

# MTX User Club Deutschland

Info 13  
10. 05. 1986

**Zweck:** Zusammentragen und Austausch von Tips & Tricks u.s.w., Hilfestellung bei allen möglichen Problemen, Aufbau einer Programmbibliothek und Basteln von Hardware-Erweiterungen.

**Programme** (nur **Selbstgeschriebenes**): Tausch von kurzen und einfachen Routinen. Gute Programme (mit Dokumentation) können über den Club an alle Mitglieder verkauft werden. Wer solche Programme an uns schickt erhält ggf. Verbesserungshinweise und eine Besprechung im Info.

**Mitglied** kann jeder werden! Keine Beitragsgebühr! Anmeldung kostet DM 1.-.

**Verpflichtungen:** Einsendung unseres Anmeldeformulars.

**Bitte:** Einsendung von Tips & Tricks, Fragen, Antworten, kurzen Routinen, Programmen, Beiträgen zum Info, Hinweisen auf preiswerte Hard- und Software, und was noch so zusammenkommt und andere interessieren könnte.

**Club-Info**, unser Blatt, verschicken wir ca. 6-wöchentlich. Inhalt ist alles was uns über den MTX/FDX (ohne Copyright) in die Hände fällt. Es kostet nicht über DM 12.- (90 Seiten) je Exemplar. Jeder kann dazu Beiträge liefern und hier gratis Kleinanzeigen veröffentlichen.

**Kosten:** Wir berechnen ausschließlich Selbstkosten und verschicken **nichts**, wenn's Guthaben nicht reicht! (s.u.)  
Schüler, Studenten, Auszubildende, Rentner und Arbeitslose erhalten einen Nachlaß von 40% auf die zukünftigen Infos nach Einsendung einer entsprechenden Bescheinigung.

**Geld/Konto:** Für jedes Mitglied führt Herbert Herberg ein Konto, von dem die jeweils entstehenden Kosten abgehen. Der Kontostand wird bei **jeder** Sendung mitgeteilt (**er steht über der Anschrift**), und kann selbstverständlich jederzeit erfragt werden! Wir verschicken nur gegen Vorkasse!

Einzahlungen bitte auf's Club-Konto: (oder V-Scheck)  
(**Absender!** incl Name **und** Anschrift nicht vergessen!)

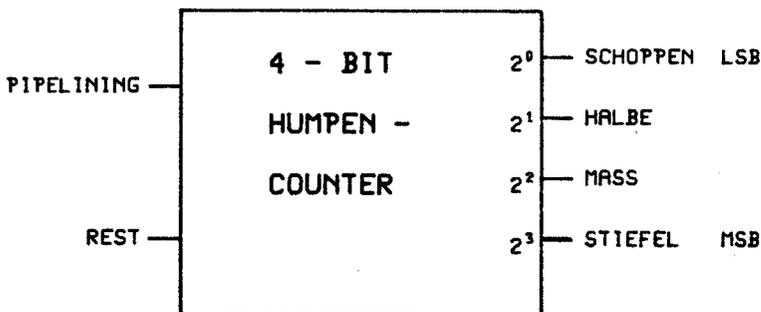
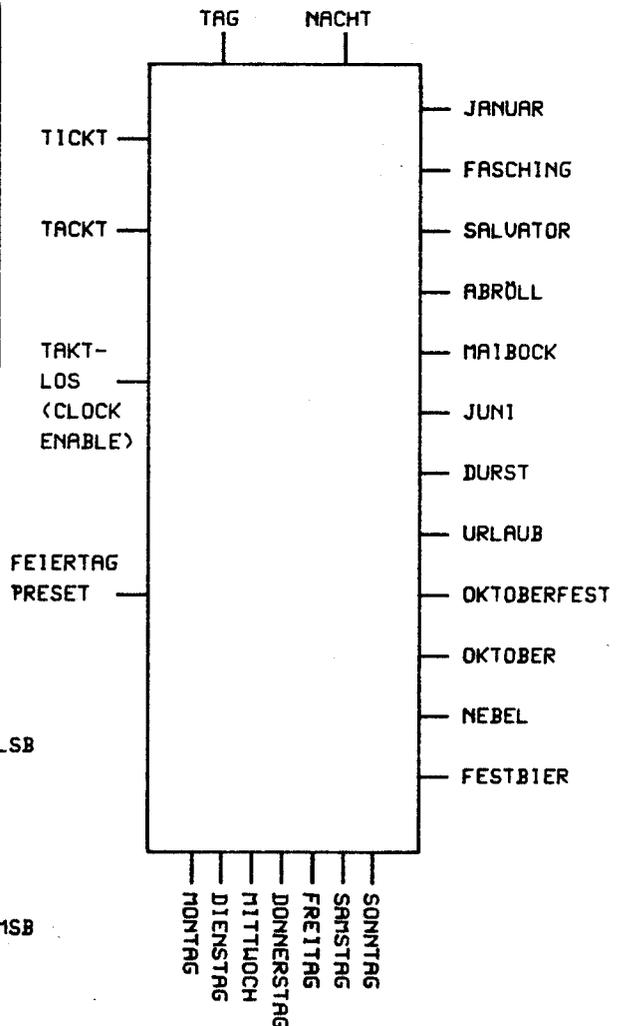
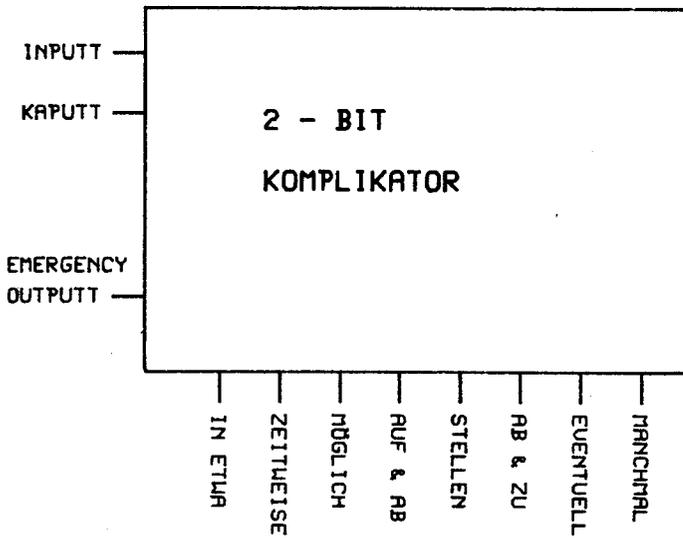
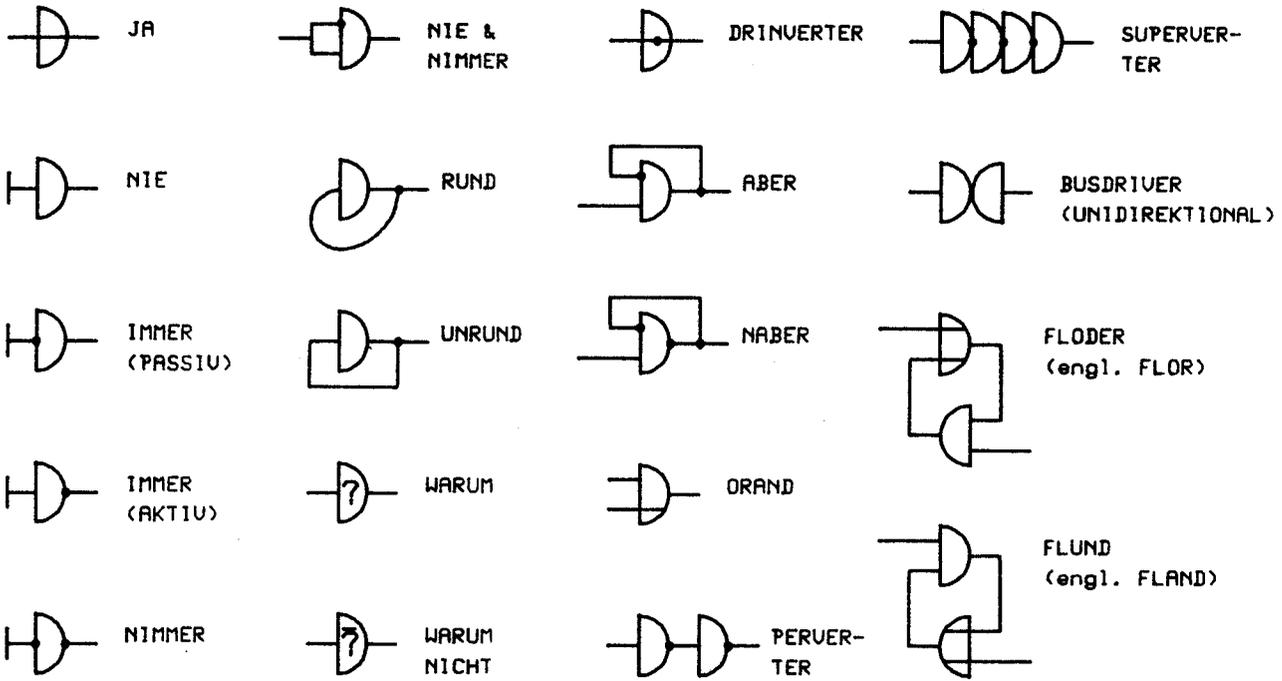
Postgiroamt Hamburg, BLZ 200 100 20,  
Herbert Herberg, Sonderkonto C, Nr. 3480 00-200

**Kontaktadressen:** (nach PLZ geordnet)

Herbert Herberg Sonnenau 2 2000 Hamburg 76 (040) 200 87 04	Frank Bueschler Am Ochsenzoll 3 2000 Norderstedt (040) 527 75 81	Christian Löhrmann Grevenbleck 24 3005 Hemmingen 1 (0511) 41 78 77	Detlef Harms Feldstr. 26 4902 Bad Salzufen (05222) 84 0 82
---	---	---	---

Thomas Pflaum Leipziger Platz 1 8500 Nürnberg 20 (0911) 51 35 21	Martin Hofmann Erikaweg 8 8580 Bayreuth (0921) 5 41 46	Volker Griener Sophienstr. 7 8581 Donndorf (0921) 3 24 27
---	---	--

# BAVARIA - LOGIKFAMILIE



Inhaltsverzeichnis**C L U B:**

Dies und Das von Herbert Herberg	Seite 1
Korrektur & Nachtrag / Wer tut Was / Kleinanzeigen	Seite 2
Leserbriefe	Seite 52
Mitgliederliste	
Formular CLUBTREFFEN	

**H a r d w a r e:**

ECB und MTX/FDX	Seite 3
Schnee	Seite 5
Farbmonitor	Seite 6
Digitalizer	Seite 7
MTX schneller / Maus	Seite 8
Maus	Seite 9

**T U R B O:**

True & False / Sprites	Seite 10
SPRITE-Routinen	Seite 11

**B A S I C:**

BASIC.COM / Seikosha / RND & INT	Seite 12
SELECT/MERGE	Seite 13
CONINC (Sprite) / HCHAR	Seite 16
Gelenkte Ausgaben	Seite 19
Gelenkte Ausgaben / Noddy-Fehler	Seite 20
Selbstklebeetiketten	Seite 21
Suchprogramm	Seite 22
Fourier-Analyse	Seite 23
Zahlendarstellung Float	Seite 24
Tasturentprellung	Seite 25
Cassetten-Inhaltsverzeichnis	Seite 26

**N e w W o r d:**

Printer-Initstring	Seite 28
Printer-Overlay	Seite 29
Printer Overlay / deutsches Handbuch	Seite 31
Rette Text nach Bdos Err / Verzögerungszeiten	Seite 33

**C P / M:**

Disc-Utility zum retten gelöschter Dateien	Seite 34
Disc-Utility zum retten gelöschter Dateien / SuperCalc	Seite 37
.SUB-Kommentare / RAM 3	Seite 38
Zum Thema CP/M 3.0	Seite 39

**A s s e m b l e r:**

Quadratwurzeln	Seite 41
Tips	Seite 42
IX- und IY-Register	Seite 43

**M a i l b o x:**

MTX-Mailbox	Seite 44
Mailboxliste	Seite 47

**S o f t w a r e:**

Neuheiten	Seite 51
-----------	----------

(Diesmal sind die Seitenzahlen wohl richtiger!)

Preis für dieses Info: DM 11,50

C L U B: Dies und Das von Herbert Herberg

Liebe Clubmitglieder,

da ich tatsächlich in **Hamburg** stationiert bin, könnt Ihr mich jetzt auch Donnerstags erreichen. Allerdings stimmen die Telefonzeiten von Donnerstag nicht: erst ab 18.00 Uhr (solange habe ich schließlich Dienst).

Da sich von Euch (insbesondere die neueren Mitglieder) es vielleicht nicht wissen: Ich habe mich für die Anrede **DU** im Info entschlossen, da ich SIE für zu gestelzt halte, und es noch unschöner finde, wenn das Info unpersönlich ist.

Bitte bedenkt bei Bestellungen, daß wir nichts verschicken, wenn Ihr nicht genug auf Eurem **Konto** habt!!! Wenn Ihr Software bei anderen Clubmitgliedern bestellt und kein Geld beifügt, ... Frank Bueschler und Christian Löhrmann fragen bei fehlendem Geld bei mir nach, ob das Geld auf dem Konto reicht!

Unsere nächste **Bestellaktion** gibt es im Juli/August. Wer nicht so lange warten möchte sage uns, was er haben möchte, und wir versuchen es möglichst preiswert zu finden!

Frank hat in Kürze einige Prüfungen und muß obendrein Software erstellen (in der Firma, in der Auszubildender ist), und hofft, daß er eine Chance hat übernommen zu werden, falls die Software gut ist. Demzufolge hat er wenig Zeit und leidet unter Streß. Daraus ergeben sich: 1. Wer bei Frank etwas bestellt und noch nicht erhalten hat, möge die Bestellung bitte wiederholen; 2. Die Clubexterne Public-Domain gibt's auch bei mir.

**Clubtreffen Hamburg am 28. & 29. Juni 1986**, Anmeldung nur mittels Anmeldeformular (anbei). Ablauf wird vorwiegend in Form von Gesprächen sein (in großem oder kleinem Kreis). Ein MTX/FDX steht für Vorführungen, etc. bereit (es ist weder Frank's noch meine!). Diesmal ist der Schwerpunkt die Kommunikation und nicht das Sitzen am Computer! Welche Themen dran kommen hängt von Euch ab! Ihr macht das Programm. Jeder Teilnehmer möge bitte DM 4.- beisteuern, damit wir Kaffee/Bier/Saft/Knabberkram bereitstellen können. Ich glaube, damit ist alles gesagt.

Das **PROBLEM PROM/PAL**-Programmer ist endlich gelöst! Ich habe das Gerät, und es läuft einwandfrei, d.h. viele von Euch haben die PROMs/PALs schon erhalten. Da sich seit den ersten Bestellungen einiges geändert hat (insbes. 512k-Karte) und diese auch ziemlich lange zurückliegen, bitte ich alle, die Bestellungen (falls noch nichts erhalten) zu wiederholen! Danke. Lest bitte zuvor den Artikel:

**PROMs und PALs** (Herbert Herberg, 2000 - diese Zahl ist die PLZ)

Warum läuft beim Umbau auf MTX 512 der MTX 500 nicht mehr unter BASIC, insbesondere nicht die ROMs? Die Lösung ist einfach:

Auf der Hauptplatine sind 24k ROM, wobei diese in entweder

3 ROMs mit je 8k

1 ROM mit 8k und einem mit 16k

1 ROM mit 8k und zwei ROMs mit je 8k übereinandergestapelt eingebaut sind. Je nach ROM-Verteilung muß ein anderes PAL her, sonst stimmt die Adressierung nicht mehr!

Bitte bei Bestellungen von MTX 500 -> MTX 512 - PALs die ROM-Aufteilung angeben.

Bei 128k-PROMs und 512k-Karten bitte bei der Bestellung den MTX-Typ angeben (ein umgabauter MTX 500 gilt als 512), damit wir die richtigen Teile liefern können.

C L U B: Korrektur & Nachtrag / Wer tut Was / Kleinanzeigen

### Korrektur & Nachtrag

#### Wer tut Was

Allround	H. Herberg, C. Löhrmann, F. Bueschler
(FDX-)BASIC	A. Viebke, H. Oppmann, V. Griener, C. Romanazzi
CP/M System	B. Freusing, H. Herberg
NewWord	U. Grass, H. Herberg
Turbo-Pascal	D. Krumnow, B. Preusing, T. Pflaum
Forth	E. Gunter
SuperCalc	W. Gieger
C	H. Lauinger
Single-Board	H. Lauinger
Analog-Datenerfassung	H. Lauinger
Grafik	M. Köster, A. Schmitt, C. Romanazzi
Was gibt's wo billig	F. Bueschler
Platinen	D. Harms
Hardware	M. Flume, H. Wenzek

Wer sich auf dieser Liste fehlt am Platz oder vermißt fühlt ... schreibe mir. (Bitte nur ernstgemeinte Zuschriften, d.h. Ihr solltet im genannten Bereich "firm" sein).

Anmerkung: Wolfgang Gieger hat sich bemüht einiges zum Thema SuperCalc einzudeutschen!

#### Kleinanzeigen

##### KONTAKTE

Herbert Herberg, Sonnenau 2, 2000 Hamburg 76, 040 - 2008704 und Frank Bueschler, Am Ochsenzoll 3, 2000 Norderstedt, 040 - 5277581: Wir vermitteln jederzeit gebrauchte Geräte und Teile der selben. Außerdem wissen wir i.a. was es wo am billigsten gibt.

##### VERKAUF

Martin Wessel, Hartkrögen 43, 2000 Hamburg 56, 040 - 812662 verkauft MTX 500.

Hartmut Traber, Hohbeulstr. 8, 5270 Gummersbach, 02261 - 65399 verkauft Schaltnetzteil +-12 V/1 A, +5 V/3 A, -5 V/-zig mA (also vier Spannungen) repariert und getestet mit Unterlagen und Verbesserungsvorschlägen.

Anton Reiser, Gindels 7, 8977 Rettenberg, 08327 - 601 verkauft das Buch SuperCalc richtig eingesetzt (enthält alle Tricks zu SC, 7 praxisnahe Beisp.) Neupreis DM 58.- Gut erhalten incl. Disc mit Beispielen für DM 40.-

##### ANKAUF

Herbert Herberg, Sonnenau 2, 2000 Hamburg 76, 040 - 2008704 sucht leeres Original-FDX-Gehäuse.

Dieter Ritter, Charlottenstr. 8, 8800 Ansbach, 0981 - 2684 sucht gebrauchten Drucker, vorzugsweise DMX 80.

Ulrich Hönisch, Wachholtzstr. 8, 3300 Braunschweig, 0531 - 343167 sucht gebrauchten Drucker, vorzugsweise DMX 80.

Hagen Wenzek, Blogauer Straße 5, 5300 Bonn 1, 0228 - 660846 sucht einen preiswerten Oszillograph.

##### SONSTIGES

Hans-Dieter Müller, Starenweg 7, 7057 Winnenden, 07195 - 66979 kopiert 3''-Schneider-Disketten auf MTX-Format für DM 25.- je Disc.

Hardware: ECB und MTX/FDX**Der ECB-Bus hat eine Macke und  
So funktioniert MTX <-> FDX** (Herbert Herberg, 2000)

Endlich hatte ich den PROMmer, mit dem ich EPROMs, PROMs und PALs brennen kann, aber jeder Versuch diesen laufen zu lassen schlug fehl. Bernd Preusing hatte genausowenig Erfolg bei seinen Versuchen die 1-Meg-Karte von c't als RAM-Disc zu implementieren.

Was tun sprach Zeus?

