,000,000,014,000,010	4140	data 000,000,000,000,000,000,000
3100	data 192,000,000,014,000,000,252	4150	data 000,000,000,000,000,000,000
3110	data 000,000,000,000,000,000,000	4160	data 000,000,000,000,000,000,000
3120	data 000,000,000,000,000,000,000	4170	data 000,000,000,000,000,000,000
3130	data 100,000,121,000,000,000,000	4180	data 013,070,183,000,000,000,000
3140	data 013,074,182,000,000,150,003	4190	data 120,000,000,013,072,121
3150	data 120,000,000,013,070,102,121	4200	data 000,000,000,000,000,000,000
3160	data 000,000,140,001,252,000,000	4210	data 000,000,013,070,000,182
3170	data 000,000,013,070,000,000,000	4220	data 000,000,000,000,000,000,000
3180	data 000,000,013,070,000,000,000	4230	data 012,121,000,150,000,000,013
3190	data 000,013,210,000,210,000,000	4240	data 000,100,000,000,000,000,121
3200	data 000,000,000,000,000,000,013	4250	data 000,000,000,000,000,000,000
3210	data 170,003,210,000,000,013,174	4260	data 000,010,141,000,000,000,000
3220	data 000,200,000,000,000,013,210	4270	data 012,007,000,000,182,000,000
3230	data 100,000,000,011,040,000,000	4280	data 020,012,121,000,000,000,004
3240	data 013,210,182,000,000,000,001	4290	data 012,000,000,000,182,000,000
3250	data 200,000,000,000,000,013,074	4300	data 000,010,141,000,000,000,000
3260	data 012,121,000,000,000,000,000	4310	data 121,000,000,013,090,000,007
3270	data 220,180,000,000,000,000,252	4320	data 000,000,010,141,000,000,000
3280	data 166,835,250,000,000,000,004	4330	data 000,000,010,141,000,000,000
3290	data 000,000,013,170,013,252,000	4340	data 004,012,007,000,000,000,000
3300	data 000,000,110,000,000,013,174	4350	data 000,000,000,000,000,000,000
3310	data 000,000,110,000,000,013,000	4360	data 000,000,000,100,000,000,000
3320	data 070,000,000,000,000,000,000	4370	data 000,000,000,100,000,000,000
3330	data 252,000,000,000,192,000,000	4380	data 007,120,000,000,000,000,000
3340	data 013,166,000,252,000,000,000	4390	data 000,000,000,000,000,000,000
3350	data 010,000,000,013,170,000,000	4400	data 000,000,000,000,000,000,000
3360	data 000,000,000,000,000,000,000	4410	data 000,000,000,000,000,000,000
3370	data 174,003,170,000,000,013,170	4420	data 000,000,000,000,000,000,000
3380	data 012,121,000,000,000,000,013	4430	data 012,100,000,000,000,000,000
3390	data 070,182,000,000,000,000,000	4440	data 000,000,010,141,000,000,000
3400	data 000,000,000,000,000,000,000	4450	data 012,100,000,000,000,000,000
3410	data 120,000,000,013,000,013,100	4460	data 000,012,100,000,000,000,000
3420	data 000,000,000,000,000,000,000	4470	data 000,000,000,000,000,000,000
3430	data 210,180,000,000,000,013,140	4480	data 000,000,000,000,000,000,000
3440	data 120,000,000,000,000,013,074	4490	data 000,000,000,000,000,000,000
3450	data 192,000,000,200,012,100,000	4500	data 000,000,000,000,000,000,000
3460	data 000,019,000,000,000,000,000	4510	data 000,000,000,000,000,000,000
3470	data 002,000,000,000,000,120,000	4520	data 000,000,000,000,000,000,000
3480	data 000,000,000,000,000,000,000	4530	data 000,000,000,000,000,000,000
3490	data 000,000,000,000,000,000,000	4540	data 000,000,000,000,000,000,000
3500	data 000,000,000,000,000,000,000	4550	data 000,000,000,000,000,000,000
3510	data 174,003,000,000,010,000,000	4560	data 000,000,000,000,000,000,000
3520	data 000,000,000,013,166,000,252,000	4570	data 000,000,000,000,000,000,000
3530	data 000,000,000,000,000,000,000	4580	data 010,140,000,000,000,000,000
3540	data 030,252,000,000,019,170	4590	data 167,012,000,000,000,000,000
3550	data 000,000,000,000,000,000,000	4600	data 010,167,000,000,000,000,000
3560	data 000,000,000,013,074,000,000,000	4610	data 252,000,000,000,000,000,000
3570	data 017,070,070,000,167,019,252	4620	data 019,252,000,000,000,000,010
3580	data 000,019,000,000,000,000,000	4630	data 160,019,252,000,000,000,000
3590	data 192,000,000,000,000,000,000	4640	data 010,170,000,000,000,000,000
3600	data 000,000,240,000,000,000,110	4650	data 000,000,010,140,000,121,000
3610	data 210,070,240,000,000,000,110		
3620	data 000,121,000,000,000,000,000		



0700	data	037,000,004,251,027,251,050	9010	data	016,000,012,004,000,000,014	9230	data	006,004,012,000,010,004,010
0710	data	240,000,240,225,120,255,120	9020	data	010,010,000,010,010,012,006	9240	data	000,010,000,000,004,004,004
0800	data	225,000,000,000,224,000,224	9030	data	006,000,010,010,000,000,040	9260	data	000,010,000,004,000,000,000
0810	data	000,120,000,120,000,000,000	9040	data	000,000,000,000,000,000,000	9270	data	000,012,000,004,000,004,000
0820	data	000,000,000,000,000,000,000	9050	data	000,000,000,000,000,000,000	9280	data	004,010,000,000,000,000,000
0830	data	000,000,000,000,000,000,000	9060	data	000,000,010,010,004,000,000	9290	data	010,000,014,010,004,000,000
0840	data	000,000,000,000,000,000,000	9070	data	004,000,000,000,000,014,010	9300	data	004,000,000,000,000,010,000
0850	data	000,000,001,005,000,005,000	9080	data	000,017,000,000,000,000,000	9310	data	000,000,000,000,004,000,000
0860	data	014,000,014,000,000,000,000	9090	data	004,004,012,000,010,004,000	9320	data	004,000,012,000,010,000,010
0870	data	000,240,000,240,000,192,000	9100	data	004,004,004,000,000,000,000	9330	data	000,010,000,000,010,004,010
0900	data	192,000,120,000,120,000,000	9110	data	004,000,004,000,000,004,000	9340	data	020,010,000,000,010,004,010
0910	data	000,000,000,000,000,000,000	9120	data	004,000,004,000,000,010,000	9350	data	000,000,010,000,010,000,010
0920	data	000,000,000,000,000,000,000	9130	data	000,010,004,000,000,000,000	9360	data	000,000,012,000,000,017,000
0930	data	000,000,000,000,000,000,000	9140	data	000,010,000,000,000,000,000	9370	data	000,000,000,000,014,010,000
0940	data	000,000,000,000,000,000,000	9150	data	000,010,000,000,000,000,000	9380	data	000,004,000,000,000,004,000
0950	data	000,000,000,000,000,000,000	9160	data	000,012,000,004,000,000,000	9390	data	004,004,000,000,000,004,000
0960	data	000,000,000,000,000,000,000	9170	data	000,010,000,000,000,000,000	9400	data	004,004,004,004,004,004,004
0970	data	000,014,010,000,000,000,000	9180	data	000,012,000,000,000,000,000	9410	data	004,004,004,004,004,004,004
0980	data	004,000,012,010,000,004,010	9190	data	000,014,012,000,010,000,000	9420	data	004,004,004,004,004,001,001
0990	data	010,000,000,010,010,010,000	9200	data	004,000,004,000,000,004,000	9430	data	004,002,000,000,000,030,120
9000	data	000,000,000,000,000,000,010	9210	data	000,000,004,000,000,012,000	9440	data	004,004,000,000,000,000,000

## Textomat ST - der Nachfolger des bekannten C 64 Programms

Der Erfolg begann auf dem C64: 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 Programmiererteam.

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, wundern 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 in 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 Fiskeltasten 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 einblendend. 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 (Textanfänger, rechtes Wort, nächster Absatz usw.) möglich.

Zur lästigen Wechselsei 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 "Mausestern" aufgehält wird. Für Anwender, die andere Dinge

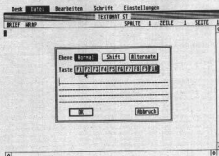


Bild 1: Funktionstasten belegen

im Kopf haben als Abkürzungen, ein Tip: Klappen Sie nach-einander 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 "B" 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 auswählen. 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 begründenswert. Enttäuschend ist es indes-nen, daß selbst die einfachsten Effekte wie Fettdruck oder Unterstreichen nicht im Editiermodus sichtbar sind. Das Programm "1st 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 C-64 noch verständlich, kann er am Atari ST nur

Herschalten zwischen den Modi erspart werden können.

Neben den üblichen Block-funktionen 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 eifrig Buchautoren sind, haben sie Funktionen zum Erstellen eines Stichwort- und Inhaltsverzeichnisses eingebaut. Die zu

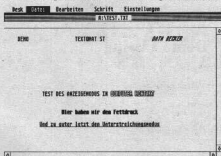


Bild 2: Einige Schriftarten

kennezeichnenden 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

der angegebenen Reihenfolge ausgedrückt, 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 "1st Word"-Texte in das Textomat-Format umwandelt. Das andere ist wohl nur für Atariäner 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 1st Word) um Längen. Einziger Mangel: der ärgerliche Anzeigemodus. Sonst stimmt alles: optimale Ausnutzung der Hardware, vorbildliches Handbuch, Bedienungskomfort, der seinesgleichen sucht, und last,

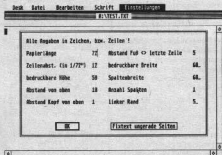


Bild 3: Einstellen des Druckformates

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

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, gemächliches Arbeitstempo und dürtigen Befehlsatz 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 Programmiersprache 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 eingegeben 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 RECHTL. Was dem gelübten Basic-Programmierer wohl als erstes auffällt, sind die fehlenden Zeilennummern. Tatsächlich werden im GfA-Basic keine Zeilennummern verwendet. Wer jetzt im Editor das ge-

wöhnte 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 Programmierung zuerst etwas näher an. Das GfA-Basic enthält einen sehr komfortablen Editor, der sehr schnell arbei-

## Demo-Listing

Die 99(20)

Einlesen der Menuestruktur

For N=0 To 30

Read M(N)

Exit If M(N)="\*\*"

Next N

M(N)\*\*

Date für das Menu. Jede Box wird mit einem Leerstring abgeschlossen

Data DESK, About Demo,-----,1,2,3,4,5,6,\*\*

Data EINTRÄGER, Text 1, Text 2, Text 3, Text 4, ENDE,\*\*,\*\*\*

Herstellen der Menueleiste in M

Menu M()

Prozedurname, die bei Menueereignis angesprochen werden soll

In Menu Gosub Menu

Eigentliches Programm (hier Endlosschleife)

Do

In Menu

Loop

Auswertungsprozedur für Menu

Procedur Menu

Menu Gf

AB="Sie haben das Punkt! !)" \*M(Menu())\*\* <<|qualit

Alert 0,AB,1,"aha",1

If M(Menu())=" ENDE"

End

Else

If M(Menu())=" About Demo"

Alert 1,"Dies ist nur ein:Demo für das GfA-BASIC!Möchten Sie weitermachen?",1,"Oh ja! Mein ",1

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 ausgetriggert 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 vermüht.

