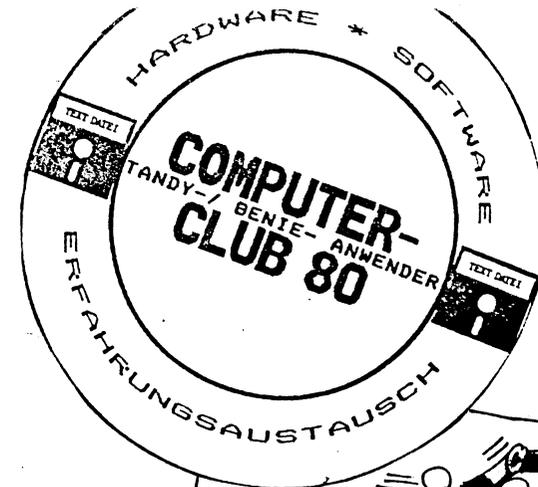


# CLUB 80



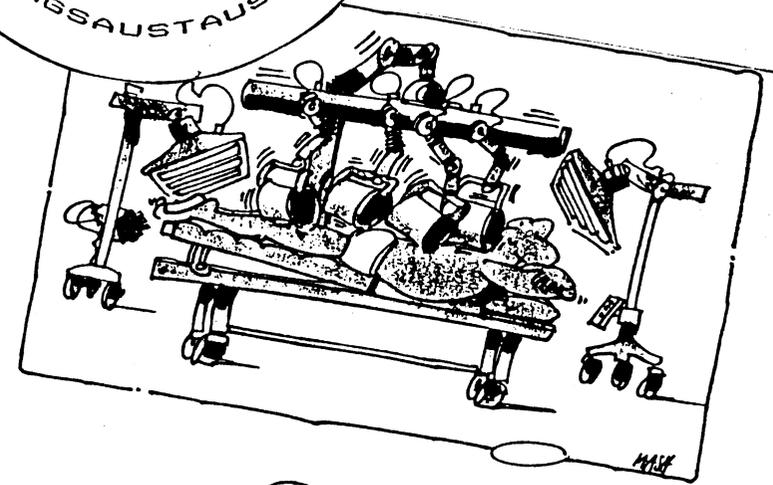
Club 80

der

TANDY -  
GENIE -

und KOMODER  
ANWENDER

26. AUSGABE



### Clubinternes

Neues vom Vorstand .....	01
Kurzbericht vom Nordlichtertreffen .	02
Gerald, Hartmut	
Happy Birthday .....	03
Jutta Obermann	
Aktuell und wichtig .....	04
Redaktion	
Vorstellungen .....	05 - 09
Jörg, Christian, Sven, Peter, Detlev	

### Software

SortICALC .....	11 - 14
Klaus-Jürgen Mühlenbein	
Was man weiß oder schon immer .....	15 - 17
Dump .....	18
Lottozahlen .....	19 - 20
Kurt Müller	
Atari-Bilder auf dem 4p .....	21 - 24
Hartmut Obermann	
CP/M Warmstart ohne Nachladen .....	25 - 27
Frank-Michael Schober	
BDOS-Replacement, was'n das ? .....	27 - 28
Es war einmal ein Datum .....	29 - 30
Alexander Schmid	
RFNL .....	31 - 35
Kurt Müller	

### Hardware

Besser erkennbare Darstellung von ..	37 - 38
EXP1-Floppy-Controller .....	38
Helmut Bernhardt	
80/40 Trackumschaltung Mitsubishi ..	39
Hartmut Obermann	
Entfernung der Kassettenschnittstell	40 - 41
Memory Mapped I/O (=) Port I/O GILs.	42 - 45
Gerald Schröder	
Darf's ein bisschen mehr sein .....	46
Alexander Schmid	
Sharp 3541 .....	47 - 51
Artikel aus Computer Persönlich	
Atari ST .....	52 - 58 (62)
Kurt, Gerald, Jens	

### Börsen

Wer hat was -- wer will was .....	59 - 61
Werbung EDU-Verbrauchsartikel .....	63 - 64

### sonstiges

Test: Brauche ich einen Computer ? .	10
Ein Segen für die Menschheit .....	34
aus "Ich hasse Computer"	
Gebrauchtcomputermarkt TANDY & Genie	65
Hartmut Obermann	
Wenn einer eine Reise tut ... .....	66
Alexander Schmid	
dBase II, III und III Plus überblick	67 - 68
Artikel aus DOS 5'88	
Auf gute Zusammenarbeit ! .....	69 - 70
Heinrich Betz	
Entwicklungen ... und wie es weiterg	71 - 72
Paul-Jürgen Schmitz	
Datenbanken richtig aufgebaut .....	73 - 77
Artikel aus Window 9'88	
Ein elektronisches Märchen .....	78
Dieter Kasper	
Ein informatisches Märchen .....	79 - 80
Kostenlos telefonieren .....	80
Alexander Schmid	
KI-Anwendung .....	81 - 82
Ein Wurm tötete Tausende Computer Iahm 83 - 84	
Artikel aus Tageszeitungen	

### Club-80 Bibliothek

Neuzugänge .....	85
Abzugeben .....	85
Werner Förster	

### Die letzten Seiten

Impressum .....	86
Schluß .....	87
Mitgliederadressenliste .....	am INFO-Ende
Redaktion	
Extrablätter:	
dBase II überblick und ALCOR-Pascal	

## Neues vom Vorstand

Hittfeld, 22.10.1988

Eigentlich wollte ich nur mal einen kurzen Besuch bei Gerald machen und den besonders guten Kuchen bei seiner Mutter genießen. Stattdessen sitze ich jetzt hier und bin dazu verdonnert, das Vorwort zum Info 26 zu schreiben (während sich Gerald hinter mir im Sessel räkelte und mir Stichworte zuruft).

Auch diesmal gibt es wieder Gutes und weniger Gutes zu berichten. Sehr gut ist, daß der Club in den letzten Wochen um 11 (in Worten: elf!) Mitglieder gewachsen ist. Besonders zu erwähnen sind in diesem Zusammenhang Gerald Dreyer und Heinrich Betz, die mit ihrer Werbeaktion erheblich an dieser Steigerung beteiligt waren.

Inzwischen ist Matthias Homann auch nicht mehr der einzige "nur-CP/M-User" im Club. Durch unseren während der Jahreshauptversammlung gefassten Beschluß, den Club reinen CP/M-Nutzern ohne Tandy-Ambitionen zu öffnen, gibt es inzwischen vier CP/M'er. Dabei sind natürlich die CP/M-nutzenden Tandy- und Genie-Besitzer, deren Zahl ständig steigt, nicht mitgezählt.

Jeder Neuzugang möchte natürlich so schnell als irgend möglich alle "alten" Infos haben. Dazu hat Jens Neueder vor einiger Zeit die Infos 1 bis 15 in Sammelordnern auf Karton zusammengefaßt. Diese Sammelordner kann jeder bei Werner Förster ausleihen. Die Sammelordner mit den Infos 16 bis 25 sind zur Zeit in Arbeit. In diesem Zusammenhang möchte ich einmal mehr auf die "Fundgruben"-Ordner hinweisen, die vor Jahren Günter Wagner zusammengestellt hat. In diesen hat er eine Unmenge Artikel gesammelt, die sich alle mit dem Thema Tandy TRS 80 bzw. Video Genie befassen. Gerade für Leute, die erst Mitte der 80'er Jahre zur Computerei gekommen sind, sind diese Ordner eine wahre Fundgrube. "Alte Hasen" finden darin z.B. viele Artikel aus der Zeit der ELCOMP und der guten Zeit der mc.

Da sich Gerald Schröder nicht mehr traut, Euch zu mehr Mitarbeit aufzufordern, bin wohl diesmal ich dran. Leute, wenn Ihr nicht mehr schreibt, wird unser Info bald nur noch aus der Titelseite, den Vorstellungen der Neuzugänge und dem Impressum bestehen. Dann wäre zwar die Vervielfältigung erheblich einfacher, das Lesen würde sich dann aber mit Sicherheit nicht mehr lohnen. Jens Neueder klagt schon, daß er trotz der geringen Auflage von nur vier Infos im Jahr 1988 fast keine Artikel mehr hat, um weitere Infos zu erstellen. Also schreibt, schreibt, schreibt!!! Schließlich wollen wir 1989, fünf Jahre nach der Gründung des CLUB 80, wieder sechs Infos auflegen, wie es sich gehört.

An dieser Stelle möchten wir uns noch bei Günter Wagner bedanken, der den CLUB 80 gegründet hat und uns Ende dieses Jahres verlassen wird. Ohne ihn wäre der Club nie ins Leben gerufen worden und hätte sich nicht zum heutigen Umfang entwickeln können. Vielen Dank an Dich, Günter!

Fröhliche Weihnachten und Frohes Neues Jahr wünschen

Hartmut und Gerald

## Kurzbericht vom Nordlichtertreffen

Am Wochenende 19./20.11 fand in Heist (nördwestlich Hamburg) das zweite Nordlichtertreffen statt. Bitte fragt uns nicht, warum die Nordlichter schon das zweite Regionaltreffen durchführen, während die Südländer noch nicht mal ein einziges auf die Beine gestellt haben (Ihr bekommt dann nämlich eine Antwort, die euch sicher nicht paßt)! Auf jeden Fall war es wieder mal ein Treff interessanter Computerfreaks und Hacker.

Es waren bei dem Treffen anwesend (zumindest körperlich):

Gerald Schröder, Bernd Retzlaff, Arnulf Sopp, Helmut Bernhardt, Kurt Müller, Jörg Brans, Gerd Rinio, Wilhelm Tornow, Christian Menk, Armin Ahlers. Als Vertreter aus dem Süden, last and least, Hartmut Obermann.

An Hardware waren zwei Model 4p, ein Genie IIIs, zwei Genie I, ein RTRS-80 und ein Toshiba T2100 (ratet mal, wem der gehört!?) vorhanden. Nicht vergessen sollte man natürlich die mechanischen und elektronischen Reparaturgeräte von Helmut, die auch ausgiebig gebraucht wurden. Zunächst mußte Arnulfs IIIs aus dem digitalen Nirwana zurückgeholt werden, dann hat sich das Bootlaufwerk von Jörgs 4p die Karten gelegt (es ist kaputt gegangen!!!) und mußte ausgetauscht werden. Für Bastelfreaks also genug Arbeit und für Hardwarelaien ein Anlaß zum Staunen.

Christian, unser z.Z. jüngstes Mitglied, hatte die Gelegenheit die Urahnen der heutigen PC's zu bestaunen. Ein Genie I ist für den Besitzer eines Sharp MZ 3541 ähnlich anzuschauen, wie eine JU-52 für einen Jetpiloten. Auch der Drahtverhau auf der Hauptplatine von Arnulfs IIIs ließ einige Ungläubige vermuten, daß die meisten Drähte nur Dekorationsfunktion hätten, was aber wirkungsvoll widerlegt werden konnte.

Am Sonntag konnten wir den Repräsentanten eines SVI-/MSX-Clubs (ca. 120 Mitglieder) begrüßen. Das Format ihrer CP/M-Disketten war schnell geknackt (Bericht in diesem oder dem nächsten Info) und somit steht einer weiteren fruchtbaren Zusammenarbeit nichts im Wege. Der Club gibt vier Infos im Jahr heraus, wobei wir einen Austausch der Artikel über CP/M anstreben.

Auch wenn das Regionaltreffen, im Gegensatz zur Jahreshauptversammlung recht kurz war, halten wir solche Treffen für sehr sinnvoll. Nur auf solchen Zusammenkünften können sich die Mitglieder persönlich kennenlernen, und der Informationsaustausch gestaltet sich mit Sicherheit im persönlichen Gespräch einfacher, als durch noch so umfangreichen Briefwechsel. Wir können den Clubmitgliedern aus dem Süden also nur raten, sich auch einmal aufzurappeln und ein Regionaltreffen zu organisieren!

In diesem Sinne alles Gute und ein bißchen mehr Initiative,

Der Vorstand\*

\*(Coproduktion Gerald & Hartmut)

Hallo meine Lieben " Clubfreunde ",

Nach Erscheinen des letzten Infos habe ich mich gleich darangesetzt, um Euch wieder zu berichten, wer alles in der nächsten Zeit Geburtstag hat. In der Hoffnung, daß dieses Info rechtzeitig bei Euch ankommt, berücksichtige ich dieses Mal nur 3 Monate, anstatt wie letztesmal, 4.

Wie Ihr unten sehen könnt, sinkt die Geburtenrate in den Wintermonaten rapide ab. Woran das liegen mag ?

Wie üblich, appelliere ich an Euch, mir Bescheid zu geben, wenn ich jemanden vergessen habe. Ich entschuldige mich deshalb schon im voraus bei den Betroffenen.

Diesmal sind die Monate November, Dezember und Januar dran.

" Novembergeborene "

- 02.11. : Krispin Michael
- 03.11. : Hentz Wolfgang
- 07.11. : Mühlenbein Klaus-Jürgen
- 08.11. : Schäfer Walter
- 28.11. : Neueder Jens

" Dezembergeborener "

- 03.12. : Rinio Gerd

" Januargeborene "

- 09.01. : Seelmann-Eggebert Jörg
- 17.01. : Betz Heinrich

Im Namen des Vorstandes gratuliere ich Euch recht herzlich zum Geburtstag und wünsche Euch alles Gute und weiterhin viel Spaß am Computer.

Ein besonders herzlichen Glückwunsch geht an unseren Inforedakteur Jens, dem ich mehr Artikel für das Info wünsche. Deshalb meine Bitte an Euch, die Arbeit des Clubs und des Vorstands mehr zu unterstützen (z.B. mit Artikeln usw.).

*Jutta*

--- Termine --- Termine --- Termine ---

Nächster Redaktionschluß ..... ca. 3 Wochen  
nach Erscheinen dieses INFO's

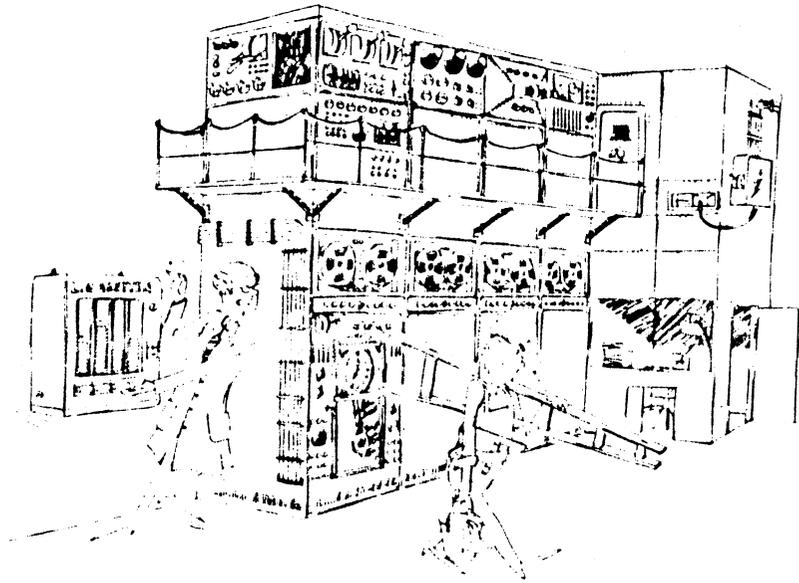
Jahreshauptversammlung ..... 25. April - 1. Mai 1989

--- Messen '89 ---

COMIT ..... Münster 8. - 15. März 1989

Internationale Computerausstellung ..... Berlin 25. August - 9. September

ESQUATRUM ..... München 7. - 11. November 1989



„Hallo, Sie - wie funktioniert dieser komplizierte Apparat?“

Hallo liebe Clubmitglieder,

ich möchte mich als neues Clubmitglied kurz vorstellen. Ich bin 45 Jahre alt, von Beruf Steuerbeamter, und arbeite (dies ist kein Schreibfehler) als Prüfer seit Dezember 1984 mit einem TRS 80 Model 4P. Das Gerät ist mit einem 280 KHz Microprozessor und 512 K ausgerüstet. Weiterhin besitze ich 4 Laufwerke, wovon 3 von 80 auf 40 Spuren umschaltbar sind. An Druckern stehen mir folgende Geräte zur Verfügung: DMP 430, HP Desk Jet, Silver Reed EX 32. Wie Ihr seht, habe ich eine ganze Menge in's Geschäft gesteckt. Die Hauptanwendung des Gerätes liegt bei Textverarbeitung (deutsche Version von Lscript / Vers. 1.70G und 1.80G) und Berechnung von Steuern (Lohn bzw. Einkommensteuerprogramme).

Durch meine Mitgliedschaft erhoffe ich mir Kontakte zu anderen Anwendern zu bekommen, und meine bescheidenen Programmierkenntnisse in Basic zu erweitern. Ich selbst kann Euch unter Umständen mit umfangreicher Literatur aushelfen, da ich sehr viele Zeitschriften besitze. Dies sind u.a. C't (komplett), 80 Micro (ab 1983), Computer persönlich (ab '83) Byte usw. Desweiteren besitze ich recht gute Kenntnisse über den amerikanischen Markt, und habe selbst schon sehr viel in Amerika per schriftlicher Bestellung (mit Visa Card) gekauft. Bisher habe ich nur die allerbesten Erfahrungen gemacht. Mit der Hoffnung, daß ich bald einige von Euch kennen lerne, verbleibe ich

Mit freundlichen Grüßen



Hallo Clubmitglieder!

Hiermit möchte ich mich gerne vorstellen. Ich heiße Christian Menk und wohne in Hanstedt (bei Hamburg). Ich bin 15 Jahre alt und besuche die 10. Klasse der Realschule. Ich werde voraussichtlich im Sommer 89 eine Lehre als Reiseverkehrskaufmann beginnen.

1982 bekam ich zu Weihnachten einen TI-99/4A von meinen Eltern und haben dann angefangen in Basic zu programmieren. Letztes Jahr habe ich mir dann von meinem Konfirmationsgeld einen SHARP dazugekauft. Seitdem stöbere ich ein wenig in CP/M herum. Meine Programmierfähigkeiten belaufen sich z. Zt. auf ~~ein~~ Basic (MBASIC 80) und ein bisschen Pascal (TURBO Pascal). Im Moment arbeite ich an einem Programm mit dem man teilen kann. Allerdings soll die Teilaufgaben ~~ein~~ ~~zu~~ Ergebnisse errechnen können die über 100. Kommastellen haben.

Mein Interesse geht zusätzlich noch zu Assembler und PL/I.  
Bis dann



Hallo Clubfreunde,

als vierter User eines reinen CP/M-Rechners möchte ich mich kurz vorstellen. Mein Name ist Sven Krahn, geb. am 05.04.1969 in Hamburg, wohnhaft in Villingen. Im Mai habe ich das Abitur gemacht und leiste seit Oktober meinen Wehrdienst ab. Danach habe ich vor, wirtschaftsingenieurwesen zu studieren. Mein Interesse für Computer begann vor etwa 10 Jahren. Nach verschiedenen Jobs und ständigem Hacken bei Freunden konnte ich mir 1984 endlich einen eigenen C64 leisten, mit dem ich Programme zuerst in Basic, dann in 6502-Maschinensprache schrieb. Kurz vor dem großen Preisrutsch konnte ich die ganze Anlage aber wieder losschlagen und mir statt dessen einen Olivetti M10 (Handheld) kaufen, der auch heute noch in meinem Besitz ist. (Der TRS80-M100 ist übrigens fast baugleich zum M10).

Zu CP/M kam ich dann durch Zufall. In einem Baseler Computershop konnte ich dem Verkäufer ein Gerät, das seit Wochen in der Reparatur stand, als nicht mehr reparierbar abschwatzen. Nach einiger Suche wechselte ich einen Kondensator aus und kam so für 50 Pfennig zu einem Computer. Nun brauchte ich noch ein Terminal anfangen. Nach einigen Monaten fand ich dann ein gebrauchtes, das mit einer Blechkiste ließ sich nicht allzuviel anfangen. finanziell meinen Erwartungen entsprach, die ganze Anlage sieht somit ziemlich antik aus. Es handelt sich um ein Micro Decisions von Morrow Designs (2 Laufwerke 5S/DD - 48tp1) unter CP/M 2.2 mit einem Terminal TAB 132/15. Aufgrund der immer weniger werdenden Informationen über CP/M bin ich noch nicht sehr weit ins System eingedrungen, ich programmiere hauptsächlich in Basic, Pascal und C, versuche mich auch mal in Assembler, meistens aber auf meinem Handheld. Vielleicht ändert sich das ja jetzt...

Meine Interessen liegen bei allem, was sich mit einem Computer machen läßt, sei es hard- oder softwaremäßig, system-orientiert oder Hochsprache, Elektronik oder Anwendungsprogramm.

Ich hoffe, daß wir gemeinsam das aussterbende Betriebssystem CP/M noch eine Weile am Leben erhalten können.

Euer



Meine Clubfreunde

Seit 1980, also zu einer Zeit, als der TRS-80 mit 16 K noch die einzige teure war, besitze ich einen solchen Computer. Allerdings hat es damit, daß ich mich (teils beruflich) zur Textverarbeitung interessiere und mir ein Videoterminal mit der ELTEC-Karte baute. Die Tastatur hatte ich sehr preiswert im DARC (Deutscher Amateur Radio Club) bekommen. Aufgezeichnet wurde mit Kassette in Schreibgeschwindigkeit.

Da ich Übersetzungen aus dem Amerikanischen machte, war die fehlende Möglichkeit des Verschiebens von Wörtern, um bei schwierigen Texten einen einigermaßen lesbaren deutschen Satz zu erhalten, doch recht hinderlich. Das brachte mich zum TRS-80. Nicht aus Überzeugung, damals war der PET sehr verbreitet, sondern durch Zufall.

Nach Jahren der Kassettschneiderei fügte ich mir ein Herz und kaufte eine leere Dr. Humann-Floppykarte, nachdem ich vorher Kleinbuchstaben aufgerüstet und auf 48 K aufgebohrt hatte. Denn lag die Karte aber noch einige Monate herum, ehe ich sie mit den Bauteilen versah. Zuerst benutzte ich drei 40-Spur-Geräte, seit 1987 2 x 40 SS/SD und 1 x 80 DS/SD. Mit Platz für Erweiterungen in einem selbstgebastelten ca. 4 Hohen-einheiten großen 19-Zoll-Gehäuse. 1986 kam ein CITIZEN 120-U dazu.

Fa ich den TRS-80 auch geschäftlich, in erster Linie jedoch als Hobby benutze, erhoffe ich mir vom Beitritt zum Club Unterstützung in Italien, in derer ich mit meinen Latein am Besten bin und das ist oft schon ziemlich schnell!!!

Im DARC habe ich Kontakt zur Arbeitsgemeinschaft Microcomputer und mehrmals hervorragende technische Hilfe bekommen. (Die 80 Microcomputer hat den TRS-80 viele Male auf 6 Europaplatten nachgebaut, eine Version kenne ich mit 7,5 M Speicher).  
Auf gute Zusammenarbeit!

*Wolfgang Winkler*

Da(s) bin ich !(:)

48-jährig, verheiratet, zwei Kinder (m/12,w/3) und nun seit kurzer Zeit Mitglied im CLUB 80.

Mein "Maschinenpark" umfaßt zur Zeit:

- 1 SANYO MBC-1150 (CP/M 2.2, 2x320k)
- 1 ATARI 1040 STM (TOS, 1x720k(940k), nur Farbmonitor)
- 1 ATARI 600 XL (2x1050, 64k-Erweiterung, Centr.-Schnittst.)
- 1 ATARI 800 XL (2x1050, Centronics-Schnittstelle, Datasette)
- 1 TANDY TRS80 COLOR-COMPUTER II (ohne Peripherie)
- 1 SINCLAIR ZX81 (16k-Erweiterung, Centronics-Schnittstelle)
- 1 9-Nadel-Drucker PANASONIC KX-F1083
- 1 9-Nadel-Drucker PRASIDENT 6313 (z.Z. nur 8 Nadeln)
- 1 9-Nadel-Drucker CP80-SHINWA (Melchers)
- 1 3 Nadel-Drucker STAR 8240
- 1 Thermo-Transfer-Flinter Brother HR5

An Programmiersprachen übe ich

BASIC, Sanyo-BASIC, GFA-BASIC (V2.2 und V3.0), GW-BASIC Turbo-PASCAL (V2.0)

LOGO

PROLOG

FORTH 83

XLISP (V1.4, V1.5)

Fit bin ich in keiner !

Regelmäßig arbeite (spiele) ich mit folgenden Programmen:

WORD-STAR	(SANYO)	1ST-Word-plus	(ATARI ST)
dbase II	(ATARI ST, SANYO)	Print-Master	(ATARI ST)
Print-Shop	(ATARI XL)	Multiplan	(SANYO)
FS2	(ATARI ST, ATARI XL)		
CALC-STAR	(SANYO)		
MS-DOS und CP/M	(ATARI ST)		

Für mich ist die Computerei kein Mittel des Broterwerbs sondern sie dient nur der (so vorhanden) Freizeitgestaltung.

Beruflich friste ich mein Dasein als Angestellter der Stadt.

(Hier habe ich auch noch Gelegenheit, ein bißchen unter den SIEMENS-Betriebssystemen BS2000 und SINIX herumzufummeln.)  
Bevor ich nun mit meiner "SELBSTDARSTELLUNG" das CLUB 80-Info in meine Biografie umfunktioniere, möchte ich an dieser Stelle (unter Hinweis auf meinen Briefkasten bei Interesse für weitere Details mich betreffend) diesen Beitrag beenden.

Mit abgebrochene Fingernägel vom Tastenquälen

rotgeränderten Augen vom Debuggen

leichten Magenschmerzen vom zu starken Bohnenkaffee und

freundlichen Grüßen an alle CLUB 80-Mitglieder sowie

aufrichtiger Anteilnahme am "Leid" deren Familienangehörigen

*Peber Schneider*

\* = regelmäßig ist auch 1x im Vierteljahr !!



Ein neuer Filename? Zählt mal die Buchstaben!  
 Alle Neune?! Also kein Filename.  
 Nur eine "Utility". (Und immer noch nicht in Maschinensprache!)  
 Natürlich für Freunde von VISICALC, wie der Name vermuten lässt.  
 Hat VC (VISICALC) nicht schon alles, was man braucht? Besonders in der 3.Version (die sich leider durch einen stark eingeschränkten Speicher "auszeichnet"... ) oder in der Modifikation "VCMOD/CMD", die das Herz jedes unverCALCTen VISIONärs höher schlagen lässt...?  
 In der Tat: VISICALC kann viel und steht seinem grossen Bruder MULTIPLAN nur wenig nach. Aber:  
 Eine einzelne Zeile oder Spalte **s o r t i e r e n** - das geht nicht; oder gar das komplette Tabellenblatt in die alphabetische oder numerische Reihenfolge gemäss einer bestimmten Zeile bzw. Spalte bringen. Kann es nicht. Eine echte Lücke!  
 In die bin ich schon einmal hineingesprungen (s.INFO 19, Juni 1987, Seite 29)  
 Dabei hab' ich mir zwar nicht gerade das Genick gebrochen, aber einen Fuss verstaucht. Unter bestimmten Bedingungen oder unglücklichen Konstellationen klappte es nämlich nicht mit diesem Programm, das unser Bibliothekar *Werner* in seiner Bekanntgabe neuer Programme versehentlich mit "VCMOD/CMD" bezeichnet hat (INFO 21, September 1987, Seite 65). Das merkte ich natürlich, als es darum ging, eine Tabelle, die im Laufe des Monats September stochastisch angefallen war, am Monatsende nach der Spalte zu ordnen, in der die Namen meiner Schüler standen. Also hiess es: Ran ans Werk und überarbeiten. Aber daraus wurde ein völlig neues Programm, das ich hiermit vorstelle und demjenigen, dem die Zeit des Abklopfens mit Recht zu schade ist, gern per Disk z.V. stelle. (Alle Formate.)  
 Voraussetzung für die Anwendung ist, das das (habt ihr's schon gemerkt? Ich befehlige mich bereits der neuen "Rechtschreibung", die z.Z. hier in Mannheim im "Institut für Deutsche Sprache" ausgebrütet wird, um uns das künftige Leben zu erleichtern... Wahrscheinlich habt auch ihr schon in der Zeitung davon gelesen! ss statt ß; das statt daß <wie logisch! Eigentlich müßte aus dem 'daß' ein 'dass' werden...) - also: Voraussetzung ist, das das zu sortierende File im sog. 'DIF-Format' (DATA INTERCHANGE FORMAT) vorliegt.  
 Um es nicht zu kompliziert zu machen, sortiert das Programm nur Spalten. Will man Zeilen sortieren, kann man die Tabelle im DIF-Format bekanntlich mit der C-Option leicht diagonal spiegeln!  
 Eine weitere Voraussetzung ist, das die Spalte, nach der sortiert werden soll, entweder nur numeri (Zahlen) oder nur strings (Wörter) enthält! Auch das ist eigentlich selbstredend - denn eine Datenfolge wie: "9640 Weinheim Am Mönchgarten 28" nach Zahlen sortieren zu wollen ("28 9640 Am Mönchgarten Weinheim") wäre ebenso sinnlos wie die Sortierung nach strings (die übrigens ganz genauso aussehen würde, da nach ASCII-Werten sortiert wird!)  
 Keiner hat das vor.  
 Wenn man nach einer Spalte sortieren will, in welcher Zahlen und Wörter gemischt sind - weil z.B. in ihrem Kopf eine Überschrift stet oder an ihrem Ende das Wort "Summe" u.ä. - so kann man den "astreinen" Teil der Tabelle, wie meniglich geüufig und gemeinhin bekannt, mit der #-Option aus der Tabelle herauschneiden, partiell speichern und dann mein Programm

```

10 * *****
20 * * V I S I C A L C *
30 * * Datei im Data Interchange Format umordnen *
40 * * <C> Klaus-Jürgen Mühlenbein, Weinheim 10/88 *
50 * *****
60 CLS: CLEAR 25000: PRINT STRING$(59, "*")
70 PRINT "*" V I S I C A L C -
*
* Datei im Data Interchange Format umordnen und speichern *
80 PRINT "*"
*
90 PRINT "*" <C> K.-J.Mühlenbein, Weinheim, Oktober 1988
*
100 PRINT STRING$(59, "*"); PRINT
110 DEFINT I, S, T, X, Y, Z: DEFSTR C, F, K, V, W: DIM K(16)
120 PRINT: PRINT "Es koennen nur Spalten sortiert werden!"
130 PRINT "(Ggf. mit der C-Option umformen.)"
140 PRINT "Die Spalten der Tabelle duerfen jeweils
entweder n u r aus Zahlen oder n u r aus Woertern bestehen!
(Andernfalls den gewuenschten Tabellenausschnitt als Hilfsdatei
im DIF-Format speichern!)"
150 PRINT "Bei Zahlenspalten darf die erste Zeile nicht leer sein
!"
(Ggf. diese Felder mit '0' belegen)": PRINT, " E N T E R !
160 Y$="": Y$=INKEY$: IF Y$="", 160
170 REM ***** Kennsatz lesen *****

180 PRINT "Welche DIF-Datei soll sortiert werden (Eingabe ohne Extension)":
INPUT F: F=F+"/DIF"
190 OPEN "I", F
200 FOR I=1 TO 16: INPUT #1, K(I): NEXT I: Z=VAL(K(7)): B=VAL(K(11))
210 CLS: PRINT "Die Tabelle hat "S" Spalten und "Z" Zeilen": DIM N(Z, S),
W(Z, S), T(S), II(Z), N1(Z), W1(Z)

220 REM ***** Datei lesen *****

230 FOR IS=1 TO S: INPUT #1, Y1, Y2, Y3, T(IS)
240 UNT (IS)+160 SUB 350, 400
250 NEXT IS: CLOSE
260 CMD "lc,n": INPUT "Welche Spalte soll sortiert werden <Buchstabe
n A...BK) "; C: CMD "LC"
270 CL=LEFT$(C, 1): CR=MID$(C, 2, 1)
280 SO=- (CR="")*(ASC(C)-52)-12
290 IFCR<>"": SO=SO-(CL="A")*(ASC(CR)-26)-(CL="B")*ASC(CR)
300 INPUT "von Zeile X bis Zeile Y <X,Y> "; X, Y
310 INPUT "Sollen die uebrigen Spalten mitsortiert werden <J/N>";
S$: SA=INSTR("JjNn", S$)

320 REM ***** Sortierschluesel definieren *****

330 UNT (80)+160 TO 440, 470
340
350 FOR IZ=1 TO Z
360 INPUT #1, N(IZ, IS): IF IZ<Z, 600 SUB 390 ELSE INPUT #1, V
370 NEXT IZ
380 RETURN
390 INPUT #1, V, T: RETURN
400 FOR IZ=1 TO Z
410 INPUT #1, N, W(IZ, IS): IF IZ<Z, 600 SUB 430
420 NEXT IZ: RETURN

```

darauf anwenden. Anschliessend baut man es - wieder mit der #-Option - wieder in die ursprüngliche Tabelle ein. Das alles ist jedenfalls wesentlich weniger anstrengend als die mühselige "MOVErei" unter Beachtung einer alphabetischen Reihenfolge!

Man kann mit dem Programm, wie gesagt, eine Spalte allein ordnen, wobei die Reihenfolgen in den übrigen Spalten erhalten bleiben, oder wahlweise <sup>die Zeile</sup> aller übrigen Spalten mit der Spalte, nach welcher sortiert wird, "mitgehen" lassen. Der Trick dabei ist, das im ersten Fall die sog. "direkte Sortierung" <CMD"D",n,A(1)>, im anderen Fall die sog. "indirekte Sortierung" <CMD"D",n,\*IX(1),A(1)> angewendet wird.

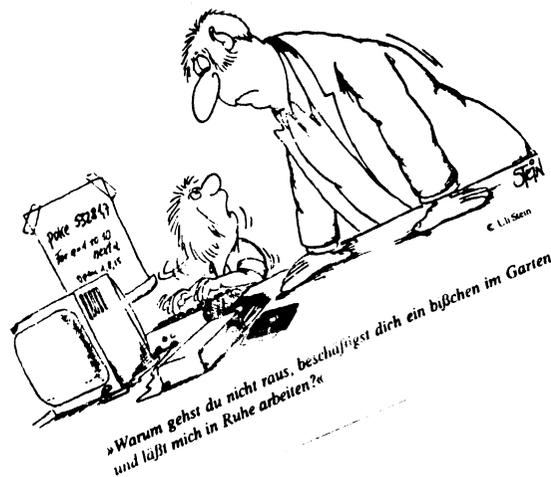
In der Tat: Ein mächtiger NEWDOS-Befehl.

Weshalb ich dieses Betriebssystem so ungern verlasse.

Oder weiss jemand ein besseres?

(Das das das kann, ist fenomenal...)

*Rafol*

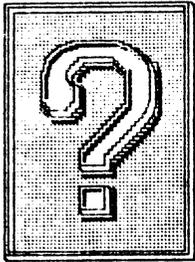


```

430 INPUT#1,T:RETURN
440 FOR IZ=XTOY:N1(IZ)=N(IZ,SO):NEXT
450 CMD"D",Y-X+1,*II(X),N1(X)
460 GOTO520
470 FOR IZ=XTOY:W1(IZ)=W(IZ,SO):NEXT
480 CMD"D",Y-X+1,*II(X),W1(X)
490 '
500 REM ***** Abspeichern der geänderten Datei *****
510 '
520 CLS:INPUT"Neuer Filename (ohne Extension und Laufwerk)";FF:INPUT
" In welchem Laufwerk speichern ";FW:FF=FF+"/DIF:"+FW
530 WW(0)="0";WW(1)="1";WW(3)="-1"
540 OPEN"D",1,FF
550 '
560 REM *** KENNSATZ ***
570 '
580 PRINT#1,"TABLE"
590 PRINT#1,WW(0);",",;WW(1)
600 PRINT#1,CHR$(34);CHR$(34)
610 PRINT#1,"VECTORS"
620 PRINT#1,WW(0);",",;STR$(Z)
630 PRINT#1,CHR$(34);CHR$(34)
640 PRINT#1,"TUPLES"
650 PRINT#1,WW(0);",",;STR$(B)
660 PRINT#1,CHR$(34);CHR$(34)
670 PRINT#1,"DATA"
680 PRINT#1,WW(0);",",;WW(0)
690 PRINT#1,CHR$(34);CHR$(34)
700 '
710 REM *** Aufbau des VISICALC-Files ***
720 '
730 FOR IS=1TOS
740 PRINT#1,WW(3);",",;WW(0)
750 PRINT#1,"BOT"
760 GOSUBB30
770 NEXT IS
780 PRINT#1,WW(3);",",;WW(0)
790 PRINT#1,"EOD"
800 CLOSE
810 PRINT:PRINT:PRINT"DOS <O> oder BASIC <ENTER>"
820 Y$=INKEY$:IF Y$="" THEN B2=0 ELSE IF ASC(Y$)=13 THEN ENDELSE CMD"BS"
830 IF X>1:FOR IZ=1TOX-1:INT(1S)+1GOSUBB90,910:NEXT IZ
840 FOR IZ=XTOY:IFT(1S)=1 THEN 930
850 PRINT#1,WW(0);",",;IF SA<3OR IS=SS, THEN PRINT#1,N(II(IZ),IS) ELSE
PRINT#1,N(IZ,IS)
860 PRINT#1,"V"
870 NEXT IZ:IF Y=Z, RETURN
880 FOR IZ=Y+1TOZ:ONT(1S)+1GOSUBB90,910:NEXT IZ:RETURN
890 PRINT#1,WW(0);",",;N(IZ,IS)
900 PRINT#1,"V":RETURN
910 PRINT#1,WW(1);",",;WW(0)
920 PRINT#1,W(IZ,IS):RETURN
930 PRINT#1,WW(1);",",;WW(0)
940 IF SA<3OR IS=SS, THEN PRINT#1,W(II(IZ),IS) ELSE PRINT#1,W(IZ,IS)
950 GOTOB70

```

# Was man weiß oder schon immer wissen wollte



"Seit Anbeginn der Zeit wartet er nun, daß..." - Ahem... nun denn. Wir haben keine Zeit zu warten. Für unsere Probleme werden andere Mittel benötigt als gerade "Warten". So etwas erzeugt nur Schwielen am Hin... ja und außerdem gerät man schnell ins Hintertreffen. Deshalb empfehlen wir einen Computer als die ultimative Lösung Ihrer Probleme.