Das, was uns beiden seit je her suspekt vorkam war die Dioden-Angelegenheit auf der Platine, die das folgende Problem lösen sollte:

Wenn der MTX auf RAM zugreift, so geschieht das innerhalb des MTX (also der Tastatur selbst: Hauptplatine und Speichererweiterung). Aber der MTX greift hin und wieder auf Ports zu. Die einen Ports im MTX (CTC u.a. für CLOCK und Cassette, VDP für 40-Zeichen-Schirm, Port 7, u.s.w.) und auf solche in der FDX (80-Zeichen-Karte und Floppy-Controller). Was aber nun mit den nicht belegten Ports? Ganz einfach: Memotech ist davon ausgegangen, das Erweiterungen, die Ports belegen in der FDX eingebaut werden. Die Möglichkeit der Erweiterung seitlich am MTX dient für ROM-Erweiterungen!

Um die Sache einfacher zu machen sind alle Port-Adressen des MTX zwischen #00 und #1F (das # vor einer solchen Zahl bedeutet hexadezimal), also unter #20, und die in der FDX von #20 bis #FF, wobei derzeit nur Ports von #30 bis #4F (wenn auch nicht alle davon) in der FDX benötigt werden. Nun stellt sich die Frage, wie funktioniert es, das je nach Port die Werte aus der richtigen Ecke (MTX oder FDX) geholt werden? Auch hier gibts eine genial einfache Lösung: Wenn die Port-Adresse unter #20 ist wird die FDX nicht angesprochen, bei Adressen zwischen #20 und #FF wird auf die FDX zugegriffen.

Auf der Interface/RS232-Karte und in der FDX sind ICs, die die Signale verstärken, damit sie das 60-polige Flachbandkabel überstehen. D.h. wenn die FDX sendet, empfängt der MTX und umgekehrt! Da aber der ECB-Bus an das Flachbandkabel zwischen MTX und FDX kommt, muß sichergestellt werden, daß wenn eine Platine auf dem ECB-Bus angesprochen wird die FDX nicht dazwischenfunkelt. Will der MTX an die ECB-Platine etwas senden, empfängt die FDX, was nicht stört - will jedoch der MTX etwas von der ECB-Platine empfangen, dann sendet die FDX und die ECB-Platine. Frage ist, wer gewinnt ...? Es muß also sichergestellt werden, daß der MTX vom ECB empfangen kann, ohne daß die FDX sendet. Das sollte diese dubiose Dioden-Konstruktion schaffen, aber das klappt nicht!!!! Meine Lösung ist vielleicht nicht das Non-plus-ultra, aber sie funktioniert einwandfrei! Wenn wir eine ECB-Platine anschließen wollen, so können wir i.a. die Port-Adressen dieser Platine über DIP-Schalter (o.ä.) bestimmen. Nun habe ich mir gedacht, daß folgende Port-Aufteilung geeignet ist: MTX #00 - #1F, FDX #20 - #7F und ECB #80 - #FF, d.h. die ECB-Platinen können Port-Adressen ab #80 belegen. Diese Lösung vereinfacht die Sicherstellung, daß alles klappt. (Angenehmerweise kollidiert dieses nicht mit den Port-Adressen von Memotech!!).

Auf dem 60-pol-Kabel von MTX zu FDX ist ein Signal, genannt 245DIR, mit dem die Richtung der Kommunikation MTX-FDX angezeigt wird (245DIR = Direction des 74LS245 = Richtung des Bustreiber-ICs, also die Übertragungsrichtung - der 74LS245 ist einer der o.g. Verstärker für die Signale). Dieses Signal wird vom MTX bereitgestellt und an die FDX übergeben. Alles was zu tun ist, ist es dieses Signal (das entscheidet, ob die FDX sendet oder empfängt) so zu ändern, daß die FDX nur dann sendet, wenn auf Ports zwischen #20 und #7F zugegriffen wird.



Hardware: SchneeNach dem Einschalten nur Schnee (Jan Brederke)

Wie ich auf dem letzten Clubtreffen hörte, kam es nicht nur bei mir vor, daß der MTX/FDX nach dem Einschalten nicht bootete, sondern nur "Schnee" auf dem Bildschirm anzeigte. Man mußte daher zum Starten immer erst die Resettasten drücken.

Dieses Problem habe ich bei mir mit nur zwei dazugelöteten Bauteilen gelöst. Laut Schaltplan sind schon parallel zu den Resettasten ein Widerstand und ein Kondensator (C11 mit 10 µF) eingebaut. Nach dem Einschalten ist die Spannung am Kondensator Null und soll dann langsam ansteigen, so daß für den Rechner die Resettasten gedrückt erscheinen. Wenn aber die Betriebsspannung zu langsam ansteigt, ist der Kondensator schon zu weit aufgeladen, sobald der Rechner endlich läuft.

Also mußte bei mir der Kondensator vergrößert werden oder ein zweiter parallel geschaltet werden. Da ich keinen Eingriff in die dichtgepackte und empfindliche Rechnerplatine machen wollte, habe

ich einen 10 µF/16 V-Kondensator auf die Tastaturplatine im Oberteil des Rechners gelötet, und zwar elektrisch parallel zu den Resettasten. Eine dafür sehr geeignete Stelle ist die Lötseite der Steckerleiste des Kabels, das zur Hauptplatine führt. Die Anschlüsse der Resettasten sind die letzten beiden Kontakte in Richtung Vorderseite, also der Seite, auf der die Leertaste sitzt. Bei mir ist der Kontakt ganz außen der Minus-Anschluß, der daneben der Plus-Anschluß.

Dies ist zu beachten, da man einen Elektrolytkondensator nicht falsch gepolt anschließen darf. Zum Schutz gegen einen zu hohen Kurzschlußstrom beim Drücken der Resettasten habe ich in Serie zum Kondensator noch einen 22-Ohm-Widerstand gelegt.

Bei dieser Gelegenheit habe ich gleich noch einige kalte Lötstellen (dSM) auf der Tastaturplatine nachgelötet, und seitdem prellt meine Leertaste auch nicht mehr.

## HAUPTSATZ DER PROGRAMMIERUNG ##

(Christian Meevissen)

" Es ist absolut unnötig, Programme zu testen oder zu dokumentieren "

# BEWEIS #

I) Konstruktives Verfahren nach Blankenburg

" Man mache die Algorithmen gleich richtig, dann weiß man, daß sie stimmen und damit erübrigt sich jeglicher Test. "

Der Vorteil dieses Verfahrens liegt darin, daß es keinerlei Dokumentation vorsieht, denn bei einem korrekten Programm wird überhaupt nichts geändert.

II) Redundanzausschluß nach Mooser

" Bei Programmen, von denen man weiß oder vermutet, daß man sie nicht braucht, entfallen Test oder Dokumentation. "

III) Ward'sches Dual-Prinzip zu II)

" Bei Programmen, von denen man weiß, daß die Aufgabenstellung falsch ist, die man jedoch entwickeln M U S S , entfallen sowohl Test als auch Dokumentation. "

Hardware: Farbmonitor

Bemerkung zu II) und III)

a) Diese beiden Kategorien von Programmen definieren den sogenannten " WRITE-ONLY-TYP " .

b) Bei Programmen vom " WRITE-ONLY-TYP " kann man unter Umständen sogar auf das " WRITE " verzichten.

IV) Richter'sches Prinzip

" Es ist vollkommen gleichgültig, ob man Programme dokumentiert oder nicht. Ein anderer versteht das Programm sowieso nicht und entwickelt es deshalb neu. "

V) Prinzip der Autoren-Abhängigkeit (Verfahren von Christian)

" Bei Programmen, bei denen man eine hohe Autoren-Abhängigkeit erzielen möchte, entfällt die Dokumentation. "

Bemerkung 1

In der Original-Version dieses Verfahrens ist auch auf die Bedeutung der Assembler-Sprache hingewiesen worden. In einer späteren Veröffentlichung weist der Autor auf Verwendbarkeit von FROTRAN 77 in seinem Verfahren hin.

Bemerkung 2

Bemerkenswert ist, daß bei diesem Verfahren kein Test benötigt wird, denn es ist immer möglich, jeden Benutzer auf seine eigenen Fehler aufmerksam zu machen.

VI) K-S-Prinzip (Klau-Schutz-Prinzip)

" Bei Programmen, von denen man weiß, daß sie über kurz oder lang geklaut werden, entfallen Test U N D Dokumentation. "

VII) Prinzip Faulheit des Universums

" Wozu Test und Dokumentation "

(Aus: H.Weber/J.Grami; " Numerische Verfahren für PRT 1 ")

User-Alltag:

" Man unterhält sich nicht mit Mikroprozessoren.  
Man sagt ihnen, was sie tun sollen und dann betrachtet man hilflos die Katastrophe, wenn sie buchstabengetreu gehorcht haben. "

Zu guter Letzt noch ein Aufruf an alle **Farbmonitorbesitzer**. Da viele Club-interne Programme für SW-Monitore geschrieben sind, kann man diese erst verwenden wenn man alle Bildschirmsteuercodes geändert hat. Aber es ist doch wirklich unnötig, daß sich jeder die Mühe macht. Vielleicht kann man eine 'Farbmonitor-Ecke' einrichten in der jeder angibt, was er schon umgeschrieben hat, was ihn interessiert, bei was er nicht weiterkommt etc.

Ich habe z.B. das Einzellaufwerks-Fastcopy von HH jetzt in Farbe und arbeite

Hardware: Digitalizer**An alle Digitiser-Geschädigten!** (Cristof Niehörster, 5060)

Als ich mich vor ein paar Tagen mal näher mit dem Bau des Digitisers von Otmar Rücker beschäftigen wollte, fiel mir ein verbreitetes Problem auf. Mehrere Club-Mitglieder versuchten den Druckkopf nach rechts zu bewegen, ohne einen Punkt zu setzen.

Persönlich habe ich zwar nichts gegen den Vorschlag von Frank Büschler, ein Farbband dafür zu köpfen (Info 12, S. 13), oder sogar ein eigenes Bauteil zum Einlesen zu kreieren (Klaus Muerling: Info 11, S. 15), es geht jedoch auch softwaremäßig:

Bei dieser Lösung, die eigentlich auch fast jede Hardcopy-Routine enthalten müßte, kann man den Druckkopf fast beliebig nach rechts fahren lassen, ohne einen Punkt zu setzen.

Man stellt den Drucker auf Grafik-Modus ein und schickt entsprechend der definierten Zeilenlänge eine Menge von chr\$(0) oder #0 oder \$0 zum Drucker.

Hierzu zwei einfache Programmbeispiele in Turbo und Basic :

Turbo-Pascal :

```

1: Procedure Leerzeile(xk);(* xk steht f}r die Zeilenl}nge *)
2: Var x:integer;
3: Begin
4:   Write(1st,#27,#65,#1);(* Zeilenabstand *)
5:   Write(1st,#10,#27,#75,Chr(Lo(xk)),chr(Hi(xk))); (Grafikmodus)
6:   For x:=1 to xk do
7:     write(1st,Chr(0))
8: End;
```

Basic :

```

0  DIM A$(480):FOR I=1 TO 480:A#=A#+CHR$(0):NEXT
10  XK=...
90  REM SUBROUTINE LEERZEILE
100 LPRINT CHR$(27);"A";CHR$(1);
110 LPRINT CHR$(27);"K";CHR$(MOD(XK,256));CHR$(INT(XK/256));
120 LPRINT A$(1,XK)
```

Erläuterungen :

Das Basic und das Pascal-Programm unterscheiden sich durch die Tatsache, das im Basic Programm die chr(0) nicht einzeln, sondern in einem Schub übertragen werden. In Basic empfiehlt es sich aus Gründen der Geschwindigkeit nicht, die Pascal-Methode anzuwenden.

In Zeile 4 bzw. 100 wird jeweils der Zeilenabstand auf eine Nadelhöhe festgelegt und in Zeile 5 bzw. 110 wird der Grafikmodus mit Zeilenlänge xk eingestellt. Ist der Drucker auf Elite eingestellt, so ist die maximale Zeilenlänge 576 Punkte und das Basic-Programm müßte im A\$ 576 mal chr\$(0) enthalten.

Zumindest bei meinem DMX 80 läuft so der Druckkopf nach rechts, ohne einen Punkt zu setzen. Allerdings muß darauf hingewiesen werden, daß das ganze natürlich nichts bringt, wenn noch etwas im Druckerpuffer stehen sollte.

Hardware: MTX schneller / Maus**MTX-Tuning** (Rudolf Gmeinwieser, 8443)

In den letzten Infos wurde ja öfters auf die Geschwindigkeitssteigerung durch Quarz-Austausch hingewiesen. Ich wollte natürlich meinem MTX auch zu einem größeren Tempo verhelfen und habe mir einen 5 MHz Quarz besorgt.

Ich lötete vorsichtshalber zuerst einen passenden Pfostenfeldstecker (3-polig, den mittleren Pin abtrennen) in die Platine, um den Quarzwechsel einfacher zu machen und steckte dann den 5 MHz Quarz ein. Leider füllte sich beim Einschalten der ganze Bildschirm nur mit verschiedenen Zeichen, aber der MTX bootete nicht.

Zufällig fand sich aber in meiner Bauteile-Sammlung ein Quarz mit dem Wert 4.4336 MHz. Nach dem Motto, Einstecken kostet nichts propierte ich nun dieses Bauteil aus. Also wieder FDX eingeschaltet und siehe da, der MTX bootete. Dieser Quarz bewährt sich nun seit zwei Wochen im MTX ohne daß es zu System-Abstürzen kam und brachte eine Geschwindigkeitssteigerung von fast 10 Prozent.

Ich empfehle aber jedem, gleich beide Quarze plus Umschalter einzubauen, da, wie Herbert schon bemerkte, manche Spiele zu schnell laufen oder eben das Modem nicht mehr mitmacht. Aber Vorsicht mit dem Verbindungskabel. Ich habe die beiden Quarze direkt an die Schalter-Anschlüsse gelötet und von hier ein 20 cm langes Kabel zu den Platinen-Anschlüssen gelegt. Ich verwende für beide Zuführungen je ein abgeschirmtes Kabel, bei dem aber die Abschirmung und der Mittelleiter eine Verbindung darstellen (Widerstand?).

**Wer hat eine bessere Idee?**

Übrigens lief das FDX-System ohne Speichererweiterung (mit MTX 512) auch mit einem 5 MHz Quarz.

Ich habe auch einmal den Strom im 5V Zweig der MTX-Platine gemessen (Sicherung FS1):

4 MHz:	1.57 A
4.4336 MHz:	1.60 A

**Maus-Dressur für MTX** (Rudolf Gmeinwieser, 8443)

Diese Tiere gefielen mir bei anderen (und teureren) Computer schon immer, nur waren mir die zahmen Nager zu teuer. So suchte ich schon seit längerem eine billigere Lösung für meinen MTX. Dank Atari kam ich nun in den Genuß einer solchen Maus. Dieses kleine Kästchen wird unabhängig von einem 250- oder 520 ST angeboten und kostet zur Zeit weniger als 150.-- DM.

Als Anschluß besitzt es einen neunpoligen Joystick-Stecker, der aber nicht für den Anschluß an den MTX zu gebrauchen ist. Die Bewegung einer Gummikugel wird durch je zwei Lichtschranken in elektrische Signale umgewandelt, die so vom Port 7 des MTX verarbeitet werden können. Weiterhin stehen auch noch zwei Tasten zur Abfrage zur Verfügung.

Hardware: Maus

Beschaltung für die Maus an Port 7:  
 (siehe dazu MTX-Handbuch Seite 240)

Port 7		Joystick-Buchse
0V	(16)	8
FIN 0	(6)	3
FIN 1	(15)	4
FIN 2	(7)	1
FIN 3	(14)	2
FIN 4	(8)	6
FIN 5	(13)	9
+5V	(11)	7

Diagramm der Maus-Werte an Port 7 für Cursor-Steuerung:

links		rechts		hoch		tief	
!	!	!	!	!	!	!	!
PIN2	PIN3	PIN2	PIN3	PIN0	PIN1	PIN0	PIN1
1	1	1	1	1	1	1	1
0	1	1	0	0	1	1	0
0	0	0	0	0	0	0	0
1	0	0	1	1	0	0	1

Da ich bis jetzt nur mit einem Programm in TURBO-PASCAL arbeite und die Abfrage in Basic zu langsam ist, hier noch Hinweise für die Assembler-Programmierung:

Um überhaupt die Richtungen unterscheiden zu können (links/rechts benutzt PIN2 u. PIN3; hoch/tief - PIN0 u. PIN1) wertet man im ersten Schritt nur die horizontale oder vertikale Richtung aus. Im nächsten Schritt wird der neue Wert gelesen und mit obiger Tabelle verglichen (siehe dazu auch das Pascal-Listing).

Beispiel:

1. Port 7 lesen --> PIN2 - LOW-Pegel >  
 > -> links/rechts  
 PIN3 - HIGH-Pegel >
2. Port 7 lesen --> PIN2 - LOW-Pegel >  
 > -> Schritt links  
 PIN3 - LOW-Pegel >

Mit diesem Programm in Pascal und in Verbindung mit Sprites (siehe weiter hinten) und dem Grafik-Paket in TURBO-PASCAL auf der CLUB.003-Diskette habe ich mir ein kleines Zeichen-Programm geschneidert, mit dem ich Freihandzeichnen kann, und das so schnell ist, das ich mit der Fläche von 30 x 20 cm für die Mausbewegung jeden Punkt des Bildschirms erreichen kann.

**Anm.d.HH** Die Programme sind auch auf CLUB.006

T U R B O: True & False / Sprites**TURBO, TURBO (kopfschüttelnd)**

(Michael Möwe, 2000)

TURBO-Pascal bietet die Möglichkeit Parameter an selbstgeschriebene Prozeduren oder Funktionen in Maschinensprache zu übergeben. Soweit so gut. Nun wollte ich einen bool'schen Wert an meine Routine übergeben und musste feststellen, daß das nicht so funktionierte, wie ich es wollte, denn um den bool'schen Wert weiterverarbeiten zu können, muß ich sein 'Byte-Äquivalent' kennen. Dazu einen Textauszug aus dem Handbuch zu Turbo:

**3.4 Boolean (Bool'sche Wahrheitswerte)**

Ein Bool'scher Wahrheitswert kann einen der beiden logischen Werte wahr oder falsch, die durch die Standardbezeichner True oder False, bezeichnet sind, annehmen. Diese sind so definiert, daß True < False ist (haha)...

So, True ist also kleiner als False, aber welchen zahlenmäßigen Wert hat True (-1 < 0 oder 0 < 255 oder ...) ?

Also schnell ein kleines Programm:

```
Begin
  Writeln(' True=',ord(True));
  Writeln(' False=',ord(False));
End.
```

Ergebnis: **True= 1**

**False= 0**

Da haben wir's also, True < False (Hihi). Die Erklärung findet sich an anderer Stelle:

**8.1 Skalare Datentypen**

... Standardtyp Boolean ... wird folgendermaßen definiert:

**Type Boolean=(False,True);** (sic est)

Das ist dann aber endlich richtig.

**Also: True>False, True=1, False=0 !!!!!!!**

**SPRITES IN TURBO-PASCAL** (Rudolf Gmeinwieser, 8443)

Da in vorhergehenden Infos schon sehr ausgiebig über den Video-Prozessor im MTX geschrieben wurde, hier nur eine Kurzbeschreibung dieses Programms (siehe auch MTX-Handbuch ab Seite 210).

Um ein Sprite erzeugen zu können muß man sich zuerst einmal ein entsprechendes Muster überlegen, das in einen Video-Ram-Bereich gepoket wird. Die Adresse dieses Bereichs wird durch das VDP-Register 6 bestimmt:  $7 * 8 * 256 = 14336$  (s. SprM\_Adr). Diese Procedure SpriteMuster ist mit GENPAT in Basic identisch.

Da es zwei Größen gibt (8\*8 und 16\*16) muß auch dieser Wert definiert werden. Dies geschieht im VDP-Register 1, wo Bit 7 diesen Wert darstellt (siehe Procedure SpriteGroesse).