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 als die gewohnten 3 Bomben, sondern eine Alarmbox, 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\$: 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 compiliertem C auf (mit Parameterübergabe). CHDIR: Wechselt den aktuellen Ordner. DEFFILL: Legt Füllmuster fest bzw. definiert eigene Muster.

```
Save Save:R Quit New Disk Size/Replace Pg up Text Left/Right Font
Load Merge List Block Disk Edit Print By Description Filter View
Alert 1, "Rechtecke", 1, "Return", 8
Berechne 1
Do
  Box 1 Mod 640,1 Mod 400,629-1 Mod 640,299-1 Mod 400
  Add 1,0
Loop
  * Dieses Demo ist als "RECHT1.BAS" auf der Programmkassette enthalten!
```

#### RECHT1.BAS geladen und gelistet



#### Ergebnis von RECHT1.BAS

```
Save Save:R Quit New Disk Size/Replace Pg up Text Left/Right Font
Load Merge List Block Disk Edit Print By Description Filter View
Print "Benchmark 8 :)"
T-Timer
K0
Kagnet
K0+1
K0+2
B=log(0.3)
C%=(0.3)
Setfil K0=1000
Print (Timer-T)/200
* Der ST benötigt für diesen Benchmark nur 1.82s
* Das "normale" BASIC von MS dagegen 3.5s
* unser Konkurrenz liefert der 1386 mit 7.6s
```

#### Benchmark-Test

```
Save Save:R Quit New Disk Size/Replace Pg up Text Left/Right Font
Load Merge List Block Disk Edit Print By Description Filter View
T-Timer
For I=0 To 1000
  Print I
Next I
Print (Timer-T)/200;" s",8
  * Für diese Schleife benötigt der ST 34.88s
  * Der 1386 benötigt dagegen nur 7.1s!
  * allerdings auf der 1386 nur 960 Bytes scrollen, der ST hingegen
  * nutzt 32000 Bytes!
```

#### Noch ein Benchmark-Test

```
Save Save:R Quit New Disk Size/Replace Pg up Text Left/Right Font
Load Merge List Block Disk Edit Print By Description Filter View
Print "Benchmark 2001 Divisionen 1387"
Print "Zeit", "Ergebnis"
T-Timer
For I=0 To 2000
  K=I/2
Next I
Print (Timer-T)/200;" s",8
  * Diese 2001 Divisionen absolviert der ST in durchschnittlich 0.8s
  * Der 1386 benötigt dagegen 2.6s!
```

Und noch ein Benchmark-Test



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 Computertechniker ist die 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 macht 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 unsere Programmdisketten zum ST, die es vielleicht bald geben wird, auch den "Run Only Interpreter" mit anliefern, 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 Hofchen 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 Einzahlung 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

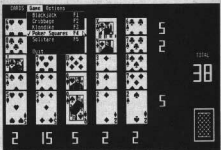
All jene, die ihrer Frau versprochen haben, daß ein Atari ST den regelmäßigen Stammtischbesuch ersetzen kann, brauchen jetzt nicht mehr auf das Kartenspielen zu verzichten. MichTron hat ein weiteres microdeal-Programm auf den Markt gebracht, das 5 (in Großbritannien) bekannte Kartenspiele in sich vereint: 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. Spiele 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 anscheinlich 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 Padorborn  
Telefon 0 52 51 3 63 96

**Flippstationen für ATARI ST®**  
**PADERCOMP FL 1** 498,-  
3,5" 1 MB, eingeb. Netzteil, NEC-Laufwerk, Abm. 240x125x40 mm, anschlußfertig mit Inklusiv-Flippstecker, großes Handgelenk, ohne Schrauben an das Gehäuse. Ein Qualitätsprodukt, das schon Hunderte von zufriedenen Kunden gefunden hat.

**PADERCOMP FL 2** 798,-  
Lineare News, Doppellaufwerk (Gehäuse), sonst wie FL 1

**NEC FD 1036 A** 3,5", 1 MB,  
32 mm Dichtung, Adressprez. 289,-  
dfo, ST-modifiziert 289,-

Inklusive Flippstecker  
jezt lieferbar nur 0,90

ST-Kabel am Shugart-Bus 3,5" 29,90

### Drucker

**Panasonic KX-P 1080**

120 ZN, NLG 648,-

**Panasonic KX-P 1091**

120 ZN, NLG 788,-

**Panasonic KX-P 1092**

180 ZN, NLG 1040,-

**OKIDATA ML 192** mit

willkür. Einseitenbetriebs 1499,-

**Glitzer 120D** der Einzeiger 595,-

**STAR NL 10** mit Interface 788,-

### Zubehör

**Eprommer für ST 2764-**

720k, Software für 90M 195,-

**3,5" Disketten** ab 3,40

**Disk-Box SS-90**

für 50 3,5" Disketten 24,90

**Druckerkabel ST**

34,90

**Preisliste kostenlos**

Alle Angebote solange Vorrat reicht. Bei

Stellung der Nachnahme oder Vorkasse

zu 30 DM. Auslieferungstermin nur gegen

Vorkasse. Eingetragene Warenzeichen:

ATARI, ST, die Flippstationen sind ge-

reg. Marken. Bitte bei Händlerbestellung

angeben.



# TI99/4A

## 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. Schwandt

## 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.

TI99er 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 waren 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 erstaufliehendem Inhalt: ein komplettes CPS-99 System, die total auseinandergenommene Hauptplatine des TI sowie alle Netzteile, der Sprach-Synthetizer und ein Modulpexpander.



## TMS 9900 Assembler auf dem TI 99/4A

Beim heutigen Bachttest 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 Ansammlung 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 laages 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 Funktionsklärung 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 Programmierlösungen 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 Nachschlagewerk 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. Schwaback

Das Buch kann über den Verlag bezogen werden. Bestellschein siehe Buchverand.

## Der Zwillingbruder 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 Brunsman, 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 32 K-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 Bildschirngrenze wurde verwirklicht, so daß man insgesamt 80 Zeichen pro Zeile unterbringen kann. Texte können wahrweise 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 FCIN-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, nummeriert 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. Brunsman  
Gigastraße 3  
4400 Münster H.-P. Schwaback

**Computer-Kontakt  
preisgünstige  
Kleinanzeigen**



# TI Special

## Tuning: 128K RAM plus Centronics-Interface

Stabiles Metallgehäuse, Versorgung über Steckernetzteil, 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-Steckernetzteil, 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, Cassettensrecorder, 32K RAM Erweiterung

**Best.-Nr. TI 5 99,- DM**

## Nützlich: TMS 9900 Assembler

DAS Nachschlagewerk für TI-Assembler-Programmierer. Der Prozessor, seine Befehle, Beispiel Listings, Speicheraufbau, Tabellen, Index...

Anforderungen: Editor/Assembler, Extended Basic

**Best.-Nr. TI 3 49,- DM**

## CK SPEZIALITÄTEN-BESTELLSCHEIN

Anzahl	Best.-Nr.	Artikel	Preis incl. MwSt.
TI 1	128K RAM plus Centronics Interface		
TI 2	TI-Maus		
TI 3	TMS 9900 Assembler		
TI 4	TI-Assembler Kurs II		
TI 5	TURBO-Assembler		
TI 6	TI-Assembler Kurs III		

Name des Bestellers

Ich wünsche folgende Bestellung

Anschrift

Nachnahme (Zuz. 5,70 DM Versandkosten)

K.L.Z./Ö

Vorauskassa (keine Versandkosten)

Datum/Unterschrift

Bei Vorauskassa bitte Scheck beilegen oder auf Postcheckkonto Kontonr. 43423 756 überweisen

Coupon ausschneiden, auf Postkarte kleben und einsenden an:  
Verlag Rätz-Eberle/CK-Software, Postfach 1640, 7516 Bredien