Obwohl dieses Programm ist nicht auf meinem Mist gewachsen ist, will ich es dennoch hier wiedergeben. Vorweg erst einmal die Quellenangabe: CHIP No. 1, 1989, Seite 235. Der Autor ist Thomas Wolf.

So, und nun zur Sache. Das Programm erscheint mir deshalb interessant, weil es recht elegant implizite Funktionen darstellt. Dabei bedient es sich der Berechnung eines Punktrasters, welches die Umrise des Funktionsgraphen abtastet und durch eine Fallunterscheidung ( $F(x,y) < 0$ ) entweder einen Punkt oder ein kleines Kreuz ausgibt. Auf diese Weise entstehen zwei Flächen, deren Berührungskanten den Funktionsgraphen abbilden. Zur Verdeutlichung ist das Rechenergebnis der im Programm per "DEF FN" eingebauten impliziten Funktion wiedergegeben. Die bei der Berechnung benutzten Koordinatengrenzen waren:

Xmin = -10      Ymin = -10      Pixelabstand = 5  
Xmax = 10      Ymax = 10

Und die Funktion:

$$f(x,y) = 9*(x-y)*(x-y-x*x) - 3*x^4 - y^4$$

Die Rechenzeit liegt bei etwa 30 Sekunden. Auf einem Genie I schätze ich so um die 5 Minuten.

*Trüßl*

```

' Darzustellende Funktion
DEFFN f(x,y)=9*(x-y)*(x-y-x*x)-3*x^4-y^4
' Umrechnung der X- und Y-Koordinaten als Funktion
DEFFN i(x)=xn-(xmax-x)/dx
DEFFN j(y)=(ymax-y)/dy
'
' X- und Y-Ausmaße des Grafikbildschirms
'
yn=400
xn=640
'
' Lage des Koordinatenkreuzes eingeben
'
CLS
PRINT TAB(19);"Darstellung implizierter Funktionen"
PRINT TAB(19);"-----"
PRINT
INPUT "  x-Achse von .....: ",xmin
INPUT "           bis .....: ",xmax
INPUT "  y-Achse von .....: ",ymin
INPUT "           bis .....: ",ymax
PRINT
INPUT "Pixelabstand (z.B. 5): ",py
'
' Berechnung einiger Schrittweiten für die Grafik
'
CLS
px=xn/(yn*1.25)*py
dx=(xmax-xmin)/xn
dy=(ymax-ymin)/yn
'
' Berechnung der Funktion als Punktraster
'
HIDEM                           ! Mauszeiger ausschalten
GOSUB koordinatenkreuz
FOR i=0 TO xn STEP px
  FOR j=0 TO yn STEP py
    IF FN f(xmax-dx*(xn-1),ymax-j*dy)<0 THEN
      PLOT i,j
    ELSE
      PLOT i,j+1
      PLOT i,j-1
      PLOT i,j
      PLOT i+1,j
      PLOT i-1,j
    ENDIF
  NEXT j
NEXT i

```

Auf eine Taste warten

```
REPEAT  
UNTIL INKEYS<>""
```

```
SHOWM  
END
```

!Mauszeiger wieder einschalten

```
PROCEDURE koordinatenkreuz
```

!Diese Variablen gelten nur hier

```
LOCAL xnull,ynull,x,y  
xnull=FN 1(0)  
ynull=FN j(0)  
LINE xnull,0,xnull,yn  
LINE 0,ynull,xn,ynull  
FOR x=INT(xmin) TO INT(xmax) STEP ABS((xmax-xmin)/100)  
  LINE FN 1(x),ynull-2,FN 1(x),ynull+2  
NEXT x  
FOR y=INT(ymin) TO INT(ymax) STEP ABS((xmax-xmin)/100)  
  LINE xnull-3,FN j(y),xnull+3,FN j(y)  
NEXT y  
RETURN
```

DU KÖNNTEST AUCH MAL WAS  
IM HAUSHALT TUN ...



NA, HÖR MAL - DU KENNST DOCH MEINE  
EINSTELLUNG: DAS IST IN EINER  
GLEICHBERECHTIGTEN BEZIEHUNG  
HEUTZUTAGE DOCH  
SELBSTVERSTÄNDLICH!



AHH...WO SCHALTET  
MAN IHN EIN...?



### Hallo Club-Freunde !

wie dem vorletzten Club-Info zu entnehmen war, hatte Kajot so seine liebe Not mit einem Textrettungsversuch. Es scheint nicht ganz bekannt zu sein, daß der DUMP-Befehl außer Maschinensprache-Programme, auch einfache Daten auf Diskette bannen kann. Hierzu muß nur ein Parameter anders angegeben werden. Die störenden Loader-Informationen werden dann unterdrückt. Nachfolgend die Beschreibung.

### DUMP

DUMP,FILENAME,Startadresse,Endadresse <,Programmstart>

Das Kommando DUMP schreibt den Inhalt des Speichers auf die Diskette unter dem Namen FILENAME. Der Dump beginnt mit der eingegebenen Startadresse und hört mit der eingegebenen Endadresse auf. Start-, End- und Programmstartadresse werden in dezimal (Wert kleiner als 65536) oder hexadezimal (Wert kleiner als 10000H) eingegeben. Für die hexadezimale Eingabe muß das 'H' angehängt werden.

Das DUMP-Kommando hat zwei verschiedene Funktionen:

- \* Wird als Programmstartadresse 65535 (OFFFh) eingegeben, dann wird ein exakter Speicherabzug auf die Floppy geschrieben. In den ersten 2 Byte steht dann lediglich die Startadresse. Diesen File kann man mit dem 'SUPERZAP' anzeigen oder ausdrucken (Funktion 'DMDB') oder auf einem anderen TRS-80 die Fehlersuche durchführen.
- \* Ist die eingegebene Programmstartadresse kleiner 65535 (OFFFh) oder wird sie nicht angegeben, so wird der Speicherinhalt im Loader-Format auf die Floppy geschrieben. Dieser File kann dann später einfach über den Loader wieder ins Memory geladen werden.

ACHTUNG: Die Startadresse darf nicht kleiner als 5200H sein, da sonst das DOS zerstört wird.

Beispiel: DUMP,FILENAME/CMD:1,5200H,89ABH,5700H

Der Speicherinhalt zwischen der Adresse 5200H und 89ABH wird auf die Floppy 1 unter dem Namen FILENAME/CMD abgespeichert und als Startadresse 5700H eingetragen. Über das Kommando

LOAD,FILENAME/CMD

kann dieses Programm wieder in den Speicher geladen werden oder über das DOS-Kommando

FILENAME <,Parameter>

direkt ausgeführt werden. Im zweiten Beispiel ist dasgestellt, wie ein Programm bzw. der gesamte Speicherinhalt im Fehlerfall auf Diskette geschrieben wird, wenn man momentan keine Zeit hat, den Speicherinhalt zu analysieren. Voraussetzung ist allerdings, daß man entweder im DOS ist oder über die Tasten 'DFG' ins Mini-DOS kommt.

DUMP, MEMORY/DMP:1,0,65535,65535

Mit diesem Befehl werden die kompletten 64K, einschließlich ROM und Display auf Floppy 1 in den File MEMORY/DMP geschrieben. Das erste und zwei Byte dieses Files sind dann 00 00 (Startadresse des Dumps). Im File sind sonst keine weiteren Kontrollzeichen. Zu einem späteren Zeitpunkt kann man dann über 'SUPERZAP' (Option 'DMDB') sich diesen Dump ansehen und auswerten. Zu beachten wäre noch, daß der DOS-Speicherbereich zwischen 4000H und 51FFH nicht mehr den gleichen Inhalt hat, da 'DUMP' aufgerufen wurde.

1987  
JANUAR  
1987

18

Hallo Clubfreunde,  
Diesmal geht es um Geld oder wie vermeide ich die Qual mit den Lottozahlen. Das weiter unten wiedergegebene Programm stammt aus meiner Mottenkisten und erblickte vor nicht ganz 10 Jahren zum ersten mal das Licht dieser schönen Welt. Damals mußte mein erster Komputter - ein 8-Bit Prozessor der auf den Namen SC/MP (National Semiconductor) hörte und in mehreren Folgen in der Bastelzeitschrift 'Elektor' erschien - für diese Aufgabe herhalten. Da das benutzte Integer-Basic ganz und gar nicht als schnell bezeichnet werden konnte, war diese eher 'mickerige' Aufgabe schon ein beachtlicher Brocken an Rechenzeit. Ich hatte deshalb an allen Ecken und Enden 'Geschwindigkeitsoptimiert' wo es nur ging. Was soviel wie Minimierung der benötigten Befehlszeilen bedeutete. Die Ur-Version enthielt sogar noch eine in Basic geschriebene Zahlenformatier-Routine, damit die Nümmerchen auch alle sauber untereinander standen. Alles in Allem war die Berechnung der 10 Ziehungen in unter fünf Minuten abgeschlossen!

Da ich auch heute - *10 Jahre danach* - immer noch keine große Lust verspüre mir die Lottozahlen selbst auszudenken, habe ich diesen Programmoldie nochmals überarbeitet und auf das GFABasic übertragen. Hier ist die ganze Aktion in ca. 2 sec. erledigt. Nur mehr gewonnen habe ich deshalb bisher nicht. All' jenen, die es genausowenig lassen können wie ich viel Spaß beim 'Ziehen der Zahlen' - und sei es auch nur auf dem verknitterten Papier aus dem Drucker.

Tschuß

*U.A.*

```
DIM z%(11)
h1$="      Die Lottozahlen der vergangenen Woche"
INPUT " Ausgabe auf den Drucker (Y/N): ";h3$
IF LEFT$(h3$,1)="Y" THEN
  f!=TRUE
ELSE
  f!=FALSE
ENDIF
CLS
DO
  ' Entscheiden ob Drucker oder nicht
  '
  IF f!=FALSE THEN
    PRINT AT(1,1);h1$
  ELSE
    LPRINT h1$
    LPRINT
  ENDIF
  PRINT AT(1,3);
  '
  ' Die sechs Zahlen ziehen und ...
  '
  FOR i%=1 TO 10
    z%(0)=INT(RND*48+0.5)+1
```

```
durchlaeufe%=1
FOR j%=1 TO 5
  INC durchlaeufe%
  z%(j%)=INT(RND*48+0.5)+1
  FOR k%=0 TO j%-1
    IF z%(k%)>z%(j%) THEN
      DEC j%
      k%=j%
    ENDIF
  NEXT k%
NEXT j%
' ... aufsteigend sortieren ...
'
FOR k%=0 TO 4
  m%=k%
  FOR j%=k%+1 TO 5
    IF z%(m%)>z%(j%) THEN
      m%=j%
    ENDIF
  NEXT j%
  SWAP z%(k%),z%(m%)
NEXT k%
'
' ...und ausgeben.
'
IF f!=FALSE THEN
  PRINT TAB(5);USING "Spiel### => ",i%;
  FOR j%=0 TO 5
    PRINT USING "#####",z%(j%);
  NEXT j%
  PRINT " [";durchlaeufe%;"] ";
  PRINT
ELSE
  LPRINT TAB(25);USING "Spiel### => ",i%;
  FOR j%=0 TO 5
    LPRINT USING "#####",z%(j%);
  NEXT j%
  LPRINT
ENDIF
NEXT i%
'
' Zehn Ziehungen sind ausgegeben, weiter ?
'
PRINT
PRINT TAB(5);"Abbruch mit 'X'";
PRINT TAB(5);"Neue Ziehung mit 'CR'";
REPEAT
  keys=INKEY$
UNTIL keys<>""
EXIT IF keys="X"
LOOP
'
END
```

Während unseres Sommerurlaubs haben meine bessere Hälfte und ich unseren Vorsitzenden Gerald Schröder besucht und bei diesem Anlaß natürlich nicht nur Hamburg besichtigt. Wie kaum anders zu erwarten, wenn zwei Computerfreaks zusammen-treffen, wurde auch ausführlich über das Hobby konferiert. Da Gerald sein IIs zu dieser Zeit gerade mittels LötKolben und Zange außer Betrieb gesetzt hatte, blieb uns zum Probieren und Programmieren nur sein ATARI 1040 ST übrig.

Nicht verschweigen will ich, daß sich das eigentlich ganz gut traf, denn mein Toshiba T2100 [ein MS-DOS (Helmut Bernhardt würde sagen DomesDOS) kompatibler Laptop], der mich auf der ganzen Reise begleitete, kommt mit den 3¼"-Disketten des ATARI sehr gut zurecht. Da mich die Grafikfähigkeiten des ATARI immer schon interessiert haben, lag kaum etwas näher, als sich ein paar Bilder vorzunehmen und diese zu übertragen.

Soweit zur Vorgeschichte; nun ans Eingemachte!

Gerald hat, wie schon erwähnt, einen ATARI 1040 ST mit einem monochromen Bildschirm. Die höchste Auflösung, die der ST hinbringt, beträgt 640 \* 400 Punkte bei zwei "Farben". Letzteres heißt im Prinzip nichts anderes, als daß nur zwischen "Punkt gesetzt" und "Punkt nicht gesetzt" unterschieden werden kann. Die möglichen Auflösungen im Farbmodus und Farbbilder interessieren uns hier im weiteren nicht.

Zunächst mußten wir herausfinden, wie die Grafiken beim ATARI abgespeichert werden. Dazu wurde mit einem der am weitesten verbreiteten Grafikprogramme (STAD ST Aided Design) eine horizontale und eine vertikale Linie gezeichnet und dieses "Bild" im sogenannten DEGAS-Format abgespeichert. Mit einem Disketteneditor wurde die Datei dann analysiert und folgendes festgestellt:

- 1. zunächst enthält die Datei 34 Bytes als Header in dem u.a. festgelegt ist, ob es sich um ein Monochrom- oder ein Farbbild handelt. Der Header für Monochrombilder sieht wie folgt aus: 00 02 07 77 00 00 00 70 00 00 07 77 07 00 00 70 07 70 00 07 07 07 00 77 05 55 03 33 07 33 03 73 07 73.

Zur Unterscheidung der Farbbild-Header:  
00 00 03 25 07 00 00 70 07 70 02 57 07 40 00 77 07 77 00 07 07 07 03 73 07 73 03 37 07 37 03 77 00 00.

Was die einzelnen Bytes und Bytefolgen bedeuten, kann ich euch leider nicht mitteilen (das wäre aber wahrscheinlich auch nur für ATARI-Besitzer interessant), aber man kann anhand des Headers immer unterscheiden, ob es sich überhaupt um ein Bild im DEGAS-Format (es gibt auch Grafikprogramme, die andere Abspeicherformate benutzen) handelt und ob ein Farb- oder ein s/w-Bild vorliegt.

- 2. Nach dem Header folgen 32000 Bytes Bildinformation, die folgendermaßen organisiert sind: je 80 Bytes repräsentieren eine Zeile; jedes Byte repräsentiert 8 Bildpunkte, wobei die Reihenfolge Bit 7 = erster, Bit 0 = letzter Bildpunkt ist. Ein einzelner waagerechter Strich über die ganze Bildbreite am oberen Bildrand ergibt also 80 Bytes mit dem Wert FFh. Ein senkrechter Strich am linken Bildrand ergibt im ersten Bildinformationsbyte den Wert 80h, dann 79 Bytes 00h, danach wieder ein Byte 80h usw. usw., d.h., die Zeilen sind in der Reihenfolge abgespeichert, in der sie auch auf dem Bildschirm stehen.

Nachdem geklärt wäre, wie die Dateien aufgebaut sind, ist es eigentlich kein Problem mehr, die Bilder sichtbar zu machen. Zunächst habe ich, noch in Hamburg, die Bilder in den Grafikspeicher meines T2100 eingelesen. Im Gegensatz zum "normalen" CGA (Color Grafik Adapter), der nur 640 \* 200 Pixel darstellen kann, beherrscht der Toshiba auch einen Grafikmodus mit 640 \* 400 Punkten. Wenn nun auch noch die Zuordnung der Speicherstellen zu den Bildschirmzeilen so einfach gestaltet wäre wie beim ATARI, wäre die Sache sehr einfach; leider ist das aber nicht der Fall! Das soll aber hier nicht weiter interessieren, wir wollen uns ja dem 4p bzw. den TRS 80-Computern zuwenden.

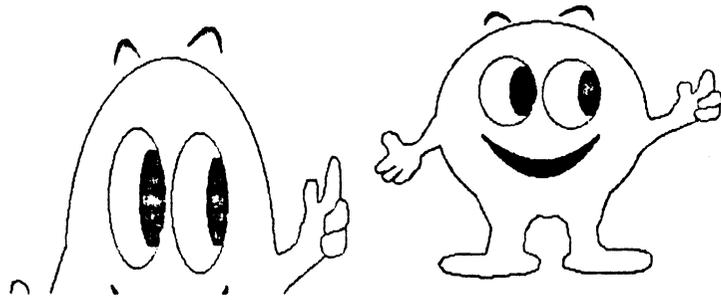
Die µLabs-Grafikkarte des Model 4p kann maximal 640 \* 240 Punkte monochrom darstellen und ist damit um 40 vertikale Punkte besser als der CGA. Nichts desto trotz können die ATARI-Bilder nicht "am Stück" angezeigt werden. Trotzdem gibt es Möglichkeiten, die Grafiken doch zu nutzen. Entweder liest man nur den oberen oder den unteren Teil eines Bildes ein, oder man wählt einen Mittelteil, der möglichst viel Bildinformation enthält. Diese Methode ist relativ einfach zu realisieren, befriedigt aber nicht so ganz, da immer Bildinformationen verloren gehen.

Erheblich eleganter ist es, das Bild zu komprimieren und so auf 640 \* 200 Punkte zu halbieren. Um das zu erreichen, könnte man natürlich einfach jede zweite (alle geraden) Zeile(n) weglassen, verliert dadurch aber erheblich an Bildinformation. Überträgt man z.B. ein Schaltbild, bei dem eine waagerechte Linie in der zehnten Zeile über den ganzen Bildschirm verläuft, wäre diese Linie (sie liegt ja in einer Zeile mit gerader Nummer) verschwunden, die Zeichnung damit unbrauchbar! Dasselbe könnte natürlich passieren, wenn man alle ungeraden Zeilen weglassen würde.

Eleganter und ohne Bildverlust ist die Methode, die Gerald Dreyer und ich schon einmal bei der Anzeige von BTP-Bilddateien (Grafikhilfsprogramm zu dBase II unter CP/M) angewendet haben. Dabei werden einfach jeweils zwei Zeilen bildpunktweise ODER-verknüpft. Hier ein kleines Beispiel:

Bildpkt: 0 1 2 3 4 5 6 7		Bildpkt: 0 1 2 3-4 5 6 7
Zeile 1: * * * * * * * \	OR ->	Zeile 1: * * * * * * *
Zeile 2: *	/	
Zeile 3: * * * * * * \	OR ->	Zeile 2: * * * * * *
Zeile 4: *	/	
Zeile 5: *	\	
Zeile 6: *	OR ->	Zeile 3: *
	/	

Das links dargestellte "F" ist rechts zwar etwas verstümmelt, zur Not aber noch als solches zu erkennen. Bei Bildern ist der Effekt erheblich besser, wie das Beispiel auf der folgenden Seite zeigt. Das erste Bild zeigt nur den oberen Teil der Grafik, das zweite die ganze, allerdings nach oben beschriebenen Verfahren komprimierte, Grafik.



Im linken Bild ist nur der obere Teil des Bitmännchens zu sehen, während rechts, der durch OR-Verknüpfung von jeweils zwei Zeilen komprimierte Mister Bit ganz sichtbar wird.



Ganz egal, ob man alle geraden oder alle ungeraden Zeilen wegfällen läßt, die Bildverluste sind immer sichtbar!

Das folgende Listing in Turbo-Pascal 3.0 stellt nicht das ganze Programm zum Einlesen von ATARI-Bildern auf dem 4p, sondern nur die eigentliche Einlese/Anzeige-Routine dar. Das ganze Programm abzudrucken lohnt nicht, es ist in der CP/M-Programmsammlung bei Andreas Rychlik zu haben. Hinzuweisen ist auf die Art, in der die Grafikdatei eingelesen wird. Normalerweise funktioniert das Einlesen einzelner Bytes unter CP/M nämlich nicht. Ich habe dazu das Bibliothekmodul BYTEFILE.BIB aus der CHIP Spezial Turbo-PASCAL Nr. 4 benutzt, welches diese Möglichkeit zur Verfügung stellt. Auch das Modul gibt es bei Andreas. Die Dateioperationen enden daher auch immer mit "ByteFile" (z.B. "AssignByteFile") und sehen etwas gewöhnungsbedürftig aus!

Ich glaube, aus den Kommentaren im Listing geht eigentlich alles wesentliche hervor, so daß sich weitere Erklärungen erübrigen. Anzumerken wäre noch, daß auf anderen TRS 80-Computern (z.B. Genie IIs) das Programm durchaus etwas anders aussehen kann. Das liegt daran, daß hier nicht direkt in den Bildschirmspeicher geschrieben werden kann, sondern für jeden Bildpunkt die PLOT-Routine aufgerufen werden muß. Das verlangsamt die Sache zwar, ändert aber nichts am Prinzip und klappt genauso gut, wie ich von Alexander Schmid weiß. Übrigens ähnelt auch die Einleseroutine für den 640\*200-CGA-Modus der oben gezeigten Routine so stark, daß nur wenig Umbauarbeiten notwendig sind, um das Programm auf PC's umzuwickeln.

```

procedure restore(name:filename;var exist:boolean);

var header : array [0..33] of byte;      [Buffer für den Header]
    zeile1, zeile2 : array [0..79] of byte; [Buffer für je eine Zeile]
    i,j:integer;
    datei:ByteFile;

begin
  clsh;                                [Löschen der HRG]
  [Si- ]                                [Abschalten der Fehlerbehandlung durch Turbo-Pascal]
  AssignByteFile(datei,name);           [Datei öffnen]
  ResetByteFile(datei);
  [Si+ ]
  if IOresultByteFile=0 then           [wenn dabei kein Fehler auftritt,]
  begin
    exist:=true;                        ["Alles OK"-Flag für die aufrufende Routine setzen]
    for j:=0 to 33 do
      ReadByteFile(datei,header[j]);    [34 Byte Header Lesen]
    port[131]:=185;                     [Steuerport des µLabs-HRG programmieren]
    for j:=0 to 199 do                   [für 200 Zeilen]
      begin
        port[128]:=0;                    [x-Register auf 0]
        port[129]:=j;                    [y-Register auf aktuelle Zeile]
        for i:=0 to 79 do                [eine Zeile = 80 Byte einlesen]
          ReadByteFile(datei,zeile1[i]);
        for i:=0 to 79 do                [noch eine Zeile einlesen]
          ReadByteFile(datei,zeile2[i]);
        for i:=0 to 79 do                [alle 80 Bytes beider Zeilen ODER-verknüpfen]
          begin                          [und in den Datenport schreiben. Es werden]
            port[130]:= (zeile1[i] or zeile2[i])
          end;                            [also immer 8 Bildpunkte gleichzeitig gesetzt!]
        end;                              [wenn 200 * 2 Zeilen gelesen und bearbeitet sind]
      end;
    CloseByteFile(datei);                [Datei schließen]
  end
  else                                  [falls beim Öffnen der Datei ein Fehler aufgetreten]
  begin
    exist:=false;                        [ist, "Alles OK"-Flag rücksetzen und zurück ins aufrufende Programm!]
    exit;
  end;
end;

```

Wem die Kompression des Bildes nicht behagt, der kann sie natürlich auch direkt auf den Drucker ausgeben. Alexander Schmid hat ein solches Programm geschrieben und wird es sicher auch in der CP/M-Bibliothek deponieren.

Bleibt nur noch die Frage zu klären, wo man ATARI-Bilder herbekommt. Ich habe es da ja relativ einfach mit meiner Methode ATARI - Toshiba T2100 - 4p, da der T2100 sowohl die ATARI- als auch die CP/M-Disketten des 4p direkt lesen/schreiben kann. Wer diese Möglichkeit nicht hat, muß auf Datenübertragung per serieller Schnittstelle zurückgreifen. Natürlich bin ich gerne bereit, Dateien (in fast jede gewünschte Format) zu übertragen. Außerdem sind auf der Programmdiskette in der CP/M-Bibliothek ein paar Bilder und ich habe auch noch ein paar, die ich, allerdings nur gegen Altersnachweis (da nicht ganz jugendfrei) weitergebe.

Viel Spaß mit den ATARI-Bildern,

CP/M-Hardstart ohne Nachladen?

Zunächst den Anger von Rudiger Sörensen im Club-Info Nr. 25 nur teilen, dann er "sich über die "Dihettenrasperei" von CP/M bei "C" beschwert. Wenn man sich dazu noch vor Augen hält, daß ein Nachladen des CCP oder gar des BDOS eigentlich heute kaum noch notwendig ist, ist das Verfahren um so ärgerlicher. Die Nachladerei ist ein Relikt aus der Kinderstube von CP/M, als in seligen Urzeiten 16 KByte RAM in einem Rechner schon den Gedanken an CP/M aufkeimen ließen. Die Generieranweisung für CP/M mit MOVECPM ist noch Heute Beweis für die Machbarkeit solcher "Mini-Systeme". Bei einem 16- oder 32-KByte-System muß natürlich alles, was nicht lebensnotwendig für das System ist, bei Start eines User-Programmes über Bord geworfen werden. Hier heißt die Devise TPA um jeden Preis!

Anders stehen die Dinge bei einem 64-KByte-System. Selbst umfangreiche Programme, wie DEBASE II oder auch NordStar, selbst mit aktiviertem internem Drucker-Puffer, laufen ohne sich am CCP oder gar am BDOS vorbeizureifen zu müssen.

Das Problem ist also nur, den Zugriff auf den Adress-Bereich des CCP zu verbieten. Wenn es gelingt den CCP generell vor Überschreiben zu schützen, wird das Nachladen überflüssig und bei Warmstart wird lediglich noch das Rücksetzen des Systems notwendig. D.h. es wird in diesem Fall kein System auf der Diskette benötigt! Der Vorteil liegt auf der Hand. Das "Blankfeilen" der Systemspur entfällt, "C" wird zur Sekunden-Sache, fremde Disketten ohne eigenes System können problemlos auch in Laufwerk A beschrieben werden, POWER wird zum TURBO-POWER...

Die Programme, die ein Überschreiben des CCP bedarfsweise durchführen, orientieren sich am Inhalt der Adresse 06-07 (BDOS-BASE) und überschreiben alles was vor dieser Adresse liegt. Der Trick liegt jetzt darin, dem Programm eine Adresse zu übergeben, die vor dem CCP liegt. Damit ist der CCP tabu!

Dieses Verfahren wird z.B. auch von ZSID oder DU benutzt, um sich vor Überschreiben zu schützen.

Man könnte man sagen, nichts leichter als das, direkt vor dem CCP wird ein JUMP auf die eigentliche BDOS-BASE eingetragen und auf den Adressen 06-07 wird die Adresse zu eben diesem JUMP abgelegt.

Die CCP-BASE beginnt immer auf einer geraden Adresse, jedoch sind die ersten 2 Byte frei und werden nicht benutzt. Dort läßt sich also der Sprung zur BDOSBASE eintragen.

```
BDOS  JUMP NEWBASE (CCPBASE)
```

```
NEWBASE: JUMP BDOSBASE
```

Leider ist die Angelegenheit so einfach nun auch wieder nicht. Bei Warmstart wird der Vereinbarungsbereich von Adr. 00 bis 07 neu initialisiert. D.h. nach einem Warmstart würde die alte BDOSBASE wieder eingetragen werden, damit ist der CCP wieder zum "Abschuß" freigegeben. Außerdem würde sich die in der Warmstart-Routine vorhandene Ladeprozedur für den CCP und das BDOS herzlich wenig um unsere Manipulation kümmern und nach wie vor den Ladevorgang durchführen.

Also muß auch hier geändert werden. Im allgemeinen beginnt die BIOS-Routine WBOOT mit dem Rücksetzen des Systems, wie Selektion DRIVE A, setzen Stackpointer usw. Dann folgt die Routine zum Nachladen einer entsprechenden Anzahl von Sektoren von den Systemspuren. Danach wird die Initialisierung des Vereinbarungsteiles vorgenommen. Abgeschlossen wird die Routine mit einem JUMP zum CCP.

Folgende Änderungen in der Warmstart-Routine müssen also vorgenommen werden:

- Änderung der Initialwerte für Adr. 06-07
- Nachladeroutine überspringen.

Man kann die Sache natürlich auch komfortabel gestalten und das Überschreiben wahlweise zulassen oder nicht. Dazu kann ein System-Flag angelegt werden, in dem beispielsweise auch das Laufwerk festgelegt werden kann, von dem nachgeladen wird.

Weiterhin ist ein 2-Byte Merkflag für die jeweils aktuelle Sprungadresse auf Adr. 06-07 einzurichten, das irgendwo in BIOS installiert werden kann.

Im folgenden ein Vorschlag für ein solches Vorgehen:

```
Aufbau SYSFLAG BIT 0-1 Laufwerks-Codes des gewünschten aktuellen Lauf-
                    werkes nach "C" (A=00, B=01...), gleichzeitig
                    das Laufwerk, von dem evtl. CCP/BDOS nachgela-
                    den werden.
                    BIT 7  Schalter für nachladen CCP/BDOS
                    0= kein nachladen
                    1= nachladen
```

Beispiel-Routine für WBOOT:

```
WBOOT: LD SP,0000 ; Stackpointer auf Standard
        LD A,(SYSFLAG) ; BIT 0-1: Laufwerks-Codes
                    ; BIT 7 0=kein nachladen, 1=nachladen
        PUSH AF ; A-Reg. sichern
        AND 03,A ; Nur BIT 0 und 1, Laufwerks-Codes DRIVE A-C
        LD C,A ; Einträge SELDISK vorbereiten
        CALL SELDISK ; Aktuelles Laufwerk selektieren
        LD HL,BDOSBASE ; Wert für Adr. 06-07 bei ungeschütztem CCP
        LD (BDSFLAG),HL ; in Merkflag laden
        POP AF ; A-Reg. zurück
        BIT 7,A ; Test auf nachladen
        JP NZ,M1 ; CCP und BDOS nachladen, Vereinbarungsteil
                    ; aufrufen und Start
                    ; Wert für Adr. 06-07 bei geschütztem CCP in
        LD (BDSFLAG),00 ; Merkflag laden
        LD A,C3 ; JP-Befehl auf NEWBASE vor CCP
        LD (BD),A ; eintragen
        LD (NEWBASE+1),HL ; Adr. BDOSBASE für JP-Befehl eintragen
        JP START ; Kein nachladen!
```

```
M1 ; Nachladeroutine CCP u. BDOS
```

```
START: ; Initialisierung Vereinbarungsteil
```

```
LD HL,(BDSFLAG) ; BDOSBASE auf Adr. 06-07
LD 0006,HL ; laden
JP CCP ; Systemstart
```

HEFT  
26  
Januar  
1989

Die Einstellungen des SYSFLAG kann z.B. noch einfach mit Hilfe des Befehls `FOHMER` über zwei die dort vorhandenen vier USER-Profile realisiert werden.

Diese Befehle führen ein Unterprogramm aus, das in `FOHMER` auf folgenden Adressen eingetragen werden muß:

```

UP1 0142-0147H ; Jeweils 2 Befehle, müssen mit RET abgeschlossen
UP2 0148-014FH ; werden!
UP3 0150-0157H ;
UP4 0158-015FH ;

```

In folgendem Beispiel wird UPI zur Einstellung Warmstart, ohne nachladen CCP/BDOS und URC zur Einstellung mit nachladen verwendet.

```

0140H LD A:00 ; kein nachladen, DRIVE A ist aktuelles Laufw.
LD (SYSFLAG),A ;
RET

```

URC

```

0148H LD A:01 ;Nachladen CCP und BDOS von DRIVE B
LD (SYSFLAG),A ;
RET

```

Das vorgestellte Verfahren ist seit einem Jahr in meinem BIOS integriert und hat sich bewährt.

Frank-Michael Schuber

### BDOS-Replacement, was'n das ?

Es dürfte sich ja allgemein rumgesprochen haben, daß CP/M inzwischen etwas betagt ist und auch etliche Kanten und Ecken hat. Einige fleißige Leute haben sich nun gedacht, selbst ist der Mann und haben für Abhilfe gesorgt. Das Original von Digital Research wurde natürlich in 8088-Assembler geschrieben und damit bringt man halt relativ wenig in den rund 3 1/2 KB unter, die dem BDOS zur Verfügung stehen.

Kurzer Einschub für die 'nur NEWDOSler': CP/M besteht aus drei Teilen. Ganz unten auf Hardwareebene ist das BIOS, das Basic In Out System. Dieses ist das direkte Interface zum jeweiligen Rechner und daher systemspezifisch. Es stellt die elementarsten Funktionen wie z.B. Zeichen ausgeben usw. Darüber kommt das BDOS, das Basic Disk Operating System. Dieses ist auf jeder Maschine gleich und führt seine Funktionen durch CALLS ins BIOS aus, stellt also eine standardisierte Schnittstelle für alle Ein-/Ausgaben dar. Als letztes kommt der CCP, der Command Console Processor, der die Schnittstelle zum Benutzer darstellt. Mit dem 'unterhält' man sich, wenn man auf der Betriebssystemebene Befehle eintippt.

Ein Weg, das BDOS zu verbessern war nun, einen leistungsfähigeren Befehlssatz, sprich den des Z80, zu nehmen. Damit gehört das Control-C nach jedem Diskettenwechsel ebenso der Vergessenheit an, wie die geheimnisvollen Fehlermeldungen. Außerdem wird eine Echtzeituhr unterstützt, damit man genau weiß, wann ein File erstellt oder zuletzt bearbeitet wurde. Weiterhin kann man Files 'public' machen, d.h. sie sind aus allen User-Bereichen zugänglich. Damit kann man z.B. den Compiler in einen Bereich und die Source-Files nach Anwendung geordnet in anderen Bereichen unterbringen.

Der Autor eines der vielen Replacements war so freundlich und hat einige gegenübergestellt:

Name	CP/M	Z8DOS+	Z8BDOS	P2DOS21	DOS+25	SUPERDOS
Author	Digital Research, Inc.	Echelon, Inc.	Carson Wilson	H.A.J. Ten Brugge	C.B. Falconer	Benjamin Ho
Derivation	Unknown	Unknown	P2DOS, SUPERDOS	Unknown	P2DOS	P2DOS
Time stamps	No	No	(*),C,U,A	C,U	C,U,A	C,U
Disks auto-login	No	Yes	Yes	No	No	Yes
Archive	No	Yes	Yes	Yes	Yes	No
Public files	No	Public user areas	F2 attribute	F2 attribute	System files at AB:	F2 attribute
Get/Use stamps	No	No	Yes	No	No	No
Get/Set time	No	No	Yes	Yes	Yes	Yes
Error messgs.	Cryptic	Clear	Legible, give function & file			
Return current DMA	No	Yes	No	No	Yes	No
Wheel-protect files	No	Yes	No	No	No	No
Set/res warm boot	No	Yes	No	No	No	No
Source code	No	No	Yes	Yes	\$50	Yes
Approx. price	\$20	\$60	free	free	free	free

(\*), C = Create, U = Update, A = Last Access

Ich arbeite zur Zeit mit dem Z8BDOS und bin sehr zufrieden. Die Sources habe ich von Gerald Schröder bekommen, der sie von Rüdiger Sörensen hat, der sie frisch über den großen Teich geholt hat. Der Aufwand ist nicht groß und lohnt sich allemal, selbst wenn es nur wegen dem Ctrl-C ist.