Um dieses Sprite nun auch auf dem Bildschirm darstellen zu können braucht man einen weiteren Video-Ram-Bereich, in dem die Sprite-Attribute eingeschrieben werden. Das VDP-Register 5 weist auf diese Adresse hin:  $127 * 8 * 16 = 16256$  (s. SprA\_Adr). Die Procedure Sprite ist mit dem Basic-Befehl identisch.

Die Procedure Sprite muß in einem Programm immer mit der ersten Sprite-Nr. beginnen (SprNr:=0), und jedes weitere Sprite muß die nachfolgende Nr. haben, also 2. Sprite: SprNr:=1, ... Wird die Reihenfolge nicht eingehalten, werden einige Sprites nicht angezeigt. BASIC hat deshalb einen Befehl, um die Anzahl von Sprites zu setzen.

Diese Prozeduren sind auch wieder auf das Grafikpaket von Olaf Krumnow unter TURBO-PASCAL auf der Diskette CLUB.003/006 abgestimmt.

T U R B O: SPRITE-Routinen**Sprite-Routinen** (Rudolf Gmeinwieser, 8443)

```
Const      SprM_Adr  = 14336;
           SprA_Adr  = 16256;
Var        Sprx, Spry: Byte;
```

```
(* Diese Prozedur ist auf Grafik-Routinen V54.1/2 von Olaf Krumnow
   abesitmt. -> CLUB.003/006 *)
```

```
Procedure VPoke (Adresse: Integer; Wert: Byte);
```

```
Begin
```

```
  Inline
```

```
    ( $2a/ Adresse/      (* Id (hl),Adresse      *)
      $cd/ WriteVDP+8/   (* call WriteVDP      *)
      $3a/ Wert/         (* Id a,(Wert)        *)
      $d3/ 1);          (* out (1),a          *)
```

```
End;
```

```
Procedure SpriteMuster (SprM_Nr, Sm1, Sm2, Sm3, Sm4,
                       Sm5, Sm6, Sm7, Sm8: Byte);
```

```
Begin
```

```
  VPoke (SprM_Adr+(SprM_Nr*8) + 0, Sm1);
  VPoke (SprM_Adr+(SprM_Nr*8) + 1, Sm2);
  VPoke (SprM_Adr+(SprM_Nr*8) + 2, Sm3);
  VPoke (SprM_Adr+(SprM_Nr*8) + 3, Sm4);
  VPoke (SprM_Adr+(SprM_Nr*8) + 4, Sm5);
  VPoke (SprM_Adr+(SprM_Nr*8) + 5, Sm6);
  VPoke (SprM_Adr+(SprM_Nr*8) + 6, Sm7);
  VPoke (SprM_Adr+(SprM_Nr*8) + 7, Sm8);
```

```
End (* SpriteMuster *);
```

```
Procedure SpriteGroesse (SprGr: Byte);
```

```
Begin
```

```
  If SprGr In [0,1] Then
```

```
    Case SprGr of
```

```
      0: Inline ( 62/ 192/ 211/ 2/ 62/ $81/ 211/ 2);
```

```
      1: Inline ( 62/ 193/ 211/ 2/ 62/ $81/ 211/ 2);
```

```
    End
```

```
  Else Begin
```

```
    Write (#7, 'Bitte richtige GröÙe angeben (0 oder 1): ');
```

```
    Readln (SprGr);
```

```
    SpriteGroesse (SprGr);
```

```
  End;
```

```
End;
```

```
Procedure Sprite (SprNr, Spry, Sprx, SprM_Nr, SprF: Byte);
```

```
Begin
```

```
  VPoke (SprA_Adr+(SprNr Shl 2) + 0, 191-Spry);
```

```
  VPoke (SprA_Adr+(SprNr Shl 2) + 1, Sprx);
```

```
  VPoke (SprA_Adr+(SprNr Shl 2) + 2, SprM_Nr);
```

```
  VPoke (SprA_Adr+(SprNr Shl 2) + 3, SprF);
```

```
End (* Sprite *);
```

B A S I C: BASIC.COM / Seikosha / RND & INT**BASIC.COM von Adreas Viebke** (Herbert Herberg, 2000)

Es kursieren verschiedene Versionen des Programmes BASIC.COM, bei dem es sich um eine verbesserte Version von FDXB.COM handelt, die auch einige neue Befehle enthält (DISC NEW, DISC LIST, ...). Damit hierbei endlich Klarheit besteht hat Andreas den Versionen von BASIC eine Anzeige der Versionsnummer eingebaut. Das BASIC V.1 ist auf den BASIC-Public-Domain-Disketten drauf.

**Seikosha-Routine** (Claudio Romanazzi, 3070)

Für Rolf Schmidt eine kleine Seikosha-Routine:

```

10 for y=27 to 0 step -1
20 lprint chr$(8);
30 for x=0 to 255
40 let a=asc(gr$(x,7*y+2,7)
50 poke irgenteine Speicherstelle,a
60 gosub 500
70 a=peek (irgenteine Speicherstelle)
80 lprint chr$(27);chr$(16);chr$(0);chr$(x);chr$(a);
90 next x
100 lprint110 next y
120 stop
500 code

```

```

ld b,7
ld a,(irgenteine Speicherstelle)
ld l,#80
a: bit 6,a
jr z,b
set 0,l
b: rla
rrc l
djnz a
rrc l
ld a,l
ld (irgenteine Speicherstelle),a
ret

```

Diese kleine Routine funktioniert auf Seikosha GP-100A.

Ich habe auch noch eine ganz in MC geschriebene Routine, die aber voll in ein anderes Prog eingebunden ist. Zum nächsten Info schicke ich sie dir bereinigt zu.

**RND / INT** (Karl-Heinz Geschwandter, 3126)

RND = Randomize, erzeugt eine Zufallszahl zwischen 0 und 0.9999999, ist ursprünglich eine Testhilfe gewesen, und wurde ersetzt durch:

RND(0), was nun eine Zufallszahl zwischen 0 und 0.9999999 liefert.

LET A = RND(0)\*101 ist eine Zufallszahl zwischen 0 und 100.9999999.

INT( ) schneidet die Nachkommastellen einer Zahl ab:

INT(12.84) = 12, INT(12.00001) = 12, INT(12.99999) = 12,

INT(-12.2) = -12.

Also:

A = INT(RND(0)\*21) liefert eine ganzzahlige Zufallszahl 0 - 20.

B A S I C: SELECT/MERGE

SELECT/MERGE für XBASIC V. 1.3

(Christian Meervissen, 5600)

SELECT saved vom User definierte Programmsegmente ab.

Aufruf: USER CALL, "SELECT"

Es gibt drei Variationsmöglichkeiten:

1) Der einzugebende Name ist mit einem Label identisch. Dann ist die erste abzusavende Zeile = Label-Zeile. SELECT sucht dann nach dem Programmende, welches durch das Label REM :(Name) END: markiert wird. Wichtig ist hierbei das Leerzeichen zwischen Name und END im REM-Statement.

2) Das Label REM :(Name) END: existiert nicht.  
Es wird nun nach der letzten Zeile des Programms gefragt.

3) Der eingegebene Name ist mit keinem Label identisch.  
Es wird nun nach der ersten und nach der letzten Zeilen des Programm-segments gefragt.

MERGE lädt mit SELECT unter (Name) gesavte Programmsegmente.

Aufruf: USER CALL, "MERGE"

Programmname und -länge werden jeweils angezeigt.  
Ist ein Programm länger als an verfügbarem Speicherplatz vorhanden, erscheint in VS0 "NO SPACE" und das Programm bricht ab.

ADAPT saved das Programmsegment SELECT/MERGE ab und gibt die Länge in Byte an. Dies dient der Anpassung von Programmen, die mit SAVE auf Band gespeichert wurden.

Aufruf: USER CALL, "ADAPT"

Nach dem Saven unbedingt Länge notieren !  
Jetzt die Anpassung:

```
NEW: LOAD "Programm"
LET I=USR(58880) ;reinitialisieren falls nötig
USER CREATE, Zeile, Länge (SELECT/MERGE) -7 ; Ass.-Zeile
USER BASIC, Zeile, VARIABLE
USER TIN, VAR ; SELECT/MERGE laden
```

evtl. Renumber und das Programm steht wieder im Speicher.  
Einer Unterprogramm-bibliothek auf Cassette steht nun nichts mehr im Wege.

Mit SELECT abgespeicherte und MERGE geladene Assembler-Programme müssen jedoch unbedingt an die neuen Laufadressen angepaßt werden, da der Befehl RELOC fehlt. Bei Verwendung der Label-Adressierung genügt das Durchgehen im Edit-Modus.

B A S I C: SELECT/MERGE

```
100 REM :SELECT/MERGE:
110 REM :SELECT:
120 USER CALL,"INIT"
130 CSR 2,18: INPUT "NAME : ";NA$
140 USER LABEL,"NXT",ERR: USER ERROR,1,ERR
150 LET DUM=LEN(NA$)+1
160 REM :NXT:
170 IF DUM=0 AND ZL1=0 THEN USER ERROR,0: USER JUMP,"NO LABEL"
180 LET DUM=DUM-1
190 USER LABEL,NA$(1,DUM),ZL1
200 USER LABEL,"NO END",ERR: USER ERROR,1,ERR
210 USER LABEL,NA$(1,DUM)+" END",ZL2
220 REM :CONT:
230 USER CALL,"LÄNGE"
240 IF SGN(LA)<=0 THEN VS 0: CLS : PRINT "FALSCHER ZEILENANGABE": PAUSE 2000: RETURN
250 USER CALL,"POSK"
260 USER DOKE,RAC+7,LA
270 USER FIND,NA$,ADR
280 USER MOVE,ADR,20,RAC+9
290 USER CALL,"ON"
300 USER TOUT,RAC+7,22
310 PAUSE 1000: USER TOUT,RA1,LA: CLS
320 RETURN
330 REM :NO LABEL:
340 CSR 2,18: INPUT "VON ZEILE NR. ";ZL1
350 REM :NO END:
360 USER ERROR,0
370 CSR 2,18: INPUT "BIS ZEILE NR. ";ZL2
380 USER JUMP,"CONT"
390 REM :SELECT END:
400 REM :MERGE:
410 USER CALL,"INIT"
420 CSR 2,18: INPUT "NAME : ";SNA$
430 USER CALL,"POSK"
440 REM :SUCH:
450 USER TIN,RAC+7,22
460 USER DEEK,RAC+7,LA
470 USER FIND,NA$,ADR
480 USER MOVE,RAC+9,20,ADR
490 VS 1: CSR 2,18: PRINT "FOUND : ";NA$(1,20),LA;" BYTES"
500 USER KILL$," ",SNA$,20: USER KILL$," ",NA$(1,20),20
510 USER POS,SNA$,NA$(1,20),DUM
520 IF (DUM=0) AND (SNA$<>"") THEN USER TIN,0: USER JUMP,"SUCH"
530 USER FREE,DUM
540 IF LA>DUM THEN VS 0: CLS : PRINT "NO SPACE": PAUSE 2000: VS 5: RETURN
550 CSR 2,18: PRINT "LOADING"
560 USER CALL,"POSK"
570 USER DEEK,RAC+7,LA
580 USER CREATE,65535,LA-7
590 LET DUM=0: USER BASIC,65535,DUM
600 USER TIN,DUM
610 RETURN
620 REM :MERGE END:
630 REM :POSK:
640 LET ZKO=0: LET RAC=0
650 USER LABEL,"KOPF",ZKO
660 USER BASIC,ZKO,10,RAC
670 RETURN
```

B A S I C: SELECT/MERGE

```
680 REM :KOPF: USER CREATE,690,22
690 CODE
```

```
8523 LD (DE),A
8524 RLCA
8525 LD D,E
8526 LD B,L
8527 LD C,H
8528 LD B,L
8529 LD B,E
852A LD D,H
852B CPL
852C LD C,L
852D LD B,L
852E LD D,D
852F LD B,A
8530 LD B,L
8531 NOP
8532 NOP
8533 NOP
8534 NOP
8535 NOP
8536 NOP
8537 NOP
8538 NOP
```

Symbols:

```
700 REM :SAV:
710 CLEAR : USER CALL,"ON": SAVE "SELECT/MERGE": STOP
720 REM :ADAPT:
730 USER CALL,"INIT"
740 USER LABEL,"SELECT/MERGE",ZL1
750 USER LABEL,"SELECT/MERGE END",ZL2: LET ZL2=ZL2+1
760 USER CALL,"LÄNGE"
770 CSR 2,18: PRINT LA;" BYTES"
780 USER CALL,"ON"
790 USER TOUT,RA1,LA
800 RETURN
810 REM :ADAPT END:
820 REM :LÄNGE:
830 USER BASIC,ZL1,RA1
840 LET ZL2=ZL2+1
850 USER BASIC,ZL2,RA2
860 LET LA=RA2-RA1-1
870 RETURN
880 REM :ON:
890 CSR 2,18: PRINT "TAPE ON ? (J/N)"
900 USER GET,"Jj"
910 CSR 2,18: PRINT "VORSPANN LÄUFT !": PAUSE 5000
920 RETURN
930 REM :INIT:
940 USER ERROR,0
950 VS 1: CLS : LET NA$="": LET DUM=0: LET ERR=0
960 LET ADR=0: LET RA1=0: LET RA2=0: LET ZL1=0: LET ZL2=0: LET LA=0
970 RETURN
980 REM :SELECT/MERGE END:
```

B A S I C: COINC (Sprite) / HCHAR*(Christian Heevissen, 5600)*

COINC vergleicht das Zusammentreffen zweier vorgegebener Sprites unter Beachtung des durch den User definierten Toleranzparameters. Bei erfüllter Bedingung wird ein Flag gesetzt. Es muß gepoket werden:

beide Sprite-Nr.	Speicherstelle	SN1,SN2
Toleranz (0-255)	Speicherstelle	TOL

Das Flag wird mit PEEK (Adr. EXP) ermittelt:

Kollision -> EXP=1, ansonsten ist EXP=0 .

Der Toleranzparameter ist identisch mit der halben Seitenlänge des quadr. Toleranzfeldes, das um den Sprite-Mittelpunkt gebildet wird. Innerhalb dieses Feldes wird jeder Kontakt mit dem zweiten vorgegebenen Sprite als Kollision interpretiert und das Flag EXP gesetzt.

Bsp.: TOL=10 entspricht einer Seitenlänge des Toleranzfeldes von 20 Pixeln (10 Pixel vor, hinter, über und unter dem Sprite-Mittelpunkt). Das Toleranzfeld kann auch größer als die Sprite-Fläche selbst sein.

HCHAR setzt ASCII-Zeichen (32-154) in die aktuelle Cursor-Position und wiederholt es auf Wunsch horizontal. Die Anzahl der Wiederholungen ist beliebig. Bei Text mit W=960 und Grafik mit W=768 wird der Bildschirm voll belegt. Sehr rascher Aufbau. Werte für ASCII-Code und Wiederholung müssen gepoket werden:

ASCII-Code	Speicherstelle	CHAR
Wiederholung	Speicherstelle	NMAL (2 Byte)

Aufruf: 1) MTX-BASIC

```
POKE Adr. NMAL+1,INT(W/256)
POKE Adr.NMAL,W-256*PEEK (Adr.NMAL+1)
GOSUB Zeilennummer
```

2) XBASIC V. 1.3 (für V. 1.1 Syntax-Unterschiede beachten)

```
USER DOKE, Adr. NMAL, W
USER CALL, "HCHAR"
```

B A S I C: COINIC (Sprite) / HCHAR

800 REM :COINC:  
810 CODE

```

8921 COINC: LD HL,(EXP)
8924 LD IX,S1Y
8928 XOR A
8929 CALL MP
892C LD HL,(SN1)
892F LD IX,S2Y
8933 CALL MP
8936 LD C,A
8937 LD HL,(TOL)
893A LD H,A
893B ADD HL,HL
893C ADD HL,HL
893D ADD HL,HL
893E ADD HL,HL
893F NOP
8940 JP NZ,#0689
8943 RST 38
8944 GO: BIT 3,C
8946 JR Z,YRT
8948 XRT: RES 3,C
894A LD IX,(S1X)
894E LD IY,(S2X)
8952 CALL PDIF
8955 JR C,NMAL
8957 BIT 2,C
8959 JR NZ,BUM
895B SET 1,C
895D NMAL: DJNZ GO
895F JR END
8961 YRT: SET 3,C
8963 LD IX,(S1Y)
8967 LD IY,(S2Y)
896B CALL PDIF
896E JR C,NMAL
8970 BIT 1,C
8972 JR NZ,BUM
8974 SET 2,C
8976 JR NMAL
8978 MP: LD L,A
8979 LD DE,8
897C LD B,E
897D MUL: ADD HL,HL
897E JR NC,NOA
8980 ADD HL,DE
8981 NOA: DJNZ MUL
8983 LD DE,65102
8986 ADD HL,DE
8987 LD (IX+0),L
898A LD (IX+1),H
898D LD DE,3
8990 ADD HL,DE
8991 LD (IX+2),L
8994 LD (IX+3),H
8997 RET
    
```

```

8998 PDIF: AND A
8999 LD H,(IX+0)
899C LD L,(IX+1)
899F LD D,(IY+0)
89A2 LD E,(IY+1)
89A5 PUSH HL
89A6 PUSH DE
89A7 SBC HL,DE
89A9 JR C,NEGA
89AB POP DE
89AC POP DE
89AD POSI: AND A
89AE LD DE,(TOL)
89B2 EX DE,HL
89B3 SBC HL,DE
89B5 RET
89B6 NEGA: POP HL
89B7 POP DE
89B8 SBC HL,DE
89BA JR POSI
89BC BUM: LD A,1
89BE END: LD (EXP),A
89C1 RET
89C2 TOL: DS 2
89C4 EXP: DS 1
89C5 SN1: DS 1
89C6 SN2: DS 1
89C7 S1Y: DS 2
89C9 S1X: DS 2
89CB S2Y: DS 2
89CD S2X: DS 2
89CF RET
    
```

Symbols:  
EXP89C4TOL89C2  
BUM89BCNEGA89B6  
POSI89ADCOINC8921  
S1Y89C7MP8978  
SN189C5S2Y89CB  
GO8944YRT8961  
XRT8948S1X89C9  
S2X89CDPDIF8998  
NMAL895DEND89BE  
MUL897DNOA8981  
SN289C6

820 RETURN  
830 REM :COINC END:

B A S I C: CONINC (Sprite) / HCHAR

```

90 REM :HCHAR:
100 CODE

8015 HCHAR: DI
8016      AND A
8017      LD A,(CHAR)
801A      CP 31
801C      JP C,ERR
801F      CP 155
8021      JP NC,ERR
8024      LD C,A
8025      LD DE,(NMAL)
8029 TEST: AND A
802A      EX DE,HL
802B      LD DE,1
802E      SBC HL,DE
8030      JP C,LEER
8033      LD DE,30
8036      SBC HL,DE
8038      JP C,KLEIN
803B      LD B,31
803D      EX DE,HL
803E      CALL AUSG
8041      JP TEST
8044 KLEIN: ADC HL,DE
8046      LD B,L
8047      CALL AUSG
804A LEER:  LD A,0
804C      LD (CHAR),A
804F      LD (NMAL),A
8052      LD (ZE),A
8055      EI
8056      RET
8057 ERR:  CALL LEER
805A      RST 28
805B      DB 18
805C      RET
805D AUSG: LD HL,OUT
8060      AND A
8061      CP 1
8063      JP NC,AUS1
8066 LOOP: INC HL
8067      LD (HL),C
8068      DJNZ LOOP
806A      INC HL
806B      LD (HL),#C9
806D      RST 10
806E OUT:  DS 33
808F      RET
8090 AUS1: LD A,128
8092      ADD A,B
8093      LD (HL),A
8094      JP LOOP
8097 CHAR: DB 0
8098 NMAL: DB 0
8099 ZE:   DB 0
809A      RET

```

Dann suche ich immer noch, trotz mannigfachen Nachschlagens nicht gefunden: Zeiger auf aktuellen Screen.