## 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 Unterprogrammen, 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
AAAAAAAAAAAA
```

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 nicht 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 Ratering

### 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$,ZEILE,ERR
DR)-----
130 !ZEILE IN DIE "GEPOKED" !WERDEN SOLL MUSS !FOLGENDERMASSEN AUS
!SEHEN :
140 DEF F(X)=X :: REM AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
150 !FUNKTIONSNAMEN FREI !WAHLBAR(NUR EIN BUCH- !STABE LANG!)
32000 SUB DEF(F$,Z,F)
32010 F=0 :: G$="" :: L=LEN(F$):: Z=1 :: IF L=0 THEN SUBEXIT
32020 IF Z>L THEN 32190 :: 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$="CO" THEN H$=CHR$(205)ELSE IF M$
="SI" THEN H$=CHR$(210)ELSE IF M$="TA" THEN H$=CHR$(212)ELSE IF M$="SO" 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$(221):: Z=Z+2 :: GOTO 32020
32150 D$="" :: XX$=CHR$(200):: S1=0
32160 IF Z>L THEN 32180 :: M$=SEG$(F$,Z,1):: A=ASC(M$)
32170 IF (A)=4B AND A<(57)OR A=4E THEN D$=D$&CHR$(A):: Z=Z+1 :: S1=S1+1 :: GOTO
32160 ELSE 32180
32180 L$=CHR$(S1):: G$=G$&X$&L$&D$ :: IF Z>L THEN 32190 ELSE 32030
32190 CALL PEEK(-31952,J,I,K,P):: J=JK256+I-65536 :: K=K+256+P-65536
32200 CALL PEEK(J,I,O):: IF I+K256+O<ZE THEN J=J+4 :: IF J>K THEN F=-2 :: SUBEXIT
T ELSE 32200
32210 J=J+2 :: CALL PEEK(J,I,O):: J=I+K256+O-65536-1
32220 D$=G$&CHR$(130)&CHR$(154)&" "
32230 Y=LEN(D$):: IF Y>13B THEN F=-1 :: SUBEXIT ELSE Z=139-(Y+1):: D$=D$&RPT$( "
",Z)&CHR$(O)
32240 FOR X=1 TO LEN(D$):: 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 Erklärungssteil 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 übrigbehaltenen 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$(119),A(30),L$(8,2),LAST(B),LASTPOS(8,2),LASTCHR(B),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,"1B1B1BFFFF1B1B1B")
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$):: CALL CHAR(37,B$)
220 CALL CHAR(91,"0044003B447C4444")
230 CALL CHAR(92,"0044007C44444447C")
240 CALL CHAR(93,"0044004444444443B")
250 C$="FFB1B1B1B1B1B1FF"
260 FOR I=96 TO 128 STEP 8 :: CALL CHAR(I,C$):: 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,      M/GLICHST VIELE W/RTER
   IN   DAS 15X15 SPIELBRETT ZU   LEGEN."
410 DISPLAY AT(6,1):"MAN DARF NUR AN BESTEHENDE W/RTER ANLEGEN, BEGINNEND MIT
   DEM STERN."
420 DISPLAY AT(10,1):"DIE W/RTER M/SSEN IN ALLEN RICHTUNGEN GELESEN EINEN   SI
   NN ERGEBEN!"
430 DISPLAY AT(14,1):"DIE ROTEN FELDER ( 2W, 3W ) BEDEUTEN ERH/HTEN WORTWERT, DI
   E BLAUEN ( 2B, 3B ) ERH/HTEN BUCHSTABENWERT."
440 DISPLAY AT(19,1):"ES DARF NUR EIN WORT GELEGT WERDEN, ABER ES K/NNEN   ME
   HRERE ENTSTEHEN, DIE ENT- SPRECHEND GEWERTET WERDEN."
450 CALL TASTER
460 DISPLAY AT(1,1)ERASE ALL:"ES STEHEN 119 BUCHSTABEN ZURVERTEILUNG, VON DENEN
   JEDER SPIELER (MAX.4) SO LANGE 8"
470 DISPLAY AT(4,1):"ST/CK ERH/LT, BIS KEINE MEHRVORHANDEN SIND."
480 DISPLAY AT(7,1):"KANN EIN SPIELER NICHT ZIE- HEN, DARF ER ALLE SEINE   BUC
   HSTABEN WECHSELN UND   SETZT AUS."
490 DISPLAY AT(12,1):"SIEGER DES SPIELES IST NA- TJRLICH DER, DER DIE MEISTENPU
   NKTE GESAMMELT HAT.": : : "VIEL SPASS..."
500 CALL TASTER
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$": ## ST # P   "%C&": ## ST # P":S,W1,Z,W2
560 NEXT I
570 DISPLAY AT(21,1):"DER JOKER (BLANK) IST PLATZ-HALTER F/JR JEDEN BUCHSTABEN!"
580 CALL TASTER
590 CALL SCREEN(15):: DISPLAY AT(3,1)ERASE ALL:"WIEVIELE SPIELER WOLLEN MIT-SPIE
   LEN?":
600 FOR I=1 TO 4 :: DISPLAY AT(5+2*I,3):I:"SPIELER :: NEXT I
610 RO=97 :: CALL LOCATE(1,97,17)
620 CALL COLOR(1,2)
630 CALL JOYST(1,X,Y)
640 IF Y=4 THEN RO=MAX(49,RO-16)
650 IF Y=-4 THEN RO=MIN(97,RO+16)
660 CALL LOCATE(1,RO,17)
670 CALL COLOR(1,1)
680 CALL KEY(1,K,S)
690 IF K<>18 THEN 620
700 PL=(RO-1)/16-2
710 CALL CLEAR :: CALL SCREEN(4)
720 B$=RPTS(" ",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 1,4,1,12,3,7,3,9,4,1,4,8,4,15,7,3,7,7,9,7,13,8,4,8,12,9,3,9,7,9,9,9,1
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):"+
860 DISPLAY AT(1,18):CHR$(104):"=3*WORT"
870 DISPLAY AT(2,18):CHR$(112):"=2*WORT"
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 FL
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=B-112*(P=2 OR P=4):: SC,SH=SL+64
1130 RO=129-32*(P>2):: CALL LOCATE(#1,RO,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,RO,SH)
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*B+1 :: MY,CY=2*B+1 :: CALL LOCATE(#5,CX,CY)
1250 MX=HX-14*B :: HY=MY+14*B
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(BTR*(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$(I),N,B#)
1660 CALL HCHAR(WX,WY+I+2,ASC(B#))
1670 CALL HCHAR(WX+1,WY+I+2,A(ASC(B#)-64)+48)
1680 CALL SOUND(50,392,7)
1690 NEXT I
1700 P=P+1 :: IF P>PL THEN P=1
1710 GOTO 1090
1720 IF ANZ=0 THEN 1090
1730 CALL HCHAR(LASTPOS(ANZ,1),LASTPOS(ANZ,2),LASTCHR(ANZ))
1740 CALL HCHAR(LS(ANZ,1),LS(ANZ,2),LAST(ANZ))
1750 CALL HCHAR(LS(ANZ,1)+1,LS(ANZ,2),A(LAST(ANZ)-64)+48)
1760 ANZ=ANZ-1
1770 GOTO 1090
1780 CALL DELSPRITE(ALL)
1790 FOR I=1 TO PL :: FOR J=1 TO 8
1800 R=18-4*(I>2):: S=2+J-14*(INT(I/2)=I/2):: CALL GCHAR(R,S,C):: IF C=32 THEN I=820
1810 POI(I)=POI(I)-C+48 :: CALL SOUND(100,124,5,-8,9):: CALL VCHAR(R-1,S,32,2)
1820 NEXT J :: NEXT I
1830 CALL CLEAR
1840 FOR I=1 TO PL
1850 DISPLAY USING "SPIELER NR. #: ### PUNKT"&CHR$(-69*(POI(I)<>1)):I,POI(I)
1860 DISPLAY
1870 NEXT I
1880 DISPLAY RPT$("*",28)
1890 DISPLAY
1900 DISPLAY "BUCHSTABENMIX..."
1910 DISPLAY :: DISPLAY
1920 END
1930 SUB SETFIELD(A$(I),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,S,^,2,0
1980 DATA EDD,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 Ssprite(#3,99,14,7*B+4*B+1,18*B+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+B*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 WORTH=1
2670 FOR I=1 TO L
2680 IF LASTCHR(I)=104 THEN WORTH=WORTH*3
2690 IF LASTCHR(I)=112 THEN WORTH=WORTH*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

```

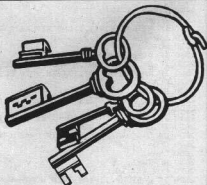
## Schlüssel

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 "KODDE (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=CB :: 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,UP,IM,CK,NE,ME,SO,US,IS,ZU,MO,NO,AN,AM
270 DATA SCH,UNG,ING,UND,SIE,WIR,IHR,VER,IEH,VOR,DEM,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,SA,S1,E1,61,6A,EB,5B,52,E2,EC,6C,63,6D,FD,7D,73,74,F4,E4,ED,ES,5E,6E
,65,7E
290 DATA 61,71,23,EA,FA,3A,4A,DB,5C,DC,6C,64,DD,3D,4D,75,3E,CE,C5
295 DATA 7A,53,DE,D1,41,6B,F1,6D,63,62,7B,72,C1,5D,54,FB,FE,FS,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$)
575 S1=1 : : FOR C=1 TO B STEP 2
580 C#=BEG$(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):"1M 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(NSP):: CALL SPRITE(NSP,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):"
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,"00C0E0F0F0CFE0FF",64,"0103070F1F3F7FFF",35,"FFFFFFFFFFFFFFF"
,35,"FF7F3F1F0F070301",37,"FFF0CF8F0E0C000")
2010 CALL CHAR(30,"1F3F7FFF0CF0F0F0F0CFE0FF3F1F0F0FF0F0F0CF0FF7F3F1F0F0F1F3FFF0FF
CF8")
2020 CALL CHAR(42,"E73F3F3F3F3C3C3C",43,"F0F0CFE0FE7E3E1E",135,RPT*( "3C",8),45,R
PT*( "1E",7)&"1F")
2030 CALL CHAR(46,"FFFFFFFFCF0F0F0",47,"F0F0F0CF0F0F0F0")
2040 CALL CHAR(50,RPT*( "F0",8)&"0301"&RPT*( "0",12)&"FCFE"&RPT*( "F",24)&"7F3F")
2050 CALL CHAR(62,"00000000000000C",63,"0000FFFFFFFFFFFFFFF",129,"F0F0F0F0F0F0CFE0
7F3F",130,"3F1F0F0703030303")
2060 CALL CHAR(131,RPT*( "03",8)&RPT*( "0F",4)&"1F3F7FFE0F0E0"&RPT*( "C0",10)&"F
0F0")
2070 DISPLAY AT(4,1)ERASE ALL: " #19# " ; " #5X# &' *+ &' , ' &'
# ";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": " 0750 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,"FF000000000000FF",143,"01010101010101FF",35,"FF010101010101FF
",36,"FF01010101010101",44,"000018243C2424")
2350 CALL CHAR(37,"01010101010101FF",38,"FF01010101010101",39,"FF000000000000",
40,"00000000000000FF",46,"00001824242418")

```



```

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)):: FARB
E$(P)=" " :: GOTO 2820
2770 NEXT P
2780 ACCEPT AT(9,3)SIZE(-2)VALIDATE(DIGIT)BEEP:F" :: IF F=2 THEN IF FARBE$(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$(4)=" " :: GOT
D 2820 ELSE IF F=13 THEN IF FARBE$(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
D 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:" : "SPIELLENDE MIT >E<" :: FOR ZEIT=1 TO 750
:: NEXT ZEIT
2860 SUBEND

```

## Katalog für Maschinen- spracheprogramme

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-Untersprogramm 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 Titelbild zurückspringen: CALL LINK ("RESET"). Aber Vorsicht: Selbstverständlich werden dabei alle Programme und Daten gelöscht!

Martin Kottala

### Assembler-Listing