Alexander Schaid

## Es war einmal ein Datum

Die meisten BDOS-Replacements unterstützen (endlich?) auch eine Echtzeituhr. Damit kann man sich entweder ausgeben lassen, wie lange man noch bis zum Kaffeezeit hat, oder alle Einträge im Directory erhalten einen 'Zeitstempel', der einem sagt, wann das File erstellt wurde, oder wann zuletzt darauf zugegriffen wurde. Das ist vor allem dann nützlich, wenn man nicht mehr weiß, welches das allerletzte Update war, wo man den nobelpreisverdächtigen Geistesblitz hatte.

Schön und gut dachte ich, als ich mit Helmut Bernhards Schaltplan so eine Zwiebel zusammengestrickt hatte, das schwierigste ist jetzt erledigt, aber es kommt ja erstens anders und zweitens als man denkt. Der RTC5032 gibt auf Anfrage seine Daten im BCD-Code her, was aber kein großes Hindernis ist. Wenn man die Register (siehe Arnulfs Artikel 'Real Time Black Box') nacheinander abklappert hat man schließlich das Datum in der Form 'Jahr, Monat, Tag, Stunde, Minute, Sekunde'. Leider scheint man in Amerika, wo die Autoren nunmal sitzen, nicht viel von dieser Notation zu halten. Das Z80DOS (und auch andere) wollen das Datum als 'Anzahl der Tage seit dem 1.1.78'. Fragt mich nicht, wie man auf sowas kommt, ich weiß es nicht. Zum Glück waren auf den Disketten auch Programme, mit denen man die Uhr stellen kann. Logischerweise mußten die eine Routine zum Urechnen haben, damit man das Datum in der 'normalen' Form eingeben kann. Das unten abgedruckte Listing ist nun der eigentliche Teil, der die Urechnung bewerkstelligt. Als Ergebnis bekommt man schließlich die Zahl im Register BC.

Arnulf möge mir verzeihen, daß ich das Listing nicht kommentiert habe, aber mich hatte im Moment nur interessiert, DASS es geht und nicht WIE. Wenn mich mal der Ehrgeiz packt, werde ich den Algorithmus vielleicht mal auseinander nehmen, aber auch nicht früher. Wer sich für die Datumsrechnung mehr interessiert, der sei herzlich eingeladen, sich dazu zu äußern.

```

; Berechnung der Tage seit dem 1.1.1978
;
; Eingabe : HL=Jahr (1978..2157)
;           DE=Monat (1..12)
;           BC=Tage (1..31)
; Ausgabe : BC=Tage erster Tag (00001) : Sonntag 01.01.1978
;           letzter Tag (65535) : Sonntag 05.06.2157
;

```

```

DAYS: LD B,0
      PUSH DE
      LD DE,1978
DAYS0: OR A
      SBC HL,DE
      ADD HL,DE
      JR Z,DAYS1
      PUSH HL
      LD HL,365
      ADD HL,BC
      LD B,H
      LD C,L
      POP HL
      EX DE,HL
      CALL LEAPYR
      EX DE,HL
      INC DE
      JR NZ,DAYS0
      INC BC
      JR DAYS0

```

29

```

DAYS1: POP DE
      PUSH HL
      LD HL,DM
      LD D,1
DAYS2: LD A,E
      CP D
      JR Z,DAYS4
      LD A,C
      ADD A,(HL)
      LD C,A
      JR NC,DAYS3
      INC B
DAYS3: INC HL
      LD A,D
      INC D
      CP 2
      JR NZ,DAYS2
      EX (SP),HL
      CALL LEAPYR
      EX (SP),HL
      JR NZ,DAYS2
      INC BC
      JR DAYS2
DAYS4: POP HL
      RET
;
; Korrektur für ein Schaltjahr
;
; Eingabe : HL = Jahr
; Ausgabe : Z = 1 Korrektur notwendig
;           Z = 0 keine Korrektur nötig
;
LEAPYR: LD A,L
      AND 3
      RET NZ
      PUSH HL
      PUSH DE
      LD A,0FFH
      LD DE,100
      OR A
LEAPY0: INC A
      SBC HL,DE
      JR NC,LEAPY0
      ADD HL,DE
      LD H,A
      LD A,L
      OR A
      JR NZ,LEAPY1
      LD A,H
      AND 3
      JR LEAPY2
LEAPY1: XOR A
LEAPY2: POP DE
      POP HL
      RET
DM: DEFB 31,28,31,30,31,30,31,31,30,31,30,31 ;Tage pro Monat

```

Alexander Schaid

HEFT  
276  
Januar  
1988

30



© Dr. Spiderman

hexadezimal (!) angeben. Hexadezimaler wird wie bei einem Assembler durch ein angehängte 'H' gekennzeichnet. Das sieht dann so aus:

12345	einfache Genauigkeit (16bit)
12345333.	doppelte Genauigkeit (32bit)
12FFH	einfache Genauigkeit (16bit Hex)
1FF088DD.H	doppelte Genauigkeit (32bit Hex)

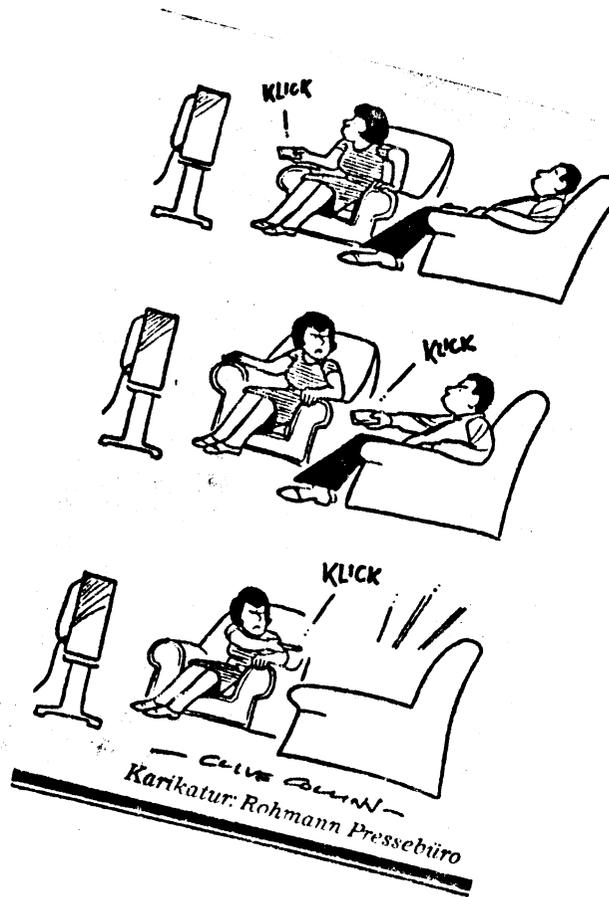
Eine Hex-Zahl in 32bit sieht etwas komisch aus, aber so ist es nun einmal. Dazu passend gibt es dann auch die entsprechenden Zuweisungsanweisungen :=(D) und ?(D). Array's mit 32bit gibt es nicht. Das wäre Platzverschwendung. Hier sind einfach die doppelte Menge an Bytes zu reservieren. Die Stringdeklaration wurde auch überarbeitet und funktioniert jetzt hoffentlich fehlerfrei. In der alten Version wurde bei fehlendem zweiten (!) alles was sonst noch kam als String vereinnahmt. Das ist zwar simpel, aber kein guter Stil. Jetzt wird dem Programmierer seine Vergeßlichkeit um die Ohren geschlagen. Variablen und Konstanten für 32 Bit werden mit DVAR bzw. mit dem Punkt am ende der Zahl definiert. Bei der Konstanten erkennt der Compiler automatisch um was es sich handelt (DPL ist dann temporär ungleich Null. S. a. weiter unten). Die Arithmetik habe ich nur auf  $32 * 16$  Bit bei der Punktrechnung erweitert. Der Aufwand ist ganz beachtlich, außerdem ist das Runtimemodul mit den ganzen Erweiterungen und Umbauten jetzt gute 3.5KByte im Umfang. Für meinen Geschmack eigentlich zu umfangreich. RPML sollte eigentlich eine 'kleine Sprache' sein, zumal es mich bei den PASCAL-Systemen immer mächtig geärgert hat, daß auch bei kleinsten Programmen stets der 12K große Runtimekern vorhanden war. Einen Vermutstropfen will ich nicht verschweigen: Die Zahlenausgabe ist etwas langsamer geworden, als beim 'Original'. Ursache ist eben jene erweiterte Arithmetik. Die normalen Multiplikations- und Divisionsroutinen sind rausgeflogen und durch die neuen ersetzt. Was bedeutet, daß die  $16 * 16$  Bit Befehle genaugenommen auch mit  $32 * 16$  Bit Genauigkeit ablaufen. Nur wird hier Vorzeichenrichtig erweitert und dann gerechnet. Eine entsprechende Umwandlungsfunktion von Single auf Double Precision ist dementsprechend eine Selbstverständlichkeit. Das sieht dann so aus: ...DOUBLE... und fertig ist die Laube. Das Vorzeichen kann sehr einfach mit folgender Sequenz übertragen werden: ... OVER SIGN oder mit ... ROT DUP >R -ROI R> DSIGN für doppelt genaue Werte. Übertragen wird stets das Vorzeichen der zweitobersten Zahl auf die zuoberste liegende Zahl auf dem Stack. Wer will kann auch einen Dezimalpunkt in die Ausgabe einbauen. Hierzu ist die Systemvariable DPL mit einem Wert größer Null zu laden. Der hier abgelegte Wert bezieht sich auf die Stellen nach dem Komma! DPL wird nicht nach jeder Ausgabe zurückgesetzt. Dies muß explizit durch das Anwenderprogramm erfolgen. Eine Ausgabe in einem Printfeld ist über die Systemvariable FLD möglich. Dazu wird dort die Feldbreite eingeschrieben. Paßt die Zahl nicht in das Feld hinein, wird FLD automatisch auf 32 (!) gesetzt. Soetwas kann immer dann passieren, wenn die Systemvariable BASE auf den Wert 2 für Ausgabe im Binärsystem gesetzt ist. Hier sollte man sich also vorher gut überlegen, was man will. Die Änderung wird nicht automatisch zurückgenommen! (Doppelt genaue Zahlen werden selbstverständlich mit PRINTD ausgegeben).

Hallo RPML-Freunde !, Wie sicherlich noch keinem bekannt ist, gilt es von diesem tollen Hochsprachencompiler inzwischen die Version 2.1. Einige Punkte wurden von mir verbessert oder abgeändert. So erscheint nun nicht mehr bei jedem 'NEW' oder bei einem Fehler diese lästige Copyright Meldung. Wichtiges hat sich jedoch im Bereich der Zahlen und im Deklarationsstil getan. Hier können jetzt 32bit Variablen definiert werden. Konstanten lassen sich ebenfalls in 32bit dezimal und

Tschuß

*Ruh*

Zu guter letzt noch etwas aus Opa's Kofferkiste: Eine Zahlenkonvertierungs-Routine, die auch vor Hexadezimal- und 32bit-Werten nicht zurückschreckt. Das Stück stammt aus der RPML-Version 2.1 Das Programm ist in der Lage, mit jeder Zahlenbasis zu arbeiten, aber wie schon beschrieben begnüge ich mich mit den beiden üblichsten Formen. Wenn diese Beschränkung nicht gefällt, kann sie ja herausnehmen. Darüber hinaus kann sich das Programm den Ziffernstring selbständig aus einem Eingabestring herausfischen. Näheres sind aus dem DECLARE-Teil am Anfang des Listings zu ersehen. Dort sind einige Teststrings angegeben. Das 'DVAR' sollte nicht verwirren, da es sich hier um die Deklaration einer 32bit-Variablen handelt. Die Vers. 2.0 kennt diese Anweisung nicht. Hier muß stattdessen mit 'ARRAY name 2' der nötige Speicherplatz reserviert werden. Das Wandlungsergebnis befindet sich zusammen mit einigen Flags (siehe Kommentare im Listing) auf dem Stack und kann mit 'D.' und 'S.' ausgegeben werden. 'D.' bedeutet 'gebe eine doppelt genaue Zahl auf dem Bildschirm aus', 'S.' bewirkt das Gleiche für einfachgenaue Zahlen.



```

*****
**          N U M B E R / R P M          **
*****

```

DECLARE

```

STRING $1 '11111'
STRING $2 '22222 '
STRING $3 ' 33333'
STRING $4 ' 44444 '
STRING $5 '55555 '
STRING $6 '66666.'
STRING $7 '1000H'
STRING $8 '10X00'
STRING $9 '100 XYZ'
DVAR  NUM

```

DEND

```

CALL $DIGIT ; ( (AC)<0 -> gesetztes carry)
D6 30 ; $DIGIT:SUB '0'
D8 ; RET C
FE 0A ; CP 9+1
38 06 ; JR C,DIG11
D6 07 ; SUB 7
FE 0A ; CP 10
D8 ; RET C
B9 ; CP C ; < BASE ?
3F ; DIG11: CCF
C9 ; RET
CEND

```

CODE PARSE

```

; (adr1 -> adr2 cnt flag)
E1 ; PARSE: POP HL ; (ADR1 -> ADR2 CNT FLAG)
46 ; LD B,(HL) ; LAENGE HOLEN
04 ; INC B ; FALLS LAENGE = 0 !
23 ; PAR1: INC HL
05 ; DEC B
28 07 ; JR Z,EPAR1
7E ; LD A,(HL)
FE 21 ; CP ' '+1
38 F7 ; JR C,PAR1
18 07 ; JR PARSE2
E5 ; EPAR1: PUSH HL ; POINTER
0E 00 ; LD C,00
C5 ; PUSH BC ; COUNT
C5 ; PUSH BC ; FLAG = TRUE BEI INHALT
18 17 ; JR PARSE3 ; FLAG = FALSE KEIN INHALT
0E 00 ; PARSE2: LD C,00 ; LAENGENZAehler SETZEN
E5 ; PUSH HL ; POINTER AUF STRINGANFANG
0C ; PAR2: INC C
78 ; LD A,B
B7 ; OR A
28 07 ; JR Z,EPAR2-1
7E ; LD A,(HL)
23 ; INC HL
05 ; DEC B
FE 21 ; CP ' '+1 ; ZEICHEN <= BLANK ?

```

```

30 F4 ; JR NC,PAR2
0D ; DEC C ; TRENNZEICH. NICHT MIT-
06 00 ; EFAR2: LD B,00
C5 ; PUSH BC
01 FF FF ; LD BC,OFFFH
C5 ; PUSH BC
CEND ; PARSE3: JP NEXT

```

CODE ?BASE

```

; cnt (adr2 cnt -> adr2 cnt base)
D1 ; QBASE: POP DE ; cnt
E1 ; POP HL ; adr2
E5 ; PUSH HL
D5 ; PUSH DE
19 ; ADD HL,DE ; POINTER AUF LETZTES STRING-
01 10 00 ; LD BC,0010H ; ... ZEICHEN ERRECHNEN
2B ; DEC HL ; SONST ZU WEIT HINTEN !
7E ; LD A,(HL)
FE 48 ; CP 'H'
E8 02 ; JR Z,BEND
0E 0A ; LD C,0AH
C5 ; BEND: PUSH BC
CEND

```

CODE (NUMBER)

```

; (adr1 adr2 cnt base -> d1 dpl true)
; (adr1 adr2 cnt base -> adr1 false)
C1 ; NUMBER: POP BC
D9 ; EXX
C1 ; POP BC
06 FF ; LD B,OFFH ; B' = DPL
D9 ; EXX ; C' = CNT
DD E1 ; POP IX ; DEHL = D1
11 00 00 ; LD DE,0000 ; BC = BASE
21 00 00 ; LD HL,0000 ; IX = ADR2
DD 7E 00 ; NUM1: LD A,(IX) ; HOLE EIN ZEICHEN
CD $DIGIT ; CALL $DIGIT
38 2A ; JR C,TSTP ; KEINE ZIFFER
F5 ; PUSH AF
CD $MULT ; CALL $MULT
F1 ; POP AF
C5 ; PUSH BC ; NEUE ZIFFER EINFUEGEN
4F ; LD C,A
09 ; ADD HL,BC
EB ; EX DE,HL
0E 00 ; LD C,00
ED 4A ; ADC HL,BC
EB ; EX DE,HL
C1 ; POP BC
D9 ; NUM2: EXX
CB 78 ; NUM21: BIT 7,B ; ZAHL MIT DEZ.-PUNKT ?
20 01 ; JR NZ,NOPNT
04 ; NUM22: INC B ; DPL = DPL + 1
DD 23 ; NOPNT: INC IX ; ADR2 = ADR2 + 1
0D ; DEC C ; CNT = CNT - 1
D9 ; EXX
20 DD ; JR NZ,NUM1
33 ; NUM3: INC SP ; ADRI VERNICHTEN DA KON-
33 ; INC SP ; VERTIERT WERDEN KONNTE
E5 ; PUSH HL ; D1 AELGEN

```

```

D5 ; PUSH DE
D9 ; EXX
48 ; LD C, B
06 00 ; LD B, 00
C5 ; PUSH BC ; DPL AUF DEN STACK
01 FF FF ; LD BC, 0FFFFH ; TRUE = ERFOLG GEHABT
C5 ; PUSH BC
18 1C ; JR NUM4 ; FERTIG !
DD 7F 00 ; TSTP: LD A, (1X)
FF 2F ; CP '.' ; DEZIMALPUNKT ?
20 08 ; JR NZ, IS1H
D9 ; EXX
00 ; NOP
CB 78 ; BIT 7, B ; MEHR ALS EIN DEZI-
28 0B ; JR Z, ERROR ; MALPUNKT
13 DB ; JR NUM22
FE 48 ; TSTH: CP 'H' ; BASE = HEX ?
20 05 ; JR NZ, ERROR ; WENN NICHT -> ERROR
D9 ; EXX
0D ; DEC C
D9 ; EXX
28 D9 ; JR Z, NUM3 ; 'H' STEHT NICHT HINTEN !
01 00 00 ; ERROR: LD BC, 0000
C5 ; PUSH BC ; FLAG = FALSE, KEIN ...
CEND ; NUM4: JP NEXT ; ... ZIFFERSTRING !

```

```

PROGRAM NUMBER ; (adr1 -> d1 true) wenn Zahl
DUP PARSE ; (adr1 -> adr1 false) keine Zahl
IF

```

```

SWAP DUP
?(B) 45 =
DUP >R
IF
INC SWAP DEC
ELSE
SWAP
ENDIF
?BASE (NUMBER)
IF
DPL :=
R> DUP DSIGN
1KUB
ELSE
TRUE DPL :=
R> DROP
FALSE
ENDIF
ELSE
FALSE
ENDIF
END

```

```

PROGRAM D.
NUM := (D)
NUM PRINTD
END

```

```

PROGRAM S.
NUM :=
NUM PRINT
END

```

```

;***** E N D *****

```

Der Computer gilt als neue industrielle Revolution. Als etwas, das in den nächsten Jahren unser Leben so total verändern wird wie noch keine Erfindung zuvor.

Tatsächlich haben die Minichips unsere Welt mehr durcheinandergebracht als Präservative, Trockenmilch und Hühnerhälften zusammen. Täglich können wir die wunderbaren Erleichterungen spüren, die durch die neue Technik in allen Lebensbereichen eingeführt werden:

- Ist Ihr verstorbener Opa in diesem Jahr auch schon dreimal zur Musterung vorgeladen worden?
- Stehen auf Ihrer Telefonrechnung seit neuestem Gespräche nach Addis Abeba?
- Soll Ihr Hund sich als Schöffe zur Verfügung halten?

- Werden auf Ihrem Konto öfter mal die Versicherungsbeiträge doppelt abgebucht?
- Bekommen Sie immer häufiger mit der Post Angebote über Sprachkurse, Sexfilme oder Parteispenden?
- Haben Sie schon mal einen Flug nach Detmold gebucht und ein Ticket nach Dallas bekommen?

Wenn Sie solche oder ähnliche Erlebnisse hatten, können Sie sich glücklich schätzen, dann haben Sie etwas vom Hauch der »industriellen Revolution« gespürt. Denn schuld an der ganzen Misere ist kein anderer als Kollege Computer.

In diesem Punkt sind sich die Wissenschaftler ausnahmsweise einig: Die elektronischen Pfliffküsse sind nicht nur die größten Datenverwalter der Geschichte, sondern sie sind auch die Weltmeister, wenn es darum geht, irgendeine Sache gründlich zu vermässeln.

Nach dem alten Motto: »Wer viel weiß, kann auch viel vergessen« sind schon so einige Datenbanken in den düsteren Keller des elektronischen Vergessens abgestürzt.

Da hat die Maschine dem Menschen wirklich viel Ärger abgenommen. Wenn früher ein Mitarbeiter Fehler machte, konnte der Chef wenigstens noch den Zeigefinger heben und mit Gehaltskürzung drohen. Beim Computer nützt nicht einmal mehr die Versetzung in die Poststelle.

Dabei kann der Schnelldenker ungleich mehr Schaden anrichten als sein lebendiger Kollege. Wenn man nur an die täglichen Fehlalarme denkt, mit denen die Pentagon-Computer seit Jahren vergeblich versuchen, den dritten Weltkrieg auszulösen.

Kurz, diese geniale Erfindung aus Chips und Kabel hat sich inzwischen in der zivilisierten Welt unentbehrlich gemacht.

Es ist also völlig falsch, diesen Gipfel aller menschlichen Errungenschaften zu verteufeln: Betrachten wir doch nur einmal die wunderbaren Vorteile, die wir dem täglichen Umgang mit unseren elektronischen Helfern in Büro und Wohnstube

verdanken: heftige, ununterbrochene Kopfschmerzen, Augenflimmern, verkrampte Finger und nicht zu vergessen die reizenden Piepstöne, die seit neuestem unsere Umgebung erfüllen. Wer wollte da noch auf den Computer verzichten?

KLUG ODER WARRD

Besser erkennbare Darstellung von Text und HRG auf dem Bildschirm

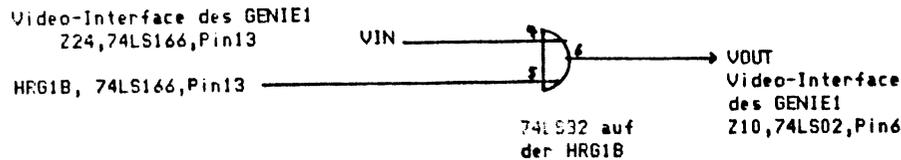
Helmut Bernhardt

Durch die Erzeugung synchroner Videosignale des normalen Text-Video-Interface und der HRG1B konnten diese beiden Signale gemischt werden, so daß beide Bildschirm-Informationen gleichzeitig dargestellt werden. Das ist einerseits ärgerlich, wenn z.B. BASICs READY ständig nach der Erzeugung eines Bildes das Werk verunziert; es kann andererseits aber auch recht nützlich sein, wenn man zu Diagrammen erläuternde Texte anzeigen möchte. Weniger gelungen ist aber die Art, wie die Video-Signale gemischt werden. Das wird auf der HRG1B durch OR-Verknüpfung der beiden Signale gemacht, was zur Folge hat, daß in hellen Flächen der HRG-Anzeige kein Text zu erkennen ist. Die OR-Verknüpfung bewirkt, daß ein Bildpunkt immer hell gesteuert wird, wenn entweder der entsprechende Punkt in der HRG oder der entsprechende Punkt eines dargestellten Textzeichens oder auch beide angezeigt werden.

Eine XOR-Verknüpfung der beiden Video-Signale steuert einen Bildpunkt nur dann hell, wenn entweder der HRG-Punkt oder der Punkt eines Textzeichens angezeigt wird. Wenn allerdings beide Video-Signale gleichzeitig einen Bildpunkt hell darstellen wollen, wird dieser dunkel gesteuert. Das hat zur Folge, daß Texte vor hellem HRG-Hintergrund dunkel dargestellt werden. Selbst wenn ein Textzeichen zum Teil vor einem hellen und zum Teil vor einem dunklen Hintergrund dargestellt wird, läßt es sich vollständig erkennen, weil sein Farbwert immer das Gegenteil der Hintergrundfläche ist.

Damit können dann nicht nur Text und HRG sehr viel besser gleichzeitig dargestellt werden; der Umstand läßt sich bei geschickter Programmierung der HRG auch dazu nutzen, inverse Zeichen darzustellen. Es müssen einfach immer die entsprechenden Felder, in denen im Text-Bildschirm ein Zeichen dargestellt wird, in der HRG hell gesteuert werden. Das bedeutet, daß pro auszugebendem Textzeichen auch 12 Bytes im HRG-Speicher auf FFH zu setzen sind. Die Zuordnung von Text-Video-RAM-Adressen zu HRG-Adressen läßt sich aus früheren Beiträgen von Arnulf Sopp zu diesem Thema entnehmen. Etwas aufwendiger wird es beim Scrollen von Texten, wenn Teilbereiche des Textes invers dargestellt sind. Dann genügt es nicht, nur mit LDIR den ganzen Text-Bildwiederholpeicher um 64 Adressen zu verschieben, dann müssen auch die hellen Felder der HRG um einen entsprechenden Offset verschoben werden, was nicht mehr mit einem einfachen LDIR zu erledigen ist.

So aufwendig das in Soft auch sein mag, in Hard ist die Änderung auf XOR-Verknüpfung der Video-Signale recht banal. Die bisherige OR-Verknüpfung sieht so aus:



37

Wenn anstelle des OR-Gatters des 74LS32 ein XOR-Gatter des pinkompatiblen 74LS86 genommen wird, ist die Sache erledigt. Leider werden von dem 74LS32 noch andere OR-Gatter für andere Zwecke verwendet, so daß man nicht einfach das IC

KLUG ODER WARRD

gegen ein 74LS86 austauschen kann. Man kann aber das 74LS86 mit den Pins 4, 5, 7 und 14 huckepack auf das 74LS32 löten, den Pin 6 des 74LS32 abkneifen und stattdessen den Pin 6 des 74LS86 mit dem Pin 15 des Sockels für den Anschluß des 24poligen Flachkabels verbinden.

So einfach ist das.

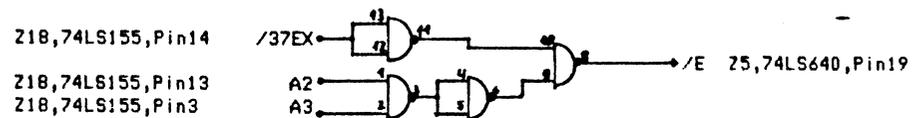
Endlich Schluß mit den Problemen beim EXP1-Floppy-Controller von RB-Elektronik

Helmut Bernhardt

Mehrmals hatte ich schon Probleme mit EXP1-Floppy-Controllern. Nach versuchsweisem Auswechseln einiger ICs war der Spuk dann meistens vorbei - leider aber immer nur für begrenzte Zeit. Inzwischen habe ich einen grundsätzlichen Fehler in der Schaltung des EXP1 erkannt, der dafür verantwortlich ist:

Die Freigabe des invertierenden Datentreibers 74LS640 wird durch AND-Verknüpfung der Signale /37ECRD (Z18, 74LS155, Pin4) und /37ECWR (Z18, 74LS155, Pin12) erzeugt. Das hat zur Folge, daß bei Schreibzugriffen auf den Floppy-Controller WD1x91 der Treiber erst im gleichen Moment freigegeben wird, wie das /WR-Signal am FDC mit seiner negativen Flanke die Daten in den Controller pustet. Durch die Gatterlaufzeit bei der Freigabe des Treibers können hinter dem Treiber noch gar keine gültigen Daten anliegen. Es ist erstaunlich, daß das Board überhaupt funktioniert hat.

Abhilfe schafft ein Freigabesignal für den Treiber, das unabhängig von /RD und /WR ist. Dieses läßt sich mit einem 74LS00 folgendermaßen erzeugen:



Das dafür nötige 74LS00 wird mit den Pins 7 und 14 auf irgendein anderes DIL14-IC (nur nicht das 74LS92) huckepack-gelötet und mit den entsprechenden Pins von Z18 und Z6 zu verdrahtet. Der Pin 8 von Z15, 74LS08 wird aus der Fassung gebogen.

Januar 1989

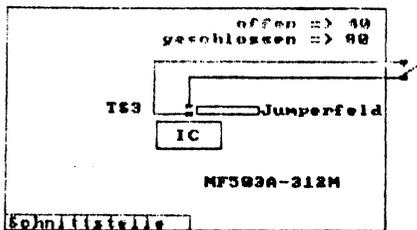
38

### 80/40-Trackumschaltung für Mitsubishi MF503A-312M

39

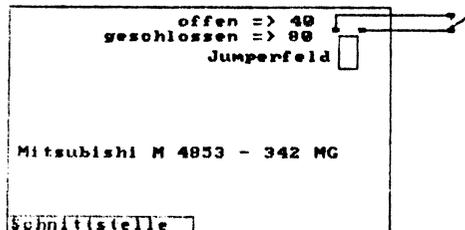
Wieder einmal ist mit ein auf 40 Spuren umschaltbares 80'er Laufwerk über den Weg gelaufen und wieder will ich kurz beschreiben, wie sich die Umschaltung bewerkstelligen läßt. Schließlich soll jeder, dem ein solches Laufwerk einmal unterkommt oder angeboten wird in der Lage sein, alle Fähigkeiten des Laufwerks für sich auszunutzen.

Wie sich schon der Überschrift entnehmen läßt, handelt es sich um ein 5 1/4"-Laufwerk der Firma Mitsubishi mit der Bezeichnung MF503A-312M. Es zeichnet sich durch Laufruhe und Betriebssicherheit aus (Jörg Brans ist schon länger im Besitz eines solchen Diskdrives). Nachteilig ist nur, daß das Laufwerk ein Auslaufmodell und damit wohl nicht mehr lange zu haben ist (außer eventuell Gebrauchtgeräte). Ein paar Drives gibt es allerdings noch, Bezugsadresse bei Jörg Brans!



Zunächst muß eine Leiterbahn zwischen den Lötstellen "TS3" getrennt werden. Das bewerkstelligt man am besten mit einem sehr scharfen Messer. Wie der nebenstehenden Zeichnung zu entnehmen ist, wird dann ein einfacher Schließer an die beiden Lötstellen angeschlossen. Dabei sollte man einen Lötkolben möglichst kleiner Leistung benutzen, da die

Punkte sehr nahe bei einem in SMD-Technik ausgeführten Microcontroller liegen. Etwas Sorgfalt ist also angebracht.



Und weils so schön war, gleich noch eine Umschaltung. Dabei handelt es sich um ein Mitsubishi-Laufwerk mit der Typenbezeichnung M4853-342MG. Laut Jörg Brans ist es das Vorläufermodell des oben beschriebenen Drives. Die Umschaltung funktioniert im Prinzip genau wie oben beschrieben, nur die Lötstellen sind an einer anderen Stelle. Wo, könnt Ihr nebenstehender Zeichnung entnehmen.

Hartmut Obermann

### Entfernung der Kassettenschnittstelle von der I/O-Karte des Genie IIS

40

Vor einiger Zeit sah ich mir den Schaltplan der I/O-Karte meines Genie IIS an und mußte dabei feststellen, daß sich gut die Hälfte der Zeichnung auf die Kassettenschnittstelle bezog. Diese habe ich noch nie benutzt und werde es nie tun. Außerdem hängt an diesem Bus auch noch die Lautsprecher-Ansteuerung; da ich keine Lust habe, den Rechner zum piepsenden Neuzötter zu machen, ist dieser Teil auch überflüssig.

Welchen Sinn hat die Aktion? Ich hatte gerade eine Erweiterung der I/O-Karte geplant (s. Artikel "Memory mapped-I/O (=) Port-I/O") und mußte dazu ein 74LS157 auf der Karte unterbringen. Das wäre normalerweise nur im Huckepack-Verfahren möglich gewesen, wobei ich die neuen Strippen oben vom IC-Pin zu IC-Pin hätte ziehen müssen. Das wäre ein undurchdringlicher Dschungel von verknüpften ICs geworden. Durch die Entfernung der Kassettenschnittstelle schaffte ich mir einen freien IC-Sockel, in den das 74LS157 paßte. Somit konnte ich alle Verbindungen auf der Lötseite herstellen, was auch einen Dschungel ergibt, wobei aber die ICs frei zugänglich bleiben. Außerdem lassen sich jetzt durch die Entfernung einer Menge Widerstände u.ä. die Leiterbahnen sehr viel besser verfolgen. Natürlich können Euch andere Gründe zu der Freisetzung ungenutzter Elektronikteile veranlassen.

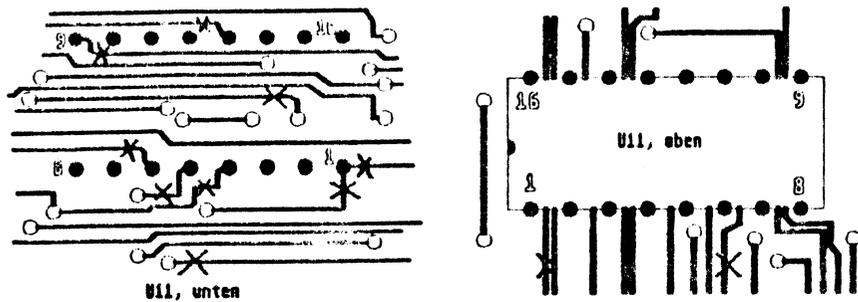
Im Einzelnen können ohne Verlust der anderen Leistungen des Systems folgende Elemente von der I/O-Karte entfernt werden:

- ICs : U11; U12; U2 Pins 6,7,9-12; U18 Pins 4-6,8-10
- Widerstände : R 4-19
- Kondensatoren: C 2-5,16
- Dioden : D 1-5
- Transistoren : T 1,2
- Reed-Relais : RE 1
- Stecker : S 1,4

Bis auf die Stecker habe ich wirklich alles entfernt bzw. die entsprechenden Pins von U2 und U18 herausgebogen. Übrigens würde ich sagen, daß sich mit den freien Pins von U2 z.B. eine zweite Centronics-Schnittstelle basteln ließe. Das wäre eine sinnvollere Anwendung als die Kassettenschnittstelle.

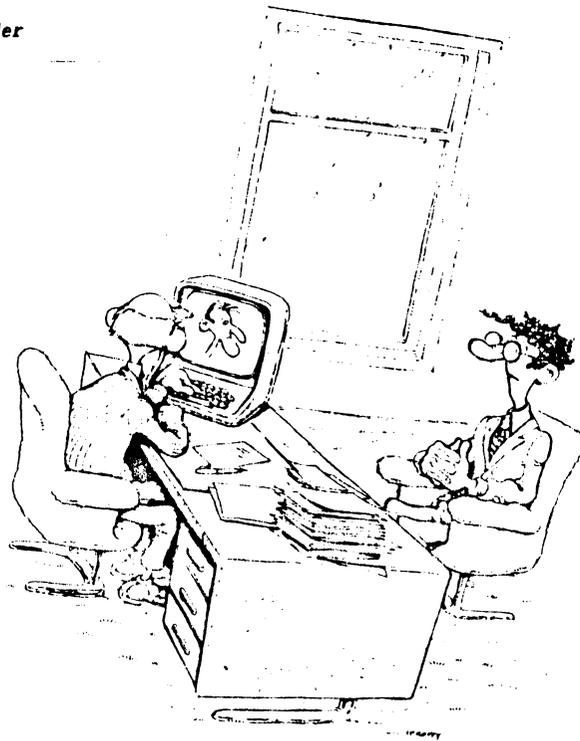
Da ich die Fassung von U11 für meine Zwecke nutzen wollte, mußten alle Verbindungen gekappt werden (bis auf GND und +5V). Seht Euch dazu die folgende Liste und die Bilder an:

Signal	Pin von U11	wo gekappt (oben=Bestückungsseite)
SR155*	1	unten, zweimal direkt am Pin
FF001*	9	unten, direkt am Pin
PH0	2	unten, an der Durchkontaktierung darüber
PH1	6	unten, direkt am Pin
MOTON	10	unten, an Durchkontaktierung Richtung U10
32C*	14	an der Steckerleiste, Anschluß A8
D0	4	unten, direkt am Pin
D1	5	unten, direkt am Pin
D2	12	oben, in Richtung U10
D3	13	oben, in Richtung U10, und unten, direkt am Pin



Leider mußten an zwei Stellen (SYSRES\* und D3) zwei Leitungen unterbrochen werden, d.h. eine führte zu U11 und eine wieder weg. Diese unterbrochenen Verbindungen müssen überbrückt werden: Für SYSRES\* von A14 (Steckerleiste) oder U8,1 (= Pin 1) oder U1,1 zu U2,3. Für D3 von C19 (Steckerleiste) zu U4,12.

Gerald Schröder



Tut mir leid - Sie entsprechen nicht unserem Anforderungsprofil.