B A S I C: Gelenkte Ausgaben

## GELENKTE AUSGABEN

Was es mit den Systemvariablen  
PRORPL, IOPR und IOPL auf sich hat

Andreas Viebke

Es soll ja Leute geben, die sich wünschen, man könnte mit einem einzigen BASIC-Befehl alle Ausgaben eines Programms zum Drucker schicken. Haargenau die gleiche Liste z.B., die man eben noch am Bildschirm bewundert hat, möchte man jetzt auf Papier bringen. Nein, bitte nicht jeden Ausgabe-Befehl zweimal ins Programm setzen, einmal PRINT und dann noch LPRINT. Das ist umständlich und kostet Platz. Stattdessen kann man für Bildschirm- und Druckerausgabe ein und dieselbe Routine verwenden.

Jetzt fehlt nur noch der Schlüssel zum Umschalten, und da BASIC-Benutzer oft in den Innereien (Systemvariablen) ihres Interpreters herumstochern, und dessen nie müde zu werden scheinen, können sie das hier auch tun. Und nun wird's konkret.

FDXB bzw. BASIC kennt drei Systemvariablen, die etwas mit der Ausgabe zu tun haben: PRORPL (#FD75), IOPR (#FD76) und IOPL (#FA8F). Im BASIC-Handbuch steht dazu soetwas Banales wie "Wenn PRORPL=1 ist, erfolgt eine Ausgabe zu der in IOPL spezifizierten Peripherie. Wenn PRORPL=0 ist, erfolgt eine Ausgabe zu der in IOPR spezifizierten Peripherie". Das hätte man auch eleganter formulieren und etwas näher beleuchten können - es ist nämlich der besagte Schlüssel.

PRINT und LPRINT benutzen dieselben Programmteile im FDXB. Führt man PRINT aus, dann notiert sich FDXB diese Tatsache, indem es eine Null in seine Systemvariable PRORPL schiebt, bei LPRINT dagegen notiert es dies mit einer Eins. Der Rest ist der PRINT-Routine egal, denn sie greift, wie gesagt, auf nur eine Routine zur Ausgabe zurück. Und diese wiederum orientiert sich an den vorher notierten Werten.

PRORPL       =0: Ausgaben gehen zur in IOPR spezifizierten Peripherie.  
              =1: Ausgaben gehen zur in IOPL spezifizierten Peripherie.

FDXB ist in diesem Punkt ganz gut strukturiert, wie mancher Benutzer des RST 10 vielleicht weiß. Denn die Ausgabe-Routine kennt für gewöhnlich drei Geräte, an die es Zeichen senden kann: Den Bildschirm, das an der Centronics-Schnittstelle, und das an der RS232 hängende Gerät. Und woher weiß sie, welches sie nehmen soll? Sie schaut natürlich in einer weiteren Systemvariablen nach, und da findet sie Werte, die sie folgendermaßen interpretiert:

0=Bildschirm, 1=Drucker, 2=RS232

Ein Beispiel: Gibt der Benutzer PRINT PI ein, dann wird dies mit einer Null in PRORPL notiert. FDXB wandelt jetzt PI in eine Folge druckbarer Zeichen um und sendet diese an die universelle Ausgabe-Routine. Die weiß anhand der notierten Null, daß sie in IOPR nachschauen muß, um zu erfahren, an welches Gerät sie das Zeichen senden soll - und tut's. Hätte der Benutzer LPRINT PI eingegeben, dann wäre eine Eins notiert worden und die Ausgabe-Routine hätte das Gerät über den Inhalt von IOPL erfahren.

B A S I C: Gelenkte Ausgaben / Noddy-Fehler

Ganz klar, daß man sowas ausnutzen muß. FDXB verändert nie die Werte in IOPR und IOPL, nur den in PRORPL (im Gegensatz zu BASIC: Es schreibt nach DISC LIST eine Eins nach IOPL, doch wer verwendet das schon in einem Programm). Jetzt kann man lenken:

```
10 LET IOPL=64143
20 POKE IOPL,0: GOSUB 50
30 POKE IOPL,1: GOSUB 50
40 STOP
50 LPRINT PI : RETURN
```

Die Ausgabe wird durch Zeile 20 auf den Schirm und durch Zeile 30 auf den Drucker gelenkt, obwohl doch nur ein LPRINT-Befehl in der Unter-Routine in Zeile 50 verwendet wird. Man kann natürlich auch IOPR verändern, doch muß IOPR immer wieder auf Null zurückgesetzt werden, bevor FDXB in den Direktmodus geht, sonst versucht es, auf dem Drucker seine Edit-Screens zu generieren, und das ergibt äußerst unschöne Bilder und Geräusche... Also besser IOPL verändern und LPRINT als Ausgabebefehl nehmen (es geht auch mit LLIST - selbstredend).

Auch kann man als Kenner der Maschinensprache und der Funktionsweise des RST 10 ein weiteres Gerät ansteuern, wenn man BASIC statt FDXB verwendet (von mir verändertes FDXB, auf allen BASIC-PD-Disks zu finden). Gerät Nummer drei ist dort die Diskette. Doch das setzt etwas Vorarbeit voraus. Ein File muß geöffnet und nach der Ausgabe mittels RST 10 auch wieder geschlossen werden. Will jemand wissen, wie das geht?

-AV

Noddy-Fehler bei großen Programmen (Jan Brederke)

Zum Problem von Michael Keßler im Info 12, Seite 28: Bei ihm stürzte ein großes FDXB-Noddy-Programm mit der Fehlermeldung SE.B ab, aber nachdem es im MTX-ROM-Modus weiter vergrößert worden war, lief es wieder unter FDXB.

Ich hatte ein ähnliches Problem: Ich schrieb ein großes Basic-Programm mit einigen Noddy-Seiten dahinter. Als dieses so groß geworden war, daß die Obergrenze von Noddy die Hex-Adresse #C000 überschritt, stürzte der Rechner ab und hängte sich sogar manchmal auf, sobald ich im Noddy editierte. Also änderte ich dort nichts mehr, sondern schrieb weiter im Basic. Als nun die Obergrenze des Basic-Programms, die auch die Untergrenze der Noddy-Programme ist, ebenfalls die Grenze #C000 überschritten

hatte, war alles wieder normal, und ich konnte wieder in Noddy editieren.

Meiner Meinung nach hängt dies mit der Seitenaufteilung des Speichers zusammen, denn exakt bei #C000 beginnt die letzte 16K-Seite. Irgendetwas muß im FDXB sehr im argen liegen, denn im FDXB-Modus stürzt der Rechner auch ab, wenn der Speicher voll ist, im Gegensatz zum MTX-ROM-Modus, in dem nur eine Fehlermeldung kommt und die auch mehr als 64K verwalten kann, wenn diese vorhanden sind.

Ich hoffe, mit diesem Hinweis den Leuten geholfen zu haben, die sich im FDXB auskennen und diesen Fehler reparieren können und möchten.

Ist es dann auch möglich, daß FDXB mehr als 64K verwalten kann?

B A S I C: Selbstklebeetiketten

```

0 REM * Das Programm bedruckt Selbstklebeetiketten der Größe 40x14 mm.
1 REM * Es müssen sich immer 2 Etiketten nebeneinander befinden, die
2 REM * zusammen einen "Druckblock" bilden. Das Programm durchläuft eine
3 REM * Schleife, in der jeweils die Anzahl der zu druckenden Druckblöcke
4 REM * angegeben werden. Dies ermöglicht insbesondere am Anfang eine
5 REM * "sparsame" Feineinstellung der Etiketten im Drucker. Etiketten
6 REM * dürfen 4 Zeilen enthalten, die definiert werden können.
7 REM * Das Papier ganz links in den Drucker einlegen, am besten erst einmal
8 REM * auf normalem Papier üben.
9 REM *
10 PRINT "=====  

11 LET BKS$="      ": GOTO 6000
12 REM "SHIFT OUT" wählt compressed-Schrift
13 LPRINT CHR$(15);
14 REM Der Zeilenabstand beträgt 9/72 Inch
15 LPRINT CHR$(27);"A";CHR$(9);
19 REM Schriftart: 0 ELITE und 1_PICA
20 LPRINT CHR$(27);"P";CHR$(1);
30 LPRINT Z1$;
35 LPRINT BKS$;Z1$
40 LPRINT Z2$;
45 LPRINT BKS$;Z2$
50 LPRINT Z3$;
55 LPRINT BKS$;Z3$
60 LPRINT Z4$;
65 LPRINT BKS$;Z4$
70 REM Der Zeilenabstand beträgt 12/72 Inch
80 LPRINT CHR$(27);"A";CHR$(12);
999 LPRINT BKS$
1110 RETURN
2000 REM Steuerung der Druckblöcke
2010 CSR 46,6: PRINT "      ";CHR$(7)
2020 CSR 0,6: INPUT "Anzahl der Druckblöcke angeben: (Ende mit '0') ";ANZ
2035 IF ANZ=0 THEN GOTO 3000
2036 PRINT : PRINT "Bin bei Druck Nr.      "
2040 FOR I=1 TO ANZ
2045 CSR 18,8: PRINT I
2050 GOSUB 12
2060 NEXT I
2070 PRINT
2080 GOTO 2000
3000 REM Enderoutine
3010 REM Drucker-Initialisierung
3020 LPRINT CHR$(27);"S";
3030 STOP
4000 REM Einlesen der Druckdaten
4010 DISC OPEN #1,R$, "I"
4020 DISC INPUT #1,Z1$,Z2$,Z3$,Z4$
4500 DISC CLOSE #1
4510 GOTO 2000
5000 REM Datenfile anlegen
5002 PRINT : INPUT "Name des Datenfiles (GROSSbuchstaben!): ";R$
5003 DISC OPEN #1,R$, "O"
5004 DIM A$(4,25): CSR 3,13
5005 PRINT "Die Zeilen dürfen max. 24 Zeichen lang sein (ev. wollen Sie zentrieren?)."
5006 PRINT "  Ihnen stehen 4 Zeilen zur Verfügung."
5008 CSR 15,15: PRINT "1...5...0...5...0...4"
5009 CSR 39,16: PRINT "<": CSR 39,17: PRINT "<": CSR 39,18: PRINT "<": CSR 39,19: PRINT "<"
5010 FOR I=1 TO 4
5020 CSR 5,15+I: PRINT "Zeile ";I;": ";
5025 LET MERK=0
5030 INPUT " ";A$(I)
5031 FOR ANZ=1 TO 24
5032 IF A$(I,ANZ)=CHR$(0) THEN IF MERK=0 THEN LET MERK=ANZ
5033 NEXT ANZ
5035 FOR J=MERK TO 24: LET A$(I,J)=" ": NEXT J
5040 DISC PRINT #1,A$(I)
5050 NEXT I
5055 DISC CLOSE #1
5056 LET Z1$=A$(1): LET Z2$=A$(2): LET Z3$=A$(3): LET Z4$=A$(4)
5060 GOTO 2000
6000 REM Herkunft der Druckdaten klären
6020 CSR 0,2: INPUT "Gibt es schon ein Datenfile ? (JA/N) ";R$
6050 IF R$="J" OR R$="j" THEN PRINT : INPUT "Name des Datenfiles (GROSSbuchstaben!): ";R$:
GOTO 4000 ELSE IF R$="N" OR R$="n" THEN PRINT : PRINT "Datenfile muß angelegt werden.": GOTO 5000 ELSE GOTO 6000

```

B A S I C: Suchprogramm

Hier noch ein kleines Suchprogramm in MC:

(Thomas Günther, 6967)

```

100 REM ***** SUCHEN *****
110 REM *** (c) 1986 by T.Günther ***
120 REM *****
130 DIM V$(8),A$(200,8)
140 LET A$(140)="SuchMich"
150 LET V$="SuchMich"
160 LET ANZAHL=200
170 LET LAENGE=8           ; muß bei V$ und A$ gleich sein
180 POKE 40001,LAENGE
190 POKE 40000,ANZAHL
210 CODE

```

```

80FE      LD DE, (#FAAC) ;PROGENDE
8102      INC DE          ; +1
8103      INC DE          ; +1 = Anfang von Variablen bei 1 Dimension
8104      PUSH DE         ; Pointer für V$ in DE retten
8105      POP HL          ; Nach HL laden
8106      LD A, (40001)   ; Laenge nach A
8109      LD C,A          ; Laenge nach C
810A      ADD A,4         ; +4
810C      LD B,A
810D L1:   INC HL
810E      DJNZ L1        ; HL = Beginn von A$
8110      LD A, (40000)   ; max.Anzahl
8113      LD B,A          ; in B speichern = Schleifenzähler
8114 LOOP: PUSH DE      ;
8115      POP IX         ; DE => IX
8117      PUSH BC        ; Schleifenzähler retten
8118      CALL VERGL
811B      LD A,B          ; wenn voll
811C      CP 0           ; durchgelaufen => gefunden
811E      JP Z,ENDE     ;
8121 L03:  INC HL        ; HL auf volle
8122      DJNZ L03       ; Länge bringen
8124      POP BC         ; Schleifenzähler laden
8125      DJNZ LOOP
8127      JP END2
812A ENDE: POP BC
812B END2: LD A,B
812C      LD (40000),A   ; I wird in 40000 gespeichert
812F      RET
8130 VERGL: LD B,C       ; B=Laenge
8131 L02:  LD A, (HL)    ; A=Inhalt von A$
8132      CP (IX+0)     ; Vergleichen mit V$(IX)
8135      RET NZ        ; Zurück wenn ungleich
8136      INC HL        ; Je eins
8137      INC IX         ;iterrücken
8139      DJNZ L02      ; bis Laenge erreicht ist
813B      RET
220 LET I=ANZAHL-PEEK(40000)+1
225 IF I=ANZAHL+1 THEN PRINT "GIBT'S NICHT !!" ELSE PRINT I

```

Das Programm sucht die Position von V\$ im Feld A\$ und speichert diese nach (40000). Man muß darauf achten, daß die Variablen V\$ und A\$ als erste in einem Programm verwendet werden und auch in der obigen Reihenfolge stehen. Bei stärkerer Dimensionierung (z.B. 3-fach) muß man die Anzahl der INCs am Anfang und den Wert bei ADD A,X erhöhen.

B A S I C: Fourier-Analyse

(Michael Pfothbauer, 5804)

Folgendes Programm zur Fourier-Analyse geht langsam aber sicher, das Feld IM(n) muss zu Beginn 0 gesetzt werden. Die Funktion muss in RE (u) stehen. n muss 2 hoch x sein, also eine Potenz von 2.

```

5 DIM RE(128),IM(128),B(128)
6 VS 4: CLS : COLOUR 3,15
7 LET K=30
8 REM ERZEUGEN EINER FUNKTION f(x)=sin x + 1/2 sin 2x + 1/4 sin 4x
9 PRINT "FAST FOURIER TRANSFORMATION"
10 FOR I=1 TO 128: LET X=(I-1)*2*PI/127: LET RE(I)=SIN(X): NEXT
30 FOR I=1 TO 128: LET X=(I-1)*4*PI/127: LET RE(I)=RE(I)+SIN(X)/2: NEXT
50 FOR I=1 TO 128: LET X=(I-1)*8*PI/127: LET RE(I)=RE(I)+SIN(X)/4: NEXT
60 FOR I=1 TO 128: PLOT I,INT(K*RE(I))+80: NEXT
70 FOR I=1 TO 128: LET IM(I)=0: NEXT

```

```

80 REM ++++++FFT+++++
90 CLOCK "000000"

```

```

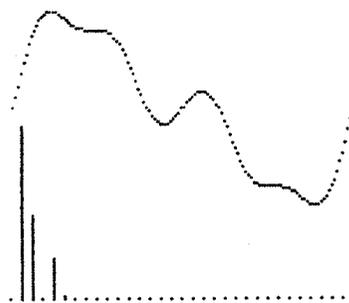
100 LET N=128
1020 LET MR=0
1030 LET NN=N-1
1040 FOR M=1 TO NN
1050 LET L=N
1060 LET L=L/2
1070 IF (MR+L)>NN THEN GOTO 1060
1080 LET MI=MR/L
1090 LET MR=L+INT((MI-INT(MI))*L+.5)*SGN(MI)
1100 IF MR>=M THEN GOTO 1170
1110 LET ZR=RE(M+1)
1120 LET RE(M+1)=RE(MR+1)
1130 LET RE(MR+1)=ZR
1140 LET ZI=IM(M+1)
1150 LET IM(M+1)=IM(MR+1)
1160 LET IM(MR+1)=ZI
1170 NEXT
1180 LET L=1
1190 IF L>=N THEN GOTO 1500
1200 LET IS=2*L
1210 LET EL=L
1220 FOR M=1 TO L
1230 LET A=PI*(1-M)/EL
1240 LET CO=COS(A)
1250 LET SI=SIN(A)
1260 FOR I=M TO N STEP IS
1270 LET J=I+L
1280 LET ZR=CO*RE(J)-SI*IM(J)
1290 LET ZI=CO*IM(J)+SI*RE(J)
1300 LET RE(J)=RE(I)-ZR
1310 LET IM(J)=IM(I)-ZI
1320 LET RE(I)=RE(I)+ZR
1330 LET IM(I)=IM(I)+ZI
1340 NEXT I
1350 NEXT M
1360 LET L=IS
1370 GOTO 1190

```

```

FAST FOURIER TRANSFORMATION DER
FUNKTION
f(x)=sin x + 1/2 sin 2x + 1/4 sin
4x
ZEITVERBRAUCH für 128 Punkte-FFT
000043 Sekunden

```



```

1500 PRINT "ZEITVERBRAUCH für 128 Punkte-FFT": PRINT TIME$;" Sekunden"
1600 REM ++++++FFT ENDE+++++
2000 FOR I=1 TO 128: LET B(I)=SQR(RE(I)^2+IM(I)^2): NEXT
2500 VS 5: FOR I=1 TO 128: PRINT I,RE(I),IM(I),B(I): NEXT : VS 4
3000 COLOUR 3,15: FOR I=1 TO 32: LET H=INT(ABS(B(I))*9#L#LINE(I,1)*4#S#(I-1)*4,H: NEXT
9000 LPRINT CHR$(27);"P";CHR$(0)

```

B A S I C: Zahlendarstellung Float

(Michael Pfothenauer, 5804)

hier die Lösung für das bei unserem FDXB verwendete Fließkommaformat.  
 Nach ein bisschen Umstellerei bei den Bytes konnte ich die Formel für  
 die Berechnung des dargestellten Wertes herausfinden:

$$\text{Wert} = (-1)^{\text{Z}} * 1.\text{Mantisse} * 2^{(\text{Exponent} - 129)}$$

Programm zum Ausdrucken und Beispiele sowie Erklärung anbei:

```

3 LET A=0: LET YH$="": LET YB$="": LET I=0: LET Y=0
4 REM "FLOTEST.BAS"
10 INPUT "Zu untersuchende Zahl:";X
15 PRINT X,
20 PROC FIND DO (X,A)
22 PRINT A
25 LPRINT X
30 FOR I=A TO A+4
40 LET Y=PEEK(I): PROC HEX$ DO (Y,YH$): LPRINT RIGHT$(YH$,2);" ";
45 NEXT
50 LPRINT " ";
60 FOR I=A TO A+4
70 LET Y=PEEK(I): PROC BIN$ DO (Y,YB$): LPRINT YB$;" ";
75 NEXT
80 LPRINT : LPRINT
100 CLS : GOTO 10
500 STOP
1000 CLEAR : LET A=0
1010 LET X=0
1020 PROC FIND DO (X,A): PRINT X,A
1030 POKE A,205: POKE A+1,204: POKE A+2,204: POKE A+3,204: POKE A+4,125
1040 PRINT X
9000 LPRINT CHR$(27);"P";CHR$(0)
    
```

				Sign		Exponent (2 <sup>n</sup> , n=Exponent-#81)
4	00 00 00 00 B3	00000000	00000000	00000000	00000000	10000011
2	00 00 00 00 B2	00000000	00000000	00000000	00000000	10000010
0	00 00 00 00 00	00000000	00000000	00000000	00000000	00000000
-2	00 00 00 80 B2	00000000	00000000	00000000	10000000	10000010
-4	00 00 00 80 B3	00000000	00000000	00000000	10000000	10000011

B A S I C: Tastaturentprellung**Basic endlich entprellt :**

Bekanntermaßen prellt bei den allermeisten MTXer die Tastatur ziemlich stark. Wer sich an eine Hardware-mäßige Lösung nicht herantraute, konnte immerhin unter CP/M das Prellen durch Tast.Com (Andreas Müller) beheben. Leider funktioniert dies nicht unter FDXB (andere Tastaturabfrage).