```

*****
* *****
* *
* * CATALOG & LABEL *
* *
* *****
*
*
* IC: Berlin Kottala 1982
*
*
* >CALL LINK("CAT") zeigt Listing für Maschinenprogramme
* >CALL LINK("LABEL","NAME",Adresse) verändert die Namenstabelle
* *****

```

ADDR 1400	DEF CAT:LABEL	
001	ORIG:001	
SCHALL 000 30010		Prozessor SCHALL-Programme
NEST 000 30010		- Sprung in den OS-Interpreter
CTI 000 31000		- Connect-Flagging-Kontroll-Software
LS1010 000 31004		- Start der Namenstabelle
WMBEF 000 31006		- Numerische Parameter übernehmen
SPREF 000 31014		- Stringparameter übernehmen
WLINK 000 31018		- ROM-Wort nach aufrufen
OSCAR 000 31020		- Textinput abfragen
WIMP 000 31024		- VSP-OSK anderer Bytes schreiben
PAC 000 31026		- Programmieren
STATUS 000 31030		- OSK-Status-Byte
OSCAR 000 31030		- OSK-Methoden
2300 0110 100		Null-Byte
TE 0110 100		Textausgabe
CSIMP 0110 10000		ROM-Wort
SPREF 0110 10020		Texteinführung bei OSK
SET 0110 10030		Zwei Geraden
WIMP 000 32		Ligne Wimp-Programme
WMBEF 0110 140,144,147,150		Texteinführung bei OSK
0110 140,150,151,152,155		WMBEF-Texten über Screen-Offset
0110 140,143,147,149,154		
WMBEF 0110 12000,12020,12050		Buffer für alle geladenen Programme
0110 12001,12020,12050		





## Sinclair

**QL-ART - das Mail und Zeichenprogramm** - Das neue Grafikprogramm für den QL komplett mit Font-Editor, Zoom, 2. Bild im Speicher, Hardcopy, Fill, Shapes, komfortable Menüführung, Kopieren von Bildern. Sehr schnell! Handbuch und Cartridge für nur 98,- DM. Auskunft ☎ 040/7 91 92 18 od. 040/7 91 74 30

**UFC-Angriff für ZX81 (Dtp)** nur 10,- ☎ 054 85/19 80

**ZX81 18K (Zusatzkart)** inklusive Software u. Lit. für 100,- bei R. Brandstätter, Frauenstr. 63, 6880 Dillingen

Wer hat eine Lösung zur Befreiung der Programm-Speicherlinge- und Programm-Lade-Schwierigkeiten? das ZX81FF, J. Becker, Beulweg 5A, 7370 B-Baden, ☎ 072 21/332 01

• Vokabeln lernen besonders leicht!

- Vokabeltrainer "Easy Learn"
- 100 % Maschinen-Code
- einmaliges Kartelastprinzip
- Speicher: 33 Tausend Zeichen
- Lesikon-Modus
- max. Suchzeit für eine Vokabel: 0,14 sek.
- Tape oder Micro-/Cassette
- Modifikation auf Beta-Disk möglich
- Preis: 19,80 DM + Porto

Mit mehrstelliger Anleitung bei: M. Roland, Postfach 5026, 4902 Bad Salzuflen, ☎ 18-211 052 22 / 05 05

## SPECTRUM-PROGRAMMSAMMLUNG (1)

5 Anwendungen, 5 Spielprogramme  
**TOOLBYTE** 100% MC, 8 KBytes; 26-Außerwöhrnt, starke u. effektive Ein-Tastenbediene. **LINE-DELETE** 50 Bytes MC; Löscht Zeilen einzeln o. blockw.  
**COMPACTOR** 122 Bytes MC; Verkürzt alle Basic-Programme, ca. 20%. **HELP** 200 Bytes MC; Verlangt laufende Programm-Anzeige d. Hst. Zeilen-Nr. **DRAW** Grafik 35 KBytes MC; 18 Befehle, 2 + Circle, Box, über 1000 versch. Schrifttypen, 3 Gummibilder, echtf-Fill etc. **TECHNO** 3 KBytes; Erstellt techn. Zeichn. in 3 Ansichten - Perspektiven mit Demo. **UBOOT-SIMULATION** 40 KBytes MC; 37 Befehle u. Kontr. + Kontrollraum, Kartenraum u. Penetik. **COSMAZICS** 5 KBytes MC; Sehr schn. Arcade. **SNAKE-LIVE** 6 KBytes MC; 48 Ebenen. **QUAZIMODO** 9,5 KBytes MC; Aufwend. Grafik, 4 Screens. **BLOCKMAN** 5 KBytes Spaß und Spannung. Alle 10 Programme in kommerziell quality aus Your Comp. 84/85 mit engl. deutscher Dokumentation **DM 25,-** an Josil Bauer, Jakobstr. 11, 8850 Hornburg. Jede weitere Kopie der Zeichn. Dokumentation DM 4,-.

**Verkaufe ZX-Spectrum 48K mit Plus-Teilset**, inkl. Demo + 10 Original-Cassetten, VB 300,- DM. ☎ 021 64/76 10

S.A.V. is coming...  
 Cassetts 25 - DM, Cartridge 30 - DM, inkl. Port. By Software, Stefan Swierger, Südimg 11a, 3440 Eschwege

## ◆◆◆◆ Endlich ist sie da! ◆◆◆◆

Die 64-Zeichen-Karte für dein Spectrum. Dieser Hardwearsatz bringt 64x32 Zeichen auf den Screen. Kompatibel zu nahezu allen Anwendungsprogrammen (Pascal, Dexpac etc.) Gratis-Info bei Ingo Kauf, Jan-Wellem-Str. 35, 5630 Remscheid

◆◆◆◆ ZX-SPECTRUM ◆◆◆◆  
 Suche Computerprogramme für Sinclair ZX Spectrum 48k. Tauschangebot an Gerald Amann, Oberweg 8, 7957 Bad Waldsee, ☎ 075 24/13 01

Kaufe alte und beschädigte Spectrums 16 + 48k, DATATEX, Lorbenweg 5, 4230 Waseil 1, ☎ 02 21/42 40 48  
 Einmalige Gelegenheit für Sinclair-Freaks: 1 ZX-Interface 1 + 1 ZX-Microdrive, fibr. neu, orig. verpackt, mit Garantie zus. 200,- DM, Schröder, Kolpingstr. 28, 4837 Lage, ☎ 052 32/805 54

**USER-CONTACT-SERVICE-SCIUC**  
 Kennen Sie ihn noch nicht? Info gegen 2-50 Pf., anfordern! Spectrum Computer Info, 7620 Heidenheim, Carl-Golzweg 17, 7620 Heidenheim

**ZX Spectrum 80KB, ISO-ROM, + Tast., Recorder, GP 90 S, Digital-Tracker, viel Software und Literatur.** Preis: VB. ☎ 07331/601 65

Suche defekten Sinclair QL oder Spectrum. ☎ 05 85 87 70 07 ab 18:00 Uhr

Zeichentisch mit dem Spectrum: 300 Byte machen aus Screens den Film und 150 Byte zeichnen ihn. Preis: 10,- DM. Peter Christmann, Gutenbergstr. 16, 7340 Geislingen

Spectrum Speicherwartung 48K 40,- DM, ZX81 + Aufsatzkassette + 18K ROM eingebaut + div. Software + Monitorauflage 100,- DM, G Pfeiffer, ☎ 0941/952 90

Verkaufe PROCEED 1 von Logtek, -Floppy- und Druckermodule 150,- DM, LIGHTPEN + dev. Anleitung 35,- DM, KEMPSTON-Joystick-Interface 35,- DM, Cassetten-Video-Interface an User-Port (aus c't Heft 2/1985), Josef Linder, Wellerswälder, 184, 6850 Neunkirchen, ☎ 06 21/12 90 0

## ◆◆◆ SINCLAIR-FREAKS ◆◆◆

Ein User-Club informiert: Seit 2 Jahren besteht der Spectrum-User-Club Wuppertal. Unsere monatlich erscheinende Zeitschrift enthält neben vielen 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 Club-Anschrift Rolf Knoke, Postfach 201 02, 5600 Wuppertal 2. • **Angebote von Händlern, die Club-Rabatte einräumen, sind erwünscht!**

Verkaufe orig. Spectrum-Software. 78 Top Progr. je 5,- DM. Außerdem andere gute Progr. ab 4,- DM. RAMBO, COMMANDO je 12,- DM usw. / Suche strategische Wargames, zähle ggf. Liste an oder anfordere für - 50 DM Rückporto bei W. Schwanz, Chemnitzstr. 12, 2120 Lüneburg, ☎ 041 31/5 97 53. Suche Reseknapp!

**Verkaufe Kempston Pro Joyst-Interface** (3 Anschl.) • **Competition-Sport 3000**, unben. zus. 70,- DM, Sporber, 6500 Bamberg, Sanjo-Genru-Str. 9a, ☎ 0951/580 28

**Spectrum 80K in Mikrotron Gehäuse mit Interf. 1 + 1 d'c., + 24 Cartridges + ZX Printer + Kempot. / programmierb. Joystick-IF + Datenrec. + Speicher ZX Mach. Code + ROM Buch + 200 Buch + mehr als 80 Kassetten.** Preis VB. ☎ 05222/7 05 05 ab 18 Uhr

◆◆ **Achtung Spectrum-Freaks** ◆◆  
 ZX Spectrum 64 K, CUB-Parformul, 452 (TL- und Spec.-Ergebn, Spec.-Interface), Interface 1, Sprint-Digi-Recorder, ZX-Print III mit Parallel- und Serialkabel, Orig.-Software usw.: Beta-Basis, Masterfile mit MF-Print, Flight, Statistik, günstigst (auch einzeln) abzugeben! P. Enders, ☎ 091 31/85-9972, 18115 oder 9927 11

**Spectrum-Software.** Verkäufe Original-Text-, Grafik-, Drucker-, Speechboard-Utility-Prog. (je 1 x). Liste anfordern! G. Schmidt, Exenstr. 43, 6230 Frankfurt 90

ZX Spectrum: Verkaufs PROCEED 1 - Floppy- und Druckermodule = 150,- LIGHTPEN mit dev. Anleitung = 35,- KEMPSTON-Joystick-Interface = 35,- CASSETTEN-Video-Interface am User-Port (aus c't Heft 2/1986) = 85,- J. Linder, Wellerswälder, 184, 6850 Neunkirchen, ☎ 05 85 87 12 90 0

**Spectrum defekt?????**  
 Kaufe Defekte Spectrol 40,- DM, Plus 50 inkl. altem Zubehör! ☎ 07321/63079, Offere bitte!

◆◆ Einkommen-Lohnsteuer 1986 ◆◆  
 Schon jetzt die Steuer 1986 berechnen und beurteilen, 5% Testauswertung möglich, Update-Service 1987 ff. Spectrum: Cassette 50,- DM, Interface 55,- DM, Info - 80 DM, AUCH C64/Apple II, Dipl. Finanzwirt U. Ulrich, Bachstr. 70, 5216 Nidenfels, ☎ 02208/481 56

Suche Programme für dein Spectrum 128, Thomas Bernhart, Hasterstr. 110, 4407 Emededter, ☎ 025 72/29 00