#### Memory Mapped I/O ( $\Leftrightarrow$ ) Port I/O beim Genie IIs

Beim TRS-80 und den meisten kompatiblen Geräten wird die Ein/Ausgabe, namentlich für Tastatur, Bildschirm, Drucker und Disk, über eine Memory Map gemacht, die bei 3700h bis 4000h liegt. Das ist mit Newdos zusammen ganz OK so, denn der Speicher wird von den Programmen erst ab 5200h benutzt. Wenn man den Rechner aber für CP/M umrüstet, gibt es Probleme, denn hier benutzen die Programme den Speicher ab 100h. Nun liegt die Memory Map plötzlich mitten im RAM runter und stört ziemlich. Beim TRS-80 gibt es eine Erweiterung, die diesen Bereich an das obere Speicherende verschiebt, wo er nicht mehr so sehr stört. Bei anderen Systemen, so auch beim Genie IIs, wird einfach die Memory Map gegen RAM ausgetauscht und nur eingeschaltet, wenn eine Ein- oder Ausgabe erfolgen soll.

Dies Umschalten birgt einige Probleme: Das umschaltende Programm darf nicht in diesem Bereich laufen, sonst dreht es sich selbst den Hahn ab; der Stackpointer könnte gerade in diesen Bereich zeigen, da aber beim Umschalten das RAM dort verschwindet, muß er umgestellt und später restauriert werden. Außerdem dürfen nun keine Interrupts mehr stattfinden, denn auch die könnten in den gesparten RAM-Bereich führen. Wenn nun ineinander verschachtelt mehrere Routinen auf die Memory Map zugreifen wollen, ist ein ganz schöner Verwaltungsaufwand erforderlich, um festzustellen, wo der SP gerade steht, ob er gerettet werden soll, ob die Memory Map eingeschaltet werden muß, ob die Memory Map wieder eingeschaltet und der SP wieder restauriert werden darf.

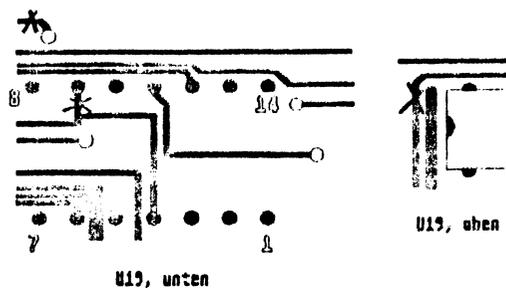
Was wird nun von der Memory Map eigentlich gebraucht? Der Bildschirm-Bereich ist oft überflüssig, denn meist wird für CP/M eine 80-Zeichen-Karte (oder eine Terminal-Karte) eingesetzt. Die Tastaturabfrage sollte ziemlich regelmäßig erfolgen, benutzt also sehr stark die Memory Map. Hier werden eigentlich nur 10 Adressen gebraucht (bei normaler Tastatur), aber eine mißene Ausdekodierung bläht diese auf 1K verbratene Adressen auf. Die Drucker-Schnittstelle braucht nur eine Adresse und wurde bei manchen Genies schon in einen Port verlegt. Die Disketten-Ein/Ausgabe belegt 5 Adressen, aber durch die Ausdekodierung werden (mit Drucker) wieder 32 daraus. Diese Adressen werden bei Diskettenoperationen sehr intensiv genutzt.

Der Z-80 stellt 256 Ports zu Verfügung, die keinerlei Speicher belegen. Warum wurden die nicht benutzt? Gute Frage, vergeben ist beim TRS-80 nur einer (FFh), bei einigen Nachbauten noch FCh, FDh und FEh. Außerdem machen Erweiterungskarten (HRG 1b, Banker) Gebrauch von den Ports, aber die 256 werden trotzdem nur zum Bruchteil genutzt. Das Bildschirm-RAM brauche ich nicht und die Tastatur-Abfrage soll demnächst die Terminalkarte übernehmen, also bleiben nur Drucker und Diskette. Warum nicht diese in Ports verlegen?

Gemacht, getan. Nachdem die I/O-Karte gesäubert wurde (s. Artikel "Entfernung der Kassettenschnittstelle") hatte ich genug Platz für den Umbau. Es wäre einfach gewesen, wenn ich nur noch den Zugriff über die Ports haben wollte, aber leider sind BOOT-ROM, GDOS samt Programmen und momentan noch mein CP/M-BIOS auf die Memory Map eingestellt. Also muß eine Umschaltung vorgenommen werden, je nach Wunsch Ein/Ausgabe über Memory Map oder Ports.

Da ich nicht 32 Ports verbraten wollte, mußte der Kram erst weit ausdekodiert werden, daß nur noch 16 Ports gebraucht werden (noch besser wäre natürlich eine Reduzierung auf 6, aber da wird's schwierig). Dazu braucht man nur ein OR-Gatter, das auch leicht zu bekommen ist: Entgegen dem Schaltplan ist das OR-Gatter von U19,8-10 überbrückt und sinnlos. Man kann stattdessen eine neue Verbindung von U16,8 zu C15 (Steckerleiste, WREQ\*) ziehen. Alsdann wird die Brücke bei U19 entfernt und die entsprechenden Pins isoliert:

Signal	Pin an U19	wo getrennt?
WREQ*	8	unten, direkt daneben an Durchkontaktierung
PLUS	9	unten, direkt am Pin
noname	10	oben, links, neben +5V



## Anmerkung:

Bei meinen Zeichnungen habe ich folgende Symbole benutzt:

- Lötspunkt (IC-Pin)
- Durchkontaktierung
- Leiterbahn
- X Unterbrechung einer Leiterbahn

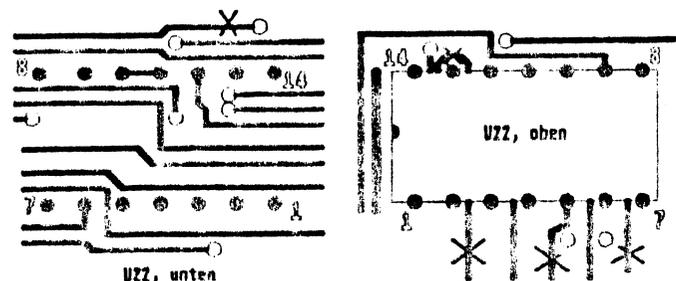
Nun kann die erste Änderung vollzogen werden: die Ausdekodierung auf 37Ex (vorher neben 37Ex auch 37Fx belegt = 32 Byte). Zuerst wird U22,13 getrennt (s. unten) und dann werden folgende Verbindungen hergestellt:

U20,9 -> U19,9 (A04)  
 U20,6 -> U19,10 (FREE\*)  
 U19,8 -> U22,13 (neu, nennen wir es xxEx\*, in Schaltplan als PEX\*)

Nach dieser Änderung muß der Rechner noch genauso laufen wie vorher, denn der gespiegelte Bereich bei 37Ex wird meines Wissens nie benutzt.

Die zweite Änderung ist der eigentliche Umbau, aber der ist nach den Vorarbeiten nicht mehr schwierig. Zuerst werden folgende Verbindungen getrennt (die erste gehört noch zur ersten Änderung, s. oben):

Signal	Pin an U22	wo getrennt?
FREE*	13	oben, ganz links in Richtung U20
		oben, ganz rechts, in Richtung U20
37EX*	8	oben, an Pin 13
MWR*	1	unten, bei Durchkontaktierung über Pin 13/14
MRD*	5	oben, in der Mitte, Richtung U20



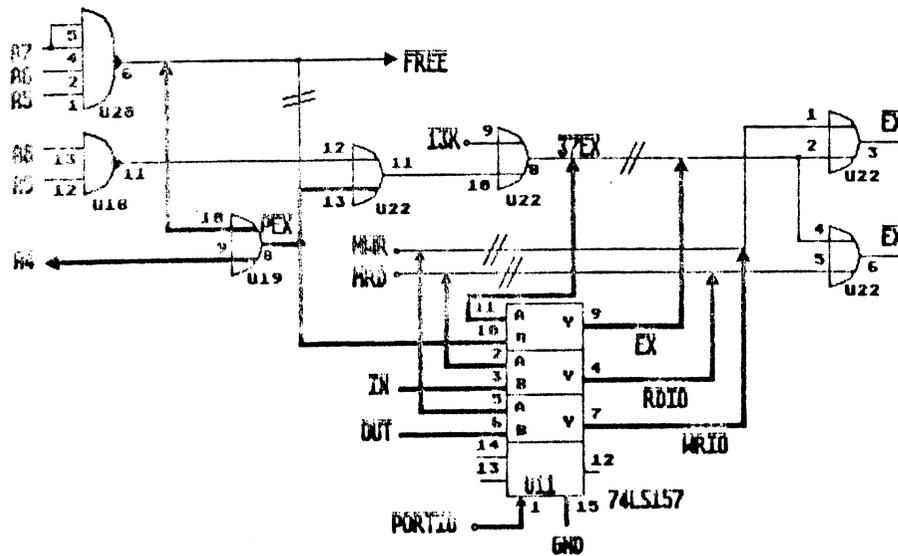
Nun wird das 74LS157 in die Fassung von U11 gesteckt und die Pins unten entsprechend der folgenden Liste verbunden. Es ist auch möglich, das IC auf ein anderes huckepack zu löten (nur Pin 8 und 16) und die Beinchen mit denen der entsprechenden ICs oben zu verbinden.

Signal	von...	nach...
FREE*	U17,12	U20,6
SPEAKER*	U11,1	U7,2
MRD*	2	A23 (Steckerleiste)
IN*	3	A21 "
RDIO*	4	U22,5
MWR*	5	A22 "
OUT*	6	A16 "
MRIO*	7	U22,1
EX*	9	U22,2
PEX*	10	U22,13
37EX*	11	U22,8
GND	15	U11,8

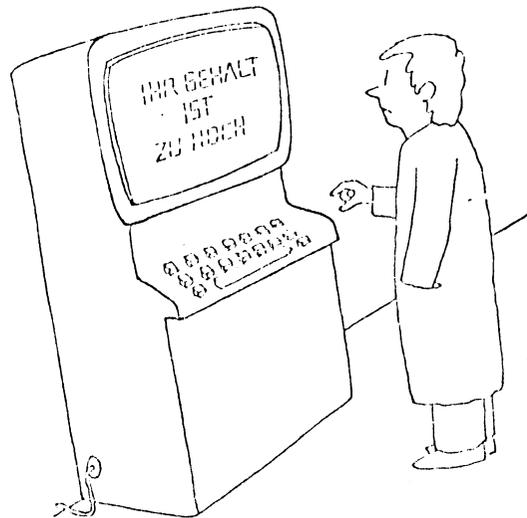
Wie Ihr vielleicht schon gemerkt habt, wird das Signal SPEAKER\* zur Umschaltung zwischen Memory Map und Port I/O benutzt. Es hatte vorher die Aufgabe, einen evtl. angeschlossenen Lautsprecher an- und auszuschalten. Das Signal bot sich für meine Zwecke einfach an (im Schaltplan unten taucht es als PORTIO\* auf).

Wie Ihr sicherlich wißt, habe ich mit diesem Artikel kein Publikum bzw. kann kaum Wachstümmer finden (wer hat schon ein Genie IIs?). Aber ich möchte Euch anregen, Eure Kiste nicht einfach kritiklos hinzunehmen, sondern sie so zu verbessern, wie es Euch gefällt. Das brauchen nicht immer die großen Hardware-Freaks zu machen, jeder kann sich kleinere Sachen selbst überlegen. Dies war mein erster Versuch in Hardware und es hat mir immerhin Mut gemacht, es wieder zu versuchen.

Abschließend noch der neue Teil des Schaltplans, wie er von Helmut Bernhardt gezeichnet wurde:



Gerald Schröder



### Darf's ein bisschenl mehr sein ? Ramdisk mit 512KB

Nachdem ich Euch kurz die c't-Solid State Disk vorgestellt hatte, nun etwas für die Feinschmecker. Platz ist dazu da, daß er ausgefüllt wird und so ist es mir dort langsam aber sicher zu eng geworden. Da auf einen vollen Bus keine zweite SSD paßt, habe ich mich als Alternative für die RAM-/EPROM-Floppy "DISI-1" der Firma Conitec entschieden. Auf die passen mit RAMs immerhin 512KB und mit den neuen gebankten Eproams sogar bis zu 2MB (!). Die Ansteuerung ist fast identisch mit der der SSD, sodaß sie problemlos ausgetauscht werden kann.

An einem Donnerstag habe ich die Bestellung abgeschickt und pünktlich am Montag war das Päckchen per UPS da. Nach pfundweise Styroporflocken kam endlich ein in Antistatikfolie und dick Schaumstoff eingewickelter zweites Paket zum Vorschein und darin war schließlich die Platine. Als begleitende Lektüre ist ein 5mm dickes, gebundenes Heft im A4-Format mit ausführlicher Anleitung und Schaltplan dabei und die Treiber-Software und einige Utilities kommen auf einer Floppy daher. Die Platine ist mit Präzisionssockeln und ICs fertig bestückt und macht einen sehr sauberen, wenn durch das Fehlen der RAMs (man kann sie auch mit RAMs bestellen) einen leeren Eindruck. Auf der Steckerleiste hat sogar jemand, ich nehme an, daß es der Entwickler ist, sein Autogramm hinterlassen. Die stehen wenigstens zu ihrem Werk. Daß ein IC (ein Treiber) gefehlt hat, darf einen bei dem fast geschenkten Preis von nur DM 412,- (incl. Mwst usw.) nicht weiter stören. Normalerweise hat man sowas ja sowieso irgendwo rumliegen und lieber sofort ohne IC, als mit und noch eine Woche warten.

Ein erster Test mit einem Eprom ging völlig in die Hose, beim Auslesen kam nur Käse raus. Ich weiß nicht, ob es an dem Eprom gelegen hat, oder ob ich die Jumper falsch gesteckt hatte, oder was auch immer, jedenfalls ging es nicht. Nach dem Einsetzen der CMOS-RAMs hat sie aber einwandfrei mitgespielt und bis jetzt sind keine weiteren Fehler aufgetreten.

An Signalen vom Rechner braucht die Karte lediglich acht Daten-, acht Adreßleitungen, WR, RD und IDRO, sodaß sich der Aufwand zum Anschließen an Freadrechner sehr in Grenzen hält; gedacht ist sie natürlich für den ECB-Bus.

Wer viel Platz auf der Ramdisk braucht und wen der doch ziemlich hohe Preis nicht abschreckt, der ist mit der DISI-1 wirklich gut bedient. Im Gegensatz zur c't SSD muß man sich höchstens einen Adapterstecker bauen und funktioniert sehr sicher.

Alexander Schmid

# Sharp 3541 — anwenderfreundlich und leicht erweiterbar

Mit dem 1. auf den Markt gekommenen Business-Computer MZ-3500 ersetzt Sharp einen etwas klöbigen Vorgänger seiner Computerserie, den PC 3201. Schon auf den ersten Blick unterscheidet sich das neue Gerät vom Vorgänger: Zentraleinheit und die beiden Slimline-Floppylaufwerke sind in einem ansprechenden Gehäuse untergebracht, der dreh- und schwenkbare Bildschirm und eine abgesetzte, sehr flache ergonomische deutsche Norm-Tastatur zeigen, daß man den Weg vom klöbigen Tastaturcomputer zum bequemen Arbeitsplatzsystem gefunden hat.

Das sollen machen zwei Leute, aber wer die letzten Ausgaben von CP aufmerksam gelesen hat, weiß, daß zu einem modernen Computer mehr gehört als eine schöne Tastatur.

Sharp hat, und das war nicht anders zu erwarten, mit der Serie MZ-3500 ein grundsolides, auf bewährten Elementen aufgebautes Computersystem entwickelt. Im Gegensatz zur Mehrzahl der übrigen Arbeitsplatzcomputer, wie jetzt Rechner der Preisklasse zwischen 5000 und 15000 Mark vielfach genannt werden, ist der MZ-3500 nicht mit einem 16-Bit-, sondern mit zwei 8080A-8-Bit-Prozessoren ausgerüstet. Dazu gehören in der Grundversion dieser Reihe, dem MZ-3541, bereits 128 KB Hauptspeicher, zwei Slimline-Floppy-Disklaufwerke mit je 380 KB Kapazität, ein fertigmacher Bildschirmcontroller und eine gepufferte Tastatur mit eigener CPU.

Daß ein im Jahre 1983 auf den Markt gekommenes Computersystem nur mit Hilfe von Erweiterungen ist, verwundert im ersten Augenblick. Beschäftigt man sich mit den heute angebotenen 16-Bit-Systemen näher, fällt bald auf, daß 16 Bit nur im mathematischen Sinn das Dop-

pelle von 8 Bit ist. Es fehlt noch an 16-Bit-Peripherie und ausgereifter Software. Auch wird es noch geraume Zeit dauern, bis die zwar auf 8-Bit-Systemen bewährten, aber im Schnellverfahren auf 16 Bit konvertieren Programme optimiert und den neueren CPUs angepaßt sind. Wen wundert das, wenn für viele 16-Bit-Computer nach kurzer Zeit Z80-Karten auf den Markt kommen, um, wie kürzlich in einer amerikanischen Computerzeitschrift zu lesen war, aus diesen Sechszehnern endlich brauchbare Computer zu machen. Dennoch hätte man dem MZ-3500 ein früheres Erscheinen gewünscht.

## Viel zu sicher

Beim MZ-3500 hat man sich augenscheinlich sehr viel Mühe mit der Ausgestaltung der Hardware gemacht. Um die Haupt-CPU nicht mit der Verwaltung des bis zu 256 KB umfassenden Speichers zu belasten, wurde ein besonderer Memory-Controller entwickelt; eine zweite Z80A-CPU übernimmt die meisten Ein-Ausgabe-Operationen und die Verbindung zu den beiden Grafikprozessoren. Auch die Tastatur mit einem 64 Zeichen umfassenden Eingabepuffer wird von einer eigenen CPU (80C49) kontrolliert.

Daß mit Spielern nicht ähnlich umgegangen wird, zeigt die Grafikerweiterung. Pro Seite werden 32 KB verwendet, was eine Auflösung von 640 x 400 Punkten erlaubt. Der Endausbau umfaßt drei Grafikseiten, die auch in Form von RGB-Speichern Farbausgabe erlauben.

## Das Wichtigste ist serienmäßig einbaubar

Sharp scheint das bisher angewandte Verfahren, alle Schnittstellen getrennt vom Grundgerät zu verkaufen, aufgegeben zu haben und rüstet den MZ-3500 mit zwei gebräuchlichen Schnittstellen aus: einer seriellen RS232C-Schnittstelle und einem Druckeranschluß, der der Centronics-Norm entspricht. Beide Schnittstellen sind jedoch nur an Platinenanschlüsse auf der Rückseite der Zentraleinheit zugänglich; man benötigt also noch ein Spezialkabel, um ein Gerät mit Normsteckern anschließen zu können. Damit sind die Anschlußmöglichkeiten jedoch noch nicht erschöpft. Neben dem Hauptmonitor kann ein zweiter, getrennt austauschbarer Bildschirm angeschlossen werden, ebenso eine weitere Diskettenstation mit zwei Laufwerken. Auf der Vorderseite des Computers befindet sich die Anschlußbuchse für einen Lichtgriffel, das erforderliche Interface ist bereits eingebaut. Auch an der Tastatur befindet sich noch eine Buchse; leider war nicht in Erfahrung zu bringen, wozu sie einmal dienen soll. Leider waren die Verbindungskabel zwischen Zentraleinheit und Bildschirm, vor allem jedoch zum Drucker, sehr kurz bemessen, was die optimale Aufstellung des Systems etwas behindert.

## Uhr mit Kalender

Geräte für den kommerziellen Anwender ist es wichtig, ständig über korrektes Datum und Uhrzeit zu verfügen. Dem wurde durch den Einbau einer batteriegepufferten Uhr mit Kalender Rechnung getragen. Auch ein Tongenerator für akustische Meldungen oder Bestätigungen ist eingebaut, er verfügt über einen Tonumfang von drei Oktaven.

## Hochoffizierende Farbgrafik

Ganz ohne Erweiterungen kommt jedoch auch der MZ-3500 nicht aus. Am wichtigsten sind wohl die Aufstockung des Hauptspei-

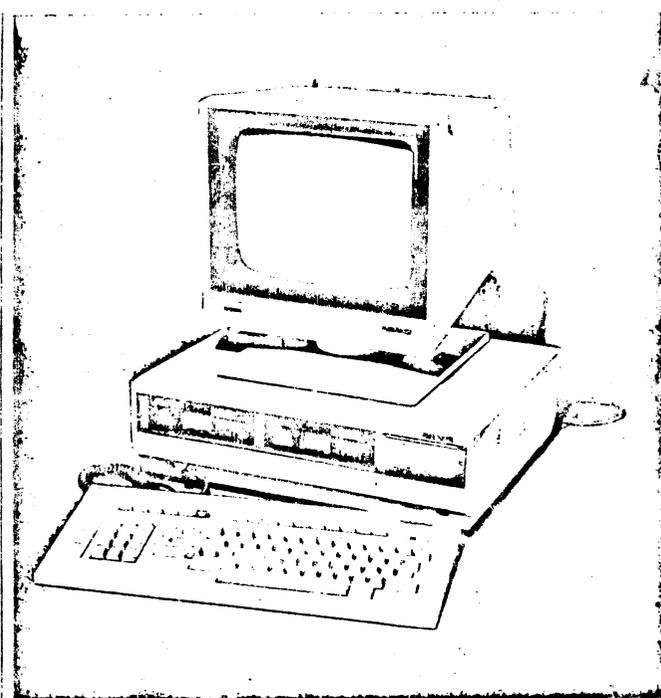
chers auf 128 KB beziehungsweise 256 KB und die Grafikerweiterung. Die Grafikplatine kann nach dem Entfernen einer Plattenabdeckung einfach unterhalb der beiden Floppylaufwerke eingesteckt werden. Sie enthält den Grafikcontroller und Steckplätze für insgesamt 96 KB dynamische RAM-Bausteine. In der Grundausstattung enthält diese Platine 32 KB RAM, also eine Grafikseite. Ausgebaut wird sie durch einfaches Einsetzen zusätzlicher Bausteine. Der Hauptspeichererweiterung findet leider im Erweiterungsgehäuse keinen Platz mehr, man muß dazu einen Erweiterungsrahmen einbauen, der dann vier Zusatzkarten aufnehmen kann. Die RAM-Karte ist im Steckfeld für zwei Speicherbänke mit je 64 KB ausgestattet, die Erweiterung erfolgt auch hier durch einfaches Einstecken der Speicherbausteine.

Zur Erweiterung der Kommunikationsmöglichkeiten wird noch eine serielle Schnittstellenkarte mit zwei Kanälen angeboten; sie ist bei Verwendung des Computers als BTX-Terminal oder innerhalb eines Mehrplatzsystems erforderlich.

## Servicefreundlicher Aufbau

Zu jedem Computertest gehört der Blick ins Innere; bei den Fähigkeiten des MZ-3500 ist dies natürlich besonders interessant. Die Abdeckhaube, wie das gesamte Gehäuse der Zentraleinheit aus Stahlblech, läßt sich nach dem Lösen einiger Schrauben leicht abnehmen. Darunter zeigt sich ein besonders übersichtlicher und damit servicefreundlicher Aufbau dieses Systems (Bild 1). Da der Rechner auf nur zwei Platinen aufgebaut ist, sind so gut wie keine Kabelverbindungen vorhanden; nur das getrennt aufgebaute und vollständig gekapselte Netzteil ist über Kabel mit der Hauptplatine und den Diskettenlaufwerken verbunden. Die Diskettenlaufwerke selbst können auf einem eigenen Chassis herausgeklappt werden und sind dann von allen Seiten zugänglich. Da sie auch in diesem Zustand betriebsbereit sind, können Einstell- und Reinigungsarbeiten leicht ausgeführt werden. Auf der rechten Seite ist nun auch der kleine Lüfter zu sehen, er ist bei geschlossenem Gehäuse kaum zu hören.

Die Systemerweiterung — das Testgerät wurde vom MZ-3530 (64 K, ein Laufwerk) zum MZ-3541 (256 K, zwei Laufwerke, Grafik) aufgerü-



stet — macht bei diesem Aufbau keine Schwierigkeiten; alle Steckplatinen sind serienmäßig eingebaut, Kabelverbindungen sind nicht erforderlich, das Gehäuse muß nicht einmal zerlegt werden. Zwei wichtige Teile der Hauptplatine sind auf der Unterseite des Geräts frei zugänglich: eine Schalterleiste zur Betriebsartenwahl und die Fassung des Zeichengenerators. Mit dem Betriebsartenschalter kann zum Beispiel die Polarität und Zahl der RS232C-Quittungssignale eingestellt und die Bildfrequenz zur Ansteuerung eines hochauflösenden Grafikbildschirms erhöht werden. Auch kann hier von der Verwendung des Dezimalpunkts (bei numerischen Eingaben) auf Dezimalkomma umgeschaltet werden (!). Die Voreinstellungen werden von Basic und EOS übernommen, können jedoch softwaregesteuert verändert werden.

Alle selbständigen Baugruppen des Rechners (unter anderem RS232C-Schnittstelle, Uhr und Tastatur) erzeugen Unterbrechungssignale (Interrupts), die zum Beispiel von Basic in den Anweisungen ON TIME, ON KEY und POL-

LING verwendet werden. Damit wird der Aufbau von zeitabhängigen Programmteilen oder die Überwachung von Peripheriegeräten zum Kinderspiel.

Leider enthalten die Handbücher keinerlei Angaben über den Schaltungsaufbau oder die direkte Steuerung der Baugruppen. Der Entwurf eigener Systemsoftware, aber auch eigener Hardwareerweiterungen, dürfte damit so gut wie unmöglich sein.

Im ernsthaften kommerziellen Einsatz wird man wohl mit den beiden Minifloppy-Laufwerken trotz der recht hohen Kapazität bald nicht mehr auskommen. Zwei weitere 5 1/4-Zoll-Laufwerke sind für rund 3500 Mark erhältlich, für rund 7500 Mark gibt es ein 8-Zoll-Doppellaufwerk mit zusammen 2 MB Speicherplatz. Auch ein Festplattenlaufwerk ist lieferbar, es hat eine Kapazität von 10 MB und dürfte mit Controller zirka 10000 Mark kosten.

## Viel Neues bei der Software

Der gute Eindruck setzt sich auch bei der mitgelieferten Software fort. Um dem Anwender ein breites

Spektrum an Möglichkeiten zu bieten, aber auch um Kompatibilität zu den Vorgängermodellen zu wahren, liegen dem MZ-3500 zwei Betriebssysteme und ein Basic-Interpreter bei. Es handelt sich dabei um das Sharp-eigene FDOS (Floppy-Disk Operating System) und das neu entwickelte CP/M-Plus-kompatible Betriebssystem EOS 3.0.

FDOS ist ein besonders anwenderfreundliches Betriebssystem. Von den acht FDOS-Befehlen befassen sich daher allein sechs mit dem Diskettenbetrieb, zum Beispiel mit der Initialisierung neuer Disketten und mit der Datensicherung. Mit dem FDOS-Befehl DEBUG kann ein kleiner Maschinenprogramm-Editor aufgerufen werden, dessen Anwendungsmöglichkeiten jedoch stark eingeschränkt sind, da sich der Zugriff auf den sogenannten Anwendermikrobereich (einem von Basic reservierten Maschinenprogramm-Bereich) beschränkt. Änderungen an Basic oder FDOS sind daher nicht möglich, der Endanwender wird daran jedoch auch kaum Interesse haben.

Unter FDOS muß auch der mitgelieferte Basic-Interpreter betrieben werden, er wird ganz einfach über den FDOS-Befehl Basic aufgerufen. Zu Basic gehören zwei deutschsprachige Handbücher mit über 400 Seiten Umfang. Beim Studium der implementierten Anweisungen und Funktionen zeigt sich bald, daß hier kein überflüssiger Aufwand betrieben wurde. Was hier aus dem Basic-Grundbefehlsatz, systembezogenen Anweisungen und einer großen Zahl aus anderen Hochsprachen entliehenen Sprach-elementen zusammengefügt wurde, ist bemerkenswert.

Ebenfalls bemerkenswert, leider im negativen Sinn, ist die Tatsache, daß Basic nur über einen Zeileneditor verfügt, das heißt jede zu bearbeitende Zeile muß mit EDIT aufgerufen werden. Warum der sonst so komfortable Interpreter damit »gestraft« wurde, ist rätselhaft.

#### Neue Variablentypen

Induzierte Variablen (zum Beispiel A\$(12)) und damit Arrays (definiert zum Beispiel mit DIM A\$(100)) werden wohl in jedem Basic-Programm vorkommen. Um nun aus einem derartigen Array ein bestimmtes Element herauszusuchen oder alle Elemente auf Diskette abzuspeichern, war man bisher gezwungen, jedes einzelne Element mittels einer For-Next-Schleife zu bearbeiten. Sharp-Basic stellt dazu einen völlig neuen Variablentyp zur Verfügung, die »Kollektive Array-Variable« (zum Beispiel A\$(\*)). Diese neue Variable besteht aus dem mit DIM deklarierten Variablennamen und umfaßt alle Array-Elemente. Im folgenden Programmbeispiel wird der Unterschied deutlich:

```
Standard-Basic
OPEN "0",#1,"PROBE"
FOR I=1 TO 100
PRINT#1,A$(I)
NEXT I
CLOSE
```

```
Sharp-Basic
OPEN "0",#1,"PROBE"
BPRINT#1,A$(*)
CLOSE
```

Der Inhalt der Datei PROBE ist in beiden Fällen derselbe.

Kollektive Array-Variablen können auch zum globalen Löschen (ERASE A\$(\*) oder Suchen (SE-

ARCH A\*(\*),"NAME",A) verwendet werden. Die Anweisung SEARCH A\$(\*),"1","SCH",A durchsucht das Array A\$(x) nach Begriffen, die mit den Buchstabenkombination "SCH" beginnen. A enthält dann den Index des Array-Elements, das dieser Bedingung entspricht. Auch die Anweisungen MAXS (Suchen des größten Elements) dienen zum globalen Suchen.

Da Basic den gesamten Hauptspeicher (maximal 256 KB abzüglich Programm) als Variablen-Speicher verwenden kann, müssen Länge und Typ der Variablen genau bestimmt werden. Dieser Umstand sorgt außerdem für sehr schnellen Zugriff und maximale Speicherausnutzung. Basic sieht daher fünf Grundtypen vor: Integer- und Real-Variablen (12 Stellen, Exponent -99 bis +99), Stringvariablen des \$-Typs (16 Stellen) und des S-Typs (16 Stellen, auch für Binärdaten mit Null-Elementen) und dimensionierte String-Variablen mit beliebiger Stellenzahl (zum Beispiel X\$(10)\*200).

#### Lokale Variablen

Eine Besonderheit sind auch lokale Variablen innerhalb von Unterprogrammen, das heißt Variablennamen können innerhalb eines Programms mehrfach verwendet werden. Die Erstellung einer Modul-Bibliothek mit ständig verwendeten Routinen bietet sich deshalb natürlich an.

#### Erweiterter Diskettenzugriff

Neben den zum Beispiel aus Microsoft-Basic bekannten Dateitypen und Zugriffsarten wurden einige systemspezifische Sonderformen angefügt. So gibt es bei sequentiellen Dateien die Zugriffsform "A" (Append), um Daten an eine schon vorhandene Datei anzufügen, ohne die gesamte Datei vorher lesen zu müssen. Einmal erstellte Bildschirmmasken können mit den Anweisungen VSAVE und VLOAD gespeichert und wieder abgerufen werden. Ähnliche Befehle gibt es zum Speichern der Bildschirmgrafik. Sogar die Tastaturbelegung kann in speziellen Dateien gespeichert werden.

#### Password-Schutz von Programmen und Daten

Programme und Daten können auf Wunsch mit Schlüsselworten gesichert werden. Derart geschütz-

te Dateien können weder kopiert noch aufgelistet werden.

#### Formatierte Ein- und Ausgabe

Neben den bekannten Anweisungen USING und IMAGE für die Datenausgabe enthält Basic sehr leistungsfähige Anweisungen zur Dateneingabe. Hier kann genau angegeben werden, welcher Datentyp eingegeben werden darf. Auch Länge und Format (zum Beispiel Anzahl der Dezimalstellen) der Eingabe kann bestimmt werden.

#### Leistungsfähige Schnittstellensteuerung

Sehr komfortabel ist auch die Steuerung der seriellen Schnittstelle ausgefallen. Zum einen können hier Drucker oder Plotter mit unterschiedlichen Übertragungsprotokollen angeschlossen werden, die Schnittstelle kann jedoch auch zur Kommunikation und zur Überwachung eingesetzt werden. Über einen Puffer und die Anweisung POLLING ist es zum Beispiel möglich, die Schnittstelle zu überwachen und erst bei Anliegen einer bestimmten Information zu reagieren. Zur Übertragung sind alle Variablentypen, auch kollektive Array-Variablen, zulässig. Als Besonderheit kann der MZ-3500 mit dem Befehl TERM in einen Terminalbetrieb versetzt werden und kann daher leicht in ein Mehrplatzsystem integriert werden.

#### Der Bildschirmkünstler

Schon die Grundgeräte (MZ-3530/MZ-3531/MZ-3541) ohne Grafikerweiterung verfügen über vielfältige Möglichkeiten der Bildschirmgestaltung. Die Darstellung kann in vier Formaten erfolgen: 40 oder 80 Zeichen pro Zeile, die Zeilenanzahl ist auf 20 oder 25 Zeilen einstellbar. Der 20-Zeilenbetrieb ist besonders übersichtlich und eignet sich vor allem zur Textdarstellung.

Jedes dargestellte Zeichen kann mit einer Reihe von Attributen versehen werden. Dazu gehört die Farbdarstellung mit getrennt wählbarer Vorder- und Hintergrundfarbe und die auch im Schwarzweiß-Betrieb mögliche Darstellungsart Blinken und Invertieren.

Jede Zeichenposition kann zusätzlich mit vertikalen und horizontalen Hilfslinien versehen werden. Mit Hilfe dieser Funktion können zum Beispiel leicht Eingabefelder oder Markierungen gezogen werden. Basic unterstützt dies mit dem Be-



Bild 2. Schriften lassen sich in beliebigen Formen in die Grafiken übernehmen

fehl TABLE. Zeichen und Attribute können getrennt voneinander gesetzt und gelöscht werden, was vor allem die Arbeit mit Bildschirmmasken erleichtert.

Da noch eine einfache Blockgrafik (SET/RESET) und eine Vielzahl von Sonderzeichen vorhanden ist, lassen sich für den kommerziellen Anwender bereits ausreichende Grafiken (zum Beispiel Balkendiagramme) erstellen.

#### 400 x 640-Punkt-Grafik mit eigenem Prozessor

Die als Option erhältliche Grafikkarte erweitert die Möglichkeiten nochmals beträchtlich. Die drei Grafikkarten mit einer Auflösung von je 400 x 640 Punkten werden im Farbmodus zum Speichern der drei Grundfarben Rot, Grün und Blau verwendet, im Schwarzweiß-Betrieb kann jede Seite getrennt angesprochen werden. Da ein Grafikkartprozessor zum Einsatz kommt, sind die in Basic implementierten zahlreichen Steueranweisungen sehr schnell und leistungsfähig.

Neben den Grundbefehlen zum Setzen und Löschen eines Punkts und dem Ziehen von Linien können Kreise und Ellipsen gezeichnet werden. Durch die Anweisung PAINT können definierte Bereiche in vier Helligkeitsstufen ausgemalt werden. Da die Bildschirmkoordinaten meist nicht mit den Werten der darzustellenden Funktionen harmonisieren, kann mit der Anweisung SCALE ein beliebiger Koordinatenursprung und eine beliebige Skalierung eingestellt werden. Eine Umwandlung von Funktions- in Grafikkoordinaten ist dann nicht mehr nötig.

Textzeichen können in 90-Grad-Schritten gedreht in die Grafik übernommen werden, auch Vergrößerungen und Verkleinerungen sind möglich. Mit Hilfe eines über die Cursortasten beweglichen Gra-

fikursors lassen sich Gebiete eingrenzen, die dann mit Basic weiterbearbeitet oder an eine andere Stelle verschoben werden können. Der Grafikkursor kann aber auch zum Zeichnen mit verschiedenen Optionen verwendet werden.

Um die Inhalte der Grafikkarten auf Papier zu bringen, werden von Sharp verschiedene Matrixdrucker in der Preislage zwischen 1650 und 2900 Mark angeboten.

#### 4-Farb-Tintenstrahl-Drucker als Zubehör

Als besonderen Leckerbissen gibt es noch einen Tintenstrahl-Drucker, der vier Farben darstellen kann. Mit ihm lassen sich auch Farbgrafiken originalgetreu wiedergeben. Die Betriebssysteme des MZ-3500 sind bereits mit den notwendigen Treiberprogrammen versehen, so daß der Anschluß des Geräts keine Schwierigkeiten bereiten dürfte.