Da aber trotz seiner Schwächen immer noch viele in FDXB programmieren, vor allem seit es das Extended-Basic von Andreas Viebke gibt, habe ich mich mal ans Eingemachte rangemacht. Mit Erfolg:

Als erstes braucht man ein FDXBas mit repariertem Discsave. Wer nicht sicher ist : #781F bis #7846 müssen NOPs sein !

Die Tastatur wird in #56B6 (Port 5) und #56DF (Port 6) abgefragt. Dann wird kontrolliert ob eine Taste gedrückt wurde. Wenn eine gedrückt worden ist springt es zu #56BC bzw. #56E7. Jetzt muß man nur hier ein Jump zu einer Warteschleife einbauen und das Basic ist entprellt !

Diese Warteschleife legen wir in den freien Speicherbereich, den wir dank repariertem Discsave ab #781F haben. Also:

```

für Port 5  56BC  JF #781F  ; Sprung zur Warteschleife
           und  781F  PUSH AF   ; Dies stand da -
                               LD A,#02   ; wo jetzt der JF steht
                               PUSH BC   ; BC retten
                               WA: LD B,#FF ; 255 mal Warteschleife
                               PUSH IX   ; Warte-
                               POF IX   ; schlei-
                               DJNZ WA   ; fe
                               POF BC   ; BC wiederholen
                               JF #56BF ; Sprung zurück

und für Port 6  56E7  JF #7831
               und  7831  AND  A
                   ADD A,#7
               dann die selbe Warteschleife wie oben
               und      JF #56EA

```

Also Diskette mit FDXBas, DDT und mindestens noch 32k Platz nehmen und :  
( /=<RETURN> )

A>DDT FDXBAS.COM/D7922/

Es muß so etwas erscheinen:

```

7922 00 00 00 00 00 00 00 00 .. ..
7930 00 00 00 00 00 00 00 00 .. ..   etc.

```

Dann: s57BF/C3/1F/78/. /s57EA/C3/31/78/. /s7922/F5/3E/02/C5/06/FF/DD/E5/  
DD/E1/10/FA/C1/C3/BF/56/00/00/A7/C6/07/C5/06/FF/DD/E5/DD/E1/10/  
FA/C1/C3/EA/56/. /^C

Und: A>SAVE 136 FDXB+.COM  
A>FDXB+

Im Panel nachkontrollieren ob alles stimmt und mal nen Probetext mit beiden Basicversionen eintippen. Der Unterschied ist schon enorm !!

*(Thomas Günther, 6967)*

B A S I C: Cassetten-Inhaltsverzeichnis

R\_o\_l\_f\_S\_c\_h\_m\_i\_d\_t

Lieber Herbert !

Anbei ein kleines Utilitie von Eurem Zombie aus Dortmund.  
 Da mein MTX mit einer defekten Centronics seit 8/85 bei VOBIS weilte, und ich  
 ihm schon Jetzt (sieh mal an) wiederhabe, konnte ich nur trocken Programmieren.  
 Leider sind nunmehr die Tasten '^', 'ü' defekt. Warum auch nicht ? (Sch...Vobis)  
 Das Ergebnis meiner Trockenübungen ist das Cassetten-Inhaltsverzeichnis, welches  
 auch die Titel cassettengerecht ausdrückt.

---

\*BITTE INITIALISIEREN SIE CAS.DIR.BAS

\*CASS.-NR. IN ZEILE 0 also Zeile 0 editieren und statt 'X' eine Zahl rein!  
 \*TITEL IN DATA-Z. 1-3 also Zeilen 1-3 editieren und Programm-Titel rein.  
 Es müssen immer 30 DATAS sein!  
 \*START MIT > RUN < , wenn fertig, RUN

Hier die Original-Copy:

```
0 LET CNR=X: REM HIER STATT "X" CASS.-NR.
1 DATA Z.B.: ,CAS.DIR.BAS,#3#,#4#,#5#,#6#,#7#,#8#,#9#,#10
2 DATA Z.B.: ,CAS.DIR.BAS,#3#,#4#,#5#,#6#,#7#,#8#,#9#,#10
3 DATA Z.B.: ,CAS.DIR.BAS,#3#,#4#,#5#,#6#,#7#,#8#,#9#,#10
```

Der Trick in Zeilen 180,190 ist auch in anderen Dateien gut zu gebrauchen.  
 Zeile 180 erkennt, ob er ein MTX 500/FDX oder einen MTX 512 vor sich hat.  
 Zeile 190 testet, ob in Zeile 0 ein 'X' (DEZ 88) steht.

Einen schönen Gruß aus Dortmund





New Word: Printer-Initstring

**Printer-Init/Uninit-String für neues NewWord**

(Uwe Grass, 3300 & Ulrich Hönisch, 3300 & Herbert Herberg, 2000)

Mit dem NWINSTAL.COM kann

Printer initialize String = Drucker-Initialisierungs-String

Printer uninitialize String = Drucker-Uninitialisierungs-String

Diese findet man im Patch-Menü (H) #3 (3) unter L bzw. M.

Wozu ?

Damit könnt Ihr dem Drucker einen selbstausgedachten String, also eine Zeichenfolge, vor (init-String) und nach (uninit-String) dem Drucker schicken. So bietet sich die Möglichkeit, dem Drucker einige Voreinstellungskommandos schicken. Leider ist der Platz dort recht begrenzt: 9 Zeichen für Init / 5 für Uninit plus ein Byte (das erste) für die Anzahl. Was soll dahin? Nun für den DMX 80 gibt's nicht allzuviel - vielleicht ein Reset: ESC #64. Interessant ist es vielleicht so einigen absonderlichen oder atypischen Typenradgeräten auf die Sprünge zu helfen. Hier nun zwei mögliche Installationen:

Init: ESC #64 ESC "C" #00 #0B Drucker-Init, 11"-Papier

Uninit: "END" #0A #0D Text END gefolgt von LF und CR

A>NWINSTAL NW

NWINSTAL 13 Dec 85

Copyright 1983,1984 Newstar Software Incorporated.

All rights reserved.

Memotech Version

MAIN MENU

What is your choice? H

PATCH MENU #1

What is your choice? 3

PATCH MENU #3

L ULINI - Printer initialize string

M ULUNI - Printer uninitialize string

What is your choice? L

057D 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

02 1B 64 1B 'C 00 0B X

PATCH MENU #3

What is your choice? M

0587 00 00 00 00 00 00 00 2C 00 00 00 00 00 00 00 .....

07 'E 'N 'D 0A 0D X

PATCH MENU #3

What is your choice? X

MAIN MENU

What is your choice? X

All changes have been made.

Newword is now installed for...

Do you want to change this? Y/N N

New Word: Printer-Overlay**Newword-Printer-Overlays** (Bernd Preusing, 2200)

Endlich habe ich es geschafft, das NWPRINT.OVR zu knacken! Das eröffnet natürlich ungeahnte Möglichkeiten endlich mal vernünftig zu drucken. Solche Probleme wie das Nicht-Unterstreichen von Leerzeichen oder daß aus 2K Text beim Drucken plötzlich ein voller 16K Druckpuffer wird, gehören somit bald der Vergangenheit an. Von 6 Customs gar nicht zu reden (^PA, ^PÜ).

Ich habe einen DATA-Treiber und einen FX80(FX100,RX80)-Treiber in das Overlay für NW 2.02 eingebaut. Zusätzlich habe ich einen Beispiel-Treiber geschrieben, der genau erklärt, welche Parameter in welcher Form übergeben werden. Wer sich seine eigenen Treiber schreiben möchte, um das Letzte aus seinem Drucker herauszuholen, kann damit vielleicht etwas anfangen(!) Beides gibt es als public domain beim Club.

Der DATA-Treiber arbeitet ähnlich wie PREVIEW, es kommt aber nur der 'nackte' Text auf die Diskette, d.h. ohne Rand, Header, Footer, etc. Das ist sehr nützlich um Daten mit Merge-Print zu bearbeiten.

Der FX80-Treiber ist ein erweiterter MX80-Treiber, der zusätzlich die Elite-Schrift (.cw10, .cw20) kennt und das Unterstreichen wirklich ein- und ausschaltet, d.h. durchgehend unterstreicht. Außerdem ist er der einzige Treiber, der das Dot-Kommando .ulon/off auswertet, d.h. Leerzeichen unterstreicht oder nicht! Die Customs ^PA und ^PÜ sind unter NW2.02 nicht installierbar; aus diesem Grund habe ich auf ^PA 'ASCII-Zeichensatz' und auf ^PÜ 'deutscher Satz' fest eingebaut. Aber Vorsicht: Sollten die alten Treiber auf diese Zeichen stoßen, ist mit herrlichen Abstürzen zu rechnen!

Der FX80-Treiber hat aber immer noch den Nachteil, daß er a) relativ langsam ist (weil er sage und schreibe jede Zeile ca. 36 mal durchgeht) und b) immer noch wahnsinnig viele Zeichen zum Drucker sendet, weil er z.B. nicht ausnutzt, daß man den linken Rand setzen oder den Druck-Modus mitten in der Zeile wechseln kann. Nebenbei werden die Customs nur dann an der Stelle gesendet, wo sie stehen, wenn die Elite-Schrift eingeschaltet ist. (???) Dieser Treiber sollte nach meinen Informationen auch einwandfrei mit dem DMX80 laufen!

Meinen Wunsch-Treiber für den FX80 zu schreiben, wird wohl noch etwas dauern, ganz so einfach ist es nicht. Eigentlich sollte man doch meinen, je intelligenter der Drucker, desto einfacher der Treiber, aber weit gefehlt! Wenn er fertig ist, könnt Ihr ihn haben.

Das NWPRINT.OVR ist folgendermaßen aufgebaut:

**0. Sektor (#0100):** Copyright und Versionsnummer (wichtig!)

Das zweite Byte gibt an, wieviele Sektoren der Header belegt, also in welchem Sektor die Programme beginnen. Hier steht normalerweise eine 3, d.h. die Daten beginnen im 3. Sektor (#0280)

**1. und 2. Sektor (#0180):** Dies ist eine Art Directory des Files. Hier stehen die Namen der Treiber und deren Adressen.

Jeder Eintrag besteht aus 3 Byte Adresse und maximal 6 Byte Name. Das erste und zweite Byte bestimmen den Sektor relativ zum Daten-Beginn, das dritte Byte ist der Offset innerhalb des Sektors.

New Word: Printer-Overlay

Um die Anfangsadresse eines Treibers zu berechnen, nimmt man das erste Byte, schiebt alle Bits 1 nach rechts und tut das herausfallende Bit in Bit 7 des zweiten Bytes. Jetzt hat man eine 16Bit-Sektornummer, zu der man den Offset des 0.Sektors addiert (also 3). Multipliziert man das Ergebnis mit 128 und addiert das dritte Byte dazu, hat man die Byte-Nummer relativ zum File-Beginn. Nehmen wir an, wir haben mit DDT NWPRINT.OVR das File geladen und finden vor dem Namen 'DRAFT' die 3 Bytes #01,#12,#34, dann haben wir die relative Sektor-Nummer #92. Diese mult. wir mit #80 und erhalten #4900, dazu addieren wir den Header (und #0100 für die TPA), also #0280 und erhalten #4BB0. Der Treiber beginnt also bei #4BB4 im Speicher.

Nach diesen 3 Bytes kommen max. 6 Bytes für den Treiber-Namen, der mit #FE abgeschlossen wird. Danach folgt der nächste Eintrag. Ist die Adresse gleich der vorigen, sind die Treiber identisch. Das Ende der Directory ist erreicht, wenn kein Name folgt, sondern nur #FE. Vor diesem #FE steht somit die erste freie Adresse des Files.

**ab 3. Sektor (#0280) die Daten (Programme)**

Jetzt kommt der Haken, die Druck-Programme sind nämlich im REL-Format abgelegt und müssen einen ganz bestimmten Aufbau haben, damit Newword sie beim Aufruf richtig laden kann (nur CSEG-Modus, gut, daß ich keinen MACRO 80 habe!).

NW hat einen eigenen Linker, der das Programm (bei NW2.02) an die Adresse #8500 lädt. Jetzt noch den Aufbau eines REL-Files erklären zu wollen, würde zu weit gehen, vielleicht hole ich es einmal nach.

Die Druck-Programme bekommen den Text zeilenweise übergeben, die Formatierung und Auswertung der Punktkommandos übernimmt Newword selbst. Zusätzlich werden etliche Informationen geliefert, z.B., ob es sich um Text, Header oder Footer handelt, ob gedruckt werden soll (falls nur Seite 7 bis 10 sollen muss der Treiber trotzdem alle Steuerzeichen auswerten) und alle wichtigen Werte wie Rand, Zeichenbreiten, Zeilenhöhen, Flags etc,etc, also alles, was durch .-Kommandos beeinflussbar ist und einiges mehr.

Beim Durchgehen dieser Routinen habe ich einige **Möglichkeiten von Newword** entdeckt, die nirgends dokumentiert sind.

Wie Ihr wisst, gibt es immer 2 Zeichenbreiten, die mit ^PA und ^PN anwählbar sind, aber damit nicht genug, diese werden für Header und Footer gesondert behandelt, d.h. die letzten .cw vor einem .he oder .fo werden für Header und Footer übernommen.

Beispiel:

```
.cw12^A
.cw24^N
.he Dies ist cw12^Aund dies cw24^N
.cw20^A
.cw10^N
.fo Dies ist cw20^Aund dies cw10^N
.cw14^A
.cw7^N
text text ...
```

Daraus lernen wir 1: ^PA und ^PN werden von NW auch hinter den .cw noch erkannt, 2: man kann im Text beliebig die cw wechseln, die Header und Footer bleiben immer gleich.

NewWord: Printer Overlay / deutsches Handbuch**Fortsetzung: NewWord Printer-Overlay**

Das .ul-Kommando legt fest, ob die Leerzeichen unterstrichen werden, was aber nur etwas nützt, wenn der Treiber das Flag auch auswertet, was in NW2.02 bei keinem Treiber der Fall ist!

^PL ist dasselbe wie .pa (also neue Seite)  
 ^PF ist ein phantom space (für Typenraddrucker)  
 ^PG ist ein phantom rubout (für Typenraddrucker)  
 ^P^ ist ein Soft-Hyphen innerhalb der Zeile (nicht drucken)  
 ^P\_ ist ein Soft-Hyphen am Zeilenende ('-' drucken)  
 ^PK, ^PP, ^PU und ^Pö bewirken nichts und dürfen nicht gedr. werden!  
 (Mit ^PP habe ich noch Großes vor, also vorl. nicht benutzen!)  
 ^PA und ^PÜ sind die zusätzlichen Customs des neuen Newword und sind bei 2.02 mit äußerster Vorsicht zu genießen!!

So, wer gibt denn jetzt nach den Spaltenblöcken (DANKE!!!) vom letzten Info und diesem Zeug noch Geld für ein neues Newword aus??

Nebenbei halte ich es für eine ausgemachte Schweinerei, wenn in 2.02 schon alles vorbereitet ist und hinterher wieder 'ausgepatcht' wurde!!

**Deutsches Handbuch zu NewWord (Ulrich Hönisch, 3300)**

Hallo NewWord-Anwender,

nachdem Uwe Grass sein NewWord Install Handbuch herausgegeben hat, habe ich mich entschlossen auch ein Handbuch herauszugeben. Es soll sich primär mit der Anwendung und seinen Schwierigkeitendabei auseinandersetzen. Eine wichtige Frage taucht dabei jedoch auf:

**Wieviele Leute haben ein Interesse daran, dieses Handbuch zu erwerben?**

Es wird sich aus zwei Teilen zusammensetzen. Zunächst werden sämtliche CTRL-Kommandos, Dot-Kommandos und alle zur Zeit bekannten Befehle in ihrer Wirkung beschrieben. Der zweite Teil wird sich mit dem Umgang mit Texten beschäftigen, insbesondere Tips und Tricks, die entweder Teile des eigenen Erfahrungsschatzes sind, oder Informationen aus dem Info wiedergeben und ergänzen.

Der Preis für diese geballte Ladung von Informationen wird sich auf schätzungsweise **DM 40.-** belaufen. Dieser Preis geht von 30 verkauften Exemplaren aus. Um zu erfahren, ob eine größere Anzahl verkaufter Exemplare den Preis noch wesentlich drücken kann, möchten wir Interessenten darum bitten entweder mir oder Herbert Herberg dieses **bis 31. Mai 1986** mitzuteilen. Wer sich sicher ist, möchte bitte gleich eine Bestellung an mich schicken. Das Werk ist dann voraussichtlich **ab Ende Juni verfügbar**. Sollte jedoch das Interesse zu wünschen übrig lassen (wird im Info vermerkt), dann gibts kein Handbuch!

New Word: Rette Text nach Bdos Err / Verzögerungszeiten

**NewWord Bdos Err** (Dietmar Gröning, 4630)

Jeder kennt das säuerliche Gefühl, wenn mitten in der Arbeit mit Newword der Mtx aussteigt. Nichts gegen Aussteiger, aber da ist dann doch der Güte ein Ende gesetzt. Ärgern nützt nichts. Was tun? (Lenin)

1. Falls eine NW Disk mit Selbststart drin ist, rausnehmen und eine mit System Config und DDT.COM reinschieben. Resettasten treten. Laufwerk C: konfigurieren.
2. DDT aufrufen. Meldet sich mit DDT VERS 2.2
3. -d9580 RET  
Bei 9580 beginnt der Textspeicher von NW. Dieser sollte den Text noch beinhalten. Er wird zwar nicht vollständig aussehen, aber er ist es, denn Wortende und anderes wird in NW durch setzen des Bit 7 angezeigt. DDT erkennt das nicht als ASCII Zeichen und setzt einen Punkt ".".
4. Mit dxxxx das Textende suchen, bzw. die Stelle finden, bis zu der der Text gerne erhalten würde.  
An diese Stelle nun mit -sxxxx 1A setzen. Dies markiert das Dateiende.
5. Nun mit Move das Ganze nach vorne schieben: -m9580,xxxx,0100
6. Textende erneut suchen. Aus der Speicherbelegung berechnet sich die Anzahl der zu savenden Seiten.

Reicht die Information im Speicher von Hex 0100 bis einschließlich:

01ff  
0200  
151A  
A0F0

dann ist die folgende Anzahl Seiten zu Speichern:

1  
2  
1\*16+5= 21  
10\*16+0=160 (Nach B.Pohl)

7. Mit ^C raus und SAVE anzahl d:nnnnnnnn.eee
8. ggf. Kontrolle ob alles korrekt gelaufen ist. (Mit dir oder d)
9. NW erneut laden und die gerettete Datei weiterbearbeiten (bis zur nächsten Störung).