## ◆◆◆ Wegen Systemwechsel ◆◆◆

Verkaufe Spectrum Plus (48k) + IF1 + Microdrive + Multiface one + Datenrecorder + Monitor (Philips TP 200) + Joystick + Joystick-IF + ZX-Drucker + 5 Rollen Metallpapier + jede Menge Software u. Lit. + RS-232-Kabel für VB 900,- ☎ 021 347/55 50 (ab 18 Uhr)

## ◆◆◆ Utilities zu Tassword Two ◆◆◆

**Tassword Two** Ergänzung: viele n. Fkt. Tassdata: sortieren beliebiger Daten, Tassformat: neues Format für Texte, Tassdel: suchen von Textteilen, Altes MC, je 20,- DM, 3 Pgm 50,- DM oder Info von Johann Weizner jun, Eichstr. 29, 6301 Hohenhain, ☎ 05 84 8/2 89

**Hallo Spectrum-User!**  
 Suche ZX-Spectrum Tauschpartner. Listen an: Uwe Gernsheim, Grüner Winkel 11, 4065 Jüchen 3, Antworte 100%ig

◆ **Hilfsprogramme zu Tassword Two** ◆  
**TASSWORD ERGÄNZUNG:** viele zus. Funkt. TASSDATA: universelles Sortieren! TASSFORMAT: Umformulieren des Textes. Alles MC, je 20,- DM, alle 3 zusammen DM 50,-, Schreck/born/NN oder -80 DM für Info an Johann Weizner jun., Eichstr. 29, 6301 Hohenhain, ☎ 05 84/2 89

## ◆◆◆ An alle Spectrum-User ◆◆◆

◆◆ **Werfen Sie ihr ROM weg!** ◆◆  
 Von mir bekommen Sie ein besseres, HM-Siem, eigener Zeichensatz und viele neue Routinen für nur 30,- DM. Info anfordern! S. Hartmann, Am Rain 6, 35033 Lohfelden 1

Suche für Spectrum zuverlässige Software-Tauschpartner (Ges.). Zuschriften und Listen an: Claudia Meuners, Postf. 1233, 40563 Jüchen 2 • Antwort 100%ig

**ZX-48:** Verkäufe Printer- und Original-Programme, suche Astrologieprogramme und Hardcopy für GP-100 Centronics-Version-V-Interface, tausche auch Original-Programme. ☎ 05 11/56 31 20

## ◆◆◆ ACHTUNG ◆◆◆

Programmverwaltung löst Ihre Probleme bei einer unübersichtlichen Softwareansammlung (mit 64 Zeichen pro Zeile und Quicksavefunktion). Info gegen Rückporto oder Programm direkt gegen 20,- DM (Schein, Schack), Jöngsfot, Postf. 10 26 10, 4040 Neuss 1

**ZX Spectrum Datenbank für Schreibpartien, M-Code.** Es spielt und speichert über 250 Partien. Info - 80 DM bei C. Spiek, Hofgasse 23a, 6117 Schaeffheim-Mosbach

**ZX-Drucker-Metal-Printer** ◆◆◆  
 Ab 3 Rollen à 7,- DM + Versandkosten 6,90 DM per Nachnahme bzw. 3,50 DM bei Vorkasse oder Vorkassenscheck, EST-Milner, Rüdgelamm 13, 3000 Hannover 21

Suche Software für dein ZX 81 16 K, Marcus Thier, Harterstr. 7, 3300 Braunschweig

## ◆◆◆ Für Sinclair QL ◆◆◆

Epronkarten mit integrierter Software. Karten mit 16 oder 32 K Exports und bis zu acht Stackpositionen. Englische OLS zu jedem Titel. Teilweise Infos bei: Armin Lang, Rheinfelderstr. 30B, 4047 Dombogen 1

## ◆◆◆ Buchungssystem SP-DV ◆◆◆

Buchungen nach versch. Schlässem, Kassenbuch, Kontogegenbuch, Jahresbericht, Sehr komfortabel und evtl. flexibel. Adressverwaltung zum Lastschriftdruck, Erhalten usw. nur 99,- DM. J. Weizner, Eichstr. 29, 6301 Hohenhain, ☎ 05 84/2 89

Verkaufe ZX-Spectrum 48K, Interface I, Microdrive + Software für 250,- DM. H. Hase, Jährstr. 5, 6330 Wetzlar 5. ☎ 064 41/2 79 82

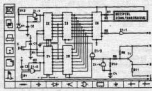
## ◆◆◆ Verkaufte wegen Systemwechsel ◆◆◆

ZX-Spectrum 48k + Aufsatzkassette + ISO-ROM + Interface 1 + Microdrive + 16 Cartridges + 2 Curran-Slots + zahlreich Software + vier Joysticks + Kempston-Centronics-Interface-E für nur 800,- ☎ 061 31/825 83

◆◆ **Hallo Spectrum-Freaks** ◆◆  
 Ich suche für dein ZX Spectrum ein Joystick-Interface mit zwei Ports. Andreas Töfler, ☎ 071 44/1 23 73

## Atari • Atari • Atari • Atari

**Elektronische Schaltungen konstruieren mit dem ATARI**  
Büchermotiviertes Arbeiten mit dem Joystick.  
Widerstand, Kondensator, Diode, Transistor, Garter,  
Inverter, ICs usw.  
- Drucker-Hörspiele  
- Läden und Säulen auf Diskette  
- Beutelebensdrehwerk führen usw.



**ATARI  
XL/XE 64K  
40.- DM**

inkl. Versand  
Bestellung:  
Schein/Scheck,  
Info kostenlos  
Nachnahme  
+ 6.- DM

**Jürgen Dörr**  
Einsteinstr. 8  
6520 Worms 26  
☎ 06241/34140

Spectrum: GP-150A mit Kempton-Centronics-Interf., voll grafikfähig 80 Zeichen/Zelle, für 300.- DM! Sound-Generator, erweiterbar zur PIO für 45.- DM. T. Stainer, Nr. 220, 2812 Hoyersgau, ☎ 042 51/36 49 (ab 18 Uhr)

MC-Programme debuggen - kein Problem mit dem 280-Micro (ZX-SF 48K) 23.- DM. Grafik auf dem Sokoloha GP 150 mit LPRINT3/0/Watadi-Centronics-IF (Anpassung für andere IFA auf Anfrage) 30.- DM. Info geg. 80 Pf. bei Jürgen Ohmardt, Eusebiusstr. 29, 5905 Freudenberg

**Verkaufe ZX Spectrum, Interface 1, Microdrive, ZX-Drucker, Bücher, Original-Software** (z.B. Desartillon, SuperPrint, Rechtschade, Boxing, Beta-Basic usw.) ☎ 071 48/61 48 (ab 17 Uhr)

Lotto-Statistik 6 aus 49: Spectrum 48K. Alle Ziehungen seit 1985! Auswertung in 10 sec.! Mit Gewähr! Bis zu 5 Blank-u-beläge viele Auswahlzahlen können untersucht werden. Cass. 45.- DM, Scheck ost + NN: G. Blatt, Wiesenerstr. 10, 5509 Schillingen. Auch für CPC 464, 964 und 6128.

**Verkaufe: ZX DATAFORM II**  
Ein selbstbeschriebenes Dataportprogramm der Spitzenklasse mit Anleitung und Cassette für 12.- DM. Eric Böhnlich, in den Beeten 80, 7121 Ingersheim 1, ☎ 071 42/15 16 61

Drucker GP 50 S zu verkaufen. Preis: 200.- DM. V. C. Riewenham, Haukm 7, 4837 Vart, ☎ 052 45/27 50

●●● Suche für ZX Spectrum ●●●  
●●● und GL ●●●

Selbstgeschriebene Programme zu kaufen (Anwendersoftware u. Spiele). Schickt erst mal eine Liste (wenn's geht) mit Beschreibung an: DATATEX, Loorbeweg 5, 4230 Wessal

Verk. Spectrum 48K für 110.- DM, Interface 1 für 40.- DM, beides defekt u. Mikro-Hefte je 1.50 DM. ☎ 08 21/96 32 84 nach 14h

**Neuheit!** Drohen Sie den Text Ihres Spectrums per Software! Anwendungsgebiet: Debugging, Ersetzen von Spielen etc. GrafikInfo oder Bestellung (Cass. und auf Anleitung + 25.- DM) bei Ingo Kaun, Jan-Willem-Str. 35, 5630 Remscheid

Spectrum 80 K mit Interface 1 und Microdrive und Software: 480.- DM. ☎ 0 91 31/38 23 02

●●● Eprom-Programmierer ●●●  
Da komplette Superding, Programm 64, 128, 256 u.a., direkt aufzutaken. Inklusiv-Software für nur 69.95 DM bei T. Kögler, Julius-Broch-Str. 13-15, 2400 Lübeck. Info gratis!

**Sokoloha GP 100 A Mark II** für VB 250.- DM, ZX Link II Interface für VB 150.- DM, zusammen für **VP 350.- DM** zu verkaufen. Wolfgang Leven, 4250 Bortrop 2, ☎ 020 45/830 74 ab 17 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: Otmar Vorholt, Schuler 5, 4416 Eversleben 11, ☎ 025 82/12 59

Top1! Opa-Floppy 439.- DM, GP 50 S + Zuhel. 179.- DM, GP 50 S def. 48.- DM, LPRINT II + Kabel 99.- DM, Centrac, E-Kabel 99.- DM, SPHINT-Recorder 99.- DM, Spectr., nur 169.- DM etc. Software ... ☎ 03 83/ 74 90

**Suche für ZX 81 series! Interface RS 232 von Memotech! ☎ 992 92/ 63 23**  
Suche prog. Joystickinterte, 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, Dahlengauß 16, 4010 Hilden, ☎ 021 03/6110 30 ab 17 Uhr

ZX 81 mit Profistatur, Repeat, Tonbox, 64 K, Highresol. Graphics, Fachliteratur für Basic und Masch.-Code, alle zusammen VB 250.- DM, Hans Adler, Schweinbühl 27, 4300 Essen 13, ☎ 02 01/55 05 83

**ULA oder kaputtier Spectrum 80-110.** Lantz Zerga, Hirsing 111, A-5201 Güssing/Verga, Meldet Zucht

Sinclair QL, RGB-Monitor (auch für andere Computer geeignet), Basic-Compiler, Bücher, 25 Cartidges mit Software, komplett 1000.- DM, einzeln Vh. ☎ 067 21/12421

**40 K-Spectrum, IF1, MDrive, Masterfile, Tasword, Leonardo-Gratik, MCTest-Tool, Sohachuch, Masterchess, Speechess, The Turk, Superchess 3.5,** ein paar kleinere Programme, Kabel, Handbücher + Literatur: 444.- DM, K5rip, Ruchshausweg 67, 4400 Münster

●●● Achtung Spectrum-User! ●●●  
Tasche Software für Spectrum 16-48 K. Suche User, die mit Discovery 720 K Erfahrung haben. Hajo Kamphausen, Postf. 1133, 4053 Jüchen 2, ☎ 021 64/ 76 10 nach 18 Uhr

●●● Neue Tasword 2 - Druckeroutine ●●●  