#### EOS 3.0 — das deutsche CP/M: keine Probleme mit Standard-Software

Als zweites Betriebssystem verfügt der MZ-3500 über das in Deutschland entwickelte EOS 3.0. Dieses System entspricht in der Grundstruktur CP/M-Plus, kann also über 64 KB hinausreichende Speicherbereiche verwalten. Dies zeigt sich schon daran, daß 63 KB als freier Benutzerspeicher ausgesprochen werden. EOS simuliert auf dem MZ-3500 zwei gebräuchliche Terminaltypen, das ADM-3A von Lear Siegler und das TeleVideo TV950. Damit bestehen für die Anpassung von Standard-Software praktisch keine Probleme, da eines der beiden Terminals sicher in den Installationsprogrammen aufgeführt ist. Entsprechende Versuche mit Wordstar, Wordmaster, T/Maker, dBase und Multiplan verliefen erfolgreich.

EOS blendet im TV950-Modus ständig eine Statuszeile ein, sie enthält Uhrzeit, Datum und Informationen über den Systemzustand. Teile dieser Statuszeile können über das Anwenderprogramm definiert werden.

Besonders angenehm fiel die leichte Belegbarkeit der Tastatur auf, alle Tasten bis auf CTRL und ESC können ohne Zusatzprogramm direkt über die Tastatur neu definiert werden. Mit einem weiteren Tastendruck kann die Originalbelegung wiederhergestellt werden.

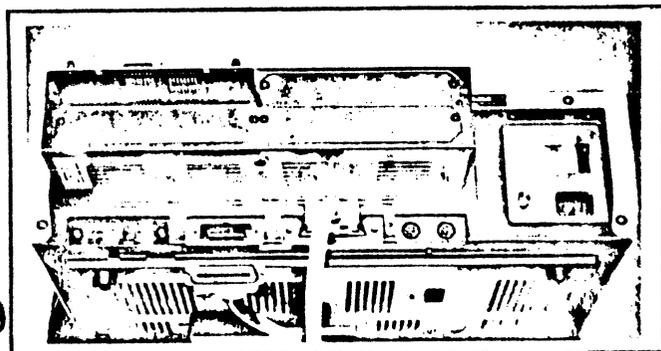


Bild 1. Unten auf der Rückseite befinden sich alle Schnittstellen, darüber der Erweiterungsrahmen mit der Speichererweiterung

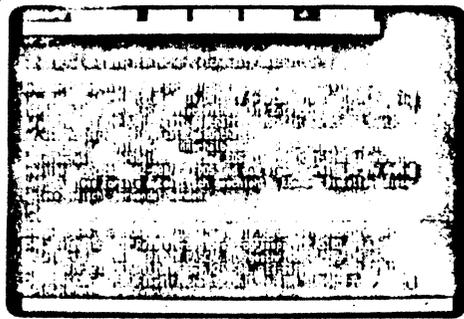


Bild 3. CP/M-Standardsoftware, hier WordStar, läßt sich einfach anpassen

Bei der Arbeit mit Wordstar und Wordmaster stellten sich in Verbindung mit der sonst sehr angenehmen Tastatur zwei Probleme heraus: Da die ENTER-Taste recht klein ausgefallen und sehr dicht bei den Cursortasten plaziert ist, kommt es bei schnellem Schreiben oft zu unbeabsichtigtem Auslösen dieser Taste. Das zweite Problem zeigte sich bei der Arbeit mit Micropro-Produkten, zum Beispiel eine Steuerfunktion, zum Beispiel eine Cursortaste, zu lange betätigt, laufen die entsprechenden Funktionen nach, der Cursor fährt zum Beispiel weit über das Ziel hinaus. Im schlimmsten Fall kann das zum »Absturz« des gesamten Systems einschließlich Datenverlust führen.

Obwohl dieses Phänomen sicher mit der Tastaturpufferung zu tun hat, tritt es bei anderen Programmen, zum Beispiel TMAKER, nicht in Erscheinung. Vielleicht sollte hier eine abschaltbare Tastaturpufferung vorgesehen werden.

Deutscher und amerikanischer Zeichensatz

Besonders Programmierer ärgern sich oft über einen Nachteil von Rechnern mit deutscher Normtastatur. Die Umlaute ersetzen die in vielen Programmiersprachen verwendeten Klammerzeichen. Die Entwickler von EOS haben da aufgepaßt und die Möglichkeit geschaffen, auf Tastendruck von deutschem auf amerikanischen Zeichensatz umzuschalten.

Um Auf- und Umsteigern innerhalb der Sharp-Computerfamilie den Systemwechsel zu ermöglichen, kann EOS mit vier Mini-Diskettenformaten arbeiten: PC 3201,

MZ-80B/A, FDOS und das eigene Format. Da die Anpassung an diese Formate automatisch geschieht, können Disketten dieser Systeme gemischt verwendet werden.

Die gelungene Anpassung von EOS 3.0 an die Hardwareeigenschaften des MZ-3500 zeigt auch die Realisierung des Programmpaketes Astro des Berliner Professors Gerd Klemm. Das in Pascal MT+ geschriebene Programm verwendet sehr viele Grafikfunktionen des MZ-3500. Es zeigt auch das einfache Einbinden der rechner-spezifischen Grafik in eine Standard-Programmiersprache. Astro erlaubt auf dem MZ-3500 die Berechnung und grafische Darstellung beliebiger astrologischer Daten auf der Grundlage einer systemeigenen Datenbank, die pro Diskette zirka 1800 Patienten aufnehmen kann.

EOS selbst stellt in vielem eine Verbesserung von CP/M dar. Besonders interessant ist die Tatsache, daß ein Diskettenwechsel nicht mehr angemeldet werden muß, ebenso gehört die stupide Meldung »BDCS-ERROR ON A« der Vergangenheit an. Man erhält eine genauere Fehlerbeschreibung und kann im Dialog entscheiden, was weiter passieren soll. Ein weiterer großer Fortschritt ist der Bildschirmeditor von EOS. Bei Eingabefehlern muß also nicht mehr die ganze Zeile neu eingegeben werden, auch müssen fast gleiche Eingaben nicht ständig wiederholt werden. Auch in der Dateiverwaltung hat sich einiges getan. Durch ein sogenanntes Stamping werden Datum und Uhrzeit der Dateierzeugung und der letzte Zugriff im Inhaltsverzeichnis mitgeführt. Zusätzlich können Attribute angegeben

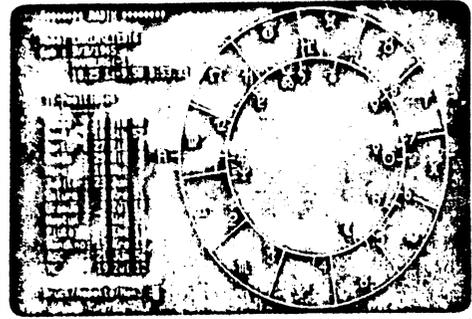


Bild 4. Rechnerspezifische Grafik mit Pascal am Beispiel eines astrologischen Programms

werden; die Organisation von Diskettendateien wird dadurch erheblich erleichtert.

Dem lobenswerten Vorsatz, Befehle und Meldungen von EOS völlig in deutsch zu halten, ist man in der beim Test vorliegenden Version leider nur teilweise gefolgt. Es ist zu hoffen, daß in der Endversion dieser Schönheitsfehler behoben ist.

**Fazit: uneingeschränkt empfehlenswert**

Mit dem MZ-3500 ist ein Rechner auf dem Markt, der sich gleichermaßen für den Endanwender wie für den Programmierer eignet.

Mit FDOS und Basic kann sehr leistungsfähige systemnahe Software erstellt werden. EOS erlaubt die einfache Anpassung des umfangreichen CP/M-Programmangebots.

Damit kann zum sehr günstigen Systemgrundpreis von rund 7900 Mark (MZ-3541; zuzüglich MwSt.) eine sozusagen schlüsselfertige Problemlösung zusammengestellt werden, die vom Anwender keine Programmier- und kaum EDV-Kenntnisse fordert.

Dem Programmierer, aber auch dem anspruchsvollen Amateur, bietet sich vor allem in Verbindung mit der Grafikerweiterung und EOS 3.0 ein leistungsfähiges Instrument an, das wohl den meisten Ansprüchen gerecht werden wird.

Ansatzpunkte zu berechtigter Kritik ließen sich, von ein paar sicher zu behebenden Schönheitsfehlern, weder bei Hard- noch Software finden. Das sehr gute Preis-Leistungs-Verhältnis tut ein übriges, den MZ-3500 uneingeschränkt empfehlenswert zu machen. (Uwe Pansow)

Hallo Clubfreunde, mein gutes altes Genie kommt nun doch so langsam in die Jahre und so stellte sich die unvermeidliche Frage: Neukauf, Ja oder Nein? Nun versieht seit ca. 4 Monaten ein ATARI 1040 ST seinen Dienst bei mir. Dieser Zeitraum erscheint mir ausreichend, um einmal einen Erfahrungsbericht für das Club-Info zu verfassen. Um dem Verdacht der Einseitigkeit vorzubeugen, habe ich aufgrund eines Hinweises von Hartmut Obermann, noch einige andere Clubmitglieder mit einem ATARI-Rechner angeschrieben und zur Mitwirkung an einem Erfahrungsbericht aus der Sicht des jeweilig betroffenen aufgefordert. Vorgeschlagen hatte ich dabei folgende Punkte:

a) Der Anlaß zum Kauf

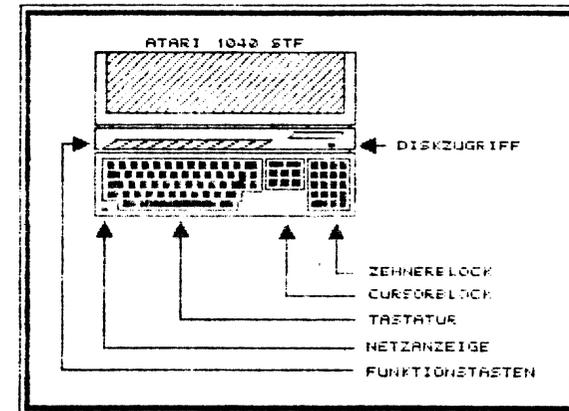
b) Erste Erfahrungen im Betrieb (Tastatur, Betriebssystem, Mausbenutzung etc.)

c) Softwareangebot und deren Qualität (Public Domain, käufliches)

d) Ein Vergleich Tandy Model I (Genie I, II, III) (-> ATARI) Hardware und Software

Was dabei herausgekommen ist, kann entweder in dieser, der nächsten oder auch schon in der vorherigen Info-Ausgabe nachgelesen werden. Hier nun mein Beitrag:

Der erste Unterschied zu meinem alten Genie ist die deutsche Tastatur und Zeichensatz, nebst einer größeren Anzahl von zusätzlichen Tasten. An die Tastatur muß man sich gewöhnen, da die Federwirkung mit zunehmendem Druck auf die Taste zunimmt! Im Grunde nicht weiter tragisch, führt jedoch bei schnellem Schreiben



hinterher ausgiebig danach suchen. Ich habe mir deshalb angewöhnt, kräftig in die Tasten zu greifen. Ansonsten macht die Tastatur einen brauchbaren Eindruck. Am Gehäuse stören mich eigentlich mehr die reichlich großzügig ausgelegten Aussparungen für die Tastenfelder. Hier hätte man durch eine präzisere Herstellung des Gehäuse-

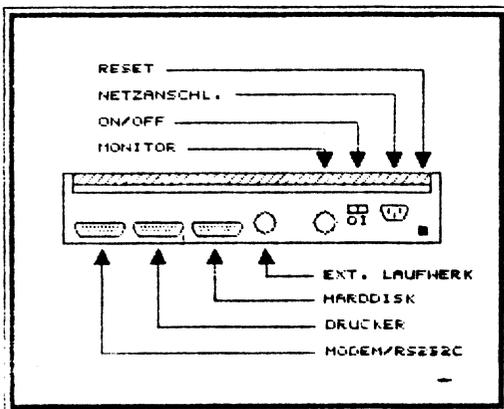
seeberteils glänzen können. Zum Punkt Schnittstellen gibt es nicht viel zu sagen, es ist alles vorhanden, was so im normalen Privatbetrieb gefragt ist (siehe Zeichnung). Das der interne Bus außen nicht verfügbar ist, sollte man wissen. Wer gern zusätzliche Hardware benutzt, könnte Probleme bekommen. Wie mir aus der c't bekannt ist, läßt sich jedoch der ROM-Port hervorragend für

diese Aufgabe mißbrauchen. Bei mir steckt an dieser Stelle ein Uhrmodul (die gibt es aber mittlerweile auch als Einbauteil !). Da ich außer dem Centronics-Port keine der anderen Schnittstellen benutze, bleibe ich den Kommentar hierzu schuldig. Soweit mir bekannt ist, handelt es sich beim Centronics-Anschluß um einen ganz gewöhnlichen 8-Bit Port des internen Sound-Chips. Wer seinen Drucker über die serielle Schnittstelle ansteuern kann, hat so einen 8bit breiten I/O-Anschluß zur Verfügung (mit der Möglichkeit diesen auch als Eingang benutzen zu können).

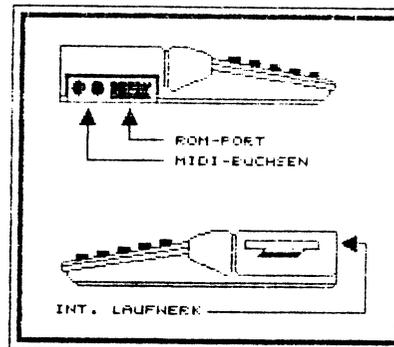
Das eingebaute 3.5" Diskettenlaufwerk arbeitet recht ordentlich. Lediglich in letzter Zeit hatte ich beim Spiel STARGLIDER Probleme mit den innersten Spuren (80... 82). Hier kam es wiederholt zu Ladefehlern. Leider benutzt der Hersteller diese, um eine Art Kopierschutz zu erzeugen. Die normale Copy-Funktion des Betriebssystems erzeugt nur unbrauchbare Kopien, da Spuren ab >80 wohl nicht mehr korrekt kopiert werden. Ansonsten muß ich sagen, das mir das 3.5" Format der Disketten mehr zusagt, als die 5.25" meines Genie's. Unglücklicherweise ist es jedoch mit der Speicherkapazität so eine Sache. Die erreichten 720K sind von der Wirkung her wie 360K bei den 8-Bitern. 16-Bit Rechner verbrauchen selbst bei "Ein-Byte Befehlen" grundsätzlich die doppelte Menge an Speicherplatz ! Der andere eigentlich schmerzlichere Punkt ist die unerwartete Speichergräßigkeit der Anwendungsprogramme: Textverarbeitung ca. 160K Byte, Grafik-Programme ca. 150K Byte, in Assmøbler geschriebene Programme (GFA-Basic und Compiler) ca. 60K Byte. Bei solchen Zahlen ist es dann kein Wunder, wenn die 720K dahinschmelzen und nach 4 bis 5 Programmen aufgebraucht sind ! Die "Speichergiganten" entstehen meist aus dem Umstand, daß die Programme in C oder anderen Hochsprachen geschrieben wurden. Die Effizienz solcher Wege ist ja ausreichend bekannt. Von dieser Tatsache einmal abgesehen, ist der verfügbare Platz für Texte und sonstigem "Spielkram" ausreichend. Es gilt sich ebend anzupassen: Basic auf eine Disk, Textverarbeitung auf eine Disk...

Über den Monitor kann ich nur positives sagen. Das Bild ist überzeugend ruhig und die Auflösung und Bildschärfe ist einfach super. Lediglich die Lösung, den Aus-Schalter an den Lautstärker-Regler zu ketten und diesen dann auch noch an die rechte Seite des Monitors zu verbannen, kann ich mir nur als Eigenwilligkeit des Designers erklären. Für den Alltagsbetrieb ist diese Lösung Käse. Ein Druckschalter im Monitorfuß und von Vorn zu bedienen, wäre der angenehmere Weg gewesen.

Dann wäre da noch das Betriebssystem. Mit stattlichen 192K ein echter Elephant (NewDOS80 ist, soweit ich mich erinnere, mit allen Overlays auch so um die 100K groß !). Dafür kann sich das innenleben allerdings sehen lassen. Während der eigentliche



Betriebssystem-Teil (TOS genannt) recht simpel gehalten ist, sind die vorhandenen grafischen Mittel spitze. Es gibt so gut wie alles, was das Herz begehrt: Linien, Kreise, Rechtecke, Polygone und diesen ganzen Klimbim um Sprites beliebige Zeichensätze und Fensterverwaltung. Demgegenüber stehen allerdings immer noch vorhandene Betriebssystem-Fehler, die sich in gelegentlichen Ungeheimheiten äußern. Datenverluste hatte ich dadurch jedoch bisher noch nicht. Das äußere Erscheinungsbild des GEM-Desktop ist recht ansprechend und gefällt mir eindeutig besser als die karge Kommando-Oberfläche des NewDOS. Die Funktionsvielfalt ist allerdings recht schmal, gemessen am Befehlsumfang von NewDOS80. Auch könnte das manchmal leidige 'runter- und wieder 'raufhängeln in der Directory-Hierarchie etwas schneller gehen. Ebenso ist die Darstellung der Dateien als Icons bzw. als Text nicht ausreichend, da eine selective Anzeige nicht möglich ist. Dafür lassen sich die Dateien aber nach vier verschiedenen Kriterien sortiert anzeigen. Die unabhängige Maus hat sich bei mir inzwischen einen festen Platz erkämpft. Dieses Tierchen ist wirklich eine feine Sache. Dank ihrer Anwesenheit entfällt bei der Textverarbeitung das lästige Arbeiten mit den Cursortasten (um von A nach B zu gelangen: Die Maus auf den Fehler, einmal anklicken und schon ist der Cursor zur Stelle).



Zum Thema Arbeits-Tempo kann ich mich kurz fassen, das Spektrum reicht bei der Textverarbeitung von einem 'uuuaaaah' bis zu einem entzückten 'ooohh'. Insgesamt hebt sich die Geschwindigkeit aber deutlich von der meines Genie ab: Eine Grafikfläche mit einem berechneten Muster füllen (384\*192 Punkte groß), ist auf dem ATARI über 10 mal schneller als auf meinem Genie (mit 3,54Mhz Taktfrequenz). Bei solchen Zahlen kommt schon Freude auf. Mit dem Speicherplatz ist es - wen wundert's - nicht viel anders. 1 Mbyte sind für Z80-Verhältnisse 'einfach umwerfend'. Es soll aber nicht verschwiegen werden, daß man die auch bitter nötig hat. Ein Text mit vielen Zeichnungen (ca. 27) und einer Länge von ca. 120KByte verschlingt glatt 400KByte an RAM ! Rechnet man das Übrige noch dazu (Bedarf des TOS, die Textverarbeitung und einige Accessories etc.), dann bleiben nur magere -300KByte RAM übrig. Das reicht gerade noch für eine kleine RAM-Disk und dann ist die Fude voll. Komfort hat so seine Dimensionen. Die Software. Ein nicht unwichtiger Punkt. Meine Sammlung besteht zu gut 80% aus PD-Software, deren Qualität zum Teil äußerst überzeugend ist. So finden sich neben makrofähigem Text-Editor, Hochsprachen und Grafikprogramme, deren Leistungen sich sehen lassen können. Ich bin auf diesem Wege preiswert zu einer guten Softwarebasis gekommen. Lediglich mit den Viren muß man aufpassen, hier ist der ATARI eine ausgesprochen heißgeliebte Spielwiese, wie ich am eigenen Leibe erfahren mußte (ich durite meine gesamte 80 Disketten umfassende Sammlung von diesen lieben Tierchen befreien). Trotz aller Fallen und Tücken bin ich mit dem Rechner zufrieden. Das Nachsehen wird auf Dauer wohl mein Genie haben, - leider - den die Kiste hat sich all die Jahre - immerhin 8 an der Zahl - wacker geschlagen.

*Incluß Kart*

## Mein Zweitrechner - der Atari ST

55 Vor ungefähr einem Jahr hatte ich diese ständige Ungewißheit endlich satt: läuft mein Genie IIs morgen oder läuft es nicht? In den Ausfallzeiten ruhte meine gesamte schriftliche Arbeit, was auf die Dauer nicht mehr zu ertragen war. Also mußte ein Zweitrechner her, damit wenigstens immer einer betriebsbereit ist.

### Nur ein Atari ST?

Das Einsatzgebiet des Rechners ist für mich die Erledigung allen Schreibkrams. Alle Briefe, Referate, Mitschriften, Artikel usw. werden getippt, weil sie sonst niemand (auch ich nicht) lesen könnte. Nebenbei habe ich noch das Anliegen, etwas zu lernen, sprich: etwas zu programmieren.

Das Budget war ziemlich klar: mehr als 2000,- sollte der Zweitrechner nicht kosten. Damit fiel mein Spitzenkandidat, der Macintosh, gleich aus der Wertung. Der Amiga war die nächstbeste Lösung, aber wegen seines Farbmonitors kostete er über 2000 DM. Außerdem sah ich nicht, wozu ich zum Briefeschreiben Farbe brauchte, ganz abgesehen davon, daß ich niemand in meinem Umfeld aufreiben konnte, der den Rechner hatte. Aber ohne einen Erfahrungsaustausch schien mir die Sache ziemlich witzlos.

Nachdem ein Prof-180X als reiner CP/M-Rechner wegen zu hoher Kosten deklariert wurde, blieben nur zwei Kontrahenten übrig: der Atari ST und die gesamte Klasse der IBM-Kompatiblen. Die IBMs kannte ich mehr oder weniger aus der Uni und diversen Zeitschriften. So war mir klar, daß die Technik nur wenig über dem Niveau eines CP/M-Rechners lag. Die 640 KB Speicher waren zwar eine Verzehnfachung, aber die Programme blähten sich immer mehr auf, weil sie auf verschiedene Grafikkarten und andere Inkompatibilitäten der angeblich Kompatiblen Rücksicht nehmen mußten. Somit war der Speicher klar zu klein gewählt. Da selbst das Betriebssystem MS-DOS nahezu vollständig vom CP/M abgekupfert wurde, war eine Steigerung kaum zu erwarten. Ein AT-Kompatibler wäre interessant gewesen, aber die Dinger kosten sogar heute noch mehr als ich bezahlen könnte. Ein PC war wegen seiner 360-KB-Laufwerke zu klein (die meisten Programme belegen mehrere Disketten), also blieb nur ein XT mit Festplatte. Bei meiner Preis-Vorgabe hätte ich wohl ein Fernost-Produkt mit 640 KB Speicher, 360-KB-Laufwerk, 20-MB-Festplatte, HerculesGrafik-Karte und (normalen) Monochrom-Monitor bekommen können. Dem stand ein Atari 1040 STF mit 1 MB Speicher (auf 4 MB ausbaubar), 720-KB-Laufwerk, 640x400-Grafik und einem absolut flimmerfreien Monochrom-Monitor gegenüber. Die Festplatte fehlte hier, dafür gaben das 192 KB große ROM mit dem kompletten Betriebssystem und das starre Konzept (nur 3 Grafik-Auflösungen etc.) zu der Hoffnung Anlaß, daß die Programme eher kurz sein würden. Somit wurde die Festplatte eher zum Luxus als zur Notwendigkeit.

Der Rest war eine reine Glaubensfrage. Mein Bekanntenkreis spaltete sich in zwei Hälften: Die eine wollte mit dem Rechner Geld verdienen und hielt sich deshalb an das, was in der Wirtschaft Standard war, also IBM. Die andere Hälfte suchte einen Rechner, von und mit dem man etwas lernen konnte und dessen Technik und Software nicht allzu rückständig war; somit herrschte hier der Atari vor (seltsamerweise nicht der Amiga). Mit dem echten 16-Bit-Prozessor Motorola-68000 und der grafischen Benutzeroberfläche GEM samt Maus bot der Atari was Neues gegenüber dem IBM, was mich reizte und letztlich den Ausschlag gab.

### Welche Erfahrungen habe ich gemacht?

96 Vom Auspacken bis zum Formatieren und Kopieren der ersten Diskette verging ca. eine halbe Stunde. Ausschlaggebend war dabei die im wahrsten Sinne des Wortes "kinderleichte" Bedienung per Maus, schon im normalen Betriebssystem. Die Tastatur ist nur sekundär und in den ersten Tagen brauchte ich sie überhaupt nicht. Ärgerlich war von Anfang an, daß (beim 1040 STF) die Tastatur gleich noch den Rechner und seitlich das Laufwerk enthält (an das man so schlecht herankommt), womit die ganze Sache unbeweglich wird. Positiv dagegen fiel mir sofort der Monitor auf: das Teil ist wirklich absolut flimmerfrei und ich möchte mit keinem anderen mehr länger arbeiten. Das schwarz-weiße Bild gibt mir oft den Eindruck, daß ich ein Blatt Papier vor der Nase hätte.

Das Betriebssystem läßt sich problemlos bedienen, was für die "echten" Programmierer ein Problem darstellt. Aber glücklicherweise haben die Designer des Systems einige Schwächen eingebaut, die bis heute nicht ausgemerzt sind. So wird das Copy-File-By-File mit einem Laufwerk zur Tortur, weil für jeden File die Disks zweimal gewechselt werden müssen (obwohl das RAM ausreicht, um alle Files auf einmal zu kopieren). Da hilft dann nur eine Ramdisk, ein zweites Laufwerk oder eine Festplatte.

Zu den (in meinen Augen) Systemfehlern gehören auch das Betriebssystem im ROM, das bekanntermaßen extrem fehlerhaft ist und in einem statischen RAM weitaus besser aufgehoben wäre, sowie die Begrenzung des Hauptspeichers auf 4 MB (neben der schlecht möglichen Aufrüstbarkeit). Da der Prozessor 16 MB adressieren kann, ist es ziemlich dämlich, das nicht auszunutzen. Aber wenigstens ist es mehr, als die IBMs bieten. Die haben allerdings (meistens) eine ordentliche Tastatur, was beim Atari nicht der Fall ist. Von der des C64 trennt sie nur wenig. Leider läßt sich nicht mal problemlos eine andere anflanschen. Dafür hat der User alle Tasten, die er sich wünschen kann, sogar eine "HELP"-Taste, die aber kaum ein Programm benutzt.

Praktisch sind die 3.5"-Laufwerke, denn die Disketten kann ich endlich in der Jackentasche transportieren, was bei den 5.25"-Disks zu leichten Problemen führte (8" sind da besser, die können schon als Kopfbedeckung dienen). Außerdem läßt sich über 720 KB nicht meckern, wenn auch mehr möglich wäre (s. Amiga oder neuerdings auch IBM).

Die Dokumentation ist sowohl für den absoluten Anfänger als auch für den versierten Programmierer nicht ausreichend. Anfänger müssen sich für teures Geld irgendwelche zweifelhaften Lehrbücher von Data Becker o.ä. kaufen, für Programmierer sieht es trauriger aus. Denn alle verfügbaren Veröffentlichungen kranken an einem Problem: Atari hat das System schlecht dokumentiert. Das Betriebssystem ist fast das Original-GEM, aber eben nur fast. Die Änderungen liegen im Dunkeln. Zusammen mit den im BIOS etc. eingebauten Fehlern ist das Chaos komplett. Ich habe bis jetzt noch kein Programm gefunden, das fehlerfrei läuft, d.h. auch mit installierten Ramdisks, verschiedenen Accessories usw. zusammenarbeitet. Bezeichnend ist, daß noch nicht einmal die "Type File"-Funktion des GEM richtig arbeitet.

### Die Software

Die Programme auf dem Atari sind fast alle ohne Anleitung bedienbar, denn die Maus ist allgegenwärtig. Leider meint jedes Programm, die Maus müßte anders eingesetzt werden, was schon zur Verwirrung beiträgt.

Leider verzichten die Programmierer darauf, die verfügbaren Tastatur-Kommandos irgendwo leicht erreichbar zu dokumentieren, obwohl sie meistens vorhanden sind. So wird man oft zu den zeitraubenden Pull-Down-Menüs gezwungen, obwohl ein Ctrl-Code kürzer und verfügbar wäre.

Außerdem scheinen die Programme auch nie den Test-Status zu verlassen. Von meinem favorisierten Textprogramm, STEVE, gibt es inzwischen Version 3.08 (nach 2.3, 2.5, 3.0), und trotzdem mußte ich schon beim ersten Einschalten einen Fehler feststellen. Ebenso sieht es in fast allen Bereichen aus:

Auch ich möchte mich, da ich den 1040 ST nun auch schon eineinhalb Jahre in Betrieb habe, kurz zum Atari-Thema melden.

Unfertige Programme werden ausgeliefert, wobei oft noch Grundfunktionen fehlen. Die Verbesserungen bestehen dann darin, irgendwelche exotischen Drucker oder Scanner zu unterstützen, während die Grundfunktionen weiter fehlen. Dafür habe ich aber eine Leistung zur Verfügung, von denen andere Rechner-Besitzer nur träumen können, während ich selber von der Peripherie träume, mit der diese Leistung erreichbar wäre.

Ebenso steigt der Umfang der Programme immer mehr an, obwohl die Ausmaße derer der IBM-Familie noch lange nicht erreicht sind. Der Grund ist mir noch unklar. Mitverursacher sind sicherlich die Einbindung in GEM, die Platz schluckt, und die Vernachlässigung der Assembler-Programmierung zugunsten von C oder Basic, was bei den miesen Assemblern, die es bis jetzt gibt, kein Wunder ist.

Interessant sind die verschiedenen Betriebssysteme und Emulatoren, die es für den Atari noch gibt: RTOS und OS/9 für Multitasking (der Benutzer als störender Rand-Prozeß), CP/M-80 für die alten Freaks, PC/MS-DOS für die Unbelehrbaren, Macintosh für die Neidischen, C64 für die Spieler.

Neben der Textverarbeitung (ca. 90%) benutze ich den Rechner höchstens mal, um eine Grafik zu zeichnen, aber da steht meine nicht vorhandene künstlerische Begabung dagegen. Oder ich ziehe aus der umfangreichen Spiele-Kiste ein Grafik-Adventure und und erinnere mich wehmütig an die Zeiten, als die Klötzchengrafik von "Forbidden City" mit dem unverständlichen Sound das allerhöchste der Gefühle war. Programmiert habe ich dagegen auf dem Atari bis jetzt nur, wenn es sich nicht verhindern ließ, und auch nur mit Pascal. Zwar bin ich inzwischen glücklicher Besitzer eines Modula, aber das Turbo-Modula von CP/M ist immer noch besser.

#### Warum verschrotte ich mein Genie IIs nicht?

Die Frage ist einfach zu beantworten: Ich benutze zwar den Atari (als reiner Anwender), aber programmieren tue ich nur am Genie IIs. Außerdem läßt es willig meine bescheidenen Umbauversuche über sich ergehen, die beim Atari unmöglich sind (gesockelte ICs? Veraltet!). Zwar sind die Anwenderprogramme auf dem Atari leicht zu bedienen und sehr leistungsfähig, aber zum Programmieren bietet die Kiste mir nicht genug. Auf meinen IIs kenne ich wenigstens noch jedes Byte persönlich und weiß genau, warum die Kiste abstürzt; beim Atari hängt das allein vom Wetter und sonstigen mystischen Dingen ab.

#### Fazit

Wer gezwungen ist, mit dem Rechner Geld zu verdienen, muß sich wohl einen IBM-Kompatiblen kaufen. Diese Leute tun mir leid, denn sie bekommen wenig für ihr Geld und müssen bei den Leistungsdaten ihres Rechners noch vor CP/M-Rechnern Angst haben.

Wer allerdings mal was ganz anderes möchte, wie eine echt bedienungsfreundliche Oberfläche (ist auch nicht jedermanns Sache), einen Super-Monitor, viele ausgeflippte Programme, ein zukunftsweisendes Betriebssystem und einen echten 16-Bitter, der sollte sich mal den Atari ansehen. Allerdings ist das Ding bestenfalls ein Prototyp für den Hausgebrauch und mehr in die Kategorie "Spiel-Kram" einzuordnen. Aber hier kann jeder sehen, wo der Weg in Zukunft langgehen wird, denn nicht umsonst zieht IBM jetzt mit "Windows" und ähnlichem Zeug hinterher. Aber die Leistung in Reinkultur gibt es wohl nur beim Amiga (Multitasking, Farbgrafik) oder dem Mac II (standardisierte Programme, durchdachtes Betriebssystem).

Gerald Schröder

Grundsätzlich kann ich mich den gedruckten Worten meiner Vorschreiber anschließen. Das Kopieren mit einem Laufwerk ist echt Quälerei! Für ein einzelnes File benötigt der Atari ca 4-5 Diskettenwechsel, die er jeweils mit einer Meldebox anzeigt. Arbeitet man mit einem Zweitlaufwerk -wie ich- oder RAM-Disk, fällt das nicht mehr ins Gewicht. Hier sind die Bytes ruckzuck umgeschaufelt.

Als einen weiteren großen Mangel finde ich die Geschichte mit den drei verschiedenen Auflösungen. Man sollte schon einen Farb-Multisync (oder je einen Farb-/SM-Monitor) auf dem Tisch stehen haben um alle Programme fahren zu können. Hier hätte man noch mehr tun können. Mindestens mit dem mitgelieferten SM-Monitor sollten alle Auflösungen (bei Farbe dann in Grautönen) darstellbar sein. Platz genug für eine entsprechend gestaltete Platine wäre schon, es ist da ja eh nicht viel drin. Vielleicht ist auch sonst nichts mehr verdient. Ich jedenfalls hatte mich damals für einen Farbmonitor entschieden. So kann ich zumindest die "mittlere" (640x200) und die "geringe" (320x200) Auflösung betreiben.

Nachdem ich mich nun gleich über die zwei -von mir aus gesehen- negativen Seiten hergefallen bin, möchte ich natürlich auch die Pluspunkte erwähnen, die mich zu der Kaufentscheidung Atari 1040 ST brachten.

Ich wollte auf jeden Fall einen modernen Rechner, mit dem man farbige Grafiken und Animationen erstellen und bearbeiten und zum Relaxen ein schickes Game usen kann. Dies fehlte mir bei dem guten TRS-80 ganz und gar. Der Tandy ist aber auch weiterhin für mich im Einsatz. Da ich auch noch nichts umgeschrieben oder angepasst habe, verwende auch noch alle Programme sei es nun für Text, Daten und den anderen Krimskrans der sonst noch läuft.

So standen für mich damals zwei Rechner zur Auswahl. Atari oder Amiga. Der Amiga ist zwar in Grafik und Sound noch besser wie der Atari, aber auch teurer. Weiterhin war der Amiga Softwaremäßig schlecht unterstützt -und wie die Zeit zeigt ist er auch heute noch hauptsächlich ein guter Spieler-.

Auf Seite 62 gehts weiter ...

# VERKAUFE

TANDY Model 4P mit 2 Laufwerken 80 Spuren, DS, DD und jeder Menge Software (Preis VB 800 DM)  
 Nadeldrucker TANDY, 9 Nadeln, Papierbreite DIN A3-  
 (Preis VB 200 DM)

Wer Interesse hat, kann sich schriftlich oder telefonisch melden:  
 Günther Wagner, Heubergstraße 8, 8201 Neubuern,  
 Tel. 08035/3927 (ab 19 Uhr oder am Wochenende)

## Verkaufe für fast nichts

- Model I, Expansion, grüner Monitor, Voice-Synthi, 3 Laufwerke, Expsion-MX-80
  - Genie (defekt), Expansion, 2 Laufwerke
  - 3 Kartons Dokumentation, 80-US, Software & Anleitungen
  - Rechnertisch
- außerdem:
- dt. Model III, 2 Laufwerke
  - Model I, Expansion, s/w-Monitor, 2 Laufwerke

bitte melden bei: Horst Schuller  
 An der Pinnau 17  
 2359 Henstedt-Ulzburg  
 Tel. 04193/4158

*Verkaufe wegen Systemwechsel:  
 TRS 80 Mod III mit  
 RS 232 (alt. Tandy) und  
 2 Laufwerken 40Track SS DS  
 + um funkt. sicherer Literatur  
 + div. Software  
 VB*

*Christian Schenk  
 Ouischerer Str. 33  
 4000 Düsseldorf 38  
 0271/493374*

Halt, hier hat sich gerade noch was gefunden. Unser ehemaliges Mitglied Harald Trapp hat noch etwas zu verkaufen. Bitte setze folgende Anzeige ins Info: DIN A3-Plotter von DENSON serielle und parallele Schnittstelle 4 Farben, Einzelblatteinzug, 1/2 Jahr alt wegen Wechsel zu A1-Plotter zu verkaufen Preis 1800,- DM (50% des Neupreises!)

- Verkaufe aufgrund Räumungsaktion:
- TRS 80, Model 4 P, 2 x 64 kB, 2 Laufwerke
  - ALCOR-Pascal (Original) für Model 1.3 u. 4 (M3-Modus)
  - 80-Micro Jahrgänge 1985, 1986 u. 1987
- Preis: VS  
 Klaus Hermann, Tel.: 07127/71945

## Er prompt...

Da ich es endlich geschafft habe, meinen Prommer zur Zusammenarbeit zu überreden, stehe ich ab sofort auch für 'brennende Fragen' zur Verfügung. Er schafft alles vom 2716 bis 27256, wenn ich einen Adapter gebaut habe auch 27512 und größer.