**NW: Verzögerungszeiten in den Menüs** (Jan Brederke)

Zur Frage von Peter Baumeister im Info 12, Seite 65: Auch mich hatte es gestört, daß Newword beim Eingeben eines Filenamens genau dann anfing, das Directory aufzulisten, wenn mir der Filenamen eingefallen war. Durch den Diskettenzugriff gingen dabei immer wieder getippte Buchstaben verloren.

Daher habe ich mittels NWINSTAL die Verzögerungszeit vergrößert. Im Menüpunkt D (Newword related items) gibt es den Unterpunkt N (Short delay time), mit dem diese Verzögerungszeit in Millisekunden eingestellt werden kann. Ich habe den Wert von 200 auf 1000 erhöht, so daß ich jetzt 1 Sekunde Zeit habe, den ersten Buchstaben in Ruhe zu tippen.

Außerdem habe ich im Unterpunkt P (Long delay time) 0 statt 3000 eingegeben, damit ich beim Aufruf von NW nicht immer 3 Sekunden lang die Begrüßungsbotschaft betrachten muß, bis das Öffnungsmenü erscheint. Jetzt erscheint sie nur noch ganz kurz, und zwar so lange, wie Newword braucht, um irgendwelche Overlays von Diskette nachzuladen.

Weiterhin gibt es noch Unterpunkt O (Medium delay time). Dort habe ich die 1000 gelassen, denn ich bin mit der Zeit, nach der die Kommando-Menüs erscheinen, ganz zufrieden. Dies betrifft zum Beispiel das Menü, das nach ^P erscheint, wenn man mit dem zweiten Buchstaben etwas wartet.

C P / M: Disc-Utility zum retten gelöschter Dateien**DU = Programm (auch) zum Retten irrtümllich gelöschter Dateien**

(Ulrich Hönisch, 3300)

Es gibt zwar auch noch eine Menge anderer Anwendungen dieses Programms, aber ich will mich hier darauf beschränken zu erklären, wie es zu bewerkstelligen ist, unabsichtlich gelöschte Files wieder zurück zu holen. Dabei werde ich auch die Möglichkeit darstellen, die User Area zu ändern. Allerdings ist das Thema User Area mit großer Vorsicht zu genießen, denn es kann vorkommen, daß ihr buchstäblich nichts wieder findet!

Zunächst ein wenig grundsätzliches dazu, wann es überhaupt sinnvoll ist, das Programm DU.COM von der Public Domain Diskette noch anzustarten:

- es fällt einem wie Schuppen aus den Haaren in dem Moment, wo man die Taste los gelassen hat, daß die gelöschte Datei doch genau die ist, die eben in stundenlanger Arbeit erstellt worden ist und nicht die, die als Programmschrott sowieso untergehen sollte!
- alle Programme, welche entweder dümmlicherweise oder als Gag auf einer anderen User Area angekommen sind (üblich ist User 0) können auf den User 0 gebracht werden.

Bei Disketten, auf welche nach dem "kleinen Mißgeschick" schon wieder geschrieben wurde, könnt ihr euch in der Mehrzahl der Fälle die Mühe sparen. Sollte der Name der gelöschten Datei noch im DU-Directory (Erklärung später) auftauchen, so lohnt der Versuch auf jeden Fall! Ist der Name dort schon nicht mehr vorhanden, so bleibt nur der Trost, daß der Tag 24 Stunden hat, um ein Programm neu zu schreiben. Notfalls kann man ja noch die Nacht dazu nehmen!

Nun zum Wesentlichen:

in dem Moment, wo ihr euer Mißgeschick bemerkt, solltet ihr jede weitere Aktion mit dieser Diskette bleiben lassen!!!! Befindet ihr euch im Newword und die "Löschaktion" hat eben erst statt gefunden, so gibt es mehrere Wege. Habt ihr einen Text in Arbeit, welchen ihr noch nicht in einer Vorversion abgespeichert habt und euch wichtiger ist als der Andere, so macht ihr am besten folgendes: mit ^KS !!! den Text abspeichern und danach den Rechner zurück setzen (RESET). Lest jetzt weiter bei: nach dem Reset!

Habt ihr einen Text in Arbeit, den ihr noch mal irgendwo habt, dann die Diskette sofort heraus und Reset! Steigt nicht "normal" aus dem Newword aus. Auch nicht ^KQ, also "Abandoning", denn alles, was da an Arbeit auf der Diskette passiert, könnte der Tod eurer Datei sein!

Befindet ihr euch nicht im Newword, sondern in irgend einem anderen Programm, dann stellen sich euch prinzipiell die gleichen Fragen. Welche Datei ist euch wichtiger? Ist es die bereits gelöschte, so bleibt nichts anderes, als die Diskette heraus zu nehmen und den Rechner zurück setzen. Im anderen Fall das wichtigere Programm erstmal an dem Punkt abzuspeichern und danach könnt ihr euer Glück zu versuchen.

C P / M: Disc-Utility zum retten gelöschter DateienNach dem Reset

Nehmt jetzt die Diskette mit dem DU.COM und startet das Programm an. Für Leute mit zwei Diskettenlaufwerken ist die Sache prinzipiell genauso wie für Leute mit einem. Sie können die zu reparierende Diskette drin lassen, müssen sich aber von dem Laufwerk, auf welchem das DU.COM angestartet worden ist, erst in das andere "einloggen". Das funktioniert in folgender Weise

I & <RET>

Für das & setzt ihr den Laufwerksbuchstaben ein, in welchem die zu reparierende Diskette sich befindet. Die Leute mit einem Laufwerk können nach dem Anstarten des DU.COM diese Diskette sofort wieder heraus nehmen. Das Programm hält sich recht hartnäckig im Speicher. Packt nun die zu reparierende Diskette in das Laufwerk. Sowohl die Leute mit einem Laufwerk als auch die mit zweien geben nun

m <RET>

ein. Es wird nun das DU-Directory auftauchen. Dort muß eure Datei mit dem Zusatz (E5) erscheinen. War die Datei größer als 32 Kbyte, so müßt ihr das DU-Directory sorgfältig absuchen, denn es müssen noch weitere Teile der Datei mit (E5) vorhanden sein, die ihr dann alle in der nachfolgend geschilderten Weise verarzten müßt. Solltet ihr nicht fündig werden, ist leider nichts (zumindestens nichts, was mit einigermaßen vertretbarem Aufwand zu bewerkstelligen ist) mehr zu machen. Für die Glückspilze, bei denen nun ihre geliebte Datei auftaucht, fängt die Arbeit erst an. Gebt jetzt folgendes ein:

g00 <RET>.

Ihr seid nun an der Stelle der Diskette, wo das Directory steht. Dazu ein wichtiger Hinweis: gelöscht werden Dateien nur in dem Directory. Solltet ihr etwas besitzen, was ihr nicht mehr haben wollt und wo ihr 100 %tig sicher gehen wollt, daß es niemand mehr zu Gesicht bekommt, so muß(!!!) die Diskette formatiert werden. Auch das Vollschreiben stellt nicht 100%tig sicher, daß nicht genau der Absatz lesbar bleibt, welcher sicher weg sollte! Soweit der kleine Hinweis an einsichtige Raubkopierer! Gebt nun

d <RET>

ein. Es wird eine für ungeübte Augen recht verwirrende Ansammlung von Zahlen auftauchen. Steht dort schon die zu rettende Datei, so lest bei "Rettungsaktion" weiter. Das kleine "e" vor eurer Datei zeigt übrigens an, das sie gelöscht ist. Die anderen müssen ihr Directory langsam "durchblättern". Dazu müßt ihr nun

+ <RET>

d <RET>

eingeben und zwar so lange bis eure Datei auftaucht. Das ist zwar recht mühselig, aber nicht zu ändern!

C P / M: Disc-Utility zum retten gelöschter DateienRettungsaktion

An der linken Seite des Bildschirms seht ihr Zahlen von 00-70. Je nachdem an welcher Position nun eure zu rettende Datei steht, gebt ihr nun ein:

**h??,00<RET>**

Natürlich gebt ihr nicht zwei Fragezeichen ein, sondern wenn eure Datei an Position 40 steht

**ch40,00 <RET>**

bzw.

**ch60,00 <RET>** für Position 60.

Es wird dadurch Hexadezimalzahl E5 durch 00 ersetzt. Zunächst aber nur im Speicher! Kontrolliert nun, ob ihr keinen Eingabefehler gemacht habt. Dazu gebt noch einmal

**d <RET>**

ein. Stellt ihr nun einen Fehler fest, d.h. an der entsprechenden Stelle findet ihr nicht 00, sondern etwas anderes, so ist das auch kein Beinbruch. Einfach

**r <RET>**

und dann:

**d <RET>**

eingeben und ihr habt wieder das, was vorher da stand. Um eine gelungene Änderung auch auf die Diskette zu bringen, müßt ihr jetzt

**w <RET>**

eingeben. Durch die Eingabe von

**x <RET>**

könnt ihr jetzt aus dem Programm aussteigen und die Arbeit mit der so geretteten Datei wieder aufnehmen.

Ändern des User-Bereiches

An der Stelle, wo ihr eben die Hexadezimalzahl E5 vorgefunden habt, steht, wenn die Datei einem anderen Userbereich zugeordnet ist, eine Zahl, z.B. 06. Diese Zahl besagt, daß diese Datei nur dem User 06 zugänglich ist. Um diese Datei entweder einem anderen Userbereich zuzuordnen oder wieder allgemein zugänglich zu machen, also dem User 00, dann habt ihr folgendes einzugeben:

**ch?,&& <RET>**

C P / M: Disc-Utility zum retten gelöschter Dateien / SuperCalc

Dabei stehen die beiden Fragezeichen für die Zeile, in welcher ihr den User ändern wollt und die Dollarzeichen für den User, den ihr da einsetzen wollt. Also 00 für den üblicherweise benutzten User 0. Demzufolge dann 04, wenn ihr diese Datei dem Userbereich 04 zuordnen wollt. Dazu noch ein kleiner Hinweis: wer nur alleine an seinem Rechner sitzt macht sich das Leben unnötig schwer, wenn er Teile seiner Dateien planlos in andere Userbereiche verschiebt. Denkt auch daran, daß ihr bei Dateien, die größer als 32 kByte sind, alle Teile mit der neuen Userkennung versehen müßt!

Wer weitere Teile seiner Programme durch "Patches" verändern will, sollte entweder das in Englisch geschriebene DU-V86.DOC der Public Domain Diskette No.1 (CPM) zu Rate ziehen oder gleich den Monitor von Olaf Krumnow für lächerlich wenig Geld erwerben.

Supercalc-Patches (Jens Ebert): Die neue Darstellung ist toll, aber blinkende und piepende neg. Zahlen? Das habe ich ganz schnell abgestellt, indem ich das Längenbyte dafür auf 0 gesetzt habe. Man sollte im Installationsprogramm als Druckerinitialisierungsstring OF angeben, um eine komprimierte Druckausgabe zu erhalten. Die Zeilenlänge von 132 Zeichen stimmt dann auch, ev. die Seitenlänge noch auf 72 Zeilen umstellen. Dies läßt sich ja alles zur Laufzeit nach Bedarf wieder umstellen (Setup-Option des /Output-Befeh[ls]).

SuperCalc - Installation (Michael Kessler)

Zuerst gebe ich mal eine Erkenntnis an Dich weiter, die ich beim ersten Durchsehen der Club-Infos hierin noch nicht gefunden habe. Es handelt sich um das SuperCalc Installationsprogramm INSTALL. Wenn man dieses Programm aufruft und schließlich ins erste Menü gelangt, so kann man unter der Überschrift "Current Terminal" die Worte MEMOTECH FDX lesen und darunter eine Reihe anderer Terminals, die installiert werden können. Am Ende dieses Menüs steht dann: Enter A-P,Y,Z or ?. Nun habe ich entdeckt, daß das Programm auch die Eingabe "X" annimmt, und dann geht's erst richtig los mit dem installieren. Es erscheint nämlich folgendes auf dem Schirm:

```
This will erase current values for new terminal definition if
you wish. (To simplify installation) Do you wish to erase current
values (y/n) ?
```

Man hüte sich davor, diese Frage mit "Y" zu beantworten, da dann die Installation des MEMOTECH FDX zerstört wird, und man sich dann wieder das Original von der Master-Diskette kopieren muß. Gibt man aber "N" ein, so wird man zu einem weiteren Menü geführt:

These are the items you may now edit:

- A. Edit screen controls
- B. Edit attribute data
- C. Edit input keys
- D. Edit GotoXY, printer init. string
- E. Edit miscellaneous data
- F. Edit terminal name
- X. Finished editing data

C P / M: .SUB-Kommentare / RAM 3

### **Kommentare ! in .SUB-Dateien**

(Michael M $\ddot{u}$ we, 2000)

Man/frau kann Kommentare in SUBMIT-Dateien einfügen, die vom CCP - kommentarlos- geschluckt werden. Was ist gemeint? Einige Kommandoangaben an den CCP werden so häufig getätigt, daß es einfacher ist diese über eine Batch-Datei erledigen zu lassen, doch kann es dann vorkomen, daß diese auch für denjenigen, der sie geschrieben hat, eines Tages nicht mehr zu enträtseln ist. Um diesem Problem aus dem Wege zu gehen, gibt es die Möglichkeit Kommentare direkt in die Kommando-Datei zu schreiben, die eine 'Online'-Information über das Geschehen liefern. Diese Kommentare können auch hinter den eigentlichen Befehlen stehen. Es gibt mehrere Zeichen, die als Kennzeichnung eines Kommentares dienen; im folgenden sollen einige Beispiele dafür gezeigt werden.

A>; **1. Kommentar**<RET>  
A>. **2. Kommentar**<RET>  
A>= **3. Kommentar**<RET>  
A>< **4. Kommentar**<RET>  
A>> **5. Kommentar**<RET>  
A>\_ **6. Kommentar**<RET>

### **Betr: RAM3 (Bernd Freusing, 2200)**

Die Wogen scheinen sich langsam zu legen, d.h. die Bestellungen von RAM3 lassen nach und die Sonderformate sind wohl alle installiert.

Das läßt mir etwas Luft, mal etwas anderes zu tun, z.B. ein Programm zu schreiben, das die gewünschten Formate aus einer Datei in RAM3 einbaut. Dieses Programm und die Datei gibt es demnächst für alle RAM3-Besitzer gratis (D&P&V). Zum anderen kann ich mir allmählich Gedanken über RAM4 machen, das sicher nicht vor dem Herbst auf dem Markt erscheinen wird. Das wird zum Teil im Boot-EPR0M sein, damit man endlich dieses blöde Format 03 ad acta legen kann. Dazu gehört dann natürlich ein komplett neues Bios.

Da ich jetzt eine 1MByte Ram-Disc(c't) und (endlich!) meine 512K-Karte habe bedeutet dies, daß der interne Speicher mit Programmen vollgestopft wird, d.h. Klick erhält eine Schnittstelle, mit deren Hilfe sich speziell dafür geschriebene Programme einbauen können, so daß sie jederzeit via Klick aufrufbar sind (Monitor, VS4-Treiber, Editor, Drucker-Steuerung etc.). Der Spooler wird so umgeschrieben, daß er nur noch mit Files arbeitet, d.h. die blödsinnige Aufteilung des Speichers entfällt und es wird kein Platz verschwendet. Für weitere Vorschläge bin ich jederzeit dankbar.

C.P. / M: Zum Thema CP/M 3.0

**Betr.: CPM3** (Bernd Preusing, 2200)

Wer das zu RAM3 gesagte richtig verstanden hat, kann sich das Lesen dieses Beitrages sparen.

Geplant ist dieser Beitrag seit einigen Monaten, aber wer hat schon Zeit; außerdem geht jede DOC-Zeit von meiner MAC-Zeit ab und Prosa war noch nie meine Stärke. Zu diesem Zeitpunkt wäre es noch ein halbwegs ausgeglichener Beitrag geworden, heute...

CPM3 ist weit besser als CPM2, und zwar um Längen, ich würde weit lieber unter ersterem arbeiten. Aber:

Ich lese nun erstmal den Leserbrief von Jobst Kilian in Info 12.

#### **1.: Preis**

Den Preis kann ich nicht bemängeln, er scheint mir für ein derartiges Software-Produkt durchaus angemessen, besonders wenn man bedenkt, daß CPM2 auf dem freien Markt über 300.-DM kostet. Außerdem glaube ich nicht, daß es für diesen Preis vollständig mit Doku und allen Dienstprogrammen ist.(?) Aber wer ist bereit, dieses Geld dafür auszugeben?

#### **2.: schnelle Dateizugriffe**

Das braucht viel Speicher, wer den hat siehe dazu RAM3. CPM3 bekommt für jedes Laufwerk einen Puffer zugewiesen, in dem es die meistbenutzten Sektoren ablegt. Das bedeutet a) daß nach jedem Warm-Boot die Puffer gelöscht werden und b) daß Laufwerke dauernd an und aus gehen, wenn nur ab und an ein Sektor von der Disk gebraucht wird. Besonders für Laufwerke mit Kopflademechanismus ist das eine Tortur!

#### **3.: Help**

Na ja, gibt es auch für CPM2

**Anm.d.HH** Das Programm HELP.COM vom CLUB.004

#### **4.: Archiv-Attribut**

Gibt es schon seit CPM1.4. Nur PIP von CPM3 nutzt es aus, aber wer benutzt schon PIP? Was uns fehlt ist ein Kopierprogramm, das solche Attribute berücksichtigt.

#### **5.: Passwort-Schutz**

Wer ein Multi-User-System auf dem MTX fährt soll sich bei mir melden, das interessiert mich sehr!

#### **6.: Hardware-Uhr**

Das ist nun mal ein wirklicher Vorteil von CPM3! Nur eine Uhr und die entsprechende Software im System zu haben reicht nämlich noch lange nicht, legt doch CPM3 Datum und Uhrzeit in der Directory ab, sodaß man jederzeit den letzten Zugriff und die letzte Änderung eines Files sehen kann. So etwas läßt sich unter CPM2 kaum machen.

#### **7.: RSX-Module**

Auch eine schöne Sache, siehe RAM3

#### **8.: IO-Umleitung**

Das Plus von CPM-Plus. Man kann unter CPM3 bis zu 12 IO-Kanäle definieren und jedem der 4 logischen Kanäle (TTY, CON, LST, AUX) zuweisen, auch mehrere!

#### **9.: Shorthand**

Das war mir neu! Ein Vorteil bei der Befehlseingabe ist, daß man die vorher getippte Zeile wiederholen und editieren kann.

**Anm.d.HH** das wird unser CP/M 2.2 auch lernen!

#### **10.: mehr BDOS- und BIOS-Einsprünge**

Sehr gut, aber nur für faule Programmierer

C P / M: Zum Thema CP/M 3.0

Nun ein paar eigene Bemerkungen:

### **11.: alte Software**

Man müßte sämtliche System-Software umschreiben oder gar ganz neu schreiben, als da wären CONFIG, SYSCOPY, RCHECK, ENTER etc,etc. Sehr viele der mittlerweile geschriebenen Programme gehen von CPM2 und unserem System aus, auch diese hätten nur noch Schrottwert.

Nun gibt es zwei Möglichkeiten:

a) man schreibt ein Bios, das voll kompatibel ist, das müßte dann auch ein eigenes Sektor-Blocking durchführen, damit z.B. TURBO noch vernünftig läuft. In diesem Fall könnte man die alte Software weiternutzen, aber was dann an Vorteilen bleibt ist die immense Arbeit nicht wert!

b) man schreibt ein Wahnsinns-Bios, das den alten MTX zu einem Turbo macht, dann folgt die ganze Umschreiberei.