Druckt einen Text mit jedem Zeichensatz (8+4 bis 8+8-Matrix) aus. Programm + 20 Zeichensätze: 15.- DM. ☎ 050 45/15 92

Verkaufe: 48K Spectrum in Orig. Verp. + Discovery Heft für 900.- DM. Eponer: Ammer 256 + Schw. 120.-, diverse Bücher bitte anfragen! Suche: Discovery-User zwecks Erfahrungsaustausch! ☎ 073.21/638 79

## Wußten

Sie erinnern sich schon, daß Sie die Fähigkeiten Ihres Spectrums nur zu einem Bruchteil auszunutzen? NEIN??? Gerade deswegen haben wir vom Spectrum-Prof-Club ein aktuelles Sonderheft entwickelt, das die besten Artikel aus unseren identischen, bisher erschienenen Clubmagazinen enthält. Der Inhalt wurde optimal zusammengefaßt, so daß möglichst viele Anwendungsbereiche aufgenommen werden konnten. Von Interfacebauplänen über Listings für anspruchsvolle Anwendungen bis zum Poka, dar den Spectrum in die ewigen, technischen Jagdgründe versetzt, ist wirklich fast alles vertreten. Dieses saure und inhaltstausche "Spectrum-Prof-Sonderheft" kann gegen 10.- DM Selbstkostenentstehung bei uns angefordert werden: bar, Scheck oder per NL, dann jedoch gegen 15.- DM! Spectrum-Prof-Club, Lärcherstr. 2, 6091 Mairnbek. Als kleine Geschmackssprobe einige Ausschnitte aus dem Einzelheft: ●●● Bauplan einer Centronics- und Kempston-Schnittstelle mit passender Software ●●● Auflistung von ROM-Routinen, Systemkopien und Soviekopien ●●● des weitesten jode Menge Basic- und MC-Listings ●●● Tips und Tricks für jede Lebenslage ●●● Computerkurse ●●● Erfahrungsbereiche ●●● 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, Roosenstr. 16, 7990 Achern oder ☎ 07841/258 09

●●● Achtung Lehrer! ●●●  
ZX Spectrum Arbeitsblätter/Lehrerführung durch Notenprogramm: verwaltet Noten-Namenlisten, Max. 10 Klassen zu max. 40 Schülern mit max. 50 Einzelnoten/Schüler; erstellt Notenablage; druckt Formulare; erstellt Leistungskurve; deutscher Zeichensatz; schnell durch MC in der Praxis erprobt... INFO mit frankiertem Rückumschlag anfordern: U. Trutter, Roosenstr. 16, 7990 Achern oder ☎ 07841/258 09

Billige Interfaces 1 + Drives gesucht! Kaufe auch defekte für 1.- ☎ 07321/638 76, others preferred!

ZX Sinclair Spectrum Videointerface für alle monoch. Monitore, hardwaremäßige Inverse Darstellung (gebar), Reset-Taste nur an der User-Port anstecken, Preis 70.- DM inkl. P+V. ☎ 071 40/ 55 80 ab 18 Uhr.

●●● VERKAUFE KOMPLETT ●●●  
Spectrum 48K + Kempston-Maus + Kompton Centronics-IF + Plotter-Schreibmaschine. Sägemehl (4 Farbt.) Joystick + IF + Software: Art Studio, Tasword 2, Planets, Laser Basic, Winter Games, 3D Chess + Datenrecorder, alles 5 Mon. alt, orig. Verp. NP 1000.- DM für 1200.- DM. ☎ 06021/692 74

●●● Kaufe ●●● Verkaufe ●●● Tasche ●●●  
Programme für Spectrum und Atari (nur Cassettes), Listen an / bei: Marcus Czymbak, Im Messeg 10, 6000 Frankfurt, ☎ 069/52 06 49

Ich programmiere alle EPROM-Typen für Spectrum, QL, ATARI ST und alle anderen Computer. Anfragen unter ☎ 09 11/40 56 96 ab 15 Uhr, D. Appel

Der SM AC wandert sich an alle Spectrum-User, die ernsthaft mit ihrem Gerät arbeiten. Kostenl. Info + Heft: D. Kompell, Waldstr. 70, 5200 Siegburg

**Suche CK 250 bis Oktober 1985 und GP 150 X Drucker (Sokoloha)** ☎ 040/7963 37, Dierr Rector, Jochemhofswinkel 5, 2104 Hamburg 92

ZX Spectrum Video Interface, anpassungsfähig (gebar), Inverse Darstellung über Hardware Resettable. Nur an den Userport stecken. Preis 70.- DM, inkl. Porto + Verpackung. ☎ 071 46/ 55 80 ab 18 Uhr

●●● Sinclair Spectrum ●●●  
Drucker Sokoloha GP 50 S + 2 Rollen Papier + 2000 Einblättl. MultiForm + 2 Farbdrucker 145.- DM, CIB-Color Farbmonitor mit Sinclairstecker 698.- DM. Beides jeweils mit Softwarepaket. ☎ 0221/550 50 74

**Super-Angebot**  
Cursih 3+Speech, Interface 1, Microdrive, 24 Cartidges, für nur 400.- DM zu verkaufen (zusammen, auch einzeln!) ☎ 062 26/32 06 (ab 18 Uhr)

Spectrum 48; Schnelles SAVE und LOAD bei 8000 Bytes, nur 20.- DM; 2-Passives Deasx, MDW/Modk, 1000 Zeilen/Sec., Traps, alle inoff. Bef., nur 40.- DM. Bei M. Stramen, Rütcher Str. 156/1513, 5100 Aachen

●●● ZX Spectrum ●●●  
Verkaufe meine Originale: Scribble, The Chess Player, Chess, OCF Schach, Trans-Express, Strip-Poker, Survival, Ghostbusters, Chaperon-Flag, Horacejoes (King, Queen), The Hobbit je 25.- DM, nur 1 vorhanden, H. D. Ganser, Wassergasse 13, 6714 Wiesbaden

●●● ZX Spectrum ●●●  
Verkaufe meine Originale: Scribble, The Chess Player, Chess, OCF Schach, Trans-Express, Strip-Poker, Survival, Ghostbusters, Chaperon-Flag, Horacejoes (King, Queen), The Hobbit je 25.- DM, nur 1 vorhanden, H. D. Ganser, Wassergasse 13, 6714 Wiesbaden

★ Business-Software ★  
★ Commodore - Sinclair ★  
★ QL-Software & Zubehör ★

PC-FIBU ab 199.-  
PC-Basic Business Paket 588.-  
C24 FIBU ab 98.-  
G64 Keyboard-Lief. Buch. 99.-  
Viazarte-Viazart-Multiplex  
SP-Tasword II 69.50  
SP-Masterfile-PR-Print 69.-  
SP-Statistik I 49.-  
SP-Software-Compiler je 96.-  
Kempston E-Interface 149.-

Katalog DM 3.-  
(Bitte Rechner angeben)

**Fa. Lückert/CK**  
R.-Wagner-Straße 71  
6239 Krieffel

## Atari

## \*\*\*\*\* SCHWEIZ \*\*\*\*\*

Suche Atari-ST-Besitzer! Bitte send mir eine Postkarte mit Eurer Adresse. Gegenleistung: Liste der Atari-Besitzer, Infoaustausch, Postkarte an Stephan Ahrens, Haslistr. 16, CH-5630 Muri

## \*\*\*\*\* NEU \*\*\*\*\* NEU \*\*\*\*\*

Atari ST-Freelink Public Domain + Kontaktklub Gründung. Näheres unter ☎ 07321/63679 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 Zeichnung, Gewinn-Tabellen, kurz gesagt, ein Programm mit vielen Möglichkeiten. 50,- DM oder 40,- sfr., ansonsten an D. Steiger, Wittlingstr. 164, CH-4058 Basel

## Telefonuhr für 800 XL/130XE

Laufende Anzeige von Gebühr und Sprechzeit während des Gesprächs. Mikroprotokollierung in Rechnungsdatei möglich. Abrechnung der Gesamtkosten für einen selbst zu best. Zeitraum auf Disk 19,80 DM + Porto bei: M. Roland, Postfach 5008, 4902 Bad Salzungen, ☎ 18-21 h 0322/75605

**Hallo Briefmarkensammler! Atari-Freelink!** Mit diesem Super-Programm verkaufen/tauschen Sie Ihre Briefmarken garantiert. 15,- DM an Luciano Lüchli, Dorfstrasse 10, 7350 Pforzheim-Würtm. 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, Australia 15, 8567 Creussen. Suche Tips zu Chemical ☎ 0-6270/1362

Suche preiswerte 1050-Ploppy. H. G. Schmittlein, ☎ 024 35/1708

Verkaufe Atari 1027-Drucker gegen Gebot. ☎ 0228/3364538, Andreas verlangen!

## \*\*\*\*\* STAUFKÄLLER \*\*\*\*\*

Abendhauben für Atari 800/800XL/1050, C 64, 1541, Klarsicht 14,25 DM, Kunstleder 19,95 DM, VK oder NN. Bei M. Köhn, Umenstr. 16, 4100 Dülburg 17, ☎ 02136/31977, nach 17.00 Uhr. Bei Vorkasse keine Gebühr!

Atari 800XL mit 1050 (nur kpl.) mit Büchern 89,- DM zu verk. Zubehör auf Anfrage zu verk. ☎ 07321/63879, öfters probieren!

## \*\*\*\*\* Atari 400/800/XL/XE \*\*\*\*\*

Utility-Disk, readonly 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. Everts, Verloer Str. 78, 5042 Puffeln 1. Bitte Datenblätter (DZ/C) angeben!

8021, \*\*\*\*\* 924 Programme \*\*\*\*\* wegen Systemwechsel zu Schneidernissen zu verkaufen. Nur Disk. Gratisliste anfordern! Karl-Heinz Hanlmann, Coesa Gratzella, CH-8518 Gorduno, Schweiz

Verkaufe Orig.-Disk/Cass./Modul. ☎ 082 51/5329 - Stefan Bayer, Vorchwitzstr. 3, 8660 Münchenberg

Atari XL/XE: Verkaufe Software ab 50 Pf. Liste gleich bei Michael Schäfer, Padstr. 47, 4250 Bottrop anfordern. Verkaufe auch Listings ab 0,50 DM.

## \*\*\*\*\* Atari 800 XL DZ \*\*\*\*\*

Tausche und verkaufe Supersoftware-Liste an/bel Alexander Spoden, Max-Planck-Str. 5, 8660 Münchenberg. ☎ 092 51/802 90, 19.00-20.00 Uhr. Verkaufe Superdisk (DZ), readonly mit Spielen für 20,- DM / 3 Disk 50,- DM. Schein bitte belegen.