Alexander Schaid

*Für den neuesten Krimi-Happening-Sitting in & out:*

# Erfahrungsaustausch über dBASE II

gesucht!  
 Vorzugsweise mit TRS-80-User m. RAMDISK!  
 Aber auch mit anderem Freak. - Bitte melden bei:  
 K.-J. Mühlenbein, Am Mönchgarten 28, 6940 WEINHEIM  
 → Es wird spannend! ←

*Projekt: Börsenkurse etc. per VideoText in den Rechner  
 Ihr könnt das Videotextangebot (weiter. Verkehr. Nachrichten, Sport, Bör-  
 senkurse usw.) in den Rechner übernehmen. Müßig dazu ist nur ein Adapter  
 einer Schweizer Firma (TRS 1598 von Nordex AG, Postfach 670, CH-3000 Bern  
 14. Tel. CH/031/250405), eine serielle Schnittstelle und (noch) ein IBM-  
 Kompatibler.  
 Dieter Kasper (ehemaliges Clubmitglied) würde diese Sache nun auch gerne  
 für unsere Rechner (unter CP/M oder Newdos) nutzbar machen. Wer Lust hat,  
 bei dem Projekt mitzumachen, wende sich an  
 Dieter Kasper  
 Schillerstr. 30  
 8952 Marktobendorf  
 Tel. 08942/1690*



**H. Pfisterer  
Bietigheim-Bissingen**

Hermann Pfisterer · Werkzeugfabrik · D-7120 Bietigheim-Bissingen

Herrn  
Gerald Schröder  
Am Schützenplatz 14  
  
2105 Seevetal 1

**Zentrierbohrer  
Stufenbohrer  
Sonderwerkzeuge**

03.10.88 wpf

**Teletex - Modem für CP/M-Computer**

Sehr geehrter Herr Schröder,

dem Chip-Magazin habe ich entnommen, daß Sie und Ihr Computer-Club sich mit CP/M-Rechnern beschäftigen.

Wegen Umstellung auf ein neues Gerät habe ich abzugeben, und biete Ihnen oder Ihren Club-Mitgliedern an:

1 Stück Teletex-Modem von Triumph Adler für serielle Schnittstelle, für TEX-ASS Textverarbeitungsprogramm gebraucht, voll funktionsfähig, postzugelassen bisher mit PC Alphatronic P3 von TA betrieben.

Verkaufspreis: Verhandlungssache

Bitte setzen Sie sich bei Interesse mit mir in Verbindung.

Mit freundlichen Grüßen  
Hermann Pfisterer  
Werkzeugfabrik

Wolfram Pfisterer

... Fortsetzung von Seite 58

Für den Atari hingegen gibt es super (und auch funktionierende !!) Software zur Bild-Gestaltung und -Animation. Als Beispiel möchte ich hier nur die Programme der "Cyber-Serie" von Antik oder "Imagic" aus Heidelberg anführen.

Besonders für mich als Video-Filmer wird die Sache durch einen Digitizer (welcher auf den ROM-Port gesteckt wird) erst richtig interessant:

Filmen, die Bilder digitalisieren, rein damit in den Atari, verändern...ergänzen...animieren, raus als Videosignal und fertig ist ein Vorspann für eigene oder aufgenommene Videofilme.

Dies war für mich die Entscheidung!

Wer die Preise für einen Schriftgenerator und weitere wichtige Zutaten der Videofilmerei kennt, wird wissen, daß mit ST und entsprechender Software mindestens die "halbe Miete" gespart ist -wie man so schön sagt-.

Abschließend muß ich sagen, daß ich mit dem Atari und Farbmonitor sehr zufrieden bin. Am Anfang war die Software (bis auf Spiele) sehr auf SW ausgerichtet, aber jetzt laufen die meisten Sachen auch in der mittleren Auflösung. Es ist also auch kein Nachteil mehr "nur Farbe" zu haben.

Wenn man dann seine eigenen Animationen laufen hat und dann zur Entspannung noch etwas "gamet" möchte man fast gar nicht wieder zurück zum Tandy, der sich bernsteinfarben auf dunkelbraunem Grund mit einem "NEWDOS 80 READY" meldet.

Auf ein fröhliches, farbiges Pixeln  
Jens Neuerer

HEFT  
26  
Januar  
1989

Oktober 1988

Sehr geehrte Damen und Herren,

der Bedarf an EDV-Verbrauchsartikeln steigt ständig an. Speziell für Computer-Club's haben wir daher eine Preisliste gängiger Farbbänder unseres Vertriebsprogramms zusammengestellt. Unser Lieferumfang umfaßt ca. 1000 Schreib- und Korrekturbänder sowie etwa 100 verschiedene EDV-Etikettenformate und Abpackungen. Wir liefern sowohl Endlosetiketten für Tintenstrahldrucker und Matrixdrucker, wie auch Einzelblattware für Laserdrucker.

Hier zwei Preisbeispiele:

<b>ENDLOSETIKETTEN</b>		für Matrixdrucker	
=====			
Format	70,0 x 70,0 mm	z.B. für 3,5 Zoll-Disk.	
Farbe	weiß		
Kartoninhalt	2.000 Etiketten		
Preis je Karton bei Abnahme von:			
	1 Karton	DM 54,72	inkl. Mwst.
	ab 3 Karton	DM 51,98	inkl. Mwst.
	ab 6 Karton	DM 49,25	inkl. Mwst.

<b>Format</b>		89 x 36 mm		z.B. als Adressetikett	
<b>Farbe</b>		weiß			
<b>Kartoninhalt</b>		4.000 Etiketten			
Preis je Karton bei Abnahme von:					
	1 Karton	DM 43,32	inkl. Mwst.		
	ab 3 Karton	DM 41,15	inkl. Mwst.		
	ab 6 Karton	DM 38,99	inkl. Mwst.		

Gern senden wir Ihnen auf Wunsch weitere Informationen zu.

Mit freundlichen Grüßen

Die Lieferung erfolgt ab DM 200 - Warenwert frei Haus, ansonsten DM 5,70 Versandkostenpauschale. Lieferung innerhalb 10 Tagen - 2% Skonto oder 30 Tage Kasse. Alle belieferten Waren bleiben bis zur vollständigen Entgegennahme Eigentum.

An alle Computerclub's und Vereine  
im Bundesgebiet

Auszug aus unserem Angebot:

Stand 10/88

**Qualitätsfarbbänder**

- made in germany
- Einzelverpackung
- Lieferzeit i.d.R. 7 Tage
- Zahlbar nach Rechnungserhalt 10 Tg. -2% ... 30 Tg. netto

**Brother**

M 1409 ==>DM 10,75 /// M 1509/1709 =>DM 12,70/// M 1724 L ==>DM 13,90

**Citizen**

LSP 120 D ==>DM 10,10 /// MSP 120 D ==>DM 10,10/// MSP 10 ==>DM 8,50  
MSP 10 E ==>DM 8,40 ///

**Commodore**

MPS 801 ==>DM 8,50 /// MPS 802 ==>DM 9,20/// MPS 803 ==>DM 8,20  
MPS 1000 ==>DM 7,40 /// MPS 1200 ==>DM 10,10/// MPS 2000 ==>DM 11,35  
MPS 2010 ==>DM 13,15 /// MPP 1361 ==>DM 8,50/// VC 1515 ==>DM 12,20  
VC 1525 ==>DM 12,20 ///

**Epson**

FX 80 ==>DM 8,50 /// FX 100 ==>DM 10,20/// GX 80 ==>DM 7,40  
LQ 800/850 ==>DM 9,50 /// LQ 1000/1050 >DM 10,95/// LQ 1500 ==>DM 10,95  
LQ 2500 ==>DM 10,40 /// EX 1000 ==>DM 9,85///

**Fujitsu**

DL 2300 ==>DM 27,20 /// DX 2100/2200 >DM 11,50/// DX 2300/2400 DM 11,65

**IBM**

4201 (6328829) DM 10,65 /// 4202-001(1040150) 14,15/// 4207 XL 24 DM 11,90  
4208 XL 24 DM 14,85 ///

**NEC**

P 5 ==>DM 11,50 /// P 6 ==>DM 11,35/// P 7 ==>DM 13,15  
P 2200 ==>DM 12,05 /// P5/9XL (Multist.) 13,00///

**OKI**

ML 182/192/194 DM 8,50 /// ML 292 schwarz DM 17,50 / ML 292 color DM 26,00  
ML 293 schwarz DM 19,20 /// ML 293 color DM 29,40/// ML 390/391 >DM 13,00

**Panasonic**

Kx-P 1080... DM 11,15 ///

**Schneider**

DMP 2000/3000 DM 9,30 /// NLQ 401 ==>DM 8,20/// Joyce Synth. DM 10,95  
DMP 4000 ==>DM 13,25 ///

**Seikosha**

SP 800/1000 A DM 10,95 /// GP 80 ==>DM 12,20/// GP 100 ==>DM 12,20

**Star**

NB/ND/NL 10 =>DM 10,40 /// LC 10 ==>DM 9,25/// NB 24-10 ==>DM 12,30  
NB 24-15 ==>DM 13,55 /// ND/NR/NX 15 =>DM 13,75///

Lieferung ab Warenwert DM 200 frei Haus, sonst DM 5,70 Versandkostenanteil. Unser Lieferumfang umfaßt etwa 1000 verschiedene Schreib- und Korrekturbänder. Bitte fordern Sie weitere Unterlagen.

## Wenn einer eine Reise tut...

...kann er viel berichten. Wie Hartmut ja schon geschrieben hat, habe ich einen Akustikkoppler und bin öfters in den Münchner Mailboxen unterwegs. Hin und wieder stößt man dabei auf ganz witzige Sachen, die es durchaus wert sind, auch woanders veröffentlicht zu werden. In loser Reihenfolge werden hier also ab und zu solche Fundstücke auftauchen, wenn sie halbwegs einen Bezug zur Computerei haben.

"humor": Software verhindert Maschinenfehler  
 --- Di, 18.10.88 / 09:18:16 von LORC

Wie aus den jüngsten Veröffentlichungen der Amerikanischen Gesellschaft für digitale Impulsunterdrückung (ASPIC) hervorgeht, ist es nach jahrelanger Forschungsarbeit gelungen, ein Software-Programm zur Ausschaltung von Maschinenfehlern zu entwickeln. Die Routine wird in Fachkreisen als Meilenstein in der Geschichte der Systemtechnik bezeichnet, da sie die immensen, durch Maschinenfehler bedingten Zeitverluste weitgehend ausschaltet.

Der Programmname "Oreana" ist vom lateinischen oreans (lasset uns beten) abgeleitet. Die Routine ruft in bestimmten Zeitabständen Gebete ab, in denen himmlischer Beistand für die Bewahrung des Arbeitsspeicherinhalts, die Festigkeit von Magnetisierungsschichten und solcher Lötstellen erfleht wird, die besonderen mechanischen Anfechtungen ausgesetzt sind.

Die liturgische Struktur der Oreana-Routine gestattet den wahlfreien Aufruf wirkungsvoller, extern gespeicherter Stoßgebete in Lateinisch, Hebräisch oder Cobol. Durch die Oreana-Unterroutine "Misere" werden dreimal täglich sogenannte zyklische Messen zelebriert (Permanentanzeige auf dem Bildschirm). Die ministrierenden Benutzer beschränken sich während dieser Zeit lediglich auf das Einlegen der Diskette und das Eingeben sakraler Nachrichten über die Tastatur (zum Beispiel "... und mit Deinem Geiste" oder "Kyrie eleison").

Litaneien in Hebräisch oder Cobol können direkt über einen freien Kommunikationskanal (FKK) in die Zentraleinheit eingegeben werden. Lateinische Psalmen (Kennsatz: "te.deum") sind jedoch peripher über Diskette einzuspeichern. Hebräische oder Cobol-Litaneien haben sich als besonders wirkungsvoll erwiesen. Obwohl von den Herstellerfirmen bereits fertige Gebetsdisketten für alle denkbaren Maschinenfehler geliefert werden, kann der versierte Programmierer von sich aus hinter jedem Amen-Block Prozeduren mit Eigenerflehung (Kennsatz: "jaul") eingeben. Spezielle Nothelfergebete stehen für Multitasking und wichtige Tests zur Verfügung. Nach letzten Testläufen mit Oreana wird die durchschnittliche Maschinenausfallzeit um 98,2 Prozent reduziert. Wie aus gutunterrichteten Kreisen verlautet, wird Oreana in absehbarer Zeit auch für die Ausmerzungen von Hardwarefehlern eingesetzt werden können.

Vom Verteidigungsministerium wurde inzwischen ein kybernetisch-theologischer Arbeitskreis eingesetzt, der sich mit der Entwicklung eines Software-Systems zur Abschaffung des menschlichen Versagens befaßt. Die Fertigstellung dieser Programme (Arbeitstitel "Neaculpa") wird allerdings noch geraume Zeit in Anspruch nehmen.

(Hans Michael Engelke, DOS International 11/88, Seite 14)

Mancher geplagte Model-I-Besitzer wäre wohl froh um so ein Programm...

Alexander Schaid

## Tandy- und Genie-Geräte auf dem Gebrauchtcomputermarkt

Zufällig fand ich in einer CHIP 8/88, die ihren Weg vom Vorzimmer eines Zahnarztes über Harald Mand zu mir gefunden hat (kaufen würde ich mir die CHIP zur Zeit nicht!), eine Zusammenstellung von Preisen für Gebrauchtcomputer. Mit Marktanalyse oder ähnlichem hat die Zusammenstellung sicher nichts zu tun. Vielmehr haben die CHIP-Leute die Anzeigenteile ihrer Zeitschrift durchsucht und jeweils den niedrigsten und den höchsten Preis für ein bestimmtes Gerät ermittelt. Also kein statistischer Schnitt und auch keine tatsächlich erreichten Verkaufspreise, sondern nur Forderungen. Trotzdem hier die Ergebnisse für ein paar Rechner:

Modell	Ausstattung/Zubehör	Gebrauchtpreislage im Markt			
		1985	1986	1987	1988
TRS 80 M.1	48k, LW, ERG, SW	550-1500	800-2250	650-1100	600-1600
TRS 80 M.3	48k, 2 LW, SW	1000-3300	1200-1800	490-1200	400- 900
Genie I	48k, LW, Mon., Dru.	1600	500-1700	350-1000	350- 700
Genie II	64k, 2 LW, Mon., SW	500-2600	1000-2000	960-1100	750-1000

Soweit die Tandy- und Genie-Maschinen. Nun ein paar vergleichbare Rechner.

CPC 464	64k, LW, SW	750- 800	500- 790	350-2000	370-1000
TA P2	48k, Drucker, SW	2800-3200	2000-2600	1200-2000	800-1700
TA PC	64k, 2 LW, Mon., SW	1200-2700	950-2800	700-1300	700-1500
Apple II+	64k, 2-3 LW, Monitor, Drucker, SW	1500-3200	750-3300	450-2000	700-1350
C64	LW, Mon., Dru., SW	500-1400	600-1200	320-1300	-1200

(CPC 464 von Schneider, TA's von Alphatronic)

Natürlich sind die Preise bei allen Maschinen sehr von der Ausstattung, dem Zustand und der mitgelieferten Software und Dokumentation abhängig. Einen Reim auf die angegebenen Zahlen muß sich jeder Leser also selbst machen!

Diese Liste soll übrigens keineswegs eine Aufforderung sein, sich seines Rechners zu entledigen! Auch bei mir laufen zwei Rechner und haben durchaus zu tun. Zudem gibt es Dinge, die ein (man mit einem) TRS 80 oder Kompatiblen durchaus besser (machen) kann als so mancher AMIGA/ATARI ST oder PC-Clone. Und wenn es nur der Umstand ist, daß man an den "alten" Z80-Geräten auch einmal Hand (sprich "Lötkolben") anlegen kann, ohne gleich Angst haben zu müssen, daß der geliebte Computer nie mehr zum Leben erwacht.

In diesem Sinne viel Spaß beim Computern und Basteln,

Hartmut Obermann

# dBase II, III und III Plus im Überblick

67

Auch diesen Monat befaßt sich das DOSSier mit dBase. Nachdem in der letzten Ausgabe der gesamte Befehlssatz von dBase abgedruckt war, folgen diesmal drei Tabellen, die sämtliche Funktionen von dBase sowie alle »set« und »set to«-Befehle mit Voreinstellung und Bedeutung auf-führen.

Dieses DOSSier ist die Fortsetzung zum DOSSier aus der letzten Ausgabe von DOS International. In der ersten Tabelle sind sämtliche Funktionen des Datenbanksystems dBase übersichtlich mit Bedeutung aufgelistet. Zusätzlich ist angegeben, in welchen dBase-Versionen die jeweilige Funktion existiert. Tabelle 1 enthält in der letzten Spalte zu jeder Funktion auch die Angabe, zu welcher Gruppe diese Funktion gehört.

Es ist nicht einfach, die dBase-Funktionen in Gruppen zu unterteilen, weil sich die Bereiche oft überschneiden. Die Abkürzungen für die in Tabelle 1 verwendeten Funktionsgruppen sind:

- D - Datum-/Zeitfunktion
- E - Environment- beziehungsweise Umgebungsfunktion
- F - Dateifunktion
- N - numerische Funktion
- S - Zeichenkettenfunktion

Auch die in der Tabelle verwendeten Bezeichnungen »A«, »D« und »N« haben eine bestimmte Bedeutung:

- »A« steht für einen Ausdruck oder eine Zeichenkette,
- »D« bezeichnet einen Datumsausdruck oder eine Datumsvariable, und
- »N« repräsentiert einen numerischen Ausdruck.

Bei Bedarf sind an diese Kennungen als Indizes Kleinbuchstaben angehängt.

Die Tabelle der »set«-Schalter (Tabelle 2) beinhaltet Angaben zu den Voreinstellungen der jeweiligen Schalter, das heißt, die Schalterpositionen, in denen sich die Schalter zu Beginn einer dBase-Sitzung befinden.

Tabelle 3 faßt die Bedeutung der »set to«-Befehle zusammen. Achten Sie darauf, daß Sie zur Vereinfachung Ihrer Eingaben sowohl im Direkt- als auch im Programmmodus auf die Befehlsabkürzungen zurückgreifen können, die sich jeweils aus den ersten vier Buchstaben eines Befehls-wortes zusammensetzen. Beispielsweise hat die Abkürzung

disp  
die gleiche Wirkung wie  
display  
(Ulrich Spranger/ma)

Funktion	Bedeutung	dBase II	dBase III	dBase III Plus	Funktionsgruppe
*	prüft, ob Datensatz zum Löschen markiert ist	x	-	-	F
act(A)	aktuelle Datensatznummer	x	-	-	F
alt(A,Nb)	wandelt »A« in Großbuchstaben um	x	-	-	S
asc(N)	Zeichenkette aus »A«, beginnend bei Position »Nb« und »Nb« Zeichen lang	-	-	-	S
abs(N)	absoluter Wert von »N«	-	-	-	N
asc(A)	gibt ASCII-Code des ersten Zeichens von »A« aus	-	x	x	S
boff	prüft, ob Datensatz vorliegt	-	x	x	F
odow(D)	gibt Wochentag von »D« als Wort aus	-	x	x	D
chr(N)	wandelt Code von »N« in ASCII-Zeichen um	x	x	x	S
omonth(D)	gibt Monat von »D« als Wort aus	-	x	x	D
col(D)	aktuelle Bildschirmspalte	-	x	x	E
ctod(A)	wandelt Zeichenvariable in Datumsvariable um	-	x	x	D
date()	Systemdatum	x	-	-	D
day(D)	gibt den Tag von »D« als Zahl aus	-	x	x	D
dbf()	Name der geöffneten Datenbank	-	-	-	F
deleted()	prüft, ob der aktuelle Datensatz zum Löschen markiert ist	-	x	x	F
down(D)	gibt den Tag von »D« als Zahl bezüglich der Woche aus	-	x	x	D
dtoc(D)	wandelt Datumsvariable in Zeichenvariable um	-	x	x	D
diskspace()	freier Speicherplatz auf aktuellem Laufwerk	x	-	-	E
eof	prüft, ob Datensatz vorliegt	-	x	x	F
eof()	prüft, ob Datensatz vorliegt	-	x	x	F
error()	Nummer des letzten Fehlers	-	-	-	E
exp(N)	natürlicher Exponent von »N«	-	x	x	N
field(N)	Name des Feldes, das mit »N« korrespondiert	-	x	x	N
file("Dateiname")	prüft, ob Dateiname existiert	x	x	x	E
fldsel(N)	Funktionsstasname, der mit »N« korrespondiert	-	x	x	E
lmax()	maximale Anzahl der Funktionsstas, die programmierbar sind	-	-	-	F
found()	prüft, ob der letzte Suchbefehl Erfolg hatte	-	-	-	F
getenv(A)	Systemumgebungsvariable	-	-	-	F
intkey()	letzter Tastendruck als ASCII-Zeichen	-	-	-	E
int(N)	gibt »N« ohne Nachkommastellen aus (Integerfunktion)	x	x	x	N
isalpha(A)	prüft, ob erstes Zeichen von »A« ein Buchstabe ist	-	-	-	S
iscolor()	prüft, ob ein Farbbildschirm angeschlossen ist	-	-	-	S
islower(A)	prüft, ob erstes Zeichen von »A« ein Kleinbuchstabe ist	-	-	-	S
isupper(A)	prüft, ob erstes Zeichen von »A« ein Großbuchstabe ist	-	-	-	S
left(A,N)	bildet aus dem ersten »N« Zeichen von »A« eine neue Zeichenkette	-	-	-	S
len(A)	gibt Anzahl der Zeichen von »A« aus	x	x	x	S
log(N)	natürlicher Logarithmus von »N«	-	-	-	N
lower(A)	wandelt »A« in Kleinbuchstaben um	-	x	x	S
ltrim(A)	formt »A« in eine linksbündige Zeichenkette um	-	-	-	S
lupdate()	gibt an, wann die Datenbank zuletzt geändert wurde	-	-	-	F
max(Na,Nb)	stellt den größeren Wert von »Na« und »Nb« fest	-	-	-	N
message()	letzte Fehlermeldung	-	-	-	N
min(Na,Nb)	stellt den kleineren Wert von »Na« und »Nb« fest	-	-	-	N
mod(Na,Nb)	ergibt den Rest der Division von »Na« durch »Nb«	-	-	-	N
month(D)	gibt Monat von »D« als Zahl aus	-	x	x	D
ndx(N)	Indexdatei-Name, der mit »N« korrespondiert	-	-	-	E
os()	gibt aus, welches Betriebssystem benutzt wird	-	-	-	E
pcol()	aktuelle Druckerpalette	-	x	x	E
prout()	aktuelle Druckerzeile	-	x	x	E
rank(A)	gibt ASCII-Code des ersten Zeichens von »A« aus	x	-	-	S
readkey()	Tastendruck, durch den der bildschirmorientierte Befehl verlassen wurde als numerische Ausgabe	-	-	-	E
reccount()	Gesamtzahl aller Datensätze der aktuellen Datenbank	-	-	-	F

68

reccount()	Datensatzgröße	-	-	x	F
recho()	aktuelle Datensatznummer	-	-	x	F
replicate(A,N)	wiederholt »A« -N-mal	-	-	x	S
right(A,N)	bildet aus den letzten »N« Zeichen von »A« eine neue Zeichenkette	-	-	x	S
round(N,Nb)	rundet »N« auf »Nb« Nachkommastellen	-	-	x	N
row()	aktuelle Bildschirmzeile	-	-	x	E
rtrim(A)	kürzt »A« um alle rechten Leerstellen	-	-	x	S
space(N)	gibt »N« Leerzeichen aus	-	-	x	S
sqr	Quadratwurzelfunktion	-	-	x	N
str(Na,Nb,Nc)	Zeichenkette aus »Na« mit einer Länge von »Nb« Zeichen und »Nc« Dezimalstellen	x	x	x	N
stuff(Aa,Na,Nb,Ab)	»Ab« wird über »Aa« geblendet, beginnend bei Position »Na« für »Nb« Zeichen	-	-	x	S
substr(A,Na,Nb)	Zeichenkette aus »A«, beginnend bei Position »Na« mit einer Länge von »Nb« Zeichen	-	-	x	S
test(A)	testet Ausdruck	x	-	-	E
time()	Systemzeit	-	-	-	E
transform(A,Aa)	Umformung von Ausdruck »A« in das Format von »Aa«	-	-	x	N
trim(A)	kürzt »A« um alle rechten Leerstellen	x	x	x	S
type(A)	prüft Ausdruck	-	-	x	S
upper(A)	wandelt »A« in Großbuchstaben um	-	-	x	E
val(A)	ergibt numerischen Wert von »A«	x	x	x	N
version()	Versionsnummer von dBase	-	-	x	E
year(D)	gibt Jahr von »D« als Zahl aus	-	x	x	F

Tabelle 1. Die Funktionen der Datenbanksysteme dBase II, dBase III und dBase III Plus

Schalter	Voreinstellung	Bedeutung	dBase II	dBase III	dBase III Plus
set alternate	off	Protokolldatei ein/aus	x	x	x
set bell	on	Tonsignal ein/aus	x	x	x
set carry	off	Übernahme der Daten des letzten Datensatzes ein/aus	x	x	x
set century	off	Datumanzeigen vierstellig ein/aus	x	-	-
set confirm	off	[Return] vor Sprung ins nächste Feld ein/aus	x	x	x
set console	on	Bildschirmausgabe ein/aus	x	x	x
set debug	off	Fehlermeldungen an Drucker leiten ein/aus	x	x	x
set deleted	off	zum Löschen markierte Datensätze nicht berücksichtigen ein/aus	x	x	x
set delimiter	off	achtet Feldbegrenzungen ein/aus	-	-	x
set dohistory	off	Programmbeehle in Historydatei protokollieren ein/aus	-	-	x
set echo	off	aktuelle Programmzeile anzeigen ein/aus	x	x	x
set escape	on	Funktion der Escape-Taste ein/aus	x	x	x
set exact	on	exakte Zeichenkettenvergleiche ein/aus	x	x	x
set fields	off	Benutzung der Feldliste ein/aus	-	-	x
set fixed	off	regelt Dezimalstellenausgabe	-	-	x
set heading	on	Feldüberschriften ein/aus	-	-	x
set help	on	Hilfsfunktionen ein/aus	-	-	x
set intensity	on	inverse Menüerstellung ein/aus	-	-	x
set menus	on	Hilfsmenü ein/aus	-	-	x
set print	off	Druckausgabe ein/aus	x	x	x
set safety	on	Sicherheitsabfragen ein/aus	-	-	x
set scoreboard	on	Statuszeile ein/aus	-	-	x
set status	on	Statuszeile unten ein/aus	-	-	x
set step	off	Einzelanschrittmodus ein/aus	x	x	x
set talk	on	Dialog-Modus ein/aus	x	x	x
set title	off	Anzeige des Dateinamens ein/aus	-	-	x
set unique	off	Identische Indexschlüssel ein/aus	-	-	x

Tabelle 2. Die »set«-Befehle dienen zum Umstellen der dBase-Schalter, ...

Befehl	Bedeutung	dBase II	dBase III	dBase III Plus
set alternate to	Protokolldatei definieren	x	x	x
set catalog to	Katalogdatei eröffnen	-	-	x
set color to	Farbeinstellungen	x	x	x
set decimal to	Zahl der Dezimalstellungen	-	-	x
set default to	voreingestelltes Laufwerk wählen	x	x	x
set delimiter to	Feldbegrenzungszeichen festlegen	x	x	x
set device to	Ausgabegerät festlegen	-	-	x
set fields to	Feldliste festlegen	-	-	x
set filter to	Datenfilter setzen	-	-	x
set format to	Auswahl einer Formatdatei	-	-	x
set format to	Auswahl der Ausgabereinheit	x	-	-
set function to	Funktionsstasbelegung	-	-	x
set history to	Anzahl der Befehle im History-Modus	-	-	x
set index to	Indexdateien eröffnen	x	x	x
set margin to	linker Rand für Druckausgabe	x	x	x
set memowidth to	Breite eines Memofeldes	-	-	x
set message to	Nachrichtenzeile festlegen	-	-	x
set order to	Reihenfolge der Indexdateien	-	-	x
set path to	Angabe von Suchverzeichnissen	-	-	x
set procedure to	Prozedurdatei öffnen	-	-	x
set relation to	Relation zweier Datenbanken festlegen	x	x	x
set typeshead to	Größe des Tastaturpuffers festlegen	-	-	x
set view to	Viewdatei öffnen	-	-	x

Tabelle 3. ... während die »set to«-Befehle dBase an Ihre individuellen Bedürfnisse anpassen

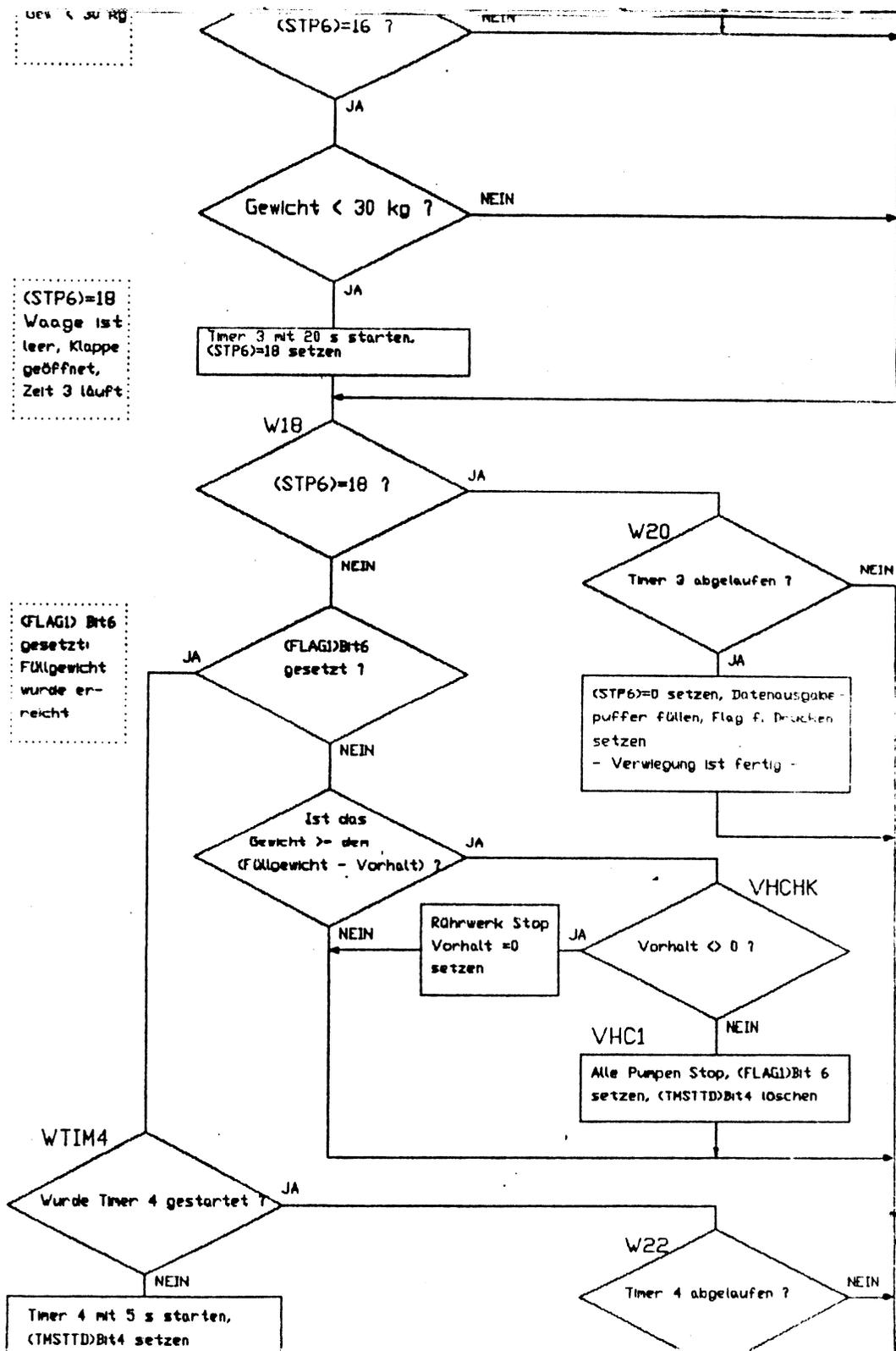
AUF GUTE ZUSAMMENARBEIT !

Tandy 4p erstellt das Programm, PC zeichnet das Flußdiagramm

Ein gut strukturiertes Programm braucht keine Hilfsmittel wie Flußdiagramme oder Struktogramme. So kann man es jedenfalls manchmal in der Literatur lesen. Aber, ehrlich gesagt: wenn ich mir eines meiner selbst geschriebenen Assemblerprogramme nach einem halben Jahr anschau, brauche ich trotz der Kommentare im ZEUS-Sourcecode eine ganze Menge Zeit, um mich in meine früheren Gedankengänge wieder hineinzufinden. Diese Arbeit wird sehr erleichtert, wenn man aus einem Flußdiagramm den Ablauf und die Bedeutung der Timer, Schrittzähler und Flags ablesen kann.

Wer schon einmal Flußdiagramme gezeichnet hat, weiß, wie sehr man auf die räumliche Aufteilung auf dem Zeichenblatt achten muß, wie leicht Zeichenfehler vorkommen und wie schwierig nachträgliche Änderungen und Ergänzungen sind. Als Hilfe für die Dokumentation meiner auf Single Board Computern laufenden Prozeßsteuerungsprogramme verwende ich das Zeichenprogramm 'Autosketch' der Schweizer Firma Autodesk. Mit etwas Übung kann man damit Zeichnungen per Maus genau so schnell am Bildschirm eines MSDOS-Rechners erstellen, wie mit Bleistift und Lineal. Der Vorteil tritt erst richtig in Erscheinung, wenn man vorhandene Zeichnungen ändern muß. Mit wenigen Bewegungen der Maus lassen sich ganze Zeichnungsteile löschen, verschieben, kopieren oder neu erstellen. Das Ergebnis ist eine saubere und übersichtliche Dokumentation. Als Beispiel zeige ich einen Teil eines insgesamt 1,20 m langen Flußdiagramms.

Autosketch ist als Einstieg in das viel umfangreichere CAD-Programm AUTOCAD gedacht und trotz seines bescheidenen Preises schon recht leistungsfähig. Wenn man vernünftig damit arbeiten will, braucht man einen PC mit mindestens 640 KByte RAM und einer Festplatte. Ein AT muß es nicht unbedingt sein. Auch auf einem 4,7 MHz-XT mit Coprozessor läuft das Programm ausreichend schnell. Mit einer Hercules-Karte ist die Auflösung auf dem monochromen Bildschirm genügend fein, die Vorteile der Layer-Technik kommen jedoch erst mit der EGA-Farbgraphik zur Geltung. Zur Ausgabe der Zeichnungen eignet sich ein ganz gewöhnlicher 9-Nadel-Drucker (meine Zeichnung wurde auf einem Star LC-10 erstellt), oder - mit erheblich besserem Ergebnis - ein HP-kompatibler Plotter.



(STP6)=18  
Waage ist  
leer, Klappe  
geöffnet,  
Zeit 3 läuft

(FLAG1) Bit6  
gesetzt;  
Füllgewicht  
wurde er-  
reicht

Der erste elektronische Computer auf Basis von Tausenden von Vakuumröhren wurde 1946 gebaut und hieß ENIAC. Er füllte einen großen Raum, verbrauchte viel Strom, war häufig defekt aber rechnete immerhin viel schneller, als dies zur damaligen Zeit manuell oder elektro-mechanisch möglich war. Im Jahre 1981, dem fünfunddreißigsten Jahrestag der Geburt des Computers, führen einige Studenten den alten ENIAC hoch, der sich nun im Smithsonian Institut befand und verglichen dessen Geschwindigkeit mit der eines damals modernen Mikrocomputers. Der Mikro - in der Größe einer kleinen Schachtel und mit IC's bestückt, die Hunderttausende von Transistorfunktionen enthalten, jeder das Gegenstück zu einer Vakuumröhre - war tatsächlich etwas schneller.

Heute steht auf dem Schreibtisch in vielen Büros ein Rechner mit einer Rechenleistung und Speicherkapazität die Anfang der 70er Jahre noch einem 500.000 DM teuren Minicomputer z.B. einer DEC PDP 11 Ehre gemacht hätte. Trotzdem, die Ansprüche wachsen und man beschwert sich, daß man zu lange warten müßte, alles so mühselig sei und es manchmal viele Minuten braucht, bis eine komplizierte Aufgabe wie die Sortierung einer umfangreichen Adressendatei bewältigt ist.