### **12.: Erkennung von Disk-Formaten**

In einigen Köpfen geistert die Idee herum, man könnte ein CPM3-Bios schreiben, das Disketten-Formate selbstständig erkennt. Das ist richtig, aber falsch. Es kann zwar erkennen, ob ein Format single- oder double-density ist oder wieviele Spuren und Bytes pro Sektor es hat, aber dann ist Schluß. D.h. es kann aus einigen verschiedenen Formaten eins herausfinden, aber ich möchte den sehen, der ein Programm schreibt, das den Unterschied zwischen MTX03 und Triumph Adler erkennt! Außerdem hat dieses Verfahren den Nachteil, daß das BIOS nach jedem Warm-Boot erstmal die Diskette identifizieren muß, was die Belastung stark erhöht. Da bleibe ich lieber bei meinem guten alten CONFIG.

### **13.: BDOS-Error on .. R/D**

Ein weiteres absolutes Plus ist, daß diese ekelhafte Meldung nicht mehr erscheint, wenn man vergaß, eine Diskette einzuloggen. Nebenbei gibt das BDOS auf Wunsch ausführliche Fehlermeldungen mit der Angabe, bei welchem File und welchem BDOS-Aufruf der Fehler auftrat.

### **14.: BIOS-ERROR on Track.., Sektor..**

Das kann man auch in CPM2 machen, wird auch in RAM3 geschehen.

### **15.: Dienstprogramme von CPM3**

Da gibt es ganz gute Sachen, die die Features von CPM3 unterstützen, aber man stelle sich vor: es gibt z.B. ein DIR.COM, das auf der Disk sein muß, wenn ich eine erweiterte Form von DIR aufrufe! Und von dieser Sorte gibt es mehrere, d.h. die Systemdiskette ist nur von den Dienstprogrammen schon fast voll.

### **Das Letzte:**

Ich habe zwar mal eine Art Vorführversion von CPM3 hergestellt, aber die Tatsache, daß ich noch nie damit gearbeitet habe und sie nur ab und zu zu einem Clubtreffen aus der Versenkung hole, sollte doch zu denken geben! Es ist einfach eine Wahnsinnsarbeit, das BIOS anzupassen, und was dagegen spricht siehe 11.

Wer sich ernsthaft für CPM3 interessiert, sollte unbedingt die letzten Ausgaben der Zeitschrift c't lesen, dort war eine sehr ausführliche Serie über CPM3.

Das alles liest sich wie ein Werbespot für RAM3, tut mir leid, aber ich schreibe Programme nicht, weil ich sie verscherbeln will, sondern weil ich sie haben und nutzen möchte. Wer viel Speicher hat, will ihn auch möglichst effektiv nutzen, und das macht RAM3 weit besser als CPM3.

Assembler: Quadratwurzeln**Quadratwurzel in Assembler**(Michael M $\ddot{u}$ we, 2000)

Hin und wieder mag es vielleicht interessant sein die Quadratwurzel aus einer 16-Bit Integer-Zahl zu ziehen, und dann weiss man nicht, wie es zu bewerkstelligen ist. Deshalb hier eine kleine Routine samt kurzer Erluterung.

An das Unterprogramm wird, im DE-Registerpaar, eine 16-Bit Integer-Zahl ubergeben, aus der die Quadratwurzel gezogen werden soll. Das Verfahren ist iterativ, das heit, da das A-Register, von 0 ausgehend, solange incrementiert und quadriert wird, bis  $A^2 \geq DE$  ist.

Um das alles nicht allzu lahmarschig werden zu lassen, wird  $A^2$  aus  $(A-1)^2$  durch Summation errechnet.

Dazu eine Gleichung:

```
X^2=(X-1)^2+Cx
X^2=X^2-2*X+1+Cx      ; (X-1)^2 ausmultipliziert
0=-2*X+1+Cx           ; X^2 subtrahiert
Cx=2*X-1               ; Gleichung umgestellt.
```

Cx nun ist das hupfende Komma, zum -bekannten- Quadrat von (X-1) addiert ergibt es  $X^2$ :

```
0^2=0                  ; allgemein bekannt
1^2=0^2+2*1-1=1       ; scheint richtig zu sein
2^2=1^2+2*2-1=4       ; dito
3^2=2^2+2*3-1=9       ; is' echt phantastisch
u.s.w.
```

Das Unterprogramm macht also folgendes:

```
LET A=INT(SQR(DE)) : LET HL=A^2
```

```
SQRT16:  ; ZIEHT AUS DEM INHALT DES DE-REGISTERPAARES DIE
        ; QUADRATWURZEL.
        ; A:=INT(SQR(DE))
        ; HL:=A^2
        ; DE-REGISTERPAAR UNBEEINFLUSST.
        ; X^2=(X-1)^2+Cx , Cx IST EINE ZU BESTIMMENDE KONSTANTE
        ; X^2=X^2-2*X+1+Cx
        ; Cx=2*X-1 !!! SIEHE UNTEN
        XOR A                      ; A:=0 CY:=0
        SBC HL,HL                  ; HL:=0
        PUSH HL
SQRT#1:  LD H,0
        INC A
        LD L,A                      ; HL:=A
        ADD HL,HL                  ; HL:=2*A
        DEC HL                      ; HL:=2*A-1
        POP BC                     ; (A-1)^2
        ADD HL,BC                  ; HL:=A^2
        PUSH HL
        AND A                      ; CY:=0
        SBC HL,DE                  ; HL:=A^2-DE
        JP C,SQRT#1                ; DE>A^2 => SQRT#1
        JP Z,SQRT#2                ; A^2=DE => SQRT#2
        DEC A                      ; A IST UM 1 ZU GROSS
        POP HL                      ; STACK SAEUBERN
        PUSH BC                    ; (A-1)^2
SQRT#2:  POP HL                     ; HL:=A^2
        RET                        ; ENDE SQRT16
```

A s s e m b l e r: Tips**Kleine Assembler-Tips** (Michael Möwe, 2000)

Und da wir gerade dabei sind, hier noch ein paar kleine Tips, die hin und wieder hilfreich sind.

```

    AND A    ;Carry löschen, Z-Flag gesetzt falls 0,
            ;Akku unbeeinflußt.
;
    XOR A    ;wie oben, aber Akku:=0.
;
    CALL OVER
    DB 'DIESER TEXT WIRD AUSGEGEBEN',13,10,'#'
OVER:  POP DE    ; Zeiger auf den Text
       LD C,9    ; Ausgabe einer Zeichenkette
       CALL 5    ; BDOS-Aufruf

```

Diese kleine Sequenz kann man an jeder beliebigen Stelle im Programm einfügen, an der man möchte, daß ein Text ausgegeben wird. Die Marke 'OVER' darf natürlich nicht mehrmals verwendet werden.

**Anm.d.HH** Es muß der direkt nach dem CALL der Text folgen. Nach dem CALL 5 arbeitet das Programm dahinter weiter. Ein RET darf nicht folgen!, da die Rücksprungadresse durch POP DE (wichtig für Ausgabe) entfernt wird - außerdem würde das RET nur auf den Text springen!

```

;
; jetzt den Akku mit 10 multiplizieren
    ADD A,A    ; A:=2*A, ist schneller als SLA A !!!
    LD x,A     ; x ist ein beliebiges Register
    ADD A,A    ; A:=4*A
    ADD A,A    ; A:=8*A
    ADD A,x    ; A:=8*A+2*A = 10 * A
    ; das war's schon
    ; ADD A,A ist doppelt so schnell wie SLA A !!!
;

```

Hier oder weiter hinten sollten eigentlich noch ein paar Seiten erscheinen, die die Funktionsweise der Z80-SID erläutern. Der DART funktioniert -soweit ich weiß- genauso, nur daß er die synchrone Betriebsart nicht beherrscht.

Bis demnächst auf dieser Welle,

Michael

d.w.

Assembler: IX- und IY-Register**Index-Register der Z80-CPU**

(Michael Möwe, 2000)

Vielleicht hat sich schon der eine oder andere gefragt, was sollen eigentlich die beiden Index-Register IX und IY ? Denn schließlich muß hier immer ein Offset angegeben werden und der Befehl erzeugt sowieso 1 Byte mehr Code \$DD, \$FD u.s.w. . Das Stichwort scheint mir zu sein: **Stackpointer-relative-Adressierung** (SrA). Watt soll ich nu' damit anfangen ? Sinnvoll wird die SrA, wenn man Prozeduren schreiben will, die viele Parameter auf dem Stack übergeben bekommen -auch für die Rekursion eignet sich die SrA hervorragend. Dazu ein kleines Beispiel:

```
( Procedure UNTPRO(P1,P2:Integer;Var Erg:Integer);external )
```

```
UNTPRO: ; KLEINES UNTERPROGRAMM, DAS 2 WERTE ALS 'VAR'-PARAMETER
; AUF DEM STACK ERWARTET UND EINEN ZEIGER AUF DEN
; SPEICHERPLATZ, WO DAS ERGEBNIS ABGELEGT WERDEN SOLL.
LD HL,0 ; HL:=0
ADD HL,SP ; HL:=SP D.H.: HL ZEIGT AUF DIE
; RUECKSPRUNGADRESSE

PUSH HL
POP IX ; IX:=SP
; ES BESTEHT JETZT FOLGENDER ZUSAMMENHANG:
; (IX+2) ZEIGT AUF DIE ADRESSE (!) DES ERGEBNISSES
; (IX+4) ZEIGT AUF PARAMETER 2 (16-BIT)
; (IX+6) ZEIGT AUF PARAMETER 1 (16-BIT)
LD L,(IX+4)
LD H,(IX+5) ; HL:= PARAMETER 2
LD E,(IX+6)
LD D,(IX+7) ; DE:= PARAMETER 1
ADD HL,DE ; HL:= PARAMETER1 + PARAMETER2
EX DE,HL ; SWAP(HL,DE)
LD L,(IX+2)
LD H,(IX+3) ; DAS IST ETWAS KOMPLIZIERT, HL
; ZEIGT AUF DIE SPEICHERSTELLE, WO
; DAS ERGEBNIS HIN SOLL.

LD (HL),E
INC HL
LD (HL),D ; DAS WAR'S
POP DE ; DE:= RETURNADRESSE
LD HL,-6
ADD HL,SP
LD SP,HL ; STACKPOINTER WIEDER 'GERADEGEBOGEN'
EX DE,HL ; HL:= RETURNADRESSE
JP (HL) ; ZURUECK ZUM AUFRUFENDEN PROGRAMMTEIL
```

Dieses Beispielprogramm ist ein wenig aufwendig für das, was es tun soll. Aber es ist ja auch nur ein Beispiel, für den sinnvollen Einsatz der Index-Register. Wenn man mit mehreren Parametern arbeitet, oder die Prozedur rekursiv sein soll, macht sich der Einsatz der Index-Register, gegenüber einem endlosen hin und her 'geFUSHe' und 'gePOPe', durch mehr Uebersichtlichkeit und Codereduktion (?) angenehm bemerkbar.

M a i l b o x**The Box**

(Michael Möwe, 2000)

Es geisterte wohl schon länger in einigen Köpfen der Gedanke herum 'man müßte eine "Mehlbox" für MTX-User aufmachen'.

Sooo weit ist es noch nicht, aber dafür habe ich in einer Hamburger Box eine Ecke für uns reservieren lassen, die zumindest solange benutzt werden kann, bis wir eine eigene Box haben (Jubel !).

Die Mailbox heißt **M.C.S.-Box** und ist unter folgender Nummer zu erreichen: **040/ 251 23 71**.

Die Box meldet sich mit 7N1, was bei mir bedeutet, daß zuerst eine Menge Schrott auf dem Bildschirm erscheint, aber **keine Panik** nach ein- oder mehrmaligem Drücken der RETURN-Taste schaltet die Box auf 8N1 um und die Welt ist wieder in Ordnung.

**Anm.d.HH** Einen Login-Dialog, wie er auf Euch zukommt, wenn Ihr in der Box anruft findet Ihr auf CLUB.007. Hier ist ein verkürzter Dialog, damit es nicht so lang wird!

\* Master-Control-Hauptmenue \*

IN-MB-USZ-RC-UV-CCC-TEL-MCS-UID-SY            (?-HELP-PRO-AMA-SMA-BE-LF-ST-T-LOGOFF)

15:30/Gast)>UID MI-MOE

Passwort >-----

Login : User, aktiv !

Es ist eine Express Mail fuer dich da !!

SYSOP    23.03.1986 / 13:53:04

Hallo Miachel !

So, hier nun Deine Befehle fuer die Gemeinschaftsfaecher fuer die MTX-Usergroup:

Der Fachzugriff wird mit dem Passwort 'FACH' aktiviert.

Ablauf ist: SY anwaehlen und Passwort angeben.

Deine, bzw. Eurer Faecher koennen folgernermassen aufgerufen werden:

MTXINFO	Infos fuer alle user, aber nur readzugriff MTX-Usergroup kann lesen und schreiben !
MTXFRA	Fragen an die Usergroup, kann von Usern nur beschrieben werden, Usergroup kann beides !
MTXANT	Usergroup antwortet, kann von Usern nur gelesen werden, Usergroup kann beides. (wie MTXFRA)
MTXINTERN	Usergroup, Internes File, kann nur von euch gelesen und beschrieben werden.

Folgene Befehlsmoeglichkeiten koennen noch benutzt werden:

LMTXINFO	lesen MTX-Info
EMTXINFO	eingabe in MTX-Info, nur von Usergroup benutzbar !
LMTXFRA	.....
EMTXFRA	.....
	etc.....

Es kann eine Uebersicht der Gemeins. Faecher mit dem Befehl 'GDIR' aufgerufen werden. Dort stehen die Aktuellen Updatewerte drin.

Ich hoffe dir damit gedient zu haben und wuensche dir, und der Usergroup natuerlich auch, viel Spass im MCS-System.

M a i l b o x

\* Master-Control-Hauptmenue \* 'User'  
20:57/MI-MOE)>MB

\* Mailbox - Menue \* 'User'  
20:16/MI-MOE)>?  
MA Message Ausgabe  
ME Message Eingabe  
PML Persoenlich Message lesen  
PME ..... Persoenlich Message eingabe  
ULSM Userliste-Suchmodus  
MDI Message Directory  
UV Userverwaltung  
IN Infomenue  
RC Rechner Corner  
CCC Chaos-Computer-Club  
HM Hauptmenue  
LOGOFF = Verbindung abbrechen

19:57/MI-MOE)>MA

\* Message Ausgabe Menue \* 'User'  
19:53/MI-MOE)>?

AL Allgemeines  
SU Suche  
BI Biete  
KO Kontakte  
TI Tips und Infos  
GE Geruechte  
WI Witze  
MDI Message Directory  
ME Message eingabe  
MB Mailboxmenue  
UV Userverwaltung  
IN Infomenue  
HM Hauptmenue  
LOGOFF = Verbindung abrechen

19:35/MI-MOE)>AL

Ausgabe Allgemeines  
An HARPER, aber auch alle anderen, die es nicht wissen:

Wozu intensiv suche, wenn wir eine riesige Help-Liste haben. Mit dem Befehl 'Help' ist eine Uebersicht ueber alle (fuer den normalen User) moeglichen Befehle erhaeltlich und mit 'Help Befehl' kann man sich dann die Help-Funktion zu den entsprechenden Befehlen geben lassen.

Wenn man ueberhaupt keine Ahnung hat, kann man sich an den Menues orientieren, bis einem z.B. in dem Menue 'Messages schreiben' der Befehl 'pme' in's Auge faellt, wenn man dann auch noch 'Help pme' eingibt, erfahrt man sofort, dass man mit diesem Befehl mit 'pme sysop' z.B. eine Nachricht an den Sysop (THW) dieses Systems schreiben kann. Ich habe bisher in keiner privaten Mailbox (mit Ausnahmer der Inka-Kopien) derartige Helplisten gesehen, nutzt sie !

M a i l b o x

\* Message Ausgabe Menue \* 'User'

18:35/MI-MOE)&gt;GF

Gemeinsschaftsfaecher (JETZT Aktiv) 'User'

18:28/MI-MOE)&gt;?

CPB	CPM-User Group
CPBFRA	Fragen an CP/M-Bau
CPBANT	Antworten von CP/M-Bau
CPBINFO	Infos von CP/M-Bau
HM	Hauptmenue

18:17/MI-MOE)&gt;C

PEINEN MOMENT BITTE....

ATTENTION USER !

HABE NOCH EBEND WAS GESEHEN , WAS ICH NOCH AENDERN MUSSTE

Gemeinsschaftsfaecher (JETZT Aktiv) 'User'

16:16/MI-MOE)&gt;?

CPB	CPM-User Group
CPBFRA	Fragen an CP/M-Bau
CPBANT	Antworten von CP/M-Bau
CPBINFO	Infos von CP/M-Bau
MTXINFO	Infos von der MTX-Group
MTXFRA	Fragen an die MTX-Group
MTXANT	MTX-Group antwortet
HM	Hauptmenue

16:01/MI-MOE)&gt;SO SIEHT ES SCHON BESSER AUS

Befehl 'SO SIEHT ES SCHON BESSER AUS' nicht bekannt oder falsche Definierung !

Gemeinsschaftsfaecher (JETZT Aktiv) 'User'

15:50/MI-MOE)&gt;SOR

\*\*\*\*\* ANFORDERUNG EINES DIALOGES \*\*\*\*\*

Bitte warten oder mit ^X abbrechen !

\*\*\*\*\* DIALOGBEGIN MIT DEM SYSOP \*\*\*\*\*

Ins Menue kommst Du wieder mit X &lt;CR&gt; ,

wenn Du das Gespraech beenden willst.

Was kann ich fuer dich tun ?

ENDLICH

&gt;ICH HABE DICH NICHT GERUFEN, FREUE MICH ABER, DASS DU DICH MELDEST.

MEINE FUNKTIONSTE HAENGT SCHON WIEDER, SO MUSSTE ICH MAL WIEDER

DEN UMWEG UEBER SOR NEHMEN

&gt;ICH WOLLTE NUR 'MAL KURZ IN DIE BOX SCHAUEN, UM ZU SEHEN, OB MAN 'UNS'

SCHON SIEHT.

TSCHAU...

Gemeinsschaftsfaecher (JETZT Aktiv) 'User'

12:02/MI-MOE)&gt;LOGOFF

Dfue end.....