Verkaufe Atari Diskdrive 810 inkl. DOS 2.5. 1a Zustand. 500,- DM (+ Porto). Drucker SEIKOSHA GP-250X, Centronics Interf. für Atari + Commodore, Endpapier, grafikfähig, 4 Schriftgrößen: 550,- DM (+ Porto). Anfragen: Marco J. Meyer, Kreuzweg 15, CH-4144 Arlesheim/Schweiz

## \*\*\*\*\* Hallo Freelink \*\*\*\*\*

Verkaufe Atari 800 XL, Recorder, 2 SpieleKocher, Software (zum Bsp. Bruce Lee one on one soccer etc.), Preis VS, ☎ 062 02/3430

ATARI 560 ST: 1 Megabyte Aufhebung, stockbare Platine ohne Lötlösungen, Eckenstockkarte und User-Port-Karte. Für 800 XL Oldtimerkarte mit 800/800 XL OS schaltbar. Für 400/800: OS Rev. B und es laufen alle Programme. Happy 1060 und Highchip, Basic XL/XE, Action ☎ 02 02/402614

Verkaufe Atari 800 XL + 1010, Programm, diverse Bücher, Stock, VB 500,- DM. ☎ 02129/50105

Atari-XL/XE-Software! Verkaufe wegen Systemwechsel meine kompletten Programme zu 30-70% des NP. Beispiel: Print Shop, Typewriter, Ultima 3, Arcade Match. Nur Originale, keine Raubkopiert! Liste anfordern bei: M. J. Meyer, Kreuzweg 15, CH-4144 Arlesheim/Schweiz ☎ ATARI!

Atari 800 XL: Suche Programme aller Art (Disk), R. Möller, Kiefernweg 16, 6946 Gonselmental

Verkaufe Computerzeitungen, drucke ATARI-Listings. Listen gegen 2,50 DM in Briefmarken von H. W. Herrmann, Mohweg 21, 9628 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. Ichen, Niederfelder Str. 44, 8072 München, ☎ 084 59/1669

## ATARI 400, 800, XL, XE

Interf. für Rechartech. computing; Schalterinterf. 3-8 Kan.; Scheitler; VC-Karte CE/BA; Alarmanlage; Sondert. 12 Tasten; A/D-Wandler 8 Bit - alles im Gd. inkl. Software - Info bei K. Huber, Kreuzstr. 6, 3583 Dausphthal 5

Suche Tauschpartner für Software aller Art! Schickte auch Listen an: F.-J. Kolback, Wattersteige 11, 2842 Kroge

The Magic Card für Floppy 1050. Bolkup jeder Disk durch HWA und 7,0 184 K pro Seite, bis 18fache Geschwindigkeit beim Lesen und Schreiben, Happy- und Avion-Kompatibel, nachrüstbar mit Track/Display, Controller und Happy-Archiv. Suche! Hardware. Christian Wendt, ☎ 02 02/402614

\*\*\*\*\* Logo-Software präsentieren \*\*\*\*\* SuperUTL 1 + 2 mit 80 Zeichen, Prog-Schützer, ...Außerdem von LOGO-Software: Programm-Liter, Mathe-Star, ...Spotprezise Info anfordern: CH. Kanja, Mainzer Straße 11, 6117 Schwelm, ☎ 09073/60981

# 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 3-8023-0677-8

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 280-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 Lösungswegs 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 Spielesreks  
224 Seiten, 25 Abb.  
33,- DM  
ISBN 3-8023-0870-0

Sie erhalten bei Ihren Buch- und Computer-Verschäner kostenlos das neue Verzeichnis 'VOGEL Computer-Bücher' mit ca. 166 Titeln.

VOGEL-BUCHVERLAG  
WITZLAUB

VOGEL - Computerbücher  
mehr wissen, mehr leisten

SF 554 (+Garantie) 348.- • VCS 2000 + 2 Joyst. + 12 Cass. 207.- • Progr. 1. KOMM.; Disk: A. Schreiber, NATO 800XL; 180.-; Startextor 25.-; Scram 10.- • ROM: Progr. 24.- • GASS. Rhythmus, Jet Boat, Jack, Spill, Ace, Tank Comm. je 12.-; Alstrik, Sp. Inv. je 10.- (Phase inkl. Versandkosten, Progr. = Orig.) • 07 31 / 74 239

Hallo Atari-800XL-Software. Liste bei Raff Petzka, Poststr. 47a, 4250 Bopstrop, 80 Pl. Rückporto!

Atari 1050; Suche Master DOS 2.0 und oder 2.5 Disk für 15.- DM pro DOS, Konrad Flachs, Müßlinger Str. 21, 7000 Stuttgart 40

Habe Lösungen zu: The Dark Crystal, Antares, Cromwell-House, The Mask of the Sun; id. + engl. 1. Null Grid Nord, Schloß des Grauens, Abenteuer im Weltbaum, The Serpent's Stay, The Dalis Quest, Nibelungen, Ja 10.- DM. Schain belegen! Zuschriften an: D. Klok, Waldweg 2, 8590 Marktredwitz

••••• Hallo Atari-Freaks! ••••• Wer tauscht mit mir Software für 800XL. Nur Disk! Franz Bachhuber, Lana-Christ-Str. 17, 8047 Karlfeld

••••• Hallo Atari-XL/XE-Freaks! ••••• Meine neue Spiele-Superdiskette ist da! Beidseitig mit ca. 40 Spielen! 20-DM-Schein an: J. Schwamer, Th.-Haus-Str. 37, 8960 Mönchberg 100 Sp. = 30.- DM!

••••• Super-Utility-Disk V.1.2 ••••• COPY 1040 + Assembler (Labels.) + Diass. (Labels!) + Diskmonitor 1040 + Boot-Sort-Converter + Micro-Bit-Load; nur 2 Sek. (DOS hat 79) + DL-Designer + Stripper + Super RAM-Disk XL bis 16 Files + ... nur 25.- DM! Schein/Scheck an Raff Devid, Günsterweg 13, 4700 Hamm, • 023 85 / 18 75

••••• Adventure ••••• Deutsches Reisen-Adventure (über 130 Kl). Eigenproduktion 20.- DM (Schein). Stefan Bauer, Freizeitweg 102, 5100 Aachen. Für Atari 800XL + Cass.

••••• ATARI ••••• Suche Diskkatalogation 1050 VHS 200.- DM • 045 23 / 41 80 (ab 15.00 Uhr)

Verkaufe Module: PacMan, Jumpman Jr., Peppy. Ich würde gerne Software auf Disk tauschen (80.- XL + 1050). • 052 22 / 7 26 05

••••• ATARI ••••• Suche preiswerte Atari-Computer, auch defekt + Floppy 1050 oder 810. M. Kammerlander, Aller Postweg 22, 4970 Bad Oeynhaus 1, • 057 31 / 53 03

ATARI 800 XL. Wer sendet mir zu: "The Eidolon" und "Spy VS. Spy" (2) Spielentwürfe? Dirk Wolf, Heydnstr. 5, 6908 Wiesloch

••••• ATARI ••••• Software für Erwachsene (Porno), Rhythmus, Ravens, Softair u.v.m. Info anfordern bei: Soft, Postfach 1619, 4930 Gütersloh

Verkaufe Atari 130 XE + Diskettenlaufwerk 1050 + Cassettenspieler XC11 + Software und einige Handbücher + leere Disketten NP 1232.- DM, VB 800.- DM, PS: Noch 5 Monate Garantie für Diskettenlaufwerk. • 023 81 / 2 59 32

••••• Ataris Achtung ••••• Tausche Supersoftware auf Disk. Liste mit Preisanschlag noch heute an: Burkhard Lieb, Am Wingart 18, 6080 Groß-Gerau 3

Suche Software!! Ich suche Software für meinen Atari 800 XL. Bitte nur auf Cassette. Preis nach Absprache. Frank Büdick, Kriegerdweg 50, 7000 Stuttgart 31

Suche auf Disk: Summergames 2, Wintergames, Boulder Dash 2, Knock out und andere MS-Programme. Kaufe und tausche. Angebote und Listen an Raff Lechner, Dreibrunnen Str. 6d, 8200 Schwainhart, • 09 27 1 / 64 83

800 XL Atari: Suche für meinen Atari Basic-Module sowie Software. Wenn sie geht, nur auf Cassette. Schreibt Ihre Angebote an Jörg Alfeldt, Bahnhofstr. 4, 4100 Duisburg 18

Verkaufe Original-Software für Atari 800 XL. • 02 21 / 51 98 44

Verkaufe/tausche Software für Atari 800 XL. Disk. Liste anfordern oder schicken. 100%ig Antwort. A. Quast, Hauptstr. 4, 5161 Merzenich 3

••••• Atari-800 XL/800 XL ••••• 256 Kb Speichererweiterung, XBI kompatibel, inklusive Software: a) RAMDISK (als Atari-Formate) b) Kollisionsprogramm (1 Diskette) c) Adressverwaltung (2000 Adressen). Preis: 278.- DM. Info gegen Rückporto bei: Völkner Schenk, Gallesstr. 22, 4970 Bad Oeynhaus 1, • 057 31 / 23 20

An alle Atari-Freaks! Suche Italienische Jagdflug Art und Partner zum Tauschen von Software. Heinz Dieter Pöger, Im Höhenröden 3, 4923 Ebstedt 1, • 057 54 / 13 41

Atari-Happy-Board für Floppy 1050, superschnell, kopiert alles, Double Density, 130 K, kein Löten, mit Anleitung + Software 7.0. Wagen Systemwechsel nur 149.- DM, mit Density-Duplyer 179.- DM. • 042 35 / 82 21

Gelegenheit! Wegen neuem Computer (520 ST) verkaufe ich meinen Atari 130 XE. 3 Monate alt und wenig gebraucht ist für 300.- DM zu haben. • 072 02 / 81 28 (16 bis 20 Uhr)

Public Domain Software für Atari 800 XL/130 XE. Liste gegen -80 DM Rückporto. Bei Frank Knöber, Kennedystr. 22, 7134 Krißlinden

User-Port für Atari 800 XL/800 XL. Programmierbar in Basic, Assembler u.a. zu 30 UD Leistungen mit 8522 I/O-DMA, Universal 50 pol. Anschlußkabel 40.-DM. System erweiterbar auf 9 Platinen mit Universalbusplatine und Treiber 130.-DM. Tausche MAC 85 gegen ACTION-Modul. • 099 75 / 7 50 ab 16 Uhr

••••• ATARI GAMES ••••• Der Nachfolger zu XELX ist da! Games 2 - 100% M-Code nur 90.- DM - Super Grafik, Spitzenantrieb! Versand per Nachnahme + 1.50 DM Porto. Lutz Malschun, Pehweg 5a, 3258 Aerzen 1