Ich selbst bin ja seit 1983 "EDV-Anwender", seit damals mit dem Genie ein preiswerter Rechner auf dem Markt angeboten wurde, der allerdings im kompletten Ausbau auch mehr als 5.000,- DM gekostet hat. Es war immer mein Interesse darin auch praktische Unterstützung z.B. beim Schreiben von Berichten zu erhalten, weshalb ich mich um Anwendungsprogramme und zuletzt auch um CP/M bemühte. Ich erinnere noch, mit welchem Stolz ich die ersten Serienbriefe druckte. Dies war nur durch mühselige Vorbereitung nach einer langsamen Verarbeitungsprozedur möglich. Dabei galt es immer zu hoffen, daß nicht zwischendurch ein Fehler auftrat und neu begonnen werden mußte. Immerhin, war ich damit viel schneller, als bei der manuellen Methode und der Aufwand lohnte sich (zumindest für den erprobten wiederholten Durchlauf).

Mein Genie steht noch hier, aber kann sich für meine Aufgaben kaum noch erwärmen, denn seit ich meinen PLANTRON PC-XT mit MS-DOS ausgebaut habe und FRAMEWORK benutze, finde ich darin eine Umgebung vor, die nahezu alles abdeckt, was mir früher nur mit Einzelprogrammen möglich war. Mir ist dabei bewußt geworden, welche ungeheure Entwicklung sich hier vollzieht - schließlich hat es bei den IBM-PC Kompatiblen auch rund 5 Jahre gedauert, bis die Programme einigermaßen ausgereift sind. Mein PC verarbeitet mit rund 5 MB Arbeitsspeicher (704 kB unter DOS plus 4 MB Speichererweiterung) rund 100 mal größere Datenmengen, als es in den 48 kB des Genie möglich war und läuft dabei auch wesentlich flotter und zuverlässiger. Die Software ist - wie angedeutet, inzwischen auch so leistungsfähig geworden, daß sie die Hardware optimal auszulasten vermag.

Gleichwohl sind inzwischen Rechner mit 80386 CPU und 16 oder 20 MHz die "aktuellen" Maschinen, die vom Profi gekauft werden, noch rund 10 mal mehr Power entfalten. Neue Maschinen mit noch mehr Leistung stehen vor der Tür, so werden die ersten 80386er Rechner mit 33 MHz und die 80486 CPU (mit der Leistung von einigen MIPS, vergleichbar einer VAX 785) bereits getestet.

72

Als interessante Entwicklung dürften wohl die Transputer anzusehen sein. Inzwischen werden vergleichsweise preisgünstige Karten angeboten, die sich an einen PC anschließen bzw. einbauen lassen und diesen Rechner nur für die Ein- und Ausgabevorgänge benötigen, ansonsten aber die Apfelmännchen schon fast in Echtzeit produzieren könnten. Wie ich der aktuellen c't entnehme, lassen sich eine Reihe solcher einzelnen Transputer auch miteinander verknüpfen, so daß die Rechenaufgaben auf einzelnen CPU's parallel verarbeitet werden können, wobei sich die Rechenleistung potenziert! Es läuft wohl darauf hinaus, daß die Leistungsmerkmale der heutigen "ENIAC", nämlich eines GRAY Supercomputers in wenigen Jahren auf dem Schreibtisch verfügbar werden könnte. Damit können die Möglichkeiten der Software noch erweitert werden und Lösungen für komplexere Problemstellungen (Sprachein- und -ausgabe, automatische Übersetzung, Expertensysteme, Bilddatenerfassung etc.) in greifbare Nähe rücken. Übrigens ist es heute schon auf dem PC möglich, Texte aus Zeitschriften, Büchern oder von der Schreibmaschine per Scanner zu digitalisieren und dann per Software in ASCII-Zeichen zu übersetzen und dann mit der Textverarbeitung weiter zu verarbeiten. Hier wird es auch wohl in Zukunft noch weitere Entwicklungen geben.

In Anbetracht des wachsenden Marktes muß ich aber auch selbstkritisch feststellen, daß ich durchaus nicht immer ein Maximum an Leistung brauche, um z.B. "diesen Text zu schreiben".

Mir hat bislang die Erkenntnis aus dem Umgang mit dem Genie und den vielen dafür verfügbaren Programmen bis hin zu KaJot's und Arnulf's "Reiterlied" geholfen, ein grundlegendes Verständnis von diesen Abläufen und technischen Chancen und Grenzen zu erhalten. Ich würde es aber auch sehr begrüßen, wenn wir neue Themen, bis hin z.B. zur Frage der Programmierung eines Transputers vom Genie aus (?) aufgreifen könnten. Zumindest würden mich solche Themen interessieren und ich würde mich über solche Beiträge freuen.

Paul-Jürgen Schmitz  
30. Oktober 1988

# Datenbanken richtig aufgebaut

Relationale Datenbanksysteme gehören aufgrund der Verkaufszahlen zu den Spitzenreitern der PC-Software. Ihre meist leichte Bedienbarkeit verführt viele Anwender dazu, große umfangreiche Dateien aufzubauen. Zu spät überrascht manchen die bittere Erkenntnis, daß man nicht nur die Bedienung der Software beherrschen muß, sondern auch eine Reihe einfacher, aber wichtiger Entwurfsregeln beachten sollte, um nicht im »elektronischen Chaos« zu versinken.

In einem Datenbanksystem ist wie bei einer Schlagbohrmaschine: In der Hand des erfahrenen Benutzers ist es ein leistungsfähiges Werkzeug; in der Hand des ahnungslosen Laien läßt sich damit großer Schaden verursachen. Natürlich hinkt dieser Vergleich wie alle: Mit einem »relationalen Datenbanksystem« (zum Beispiel dBase III, K-Man, Rbase oder die entsprechenden Komponenten integrierter Pakete) können Sie nichts in Ihrem PC kaputt machen. Aber Sie bereiten sich auf, wenn Sie Ihre Datenbanken wie geschichtliche Informationen erheben, nicht ernst genug zu nehmen und sich nicht erlauben, die Auswertungen nicht zu erlauben. Dabei ist alles – wie immer – ganz einfach, wenn man die Regeln zum Aufbau relationaler Datenbanken kennt.

Zuerst sei jedoch definiert, was eine Datenbank ist. Einfach formuliert ist eine Datenbank eine Anzahl von zusammengehörigen, miteinander verknüpften Dateien. Eine Datei wiederum ist (aus Sicht des Anwenders) eine simple Tabelle mit Zeilen und Spalten, die mit einem Namen versehen und über diesen ansprechbar ist, wie Bild 4 zeigt. Die »Verknüpfung« besteht darin, daß Daten einer Datei (Tabelle) zu denen der anderen zuzuordnen sind – eine zweite Tabelle würde zum Beispiel mit der aus Bild 4 eine »Datenbank« bilden, wenn sie Angaben über die »Gebäude« enthielte (Größe, Lage, Adresse); eine mit Angaben über Automarken aber nicht.

An einem simplen Beispiel sehen Sie, was gemeint ist: Ein Makler will seinen PC benutzen, um Daten über Eigentumswohnungen seines Angebotes zu speichern. Für jedes

Objekt möchte er beispielsweise aufnehmen: die laufende Nummer, die Adresse (Nummer des Bezirkes und Straße) die Wohnfläche, den Preis, eventuelle Extras, die Telefonnummer des Verkäufers und seinen Namen (diese Liste ist in der Praxis allerdings noch viel umfangreicher). Zur Auswertung wünscht er einen Bericht wie in Bild 1.

## Dateien locker aus dem Handgelenk entworfen... aber leider falsch

Nichts ist leichter, als daraus eine Datei (wie gesagt, in Form einer Tabelle) für den PC zu machen, die ein Dateiverwaltungssystem wie dBase III anstandslos akzeptieren würde. Den meisten Anwendern ist bekannt, daß die Definition eines »Schlüssels« bei einer Datei (oder Datenbank) notwendig, zumindest aber sinnvoll ist. Ein Schlüssel ist eine Datenspalte oder ein Feld, das jeden Satz in der Datei eindeutig identifiziert (also nicht zweimal oder noch öfter vorkommt). Zudem ist die Datei meist automatisch nach ihrem Schlüssel sortiert – ein Service der Datenbank-Software.

Schlüssel des Satzes in der Wohnungsgate ist aber die Objekt-Nr., das jede Wohnung

eindeutig identifiziert. Für die Extras bereiten Sie mehrere Felder vor, zum Beispiel drei, so daß Sie maximal drei verschiedene Extras speichern können. So geht es weiter, und schon haben Sie eine einfache Felddarstellung für Ihre Datei. Ein weiterer Service der Software ist, daß eine »Erfassungsmaske« für die Datei automatisch daraus hergeleitet wird. Darin tippt der Benutzer komfortabel seine Daten ein, wie es Bild 2 zeigt. Gehen Sie davon aus, daß in der Praxis die Längen der Felder nicht so eng bemessen sind wie in unserem Beispiel – es ist ja allgemein bekannt, daß hier bei der Anwendung oft Schwierigkeiten auftauchen. Sollte also als Verkäufer die bekannte »Donaudampfschiffahrtsgesellschaft« in Frage kommen, so muß das Feld entsprechend groß gewählt werden.

Aber leider ist inzwischen schon ein viel gravierenderer Fehler unterlaufen. Alle DV-Hasen kennen das Naturgesetz: »Es ist immer einer mehr oder weniger, als man gedacht hat!«. Irgendwann merkt besagter Makler: »Drei Spalten für Extras sind zu wenig. Nehmen wir lieber fünf.« Was ist zu tun? Er muß die Datei erweitern (neue Felder kommen hinzu) und umladen. Zudem muß er – wegen des geän-

dernten Dateiaufbaus – die Auswertprogramme ändern. Und wehe, das erste Objekt mit sechs Extras soll in der Datei gespeichert werden. Das Ganze wird immer ärgerlicher, ist aber noch lange nicht das Schlimmste, was einem passieren kann.

Stellen Sie sich vor, die Datenbank ist ziemlich voll – sagen wir einige hundert Sätze (Dies ist beim Speichervermögen heutiger Systeme auf dem PC ein Klacks). Ein neues Objekt wird bekannt und sofort in seine Erfassungsmaske eingegeben. Dabei passiert leider ein Mißgeschick:

VERK. TEL. > 2346578 <  
VERKÄUFER > GeSoBau AG <

Was ist passiert? Die Datenbank enthält eine Ungenauigkeit, einen Widerspruch – zumindest unter bestimmten Annahmen. In unserem Fall soll es so sein, daß in der Realität ein Verkäufer nur eine Telefonnummer hat. Dann ist der neue Eintrag auf jeden Fall unrichtig: Die »GeSoBau AG« hat die Nummer »234 56 78«. Die Datenbank ist »inkonsistent« geworden. Wie läßt sich das bereinigen? Im ersten Fall sind die »Extras« sogenannte »Mehrfachfelder« (weil ein Feld mehrfach pro Tabellenzeile beziehungsweise Datensatz vorkommen darf). Regel 1 des »methodischen Datenbankdesigns«, das von Informatik-Wissenschaftlern erarbeitet wurde, verbietet solche Mehrfachfelder. Wie hätte man es machen müssen?

## Kombinationsschlüssel schaffen Ordnung, aber auch neue Probleme

»Mehrfachfelder« werden in einer gesonderten Datei gespeichert und nicht in der Originaldatei. Und zwar pro Feld ein Satz, so daß nun beliebig viele Extras (0, 1, 2, ..., viele) pro Wohnung zu erfassen sind. Diese Datei sieht dann beispielsweise aus wie in Bild 3.

Bevor Sie diese Lösung genauer betrachten, schauen Sie sich das Feld »Adresse« noch einmal an. Auch hier wurde ein im wahren Sinne des Wortes »elementarer« Fehler gemacht.

Als Feld in einer Datei, als Spalte in einer Tabelle sollte immer nur ein

»Datenelement« dienen. Ein Element ist wie in der Chemie eine Einheit, die nicht weiter zerlegt wird (und manchmal gar nicht zerlegbar ist). Nur ein solches Datenelement läßt sich im allgemeinen gesondert betrachten. Unsere Datei läßt sich zum Beispiel hinsichtlich des Bezirkes oder der Straße gar nicht auswerten, zumindest nicht ohne Schwierigkeiten. Das aber ist ja gerade die Stärke von Datenbank-Software, daß Sie Fragestellungen wie: »Welche Objekte im Bezirk 13 sind billiger als 250000 Mark?« mit einem einfachen Befehl der Art

list for bezirk = 13 and preis < 250000 beantworten. Allerdings erlauben es die meisten Softwarepakete, Felddaten zu zerlegen. In unserem Fall könnte man durch die Suche nach dem Komma den Bezirk von der Straßenangabe leicht trennen. In der Praxis ist es jedoch in jedem Fall sauberer, daraus zwei getrennte Felder zu verwenden. Widmen Sie sich noch einmal den Schlüssel: Felder, die einen Satz in der Datei (eine Zeile in der Tabelle) eindeutig identifizieren. Was passiert, wenn es kein einzelnes Feld gibt, das dies leistet?

In diesen Fällen benutzt man in der Praxis »Kombinationsschlüssel«: zwei oder mehr Felder, die in Kombination miteinander eindeutig sind. Will man also die monatlichen Gehälter von Mitarbeitern für ein Jahr speichern, so hätte man die Kombination der Felder »Personalnummer« und »Monat« als Schlüssel; wollte man mehrere Jahre speichern, bräuchte man die Jahreszahl als dritte Schlüsselkomponente.

Im Beispiel der »ausgelagerten« Extras in Bild 3 sehen Sie, daß die Datei sogar nur aus ihrem Schlüssel besteht: der Kombination aus »Nr.« und »Extra«. Außer dem Schlüssel ist die Datei hier zufällig leer, es könnten aber genauso gut noch (von Wohnung zu Wohnung verschiedene) Daten gespeichert sein, die sich aber unbedingt auf die Kombination aus beiden Schlüsseln beziehen müssen.

In der Praxis kommt es häufig vor, daß eine Datei nicht nur einen einzigen Schlüssel (wie hier die »Nr.« der Wohnungsdatei) hat, sondern

daß eine Kombination aus mehreren Feldern zur Identifizierung eines Satzes herangezogen wird. Vielfach ist dabei das Datum im Spiel. Denken Sie sich eine Datei, in der einzelne Besichtigungen der Wohnungen durch die Interessenten gespeichert werden sollen. Sie hätte als Schlüsselkomponente das Feld »Datum« und zusätzlich... Nun, überlegen Sie selbst, welche Spalten Sie außerdem zur Identifizierung eines Satzes benötigen. Diese aus mehreren Datenelementen (Tabellenspalten) bestehenden Schlüssel, die »Kombinationsschlüssel«, leisten also gute Dienste beim sauberen Entwurf von Dateien – haben aber bei unbedach-

NR.	> _____ <
ADRESSE	> _____ <
Q#	> _____ <
PREIS	> _____ <
EXTRAS1	> _____ <
EXTRAS2	> _____ <
EXTRAS3	> _____ <
VERK. TEL.	> _____ <
VERKÄUFER	> _____ <

Bild 2. Felder der Wohnungsdatei in der Erfassungsmaske

NR.	EXTRA
001	Balkon
001	Garage
001	Wintergarten
002	Anlagesteg
002	Garage

Bild 3. Daten der Extras der jeweiligen Wohnungen

NAME	ART GEBÄUDE	TEL.
Anders	A12	04 1233
Baier	B02	01 5133
Charcoal	C15	12 1314
Demmer	C15	12 1210
Eaer	B02	01 5411
Eaer	A12	04 1100

Bild 4. »Ausgelagerte« Zusatzdaten in einer Datei

NR.	ADRESSE	Q#	PREIS	EXTRAS	VERK. TEL.	VERKÄUFER
001	13, Bundesallee 33 a	120	240000	Balkon Garage Wintergarten	123 45 67	Dr. Müller
002	12, Am Weiher 15	85	195000	Anlagesteg Garage	234 56 78	GeSoBau AG

Bild 1. Liste der Eigentumswohnungen

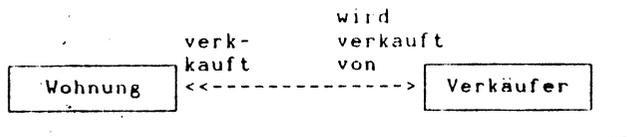


Bild 5. Relationen zwischen zwei Objekten

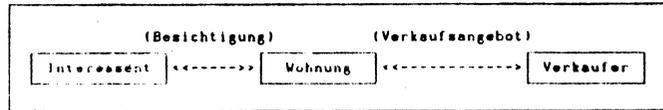


Bild 6. Relationen zwischen drei Objekten

tem Gebrauch auch ihre Tücken. Betrachten Sie folgendes Telefonverzeichnis (Bild 4), bei dem – wie Sie an dem letzten gezeigten Eintrag sehen – nur die Kombination aus »Name« und »Abteilung« eindeutig ist (unter der Annahme, daß nicht zwei »Esser« in derselben Abteilung vorkommen). Sieht doch gut aus, oder? Leider ist es ein Verstoß gegen Regel 2 des Datenbankdesigns: Ein Feld darf nicht von nur einem Teil des (Kombinations-)Schlüssels abhängen! In unserem Fall hängt die in Ziffernform abgekürzte Angabe des Gebäudefeldes von »Abt.« ab. Warum ist

das so schlimm? Niemand hindert uns daran, bei einem Neuzugang eine Zeile

Huber B02 04 8613 mit aufzunehmen – mit falschem Inhalt, da doch die Abteilung »B02« in Gebäude »01« angesiedelt ist. Die Datei enthält somit einen logischen Widerspruch zur Realität. Von der »redundanten« (das heißt mehrfach identisch vorhandenen) Information ganz zu schweigen. Sie kostet uns zumindest Speicherplatz. Die Aussage »Abt. B02 ist in Gebäude 01« wird ja überflüssigerweise in vielen Sätzen gespeichert, obwohl ein Mal reichen würde

LEISTUNGSMESSUNG									
Datum	Vers. Nr.	Ser. Nr.	Typ	Bezeichnung	Sp. [V]	Str [A]	Leistung		
18.05.88	2136	712577	GE01	Generator I	220	102	22440		
					222	100	22200		
					218	99	21681		
					Durchschnitt [W]		22107		
Meßabweichg. [%]		3.43							
18.05.88	680455	GE02	Generator II	115	90	10350			
				111	93	10323			
				110	93	10230			
				Durchschnitt [W]		10301			
Meßabweichg. [%]		1.16							
19.05.88	2138	712577	GE01	Generator I	219	104	22776		
...	...	...	...	...	...	...	...		
...	...	...	...	...	...	...	...		

Bild 7. Auswertung von Leistungsmessungen

(nämlich in einer gesonderten Tabelle). Sie werden es schon gemerkt haben: Schon öfter wurde der Ausdruck »unter der Annahme, daß...« gebraucht. Diese Annahmen sind nicht Vereinfachungen unserer Beispiele, sondern wichtige, in der Praxis festzustellende Fakten. Denn wenn es erlaubt ist, daß eine Abteilung über mehrere Gebäude verstreut ist, dann wird aus dem diagnostizierten Designfehler plötzlich eine korrekte Abbildung der Realität. Dieser Sachverhalt läßt sich nur durch Studium der in der Realität herrschenden Regeln und Gesetze feststellen und hat einen bestimmenden Einfluß auf den Entwurf der Datenbanken. Der Fachmann redet hier von »Objekten« (zufällig derselbe Ausdruck, den der Makler gebraucht). Objekte (oder »Informationsobjekte«) sind Dinge, über die jemand Daten (in Form von Datenelementen) sammelt, und die deshalb durch Schlüssel voneinander unterschieden werden müssen. Die wichtigste Frage ist die nach den Beziehungen der Objekte untereinander – hauptsächlich, ob zu einem Objekt immer und ausschließlich genau ein anderes gehört (und umgekehrt), oder ob es mehrere sein können. Beziehung heißt vornehm (und englisch) »Relation«. Merken Sie etwas? Weil das die wichtigste Aussage in der Datenbank ist, nennt man sie »relational«.

Die Relationen zwischen zwei Objekten müssen immer in zwei Richtungen untersucht werden: hin und zurück. In Bild 5 sehen Sie dies: Eine Wohnung wird von genau einem Verkäufer verkauft (wo käme denn der Makler hin, wenn mehrere Verkäufer dasselbe Objekt anbieten würden?), aber ein Verkäufer verkauft unter Umständen mehrere Wohnungen. Solche Beziehungen nennt der Fachmann »N:1-Beziehungen«. Sie sind von besonderer Bedeutung, wenn sichergestellt ist (per Vorschrift, Gesetz oder gar aufgrund eines Naturgesetzes), daß wirklich nur ein einziges Objekt mit mehreren eine Relation bilden kann. Die Lehrbücher verwenden hier übrigens zur Illustration oft das naheliegende Beispiel »Ehe« zur Darstellung der Varianten »1:1« (Einehe), »N:1«, »1:N«

und »N:N« (letzteres als »Chaos« definiert). Diese korrekte Abbildung der Realität ist die wichtigste Voraussetzung für ein einwandfreies Design von Datenbanken. In Bild 6 sehen Sie eine Erweiterung: Ein neues Objekt namens »Interessant« wird gespeichert. Es steht zur Wohnung in einer N:N-Beziehung: mehrere Interessenten besichtigen ein und dieselbe Wohnung, ein einziger Interessent besichtigt aber seinerseits mehrere Wohnungen. Es gibt aber noch weitere Regeln zum korrekten Dateientwurf. Ein Problem war ja entstanden, wenn ein Feld nur von einem Teil eines Kombinationsschlüssels abhing. Ein ähnliches Problem tritt auf, wenn Dateifelder nicht von Schlüsselteilen, sondern sogar untereinander abhängig sind. Todsünde Nummer 3! Ein Beispiel zeigt bereits Bild 1. Hier sind »Verkäufer« und »Verk.Tel.« untereinander abhängig. Wieder liegt Redundanz vor und die Möglichkeit zu widersprüchlichen Dateieinträgen, wie sie schon dargestellt wurden. Dabei führen nicht nur Einschübe (»insert«), sondern auch Löschungen (»delete«) und Änderungen (»update«) zu Schwierigkeiten. Nachdem Sie nun erfahren haben, was man alles falsch machen kann, sollen Sie auch die richtigen Lösungen präsentiert bekommen.

### Drei einfache Regeln führen zu sauberem Datenbankdesign

Zum Design einwandfreier »Relationen« (Tabellen, Dateien mit nur einer Satzart) gibt es drei Regeln:

- 1) Jedes Feld darf im Datensatz nur ein Mal und nicht mehrfach vorkommen.
- 2) In Dateien mit einem Kombinationsschlüssel muß jedes nicht zum Schlüssel gehörende Feld vom gesamten Kombinationsschlüssel abhängen und nicht nur von Teilen davon.
- 3) Felder, die nicht Teil des Schlüssels sind, dürfen nicht untereinander abhängig sein.

Die Beachtung dieser drei Regeln liefert Dateien (Relationen), die redundanzfrei sind, und in denen in-

haltlich widersprüchliche Sätze nicht vorkommen.

Wäre die Wohnungsdatei nach diesen Regeln entworfen worden, hätten bei Änderung der Zahl der Extras-Felder weder Dateien noch Programme geändert werden müssen. Abgesehen davon lassen solche korrekten Dateien Fragestellungen und Analysen zu, die mit »unsauberen« Dateien nicht realisierbar sind.

Die Vorgehensweise beim Dateientwurf kann nun als Methode wie ein Kochrezept beschrieben werden. Sie sei hier nicht im Detail ausgeführt, aber kurz anhand eines Beispiels verdeutlicht. Dieses wurde bewußt aus einem Bereich gewählt, der den meisten von Ihnen fremd sein wird. Sie werden sehen, daß die Vorgehensweise unabhängig von einer gründlichen Kenntnis des Fachgebietes funktioniert.

In einem Maschinenbaubetrieb soll eine Liste produziert werden, die Meßwerte von Generatoren enthält (Bild 7). Wie gehen Sie nun methodisch an den Dateientwurf heran?

Schritt 1: Welche »Informationsobjekte« (oder »Objekttypen«) sind erkennbar? Auf deutsch: Über welche Gegenstände (im weitesten Sinne) werden Daten gesammelt, und wie (durch welche Schlüssel) werden sie identifiziert?

Im Beispiel erkennen Sie die Schlüssel »Versuchsnr.« und »Seriennr.«. Daraus entnehmen Sie den Objekttyp »Versuch« (Schlüssel »Versuchsnr.«) mit den Daten »Datum«, »Durchschnittsleistung«, »Meßabweichung« und (nicht vergessen!) dem Hinweis auf die »Seriennr.«. Beachten Sie, daß nach Regel 1 Mehrfachfelder nicht erlaubt sind. Es dürfen also nur Datenelemente zugeordnet werden, die pro Informationsobjekt »Versuch« genau ein Mal vorkommen.

Als nächsten Objekttyp identifizieren Sie die »Maschine« (Schlüssel »Seriennr.«) mit den Daten »Typ« und »Bezeichnung«.

Wie Sie den Fall des »Mehrfachfeldes« behandeln, haben Sie schon erfahren – eine separate »Relation« (das heißt eine Tabelle). Das einzige Problem ist der Schlüssel. Die »Versuchsnr.« reicht ja nicht

Tabelle »Versuche«:					
SCHLÜSSEL					
Vers. Nr.	Datum	Seriennr.	Durchschnitt	Meßabweichg.	
2136	18.05.88	712577	22107	3.43	
2137	18.05.88	680455	10301	1.16	
2138	19.05.88	712577	.....	.....	

Tabelle »Maschine«:		
SCHLÜSSEL		
Seriennr.	Typ	Bezeichnung
712577	GE01	Generator I
680455	GE02	Generator II

Tabelle »Messung«:				
---SCHLÜSSEL---				
Vers. Nr.	Lfd. Nr.	Spannung	Strom	Leistung
2136	1	220	102	22440
2136	2	222	100	22200
2136	3	218	99	21681
2137	1	115	90	10350

Bild 8  
Erster Entwurf  
von Relationen

aus, jede einzelne Messung zu identifizieren. Also nehmen Sie eine (auf der Liste nicht sichtbare) laufende Nummer mit hinzu. Daraus entsteht der Objekttyp »Messung« mit dem Schlüssel »Versuchsnr.« und »Lfd.Nr.« sowie den Daten »Spannung«, »Strom« und »Leistung«. Eine übersichtliche Darstellung der Ergebnisse des ersten Schrittes zeigt Bild 8 in Form von »Relationen« (so nennt man auch diese tabellarische Auflistung).

Die Klärung der Beziehungen zwischen verschiedenen »Objekten« ist nicht nur im menschlichen und politischen Leben wichtig. Auch für den Entwurf von sauberen Dateien ist es von höchster Bedeutung, sie korrekt zu berücksichtigen.

Schritt 2: Welche Beziehungen und

Häufigkeiten bestehen zwischen den »Objekttypen«?

Im Beispiel erkennen oder erfragen Sie folgende Sachverhalte:

- Es gibt zwar mehrere Messungen pro Versuch, aber jede Messung gehört zu genau einem Versuch (eine »1:N«-Beziehung).

Die präzise Klärung der Beziehungen zwischen den »Objekttypen« ist wichtig

- Es gibt zwar mehrere Versuche pro Maschine (Versuche können wiederholt werden), aber jeder Versuch gehört zu genau einer Maschine.

Ein noch unentdeckter Sachverhalt wird geklärt:

- Es gibt zwar mehrere Maschinen pro Typ, aber jede Maschine gehört zu genau einem Typ.

Schritt 3: »Normalisieren« der Relationen, das heißt Anwendung der drei Regeln auf den ersten Entwurf, um die aufgrund der nun geklärten Abhängigkeiten entdeckten Regelverstöße zu entfernen.

In dem Beispiel erkennen Sie, daß das Vorhandensein der »Bezeichnung« in der Relation »Maschine« ein Verstoß gegen Regel 3 ist - ebenso wie die Speicherung der »Leistung«, da sie ja aus Strom und Spannung direkt errechnet wird. Damit erhalten wir die sauberen Dateientwürfe, wie sie in Bild 9 zu sehen sind. Wären Sie intuitiv, ohne Methode, darauf gekommen? Diese Relationen sind nicht notwendigerweise vollständig; es könnte etwa zum Maschinentyp ein Datenelement »Nennspannung« und zur Maschine direkt ihr »Herstelldatum« gehören.

Aber Sie sehen sehr schön, wie jetzt durchschaubare Verhältnisse entstanden sind, zum Beispiel durch den »Fremdschlüssel« (das Feld »Seriennr.«) in der Tabelle »Versuch«, der zu weiteren Angaben in der Tabelle »Maschine« führt. Damit läßt sich auch durch Befragen der Fachleute des jeweiligen Anwendungsgebietes sehr leicht der eigene Entwurf überprüfen; hier etwa durch die Frage: »Ist es zutreffend, daß eine Maschine einer bestimmten Seriennummer mehrmals, in mehreren Versuchen gemessen werden kann?«

Die Vorteile der »sauberen« Dateien sind im wesentlichen:

- Daten werden nicht mehrmals (»redundant«) gespeichert. Das spart Platz und (wichtiger!) verhindert sich widersprechende Dateiinhalte.

- Beliebige Fragestellungen sind realisierbar, die in der Regel durch eine in das Datenbanksystem integrierte »Abfragesprache« beantwortet werden können (zum Beispiel: »Welche Leistung(en) wurden in eventuell mehreren Versuchen für Maschine 712577 gemessen?«)

Das interessante Gebiet des methodischen Datenbankdesigns konnte hier nur kurz dargestellt werden. Sicher haben Sie daraus erkannt, wie wichtig es ist, von Anfang an klare und sachlich richtige Verhältnisse herzustellen.

(Jürgen Beetz/zi)

## Ein elektronisches Märchen

Es war einmal zur Zeit  $T=0$ , da lebte ein armer, aber recht-schaffender Vierpol namens Eddy Wirbelstrom. Er bewohnte einen bescheidenen möblierten Hohlraum mit Dielektrikum und fließend kaltem und warmen Sättigungsstrom. Leider mußte er während der kalten Jahreszeit für die Erwärmung der Sperrschichten noch extra bezahlen. Seinen Lebensunterhalt bestritt er mit einer Verstärkerzucht auf Transistorbasis.

Eddy Wirbelstrom liebte mit der ganzen Kraft seiner Übertragungsfunktion Ionchen. Ionchen, die induktive Spule mit dem kleinsten Fehlwinkel im ganzen Kreis, war die Tochter der einflußreichsten EMK. Ihr permanenter Ferritkörper, ihre symmetrischen Netzintegrale und ihre überaus harmonischen Oberwellen beeinflussten selbst die Suszeptibilität ausgedienter Leidener Flaschen (was viel heißen will!). Ionchens Vater, Cosinus Phi, ein bekannter Industrie-Magnet und Leistungsfaktor, hatte allerdings bereits konkrete Schaltpläne für die Zukunft (T>0) seiner Tochter. Sie sollte nur einer anerkannten Kapazität mit ausgeprägtem Nennwert angeschlossen werden. Aber wie so oft während der Lebensdauer Groß W, der Zufalltrieb wollte es anders.

Als Ionchen eines Tages, zur Zeit  $T=1$ , auf ihrem Picofarad vom Frisiersalon nach Hause fuhr - sie hatte sich die neue Sinus-stehwelle anlegen lassen - da geriet ihr ein Sägezahn in die Filterkette. Aber Eddy, der die Gegend periodisch frequentierte, eilte mit minimaler Laufzeit hinzu, und es gelang ihm, Ionchens Kippschwingungen noch vor dem Maximum der Amplitude abzufangen und gleichzurichten.

Es ist sicher nicht dem Zufall zuzuschreiben, daß sie sich schon zur Zeit  $T=1$  widersahen. Eddy lud Ionchen zum Abendessen ins »Ringintegral« ein. Aber das »Ringintegral« war leider geschlossen. »Macht nichts«, sagte Ionchen, »ich habe fest 0.2 KHz zu Mittag gegessen und die Sättigungsinduktion hat bis jetzt gehalten. Außerdem muß ich auf meine Feldlinien achten!« Unter irgend einem Vorwand lud Eddy sie daraufhin zu einer Rundfahrt im Rotor ein. Aber Ionchen lehnte ab, ihr würde bei der zweiten Ableitung immer so übel.

Und so unternahmen sie, entgegen den Schaltplänen von Vater Cosinus Phi, einen kleinen Frequenzgang ins nahe gelegene Streufeld.

Der Abend senkte sich über die komplexe Ebene, und am Himmel erglänzten die Sternschaltungen. Eddy und Ionchen genossen die Isolierung vom laufenden Getriebe der Welt. Nur ein einsamer Modulationsbrummer flog vorbei. Sanft plätscherten die elektromagnetische Wellen an Gestade und leise rauschten die Röhren.

Als sie an der Wheatstone-Brücke angelangt waren, nahm Eddy seinen ganzen Durchgriff zusammen und emittierte: »Bei Gauß, mein Ionchen, deine lose Rückkopplung hat es mir angetan!« Der Informationsgehalt dieser Nachricht durchflutete Ionchen mit Groß-Psi. Die Summe über alle  $n$  von  $n=1$  bis  $n$  dieser Ereignisse war zu überwältigend und sie entglitt der Kontrolle ihrer Zeitkonstanten. Im Überschwang des jungen Glücks erreichten beide vollausgesteuert die Endstufen.

Und wenn sie nicht gedampft wurden, dann schwingen sie noch heute.

Dieter Kasper

HEFT  
26  
Januar  
1989

78

Tabelle »Versuche«:

SCHLÜSSEL				
Vers.Nr.	Datum	Seriennr.	Durchschnitt	Meßabweichg.
2136	18.05.88	712577	22107	3.43
2137	18.05.88	680455	10301	1.16
2138	19.05.88	712577	.....	.....

Tabelle »Maschine«:

SCHLÜSSEL	
Seriennr.	Typ
712577	GE01
680455	GE02

Tabelle »Maschinentyp«:

SCHLÜSSEL	
Typ	Bezeichnung
GE01	Generator I
GE02	Generator II

Tabelle »Messung«:

SCHLÜSSEL			
Vers.Nr.	Lfd.Nr.	Spannung	Strom
2136	1	220	102
2136	2	222	100
2136	3	218	99
2137	1	115	90

Bild 9.  
Sauberer Entwurf  
von Relationen

--- Fr, 21.10.88 / 03:06:15 von DR.BAKTERIUS

Es war einmal ein kleines, süßes Mädchen, das immer ein Käppchen aus rotem Samt trug. Aufgrund dieses Attributes erhielt es ein Assign unter dem symbolischen Namen "Rotkäppchen". Eines Tages sprach die Mutter:

"Rotkäppchen, die Gesundheit deiner Großmutter hat einen Interrupt bekommen. Wir müssen ein Pflegeprogramm entwickeln und zu ihr bringen, um das Problem zu lösen. Verirre dich aber nicht im Wald der alten Sprachen, sondern gehe nur strukturierte Wege. Nutze dabei immer eine Hochsprache der 4. Generation, dann geht es der Großmutter schnell wieder gut. Und achte immer darauf, daß dein Pflegeprogramm transaktionsorientiert ist, damit es die Großmutter nicht noch mehr belastet".

Da der Weg zum Haus der Großmutter reentrant war, traf Rotkäppchen den bösen Wolf. Er tat sehr benutzerfreundlich, hatte im Background jedoch schon einen Abbruch programmiert. Während Rotkäppchen einen Go-To ins Blumenfeld machte, ging der Wolf im Direktzugriff zur Großmutter und vereinnahmte sie unverzüglich durch einen Delete. Ohne zu zögern gab er sich den Anschein, kompatibel zu sein und nahm die logische Sicht der Großmutter an. Dann legte er sich in ihren Speicherplatz.

Kurz darauf lokalisierte auch Rotkäppchen die Adresse der Großmutter und trat in den Speicherraum. Vor der Installation des Pflegeprogramms machte Rotkäppchen sicherheitshalber einen Verify und fragte:

"Ey Großmutter, warum hast du so große Augen?"

"Weil ich zufriedene Endbenutzer gesehen habe."

"Ey Großmutter, warum hast du so große Ohren?"

"Damit ich die Wünsche der User besser verstehen kann."

"Ey Großmutter, warum hast du so ein entsetzlich großes Maul?"

"Damit ich dich besser canceln kann!"

Sprach's und nahm das arme Ding als Input.

Nach einem Logoff begab sich der Wolf zur Ruhe, schlief ein und begann laut zu schnarchen. Als der Jäger auf seinem Loop durch den Wald am Hause der Großmutter vorbei kam, sah er durch ein Window den Wolf im Bett liegen.

"Finde ich dich hier, du alter Error", sprach er, "ich habe dich lange gesucht!" Als Kenner der Szene analysierte er sofort, daß nach den Regeln der Booleschen Algebra die Großmutter nur im Bauch des Wolfes sein konnte. Er nahm sein Messer, teilte dem Bauch des Wolfes in mehrere Sektoren und machte, welch' Freude, Großmutter und das Rotkäppchen wieder zu selbständigen Modulen.

Als Input für den leeren Bauch des Wolfes nahmen sie viele Kilobyte Steine und beendeten die Operation mit einem Close. Als der Wolf erwachte, verursachte ihm sein dermaßen aufgeblähter Hauptspeicher solche Schmerzen, daß er an einer Storage Violation jämmerlich zugrunde ging. Da waren alle veranußt.

Das Pflegeprogramm aktivierte die Großmutter. Rotkäppchen aber dachte:

"Du willst dein Lebtag nie wieder einen Go-To machen, sondern nur noch strukturierte Wege gehen, wie dir's die Mutter geboten hat."

Und wenn sie nicht einem Systemabsturz zum Opfer gefallen sind, durchlaufen sie noch heute die Schaltkreise...

Alexander Schmid

Kostenlos telefonieren

+-----+  
! Fun With Fone !  
+-----+

aus der B.H.P. #7

Man möchte es gar nicht glauben: Man kann mit dem Telefon auch ohne irgendwelche Gebühren eine Menge Spaß haben. Als Beispiel folgen nun einige Ansagen, die alle dasselbe bedeuten, sich aber trotzdem sehr unterschiedlich anhören.

Die bekannteste Ansage dürfte wohl das gute alte 'Kein Anschluss unter dieser Nummer' sein, das einem - typisch deutsch - mitteilt, dass die gewünschte Nummer nicht existiert, z.B. 0015 - 4711. In München reicht es schon die Ziffern 25 zu wählen. Eine schöne Ansage bietet auch der hohe Norden. Versucht doch mal 04567. Anders sieht es hier im Ausland aus, wo man je nach Land von den verschiedensten Sprechern informiert wird, dass man eine falsche Nummer erwischt hat:

Nordamerika:  
001212 1234567 USA/New York  
001213 1234567 USA/Los Angeles

Südamerika:  
00503 123456 El Salvador  
00506 123456 Costa Rica  
0054328 123456 Argentinien  
005631 123456 Chile

Afrika:  
00203 1234567 Ägypten/Alexandria  
0021021 123456 Libyen  
002542 123456 Kenia  
002601 123456 Sambia

Arabien:  
009626 123456 Jordanien/Amman, sehr laut, man beachte die originelle Aussprache.  
009662 123456 Saudi-Arabien

Asien:  
00822 123456 Rep. Korea  
008865 123456 Taiwan

Anmerkung:

Die japanische Zeitansage ist unter 0081 777 777 7... (so lange eine 7, bis was zu hören ist) zu erreichen.

Alexander Schmid

## Abschied von der Tastatur

Die komplexen Prozesse zur Informationsverarbeitung, die dem Verstehen natürlicher Sprache zugrunde liegen, werden mit rechnerorientierten Methoden untersucht. Mit Computern, die Sprache verstehen können, ist der Mensch-Maschine-Informationsaustausch schneller, akkurater und effizienter durchführbar und -last, not least - „menschlicher“.

Um gesprochene Sprache zu verstehen, müssen die Schallwellen eines gesprochenen Wortes mit dem Schallmuster eines gespeicherten Wortes verglichen werden, um zunächst die syntaktische (den Satzbau betreffende) Abfolge der Wörter zu analysieren. Das Problem ist nun, daß das Schallmuster eines Wortes je nach Stellung im Satz und je nach Sprecher sehr verschieden sein kann und zudem noch aus einem Geräuschpegel herausgefiltert werden muß.

In der sprachorientierten KI-Forschung unterscheidet man neben der Syntax zwei weitere Ebenen des Verstehens: Die Semantik, die sich mit der Bedeutung des Satzes beschäftigt und die Pragmatik, die sich mit dem praktischen Umsetzen der Äußerung befaßt.

Der Vorteil von Spracherkennungssystemen liegt auf der Hand, denn Tastatureingaben von Texten bedingen die Gefahr, typographische Fehler zu begehen, die un bemerkt bleiben. Derartige Fehler, die nicht auf „logische“ Falscheingaben zurückzuführen sind, können mit den neuen Systemen verhindert werden.

Die technologischen Konsequenzen sind offensichtlich, zumal der Benutzer selbst zum Umgang mit solchen Systemen keinerlei EDV-Kenntnisse mehr, wie etwa bei der Verwendung von Terminals benötigt.

Zur Zeit werden „sprachgesteuerte Schreibmaschinen“ („voice activated typewriter“) weltweit in zahlreichen Forschungsinstituten entwickelt. Der Wortschatz dieser Spracheingabesysteme umfaßt teilweise bis zu 20 000 Wortformen. In der Bundesrepublik werden sprachverarbeitende Systeme in Berlin, Erlangen, Hamburg und Saarbrücken entwickelt. ddt

## KI-Anwendung: Einsatz von Spracherkennungs-Computern in der Medizin



**S**chwerer Verkehrsunfall. Zwei lebensgefährlich Verletzte werden in die Unfallstation des Nash General Hospital eingeliefert. Nachdem das eingespielte Team die Verletzten versorgt hat, muß der Chefarzt seine Berichte abfassen, medizinische Buchhaltung sozusagen: „Eine verantwortungsvolle Pflicht, von der Außenstehende gar nicht ahnen, daß sie bis zu 30 Prozent unserer Arbeit als Notfallarzt ausmacht“, so Dr. Arthur Chambers, Direktor der Notfallstation des Nash-Krankenhauses von Rocky Mount in North Carolina.

Bis vor kurzem diktierte Dr. Chambers seine Protokolle auf Tonband. Die besprochenen Kassetten gingen an den Schreibpool des Hospitals, wurden abgeschrieben, dem Doktor zum Gelesen vorgelegt, korrigiert, im Zweifelsfall noch einmal geschrieben - ehe sie auf den angestammten Dienstweg gebracht werden konnten. Seit neuestem steht in seinem Dienstzimmer - und in einem Dutzend weiterer US-Hospitäler - ein spezieller Protokoll-Computer, der die Arbeit des Unfallarztes erleichtert und insbesondere die Zeitspanne bis zur Ablieferung der Berichte ganz erheblich verkürzt. Der Computer versteht nämlich, was Dr. Chambers diktiert, mehr noch: Er setzt den gesprochenen Protokolltext sogleich um und schreibt ihn auf den Bildschirm.

Auch die Änderungswünsche, die der Doktor beim Korrekturlesen seiner Texte von sich gibt, führt der Computer auf der Stelle aus. Und auf das (gesprochene) Kommando „Print

this“, druckt er schließlich das fertige Protokoll formgerecht auf einer Karteikarte aus. Der Diktier-Computer repräsentiert eine der ersten praktischen Anwendungen dessen, was in der Zukunft die Computer-Leute mit den Begriffen wie „Hands-off system“ oder „Voice recognition“, einem Teilgebiet der Künstlichen Intelligenz, umrissen ist.

Es handelt sich bei Dr. Chambers Computer um einen ganz normalen PC, kombiniert mit einem

Spracherkennungsmodul aus der Werkstatt der kleinen Bostoner High-Tech-Firma Kurzweil Applied Intelligence Inc., einem Spaltprodukt des legendären Massachusetts Institute of Technology (MIT). Firmengründer Raymond Kurzweil beschäftigt sich seit fast zwei Jahrzehnten mit dem schwierigen Problem, die menschliche Sprache elektronisch zu erfassen, zu analysieren und zu computerisieren.

Durch Zufall wurde 1984 der Röntgenmediziner Alan H. Robbins auf die Experimente Kurzweils aufmerksam - und die Verbindung der beiden blieb nicht ohne Folgen. Dr. Robbins, Professor für Radiologie an

der Tufts University School of Medicine in Boston, suchte seit langem eine rationelle Methode, die bei der Analyse von Röntgenaufnahmen erkannten Befunde schnell und sicher zu erfassen: „Der Röntgenarzt ist es gewöhnt zu sprechen, während er das Bild betrachtet; man kann ihn nicht dazu bringen, seine Arbeitsweise zu ändern.“

Deshalb galten in den Augen des Professors die verschiedenen Versuche, Protokollsysteme auf der Basis von Strichcodes oder berührungsgesteuerten („Touch screen“) Computern zu schaffen, als gezeichnet, „weil der Arzt dabei gezwungen ist, den Blick vom Röntgenbild weg, dem Erfassungssystem zuzuwenden“.

„Frauen aber, die auch das fachmedizinische Vokabular beherrschen, sind selten und teuer. Robbins: „Ich muß nicht selten mehrere Tage auf das Abtippen meiner Berichte warten.“

So unterzog sich der Röntgenprofessor der Mühe, für das im industriellen Bereich - zum Beispiel bei Qualitätskontroll-Stationen am Fließband - bereits bewährte Kurzweil

Statt Sekretärin oder Diktiergerät benutzen amerikanische Mediziner - wie die Röntgenärztin im linken Bild - zunehmend Spracherkennungs-Computer, die medizinisches Vokabular verstehen. Einer der Vorträge: Die Maschine des KI-Wissenschaftlers Raymond Kurzweil (rechts) versteht selbst Englisch mit stärkstem Akzent.

FOTOS: ERNST ULRICH

## Wie man den Computer zum Diktat bittet



Voice System (KVS) medizinische Software zu erarbeiten.

Er untersuchte 12 000 Röntgenberichte und dampfte das darin vorkommende Vokabular auf einen Grundbestand von knapp 1000 Wörtern ein: Es sind die 1000 Begriffe, die der Computer verstehen muß, um in der Röntgendiagnose effizient arbeiten zu können. Inzwischen wird „VoiceRAD“, das so entstandene System für Radiologen, in den USA vermarktet und hat seine Bewährungsprobe bestanden. Es kostet etwa soviel wie neun Monate Gehalt für eine geeignete Typistin - rund 16 000 Dollar. Zuverlässigkeit der Spracherkennung: 95 Prozent.

Wer damit arbeiten will, muß sich und seine Stimme zunächst dem Computer vorstellen. Das geschieht, indem der Arzt oder die Ärztin eine Liste von 209 ausgesuchten Wörtern - von „Yes“ bis „Artherosclerotic“ - vorspricht, was gewissermaßen der Einstimmung der Maschine auf den menschlichen Partner dient. Danach folgt ein zweimaliges Einlesen des 1000-Worte-Fachvokabulars.

Aus den personentypischen „Voiceprints“ dieser Wörter baut sich der Computer ein Lexikon, das ihm später während der Routinearbeit als Vergleichsbasis dient: Sobald das Spracherkennungssystem mit Hilfe von acht speziellen Filter-Chips Übereinstimmung eines gesprochenen Wortes mit einem gespeicherten Voiceprint feststellt, schiebt es die entsprechenden Buchstaben auf den Bildschirm.

Und den Befehl „Signature“ (Un-

terschrift) quittiert das System mit ihrem vollen Namen plus Datum und Uhrzeit. Röntgenärztin Dr. Miriam Vincent: „Die meisten Protokolle sind in dem Moment fertig, in dem ich das Röntgenbild in seinen Umschlag zurückstecke.“

Aufgrund der guten Erfahrungen der Röntgenärzte hat die Kurzweil Inc. die Anwendungen der Spracherkennung auch auf andere Bereiche der Medizin ausgedehnt. So gibt es mittlerweile - von Mammographie bis Neurologie - bereits fünf spezifische Sprachmodule für verschiedene Bereiche der Röntgenmedizin. Daneben trimmte der Traumatologe Dr. Chambers zusammen mit Software-Ingenieuren das System auf die Besonderheiten der Notfallmedizin: nach erfolgreicher Testphase im Nash Hospital von Rocky Mount wird es jetzt unter dem Namen Voice-EM (für Emergency medicine) in den USA angeboten.

Geplant sind weitere medizinische

Applikationen, etwa in der Chirurgie und Pathologie, sowie eine Erweiterung der Voiceprint-Bestände auf einen Umfang von zunächst 5000, später sogar 10 000 Wörtern.

Ein unerwarteter Vorzug der KI-Computer zeigte sich schon bald nach der Installation des allerersten VoiceRAD im Veteran's Hospital in Boston. Während die Schreibdamer der Klinik stets Probleme hatten, wandfreie Transkripte von den Tonbändern des mit stark indischem Akzent sprechenden Röntgenarztes Dr. Srinivasan herzustellen, gelingt das dem Computer tagtäglich so gut wie fehlerfrei. Er ist auf die Stimme des Inders trainiert.

ULRICH BLUMENSCHNEIN

Nächsten Montag:

Gespräch mit dem technisch-wissenschaftlichen Direktor des Deutschen Forschungszentrums für Künstliche Intelligenz (DFKI), Professor Gerhard Barth.



**KÜNSTLICHE INTELLIGENZ**  
Eine Jahrhundertwissenschaft?

**Ein "Wurm" legte Tausende von Computern lahm**  
 Stocker schickte Virus ein / Programm so lange kopiert, bis die Speicherkapazität erschöpft war / Von Horst Rademacher

SAN FRANCISCO, 6. November. Russell Brand wollte sich am vergangenen Mittwoch eigentlich einen ruhigen Abend in seinem Haus in den Hügeln von Berkeley machen. Der 26-Jährige alte Informatiker arbeitet am Lawrence-Livermore-Labor. Eine der besten und am weitesten entwickelten und am weitesten entwickelten, entwickelten werden. Wie an jedem Abend schaltete der Wissenschaftler kurz vor Mitternacht noch einmal seinen Heimcomputer ein und stellte eine Verbindung mit dem Rechner in seinem Büro her. Normalerweise wartet Brand dabei Programme, die die Computer in Livermore während der Nachtstunden berechnet. Wenn er am nächsten Morgen ins Büro kommt, liegen bereits die Ergebnisse seiner Berechnungen vor.

Doch in der Nacht zum Donnerstag spielte der von Brand benutzte Computer in Livermore regelrecht verrückt. „Das System antwortete sehr langsam“, berichtete Brand. „Nachts werden die Computer sonst kaum benutzt, doch an diesem Abend waren die Rechner voll.“ Diese ungewöhnliche Aktivität machte Brand stutzig. Er schaute sich die Programme, die der Computer damals berechnete, etwas genauer an. Dabei fiel ihm ein Programm mit dem Namen „SH“ auf, das er noch nie vorher gesehen hatte. Es war offenbar von außen in die Maschine geschleust worden war. Brand rief daraufhin jene Kollegen an, die für die Computersicherheit in Livermore zuständig sind. Gemeinsam ging man der Sache nach.

Unterdessen kämpften Darren Griffith und seine Kollegen an der Universität von Berkeley schon seit sechs Stunden mit ihren Computern. Sie waren gegen 18 Uhr vom Abendessen in einer Pizzeria in das Institut für experimentelle Informatik zurückgekehrt. Ein Blick auf die Bildschirme zeigte den Experten, daß irgend etwas mit ihrem System nicht stimmte. Auch sie fanden das mysteriöse SH-Programm in ihren Rechnern. „Es blockierte unsere Rechner vollständig“, sagte Griffith.

Weder Brand noch Griffith merkten zu dieser Zeit, daß ihre Computer vom gleichen „Virus“ befallen worden waren. Den Nachtarbeitern in vielen Computercentren in allen Teilen Amerikas ging es ebenso. An den berühmten Universitäten wie Harvard, Stanford, Cornell und dem Massachusetts Institute of Technology (MIT) lief nichts mehr. Die Computer von Unternehmen liefen ebenso Amok wie einige Rechenmaschinen der Nasa und die zahlreicher militärischer Einrichtungen. Am Donnerstagmorgen waren mehr als 6000 Computer in Amerika von diesem Virus befallen. Alle diese Rechenmaschinen hatten nur eines im Sinn: Sie kopierten das SH-Programm immer und immer wieder – so lange, bis ihre Speicherkapazität erschöpft war und sie sich selbst abschalteten.

Am Donnerstag stellte sich heraus, daß nur eine bestimmte Gruppe von Computern infiziert war. Es handelte sich ausschließlich um VAX-Computer und die sogenannten Sun-Workstations. Sie alle arbeiteten unter dem Betriebssystem Unix 4.3 und waren ausnahmslos an eines der größten Datenverbundnetze der Welt angeschlossen, das Arpanet. Raymond Coladay, der Leiter der zum Pentagon gehören-

den und Arpanet betreibenden Defense Advanced Research Projects Agency (Darpa), rief den Notstand aus. Das sei der größte Angriff, der je auf amerikanische Computer unternommen worden sei, meinte er. Das ebenfalls zum Verteidigungsministerium gehörende National Computer Security Center (NCSC) in Bethesda (Maryland) richtete ein Krisenzentrum ein. Bei der „New York Times“ meldete sich unterdessen ein anonymes Anrufer. Einer seiner Freunde, ein Informatikstudent, habe das Computervirus erzeugt und in den Datenverbund eingeschleust, erklärte er. Allerdings habe er einen Programmierfehler gemacht. Was ursprünglich nur als Spaß gedacht war, legte nun den gesamten Datenaustausch zwischen 300 Universitäten, privaten und industriellen Forschungseinrichtungen sowie verschiedenen militärischen Stellen lahm.

Informatikern am Lawrence-Livermore-Labor und bei der Firma SRI in Palo Alto (Kalifornien) gelang es schließlich, das mysteriöse Programm zu isolieren und zu untersuchen. Es handelte sich um ein relativ kurzes Programm von nur 99 Zeilen Länge. Es war in der Programmiersprache „C“ geschrieben. Es wies die befallenen Computer an, das Programm selbst so oft wie möglich zu kopieren. Ein anderer Teil dieses Programms suchte unterdessen in den Dateien nach möglichen Verbindungen zu anderen Computern. Wenn es solche fand, kopierte sich das Programm selbst über Arpanet auf

andere Maschinen. Dort begann es sein zerstörerisches Werk von neuem. Das Programm sei äußerst intelligent geschrieben, meinten die Experten. Es mache sich einige bekannte Unzulänglichkeiten in jenen Systemprogrammen zunutze, die Arpanet mit den Unix-Computern vernetzt verbindet. Nachdem klar war, wie das Programm arbeitete, könnten die befallenen Computer in wenigen Stunden desinfiziert werden. Am Wochenende, so hieß es bei Darpa, habe das Netz wieder normal gearbeitet.

Viele Experten waren überrascht, daß es einem einzigen Programm gelingen konnte, in wenigen Stunden mehr als 6000 Computer lahmzulegen. Noch überraschter waren sie jedoch, als sich der Übeltäter stellte. Es handelte sich um den 23 Jahre alten Robert Morris Jr., einen Informatikstudenten an der Cornell-Universität in Ithaca (New York). Sein Vater Robert Morris Sr., ist am NSCS als Chefwissenschaftler angestellt und gilt als der Fachmann für Computersicherheit in Amerika. Als Spezialgebiet befaßt sich der Vater mit der Sicherheit von Unix-Systemen.

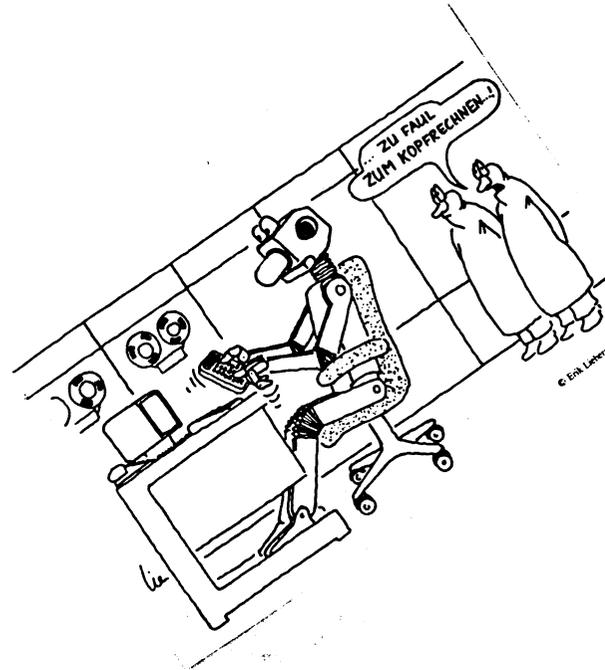
Auf den ersten Blick sieht es so aus, als habe der Sohn mit dem Computervirus seinem alten Herrn einen Streich spielen wollen. Doch viele Experten hatten seit langem damit gerechnet, daß Arpanet zum Ziel von Hackern werden würde. Erste Computer wurden schon im Jahre 1969 diesem Datenverbund angeschlossen. Da-

mit wollte Darpa, der Forschungsarm des Pentagons, Wissenschaftlern und Ingenieuren, die an von ihr geförderten Projekten arbeiteten, den Datenaustausch und die Kommunikation zu erleichtern. Im Laufe der Jahre schlossen sich immer mehr Computer an. Selbst Computer in Europa und Australien wurden angekopelt. Im Jahre 1983 entschloß sich das Pentagon, einen neuen Datenverbund einzurichten. Der Austausch geheimer Daten wurde aus Arpanet ausgelagert und wird seitdem über Milnet abgewickelt.

Von den meisten amerikanischen Universitäten ist der Zugang zu Arpanet relativ einfach. Deshalb befürchteten Sicherheitsexperten schon seit langem, daß clevere Informatikstudenten irgendwann ein Virus in dieses System einschleusen würden. Bisher blieb das Netz davon jedoch verschont. Wie Morris junior sagte, hatte er nicht vor, das gesamte Netz lahmzulegen. Er wollte lediglich einen „Wurm“ einschleusen, der sich allmählich durch die an Arpanet angeschlossenen Computer „frißt“. Dabei sollte er keine Spuren hinterlassen – bis auf eine. Morris hatte seinen Computervirus so programmiert, daß jedes Programm sich regelmäßig bei einem Computer in Berkeley melden sollte. Auf diesem Computer mit dem Namen „Ernie“ hatte Morris eine Datei angelegt. In ihr sollten die Meldungen gesammelt und damit die Ausbreitung des Wurms dokumentiert werden. Morris habe damit zeigen wollen, wie schlecht der Datenverbund gegen unbefugte Benutzung gesichert ist.

Doch durch einen kleinen Programmierfehler geriet ihm der Wurm außer Kontrolle. SH kopierte sich tausendmal schneller, als es Morris vorgesehen hatte. Da jede dieser Kopien ihren Standort an Ernie in Berkeley zurückmeldeten, hatten Griffith und seine Kollegen besondere Schwierigkeiten, ihre Rechner unter Kontrolle zu bringen.

Seit diesem Zwischenfall mit Arpanet wird in Amerika wieder über die Computersicherheit diskutiert. Zwischen August 1986 und Juni vergangenen Jahres war es zum Beispiel einem Hacker aus Hannover gelungen, sich über das Datennetz der Bundespost (Datex-P), dem vom Flugzeughersteller McDonell-Douglas betriebenen „Tymnet“ und Arpanet Zugang zu vielen amerikanischen Computern zu schaffen. Er stöberte dort in Dateien herum und suchte offenbar nach geheimen Informationen über SDI. Im September wurde erstmals in Amerika ein Programmierer wegen eines Computervirus verurteilt. Der Verurteilte, Donald Burleson, arbeitete für eine Versicherungsgesellschaft in Fort Worth (Texas). Er wurde schuldig gesprochen, kurz vor seiner Entlassung die Computer der Gesellschaft mit einem Virus infiziert zu haben, das die Informationen über 168 000 Versicherte zerstört hat. Burleson konnte jedoch nur verurteilt werden, weil er wirklich Schaden in den Dateien angerichtet hatte. Im Gegensatz dazu blockierte das Programm des jungen Morris zwar für Stunden die an Arpanet angeschlossenen Computer. Es zerstörte jedoch weder Daten noch hinterließ es – nachdem die Rechner desinfiziert waren – irgendwelche Spuren.



**CLUB-80 Bibliothek**  
-----

In der Club-Bibliothek sind folgende Neuzugänge zu verzeichnen:

- Tischcomputer-BASIC für Anfänger  
(ca. 200 Seiten)  
meiner Meinung nach nur für absolute Newcommer
- BASIC-Übungen für Anfänger  
(ca. 160 Seiten)
- Disk-Interfacing-Guide  
(ca. 50 Seiten)  
für Hardware-Freaks
- Programmieren in Maschinensprache mit Z80  
(ca. 80 Seiten - Hofackerqualität?)

Gleichzeitig haben wir 3 Bücher zu verschenken - und zwar an den, der sich als Erster bei mir meldet:

- Programmieren mit TRS-80  
(ca. 200 Seiten)  
bekanntestes Einsteigerbuch für TRS-80
- TRS-80 Disk And Other Mysteries  
(ca. 100 Seiten)  
erläutert die Diskettenzugriffe
- TRSDOS 2.3 Decoded And Other Mysteries  
(ca. 300 Seiten)  
enthält die Assembler-Listings für TRSDOS 2.3

Das war's von der Club-Bibliothek

1. Vorsitzende                   Gerald SCHRÖDER  
Am Schützenplatz 14  
2105 Seevetal 1  
☎ 04105 / 2602

2. Vorsitzende                   Bernd RETZLAFF  
Kleiner Sand 98  
2002 Uetersen  
☎ 04122 / 49551

Hardwarekoordinator           Eckehard KUHN  
Im Dorf 14  
7443 Frickenhausen 1  
☎ 07022 / 45417

Diskotheke                       Werner FÜRSTER  
Club-Bücherei                   Christoph-Krebs-Straße 9  
0720 Schweinfurt  
☎ 09721 / 21841

Redaktion                       Jens NEUEDER  
Panoramastraße 21  
7170 Michelbach /Bilz  
☎ 0791 / 42877

Es erfolgt keine Zensur oder Kontrolle der jeweiligen eingeschickten Infobeiträge durch die Redaktion.

Autoren                         Die Redaktion bedankt sich bei den im INHALTSVERZEICHNIS genannten Autoren für die Mitarbeit an der Club-INFO.

Druck                            Horst-Dieter Schroers  
Breslauer Straße 9  
8016 Feldkirchen  
☎ 089 / 9832615

Bankverbindung               des CLUB 80  
Postgirokonto Peter STEVENS  
Sonderkonto CLUB 80  
Konto-Nummer 285 491 - 465  
Postgiroamt Dortmund  
BLZ 440 180 46

Hallo Club-80er,

endlich ...

... werdet Ihr sagen, erscheint nun das neueste INFO.

Die Club 80 Redaktion ist aus Ihrer Versenkung wieder aufgetaucht !

Das übliche Sonderheft am Jahresende, mit dem Jahresinhaltsverzeichnis und der kompletten Auflistung unserer Software sowie der vollständigen Bücherliste unserer Club-80-Bibliothek, ist leider noch nicht fertig.

Werner hat Probleme mit seinem Drucker.

Es wird also alsbald noch einen Nachschlag für dieses INFO geben.

Und das Nächste soll ca. 3-4 Wochen nach Erscheinen des 26. INFO's folgen. Dazu brauche ich aber Eure Mithilfe. Ich suche noch Infomaterial. Ihr werdet ja in der Zwischenzeit schon fleißig für das 27. INFO gesammelt haben? Ich hoffe es.

Als weiteres möchte ich ein Sonderheft aus den in INFO 25 als Liste vorgestellten ELCOMP-Artikeln machen. Ich hoffe, daß das dann bis zum Clubtreffen fertig ist.

Diesem INFO liegen wieder zwei Referenzkarten am Heftende bei.

dBase II

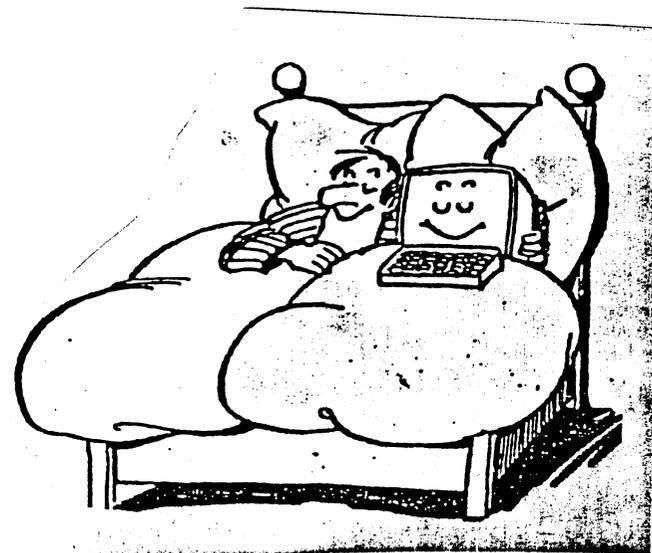
Alcor Pascal

Bis zum nächsten INFO wünsche ich Euch eine gute Zeit und vergesst nicht, ich suche noch Infoartikel für das 27. INFO !!

Termin in ca 3 Wochen.

Es grüßt Euch Euer

Jens



Name	Vorname	Straße	PLZ	Stadt	Telefon	privat	// geschäftlich
Ahlens	Armin	Rotdornstraße 13	2116	Hanstedt / Nindorf	04185 / 7819	//	-
Albers	Herbert	Zum Düwelshöpen 14	2117	Wistedt	04182 / 8799	//	-
Beckhausen	Wolfgang	Vüfelfer-Kaule 30	5060	Bergisch-Gladbach 1	02204 / 62781	//	-
Bernhardt	Helmut	Hafenstraße 7	2305	Heikendorf	0431 / 241907	//	0431 / 74047
Betz	Heinrich	St. Wolfgangstraße 13	0551	Hausen	09191 / 31698	//	09191 / 611108
Brandl	Hermann	Cäciliastraße 30	0439	Postbauer-Heng	09180 / 493	//	0911 / 219-245
Brans	Jörg	Tieloh 55	2000	Hamburg 60	040 / 6912716	//	-
Braun	Harald	Postfach 8011	2300	Kiel 17	-	//	-
Bröcker	Detlev	Eisenstraße 92	4000	Düsseldorf 1	0211 / 772436	//	-
Böckling	Ulrich	Am Sonnenhang 11	5414	Vallendar	0261 / 69522	//	02631 / 895168
Dreyer	Gerald	Am Speiergarten 8	6200	Wiesbaden-Bierstadt	06121 / 508218	//	-
Eilers	Hans-Joachim	Theodor-Heuss-Straße 129	2900	Oldenburg	0441 / 53239	//	0441 / 7983845
Förster	Werner	Christoph-Krebs-Straße 9	0720	Schweinfurt	09721 / 21841	//	09721 / 51256
Hartmann	Hans-Günter	Möwenstraße 9	2076	Barne 2	04406 / 1911	//	0421 / 2483405
Heidenreich	Ulrich	Werdenstraße 35	4300	Essen 1	0201 / 282770	//	-
Held	Manfred	Stirner Straße 22	0935	Pleinfeld	09144 / 6563	//	0911 / 2195245
Hentz	Werner	Am Tränkgarten 20	6457	Maintal 2	06109 / 66625	//	0691 / 583639
Hermann	Klaus	Forchenstr. 8	7401	Pliezhausen	07127 / 71945	//	-
Homann	Matthias	Maschener-Straße 49 A	2105	Seebet 1	04105 / 50222	//	040 / 7511-278
Issig	Günter	Teutonenstraße 20	7100	Heilbronn	07131 / 42601	//	-
Jablotschkin	Rainer	Südertor 10	4700	Lippstadt 8	02921 / 23245	//	02921 / 70431
Kostya	Mary Jo	Balberstraße 68	CH-8038	Zürich	00411 / 482894	//	00411 / 258170
Krahn	Sven	Am Affenberg 20	7730	Villingen-Schwenningen	07721 / 1785	//	-
Krispin	Michael	Schwanstraße 8	4150	Noers 3	02841 / 73690	//	-
Kuhn	Eckehard	Im Dorf 14	7443	Frickenhausen 1	07022 / 45417	//	09171 / 832665
Lachmann	Wolfgang	Am Ringofen 11	6407	Schlitz	06642 / 6940	//	069 / 832051
Lohmann	Willi	Honnorskamp 2	4300	Essen 12	0201 / 356319	//	-
Loose	Gerhard	Viefhaushof 42	4306	Essen 13	0201 / 212608	//	-
Magnus	Andreas	Pommernstraße 18	4650	Gelsenkirchen	0209 / 870230	//	-
Mahlert	Herbert	Baumschulstraße 7	4100	Duisburg 14	02135 / 81462	//	-
Mand	Harald	Kleinflintbeker Straße 7	2302	Flintbek bei Kiel	04347 / 3629	//	0431 / 3013580
May	Holger	Marienstr. 9	5768	Sundern 2	02935 / 1668	//	-
Meklenburg	Heinz-Dieter	Amruner Weg 1	2262	Leck /NF	04662 / 691	//	-
Mank	Christian	Ollsener Straße 52	2116	Hanstedt	04189 / 7825	//	-
Misioch	Waldemar	Adenauerring 25	0911	Röthenbach a. d. Pegnitz	0911 / 506051	//	0911 / 107945
Mühlenbein	Klaus-Jürgen	Am Nöschgarten 28	0940	Weinheim -Lützel Sachsen	03201 / 55052	//	-
Müller	Kurt	Solttaustraße 24a	2050	Hamburg 90	040 / 7246083	//	04151 / 8891-37
Neueder	Jens	Rudolf-Then-Straße 32	7178	Gschlachtenbretzingen	0791 / 42877	//	0791 / 44-667
Obermann	Hartmut	Schwalbacher Str. 6	6209	Heidenrod 1	06124 / 3913	//	-

Stand Januar 1993  
 Bitte überprüft eure Daten auf Richtigkeit  
 und teilt mir Ineegelmäßigkeiten mit.  
 Die Redaktion

**CLUB 80** Mitgliederadressenliste

Name	Vorname	Straße	PLZ	Stadt	Telefon	privat	// geschäftlich
Obscheringkat	Helmut	1 rue des Bruyeres	F-68360	Soultz	0033/89-762698	//	-
Piller	Walter	Rohnenstraße 8	CH-8835	Feusisberg	01	/7847418	//
Preußing	Wolfgang	Dammstraße 68	4300	Essen 14	0201	/530878	// -
Raggan	Hans	Backnanger Weg 36	7146	Tamm	07141	/603611	// 0711 /2630473
Rank	Heinrich	Frühlingstraße 2	8000	Fürstenfeldbruck	08141	/43791	//
Rensch	Richard	Bahnhofstraße 100 (Postf. 226)	7128	Lauffen am Neckar	07133	/4167	// 07133 /8415
Retzlaff	Bernd	Kleiner Sand 98	2082	Uetersen	04122	/43551	// 04103 /605310
Rinio	Gerd	Rennbahnstraße 9	2000	Hamburg 74	040	/6552630	// -
Ruschinski	Claus	Pommernstraße 21	4370	Marl	02365	/34646	// -
Rychlik	Andreas	Königsberger Allee 120	4100	Duisburg 1	0203	/331383	// 0203 /331383
Schmid	Alexander	St. Cajetan-Straße 38/VII	8000	München 80	089	/495326	// -
Schmitz	Paul-Jürgen	Bremer Straße 9	6236	Eschborn	-		// -
Schneider	Bernd	Wirtwis Straße 13	CH-8951	Fahrweid	0041/1/7400348	//	-
Schober	Frank-Michael	Weberweg 2	DDR7590	Spremberg	003	/75/4-4565	// -
Schrewe	Christian	Fliederweg 32	4000	Düsseldorf 31	0203	/740897	//
Schroers	Horst-Dieter	Breslauer Str. 9	8016	Feldkirchen	089	/9032615	// 089 /9032615
Schröder	Gerald	Am Schützenplatz 14	2105	Seevetal 1	04105	/2602	// -
Schröder	Peter	Theodor-Fahr-Straße 32	2000	Hamburg 62	040	/5311582	// -
Schut	Andre	Sanderstraße 26	1000	Berlin 44	030	/6917861	// -
Schäfer	Walter	Rathausstr. 4	8160	Miesbach	08025	/1631	// 08025 /41247
Seelmann-Eggebert	Jörg	Henri-Spaak-Straße 96	5305	Alfter	0228	/643853	// -
Sopp	Arnulf	Wakenitzstr. 8	2400	Lübeck 1	0451	/791926	// -
Stephan	Hans-Martin	Am Glasesch 9a (Postf. 1207)	4506	Hagen a.TW.	05401	/99505	// 05401 /30096
Stober	Reiner	Nelkenstraße 12	3216	Salzhemmendorf 4	05153	/1564	// -
Sörensen	Rüdiger	Thomas-Mann-Straße 3A	6500	Mainz 1	06131	/32860	// 06131 /395260
Tornow	Wilhelm	Görlitzer Straße 16	2190	Cuxhaven 13	04123	/1355	// 0431/362055
Volz	Oliver	Waldenburgstraße 73	7000	Stuttgart 80	0711	/7353817	// -
Waccus	Michel	Mühlhofweg 2a	CH-8266	Steckborn	054/612560	//	-
Wacker	Fred	Weingärtenstraße 4	7554	Kuppenheim	07222	/48931	// -
Wucherer	Jürgen	Menzelstraße 1	7750	Konstanz	07531	/54686	// -
Wulf	Hans-Otto	Im Brahmkamp 38	4250	Bottrop	02041	/688972	// -