Suche Software für Atari 800XL (Disk) Spiele und Anwendungsprogramme. Listen an: D. Westermann, Alexanderstraße 290, 2900 Oldenburg

Verkauf: Atari-800XL, Floppy 1050, Drucker Sekoipa GP-100AT, Joysticks + Software (Macro-Ass., Texteditor, Donkey Kong) + Literatur, nur gegenw. 700.- DM. • 057 32 / 1 22 95, ab 16.00 Uhr

Alle Atari-800XL-Newcomer suche ich Kontakt zu Atari-Freunden oder Clubs vorzugsweise in Berlin. • 030 / 7115 99

••••• Hallo Atari-Freunde! ••••• Superdisk 2 ist da! Random mit Grafik, Sound, Utilities und Spielen für nur 20.-DM-Schein an Franz-Josef Kolbeck, Wackerlocher 11, 2842 Krogen

Hallo ATARI-XL/XE-Freaks! Meine neue Spiele-Superdiskette ist da! Beidseitig mit ca. 30-40 Spielen! 20-DM-Schein an: Jera Schwärzer, Th.-Haus-Str. 37, 8960 Mönchberg, 3 Disketten mit ca. 100 Spielen = 90.- DM

Matrix/Drucker GP-500AT für Atari Computer, preisgünstig abzugeben. Bitte anschreiben! NP 350.- DM. Verkaufspr. VS. B. Haftrank, Mönkestr. 12, 7121 Ingersheim 1, • 071 42 / 2 03 37

••••• Das 1050 Turbo-Modul ••••• ••••• Hallo Atari-Freaks! Schaut mal in ••••• den gewöhnlichen Teil rein und ••••• laßt Euch ein Info schicken! •••••

## Commodore

30 C64-Programme auf Disk für nur 25.- DM inkl. Versand. Vorschau an Uwe Fütterer, Kantstr. 24, 7952 Durmersheim. Info nur gegen Rückporto!

C64, Floppy 1541, Speed-DOS-E, komplett 780.-, ggf. auch einzeln zu verkaufen. • 05 21 / 1 58 24 / 15 23 75

C64 + Datenset + Abdeckhaube + Peripherie + Programme = 400.- DM. • 073 21 / 6 38 75, offers problems.

C64 + Floppy 1541, mit Baschi-Formel 64, DATASETTE und Software zu verkaufen. Wenig benutzt, 1 Jahr alt. Preis VS., ab 16.00 Uhr anrufen. • 021 64 / 23 90

## albs IMMER NEUESTE HARDWARE FOR DEN TI-99/4A .... DIREKT VOM HERSTELLER

EXTENDED-BASIC (Mechatronik) mit deutschem Handbuch 198.-  
EXTENDED-BASIC II PLUS mit deutschem Handbuch 285.-  
= Extended-Basic + Grafik Extended-Basic (Apostroph) in 1 Modul

Umsatzaktion Bei Bestellung eines EXTENDED-BASIC II PLUS vergüten wir Ihnen DM 70.- bei kostenfreier Zusendung eines original amerikanischen Extended-Basic-Moduls (elektrisch/mechanisch einwandfreier Zustand!!!) Sie zahlen nur noch 215.-

Umbauaktion (gilt nur für deutschen Lizenznachbau „Mechatronik“). Wir machen aus Ihrem EXTENDED-BASIC ein EXTENDED-BASIC II PLUS mit deutschem Handbuch für nur 95.-

32-K-RAM-ERWEITERUNG mit Centronic-Interface, Kunststoffgehäuse 190 x 110 x 60 mm zum seitlichen Anstecken an den Bus, der Bus wird nach rechts durchgeschaltet, mit 5-V-Steckerleitung 289.50

Unser Paketpreis-Angebot EXTENDED-BASIC II PLUS + 32-K-RAM-ERWEITERUNG ohne Centronic-Interface für nur 490.50

128-K-RAM-ERWEITERUNG, mit Centronic-Interface und 5-V-Steckerleitung 395.50

Technische Änderungen vorbehalten. Versand gegen Nachnahme oder Vorauskasse

4-FARBEN-PRINTER-PLOTTER A 4, Centronic-Schnittstelle, DIN-A 4-Format, Direktanschluß an 32-k- oder 128-k-RAM-Erweiterung 699.-

ANSCHLUSSKABEL von 32-k- oder 128-k-RAM an PP-A 4 SLIM-LINE-LAUFWERK 5.25" 88.-

(z. B. TEAC FT 55 II) 399.90  
EINBAUSATZ für 2 Laufwerke in original TI-P-Box 95.-

DISC-STEUERKARTE (CorComp), DS/DD, für max. 4 Laufwerke 635.-

Die Weltnachricht: GRAM-Karte für die P-Box mit 128k eröffnet völlig neue Perspektiven; u.a. Entwicklungssystem für eigene „Module“ für nur 698.-

TI-MAUS - die schnelle und komfortable Cursorsteuerung mit Software auf 5.25"-Diskette, mit 5-V-Steckerknetzell 296.-

EPROMMER - zum Brennen von Proms bis 27128, alle Funktionen Software gesteuert, Programm auf 5 1/4 Zoll Diskette mit Steckerknetzell 278.90

TI 99/4A Intern von Heiner Martin (englische Ausgabe) 34.95

albs-Alltronic · B. Schmidt · Postfach 1130  
7136 Öttsheim · T. 07041/27 47 · Tx. 7 263 738 albs

## Neu: Die externe 80-Zeichen-Karte ist da!







## 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 Wolk



C 64 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 Speed-DOS und Geos. Ältere Interfaces dieses Typs können für 50,- DM in die neue Version umgerüstet werden.

Info:  
Wiesemann Mikro-computertechnik  
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 "Trauminterface" 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 erstnhafte Darstellung einreicht.

Wiesemann Mikro-computertechnik  
Winchenbachstraße 3-5  
5600 Wuppertal 2

## Inserentenverzeichnis

albs-Altronic	S. 120
Belkenheid	S. 30
C & M Meyer	S. 32
CSV Piegert	S. 3
Dörr	S. 118
Dresser	S. 30
EZS	S. 3
Holschuh	S. 6
Hüthig-Verlag	S. 7
Individual Software	S. 25
Jäckel&Klintworth	S. 9
JeppoSoft	S. 5
Jupitersoft	S. 3/19
Kunz	S. 19
Kiemmer&Schulte	S. 47
Lücker	S. 118
Merkens	S. 6
Naujoks	S. 9
Padercomp	S. 99
Rausch&Haub	S. 101
Schiffbauer	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 Ritz-Eberle GbR  
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 Anzeigenpreistabelle 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 möglich.

Grafik + Satz: Druckerei Springer,  
7143 Vaihingen/Enz

Druck: Verlags- und Industriedrucke  
GmbH + Co KG  
7730 Villingen-Schwenningen

Titelbild: Rainer Grinda

Vertrieb: Verlagsgesinn  
Friedrich-Burgius-Str. 20, 6200 Wiesbaden



# Compy - Shop Ohg



Inhaber: J. Jähn und Peter Bee, 4330 Mülheim Ruhr - Gneisenaustraße 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 19K RAM auf 64K RAM - ohne RAMdisk	96,- DM
Atari 600 XL von 19K RAM auf 192K RAM - mit 128K RAMdisk	196,- DM
Atari 800 XL von 64K RAM auf 200K RAM - mit 256K RAMdisk	246,- DM
Atari 130 XE von 128K RAM auf 192K RAM - mit 128K RAMdisk	96,- DM

*Alle Speichererweiterungen von 64K sind kompatibel zum 130 XE und werden mit einem LOGS für die RAMdisk geliefert!*

### Die Oldrunnerkarten:

Oldrunnerkarte - für XL/XE-Computer - Speicher bei 8000 frei	96,- DM
Oldrunnerkarte mit 80-Zeichenkarte - für XL/XE-Computer	146,- DM
Oldrunner mit 4K-BIBOMON - für XL/XE-Computer	178,- DM
Oldrunner mit 80-Zeichen, 4K-BIBOMON - für XL/XE-Computer	196,- DM

### Der 16K-Bibomon:

Maschinenmonitor mit eingetautem DOS, Druckerschnittstelle, Hardcopy und Hardware	440,- DM
---	----------

•• Siehe Textbericht Computer Kontakt ••

### Programmiersprachen:

kyan Pascal, für 1 Laufwerk, liefert echten Objectcode	296,- DM
--	----------

•• Testbericht CK 6/7 '86 ••	
Basic XL von OSS, eines der schlechtesten Basic	226,- DM
Basic XE, von OSS, speziell für den 130 XE	246,- DM
Achford, von OSS, superschnelle Compilingsprache	246,- DM
Mac/65, von OSS	296,- DM
Macroassembler	49,- DM
Atlas II, Makroassembler	19,80 DM
Atlas II, Toolbox zum Makroassembler	19,80 DM

### Software:

Design Master	19,80 DM
Das Mini-CAD-Programm	29,80 DM
Die Soundmaschine, Super!	29,80 DM
Monitor XI, Masch.-Monitor für's Basic	19,80 DM
Pyramids, deutsches Action-Adventure	39,- DM

**und vieles mehr! Fordern Sie unsere Preisliste an!**



Pascal Software Atari XL/XE

**kyan pascal für die XL/XE Serie ..... DM 296.-**

*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-Graphik 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

© 1986 Compy-Shop



### Speedy 1050 - Super Speed

SPEEDY 1050 N, normale Ausführung	196,- DM
SPEEDY 1050 NE, Entwicklungssystem, mit allen technischen Unterlagen	246,- DM
SPEEDY 1050 T, mit digitaler Trackanzeige	296,- 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 kopierschutzfreie Software kopieren
- Digitale Trackanzeige (SPEEDY T)
- Akustischer Fahrmelder (SPEEDY T)

SPEEDY 1050 TE, Entwicklungssystem, mit allen technischen Unterlagen	346,- DM
--	----------

**EINBAU einer SPEEDY 1050 NE/TE in Ihr Laufwerk, zgg. 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**