MailboxlisteMailboxliste

(Frank Bueschler)

Nummer	Name	Zeiten	Bemerkung
0201 - 23 73 99	EMS I	20-13h	
0201 -1 83 37 81	UNI-Essen	??-??h	Passwort notw.!
0202 - 8 81 16	MBALG	00-24h	
0202 - 46 36 78	Ronsdorfer Mailbox	00-24h	
0202 - 55 93 50	Toelleturm	00-24h	
0202 - 66 64 03	MABP	00-24h	
0203 - 40 76 89	CCD 1	18-06h	
0203 - 46 79 39	CCD 2	18-06h	
0203 - 55 82 68	Compi Box	19-09h	Atari-News
0203 - 58 88 22	D.D.D.	22-05h	
0203 - 59 61 46	MMS 2	20-06h	
0203 - 78 71 11	MMS 1	15-10h	
02051- 8 14 11	CSV	00-24h	
02054- 23 45	RCB	17-06h	
0208 - 47 17 14	MCC	18-07h	
0208 - 47 25 99	MHT	19-06h	
0209 - 27 16 66	Vollrath	00-24h	IBM-Infos
02101- 27 43 37	Neusser Mailbox	00-24h	IBM-Infos
02102- 6 80 04	RTC	17-07h	
02102- 47 54 00	RATEV	00-24h	
0211 - 27 47 72	Ultrasoft	20-02h	Atari-Userclub
0211 - 32 45 17	EDV	00-24h	
0211 - 32 82 49	RCPM	??-??h	
0211 - 34 00 71	Data Becker	00-24h	Werbe-Box
0211 - 43 53 93	P-S-I	15-22h	Sonntags ab 11h
0211 - 45 20 52	I.P.Sharp	??-??h	Userantrag
0211 - 61 37 99	L.I.S.	14-22h	
0211 - 62 23 50	Nanco	18-06h	
0211 - 68 60 15	BHD	00-24h	
0211 -3 11 39 04	UNI-Duesseldorf	18-08h	Passwort notw.!
0211 -5 04 78 65	Epson	00-24h	Werbe-Box
0212 - 4 75 11	Solinger Mailbox	18-15h	
02129- 41 10	Boston Mail	08-24h	Nur SA&SO online!
02151- 6 94 93	Datalogic	00-24h	
02151- 30 26 42	KM	20-06h	
02151- 47 65 67	KWCS/BDVI	00-24h	
02151- 56 02 43	CMB-Box Krefeld	20-06h	
02151- 56 32 57	Computer Artists KR	20-06h	
02151- 70 02 53	Lion-Box	22-06h	
02151- 80 13 39	K.I.S.	17-03h	
02152- 51 02 94	Mailbox Kempen	14-02h	
02154- 77 37	W.I.S.	22-07h	
02161- 20 09 28	SYMIC	00-24h	
02161- 58 35 57	Die Kiste	00-24h	
02161- 64 02 76	Shadow	18-24h	Atari 520'er News
02162- 5 84 57	Johnny Walker	18-06h	SA&SO 24h Online
02162- 8 06 79	Viersener Box	20-06h	
02171- 3 35 97	PIZ	00-24h	
02174- 33 71	BCO	00-24h	
02202- 5 00 33	CCB	00-24h	
02203- 3 29 21	SVDG	21-08h	Nur FR-MO online!
02203- 3 44 56	INFSYS-Koeln-Porz	24-10h	Comp. Artists
02203- 3 78 62	MASCH	15-07h	Atari User Group
02204- 5 70 25	CAC-Vax Box	00-24h	C. Artists
02204- 5 70 50	CAC-Video Box	00-24h	C. Artists
02208- 7 29 66	CCK	00-24h	
0221 - 23 72 29	KUT	??-??h	
0221 - 37 10 76	WDR-KomCom	00-24h	Multiuser Sys.!
0221 - 39 49 76	PMS	00-24h	
0221 - 51 26 40	HBK	00-24h	C. Artists
0221 - 51 46 97	K.I.S.	00-24h	
0221 - 54 17 62	HAF	22-17h	C. Artists

Mailboxliste

0221 -	55	83	36	BIT DSCHUNDEL	00-24h	C. Artists
0221 -	84	13	70	GHOSS CAC	00-24h	C. Artists
0221 -1	61	62	84	Saturn	00-24h	
02236-	6	33	71	Magic Mountain	00-24h	Schneider
02304-	4	47	70	SIS	00-24h	
0231 -	17	94	14	DMB 1	00-24h	
0234 -	50	01	73	B.M.S.	22-05h	
0234 -7	00	40	23	UNI-Bochum	??-??h	Passwort notw.!
02364-	1	38	26	H.I.D.	00-24h	
02365-	6	43	66	Marler Box	00-24h	
02373-	6	68	77	Ueding	17-09h	
02374-	1	34	20	MMB	00-24h	
02383-	5	08	66	IGS	00-24h	
02407-		83	26	HALU 1	21-03h	
0241 -	8	10	81	TH Aachen	00-24h	Passwort notw.!
0241 -	8	40	53	RBBS Aachen	00-24h	
0241 -	51	20	62	HALU 2	15-19h	
0241 -	87	05	55	AIS	18-09h	
0251 -	52	27	90	Maus	00-24h	
0251 -	61	90	54	Mickey	18-06h	
02521-		42	15	BIT	00-24h	
02594-	8	59	91	Data-Service	00-24h	
0261 -	2	44	80	SB-Koblenz	18-22h	
0261 -	4	10	81	WHFue	18-06h	
02841-	3	34	76	M.K.A.	21-04h	FR-SO 19-10h
02841-	5	73	25	M.U.B.	22-06h	
02841-	6	41	48	Moerser Mail	19-23h	
030 -	73	49	86	Titan 128	??-??h	Hardware-infos
030 -2	11	83	90	COMPAC 64 I	00-24h	
030 -3	05	26	35	Berliner Mailbox	18-08h	
030 -3	12	99	02	COMIN	20-06h	
030 -3	14	77	30	UNI-Berlin	??-??h	Passwort notw.!
030 -3	23	33	84	USSB	00-16h	C. Artists
030 -3	71	34	04	N.I.S.	20-05h	
030 -3	95	67	98	B.I.S.	17-22h	
030 -4	14	40	68	Gerb-Net	00-24h	
030 -4	53	93	33	TSCB	00-24h	
030 -4	92	57	84	Atari Box	18-22h	
030 -4	92	66	43	Telemail	00-24h	
030 -4	94	52	28	BWCC	20-06h	
030 -6	05	49	59	Friesland 64	00-24h	
030 -6	11	85	02	CIZ	00-24h	
030 -6	81	86	79	IBB	00-24h	
030 -6	87	05	05	ERGO	18-08h	
030 -7	46	68	50	Bert	15-01h	
030 -7	85	05	00	Tele Text	18-06h	
030 -8	02	42	28	Compac 64 II	00-24h	
040 -	27	87	14	C.C.H.	00-24h	
040 -	31	28	80	GGM	20-12h	
040 -	37	38	65	AUGE-Hamburg	00-24h	
040 -	57	35	16	ENIAC 2	22-06h	
040 -	81	61	32	VMSR	00-24h	
040 -2	51	23	71	Master-Control-System	00-24h	COMPAQ XT
040 -2	99	34	61	M.B.S.	00-24h	
040 -4	91	61	17	H.I.S	20-06h	
040 -5	27	70	16	Tornado	00-24h	
040 -5	29	20	10	Ghost-Box	00-24h	
040 -5	38	32	16	T.M.C.	00-24h	
040 -5	59	31	29	VMS	00-24h	
040 -6	03	76	99	S-S-B	00-24h	
040 -6	32	35	17	C.L.I.N.C.H.	00-24h	
040 -6	77	11	14	SNOBSOFT	00-24h	
040 -6	78	87	83	H.O.M.	00-24h	reinhacken!
040 -6	93	66	57	M.A.G.	20-06h	
040 -7	54	05	98	USC=	00-24h	
040 -8	80	23	83	R-A-M	00-24h	
04106-	45	15		ENIAC 1	22-06h	

Mailboxliste

04121-	42	86	67	BAM 1000	18-07h	
04131-	4	14	86	BBB	22-24h	
04141-	8	28	71	SMC Stade	19-22h	
0421 -	42	51	93	BMS	00-24h	
0421 -	59	21	64	CIA	18-06h	
0431 -8	80	45	56	UNI-Kiel	??-??h	Passwort notw.!
0431 -8	80	45	64	UNI-Kiel	??-??h	Passwort notw.!
04331-	2	90	92	R.O.M.	20-24h	
04421-	2	72	04	W.M.B	21-06h	
04431-		27	31	W.I.S.	19-22h	
0451 -	49	85	13	L.U.B.	??????	
0461 -	1	23	02	ADE II	00-24h	
0461 -	5	24	57	ADE I	18-03h	
0461 -	9	37	27	W.I.S.	00-24h	
0461 -	9	38	23	AMS	00-24h	
04683-		5	54	Comal Mail	21-02h	
0471 -	5	23	29	SKF	22-24h	
04841-		18	81	TINE	00-24h	
05063-	4	94	33	Martini	00-24h	
0511 -	40	10	24	AGMCC I	??-??h	
0511 -	40	10	25	AGMCC II	??-??h	
0511 -2	10	60	11	RRZN	??-??h	Passwort notw.!
05121-	4	57	92	Aquila	20-08h	
05151-	6	36	87	Hameln CC	00-24h	
05131-	9	52	19	Future World	??-??h	
05209-		46	83	Teuto-Box	00-24h	
0531 -	69	13	13	Bird	??-??h	
0531 -	89	21	40	Point	20-07h	
0551 -	2	18	43	UNI-Goettingen	??-??h	Passwort notw.!
0551 -	2	32	77	UNI-Goettingen	??-??h	Passwort notw.!
0561 -	40	50	90	DISCUS	16-24h	
0561 -	49	86	69	DARC	00-24h	
05693-		2	32	N.Hessen Mail	??-??h	
05931-	1	89	48	CompuCamp	00-24h	
06081-		96	77	Taunus-Mailbox	15-07h	
06102-	1	73	28	Phanter-Box	00-24h	z.Z. im Urlaub
06121-		34	09	HZD	00-24h	
06126-	5	24	92	IHB	??-??h	
06128-	7	34	98	Spy-Box	00-24h	
06136-	8	78	87	D.A.T.O.S.	19-23h	
06136-	8	84	69	C.O.P.I.S.	19-23h	
06151-	78	41	58	INF-SYS-Darmstadt	07-19h	FR24h SA-SO 00-19h
06154-	5	14	33	Decates	00-24h	Gebuehrenpflichtig!
06174-		53	55	KFC Informationssystem	00-24h	
06181-	4	88	84	Otis	00-24h	
06187-	2	58	28	Thor	00-24h	
06192-		79	60	CVCM	18-06h	
06203-	4	54	96	RNI	00-24h	
0621 -	1	23	02	Spima	00-24h	
0621 -	41	30	91	Telebox der DBP	00-24h	Posteigene Box!
0621 -	68	27	22	Mannheimer Mailbox	??-??h	
06227-	5	96	51	Dave Kert	??-??h	Passwort notw.!
0631 -	5	72	63	MB-Kaiserslautern	18-06h	Partnerver.!
06305-		82	81	Radio Peking	??-??h	
06322-		38	14	Beppos's Box	22-06h	
06352-		59	61	Donnersberg	19-24h	
06403-	7	28	61	Freak-Box	21-06h	
06421-	48	19	72	HASS	17-06h	
06432-		73	84	PC Billboard	00-24h	
06434-		62	91	CCCC	22-06h	
0641 -	7	60	32	UNI-Giessen	??-??h	Passwort notw.!
06808-	8	15	36	Fraego's Box	00-24h	
06826-		22	34	Hobby-C I	15-19h	
06826-		63	44	Hobby-C II	19-23h	
06854-		89	67	BMS	20-06h	

Mailboxliste

069	-	46	67	83	Control Data	??-??h	
069	-	49	42	01	A.U.G.E.	19-06h	Apple
069	-	55	78	44	UNI-Frankfurt a.M.	??-??h	Passwort notw.!
069	-	72	65	27	F.I.B.	??-??h	
069	-	81	67	87	Tecos	00-24h	
069	-4	30	82	13	F.I.Z.	??-??h	Passwort notw.!
069	-6	63	81	91	COMPBO	00-24h	Commodore
069	-6	68	72	57	GID	??-??h	Passwort notw.!
07024-		5	36	50	FBW	00-24h	
07031-		2	61	66	Elias	00-24h	
07042-		2	42	05	Vaihinger Box	??-??h	
0711	-	24	49	83	ALS Telebox	15-09h	
0711	-	46	10	32	PFM	19-06h	
0711	-	51	90	08	Norsak	00-24h	
0711	-	76	74	62	MCP	16-08h	
0711	-	83	76	86	Hyper-Box	00-24h	
07131-		7	81	32	Compare	??-??h	
07191-		6	27	53	Coloss	21-06h	
07191-		8	57	84	Sputnik	21-06h	
07191-		8	61	63	Infonet	00-24h	
0721	-	6	04	51	Karla Rechenzentrum	00-24h	
0721	-	6	04	52	Karlsruhe	??-??h	Passwort notw.!
0721	-	55	64	68	M.B.S.	??-??h	
0721	-	68	50	10	M.C.S.-Karlsruhe	00-24h	
07232-		8	12	78	E.I.S.	00-24h	
07247-		82	45	08	Inka KFZ	??-??h	Passwort notw.!
07247-		82	45	68	Inka	??-??h	Passwort notw.!
07452-		6	56	04	Schwarzwald-Info	??-??h	
0761	-	8	67	05	FLUG	20-06h	
0791	-	4	29	19	MicroSoft	??-??h	
08024-		68	51		Digital-Info	00-24h	Hofacker-Verlag
08031-		3	20	15	MKV	18-08h	
08151-		1	39	22	MVS	00-24h	
0831	-	6	93	30	G.E.S.	00-24h	
0841	-	5	58	87	SCHINFO-Box	18-05h	
08544-		6	57		L.I.S.A.	00-24h	
089	-	15	67	91	M.R.Box	18-21h	
089	-	39	22	89	Hitech Jr.	00-24h	
089	-	47	57	79	P.M.C.	18-06h	P.M. Werbe-Box
089	-	55	73	18	C.F.C.	00-24h	
089	-	55	76	38	HCM	00-24h	
089	-	59	64	22	Tedas I	00-24h	Francis Verlag
089	-	59	84	23	Tedas II	00-24h	Francis Verlag
089	-	83	12	88	Lauche & Maas	00-24h	
089	-	83	70	23	HDS-Mailbox	00-24h	
089	-	85	30	16	Knabber-Box	19-24h	
089	-3	61	45	26	IRATA	22-07h	Atari Club-Box
089	-4	30	18	13	VOX Mailbox	19-24h	
089	-4	48	83	62	ABAKUS	20-24h	
089	-6	09	19	20	Sitel	18-06h	
089	-6	41	48	79	CBBS McTosh	19-07h	
089	-7	93	13	32	Phoenix	00-24h	Auch Datex-P!
089	-8	12	03	38	A.C.M.	00-24h	
089	-8	54	54	02	Pro Box	00-24h	
0911	-	33	00	39	Mailbox System	00-24h	
0911	-	33	49	27	UBN	00-24h	
0911	-	57	41	60	Smurf-o-Box	00-24h	
09120-		61	03		L.M.S.	19-22h	
0921	-	6	46	87	Bayreuth Box	20-00h	
09281-		92	35	53	Magnum's Box	19-06h	

Soweit also mal wieder eine halbwegs "aktuelle" Mailboxliste, die ich aus der M.C.S. Mailbox entnommen habe. Allerdings auch diesmal wieder ohne jegliche Gewähr, da weder ich, noch andere Mitglieder die Kohle haben, all' diese Nummern durchzutesten.

Software: Neuheiten**MONI 2.00** (Olaf Krumnow, 2000)

MONI ist ab sofort in einer neuen, stark erweiterten Version erhältlich. Nicht weniger als 7 neue Kommandos sind implementiert worden, u.a. die schon mehrmals gewünschte direkte Assemblereingabe und neue Suchmöglichkeiten. Nachdem die alte MONI-Version doch ihre Schwierigkeiten mit RAM3 hatte, arbeitet MONI 2.00 ohne Probleme mit RAM3 zusammen. (VDEB tuts da ja gar nicht mehr). Aber MONI 2.00 läuft natürlich auch ohne RAM3, kann also von allen benutzt werden.

**Was kann MONI?**

MONI kann sehr viel! Das mag im ersten Augenblick eingebildet klingen, aber diese Bemerkung ist als Vergleich zu DDT oder VDEB gemeint, die ja jeder besitzt. Wer mal mit DDT oder VDEB gearbeitet hat, wird den Unterschied sehr schnell merken. Alle Eigenschaften dieser beiden Debugger sind in MONI 2.00 mit einer Ausnahme (Zugriff auf das VRAM) implementiert. Deswegen sollen hier nur die Erweiterungen beschrieben werden:

- MONI kann Dateien schreiben. Das klingt wie eine Lappalie, aber rechne doch mal aus, welchen Wert Du bei SAVE xx NAME.COM für xx angeben mußt, wenn Du von 100H bis 4346H abspeichern willst. Bei MONI brauchst Du nur die Adresse zu nennen, in der das letzte abzuspeichernde Byte steht, der Rest kommt von allein. Außerdem brauchst Du MONI zum Speichern nicht zu verlassen, wie es bei DDT und VDEB der Fall war.
- MONI kann suchen. Das ist besonders hilfreich, wenn in einem selbstgeschriebenen Programm eine bestimmte Stelle gefunden werden soll. Dann kennt man die Opcodes, aber nicht die Adresse. Auch wenn man eine Patchanleitung für (z.B.) TURBO 2.0 hat, jedoch TURBO 3.0 patchen will, muß man erst die richtige Stelle suchen. Beim Suchen können auch Platzhalter angegeben werden und Bitmuster gesucht werden. Ferner ist es möglich, Strings zu suchen.
- MONI bedient den Drucker. Alle Listings und Traceläufe können auf dem Drucker protokolliert werden. Der Druckerstatus kann direkt im Programm geändert werden.
- MONI kann den erweiterten Z80 Befehlssatz. Beschrieben ist dieser in MC 1/1982 bzw. Info 9-61. Sowohl beim Assemblieren als auch beim Disassemblieren können diese neuen Befehle benutzt werden.
- MONI merkt sich Adressen und Filenamen. Diese können bei Bedarf mit einem Tastendruck wiederholt benutzt werden, z.B. bei sequentiellm Suchen mit anschließender Korrektur.
- MONI arbeitet interruptfrei. Das heißt, er arbeitet mit RAM3 problemlos zusammen (ganz im Gegensatz zu VDEB, das unter RAM3 nicht mehr zu gebrauchen ist), ist jedoch nicht auf RAM3 angewiesen.
- MONI arbeitet mit jedem einigermaßen sinnvoll belegten Zehnerblock zusammen, ob nun original, CP/M-Patch oder als Funktionstastenblock.
- MONI verwaltet bis zu 10 Breakpoints. Damit können problemlos Programmverzweigungen auch größeren Ausmaßes beherrscht werden. Zusätzlich dazu und unabhängig davon kann der normale von VDEB gewohnte Programmlauf gewählt werden (GO Von Bis).
- MONI spricht deutsch. Alle Anweisungen, Aufforderungen und Meldungen sind ausnahmslos in deutsch.
- MONI hat eine ausführliche deutsche Dokumentation. Jedes Kommando von MONI wird ausgiebig und mit Anwendungsbeispielen beschrieben. Dem Anfänger werden Tips zur Nutzung von MONI gegeben.
- und noch einiges mehr...

Software: Neuheiten**Was kostet MONI?**

Der Preis von MONI 2.00 beträgt DM 20 excl. Datenträger und P&V, für Besitzer einer Version 1.xx (**soweit bei mir registriert!!!**) ist bis zum Erscheinen des Infos 14 das Update für DM 5 zu haben (natürlich nicht Diskette, P&V, die wie üblich mit DM 8 zu veranschlagen sind). Bestellungen bitte nur direkt an mich. Auslieferung nur bei beigelegtem V-Scheck oder gegen Vorauszahlung auf das Postgiroamt Hamburg (BLZ 20010020) Konto-Nr. 566761-201 (**Absender nicht vergessen !!**).

**CASSETTE.COM** (Rudolf Gmeinwieser, 8443)

Mit CASSETTE kann man endlich seinen Musicassetten und Datencassetten zu einer anständigen Hülle verhelfen.

Die Inhaltsangabe besteht aus Seite A und B mit insgesamt 18 Zeilen je Seite und zwei Zeilen für den Cassetten-Titel. Die Inhaltsangaben werden mit einem Full-Screen-Editor erstellt und können danach abgespeichert, ausgedruckt (EPSON oder Kompatible) oder wieder geladen werden. Der Editor kann auch in eigene Pascal-Programme eingebaut werden.

Das Programm ist in TURBO-PASCAL geschrieben und bedient sich der WINDOW-Routinen von Olaf Krumnow, die sich auf der Diskette CLUB.003 befinden. Diskette wird nur als 40-Spur, TYP 03 geliefert.

Disketten-Inhalt: CASSETTE.COM, CASSETTE.PAS (Source), CASSETTE.DOC.

Preis: DM 15.-

Software von Volker Griener

(Jens Ebert): Ich meine, auch schlechte Erfahrungen mit Software, die von Mitgliedern des Clubs erstellt **und verkauft** wird, sollten geäußert werden dürfen. Durch die Kurzbeschreibung im Info 11 war ich auf das NLQ-Programm von Volker Griener neugierig geworden, welches "Schönschrift" auf dem DMX ermöglichen sollte. Der Preis von 10 DM erschien mir zwar etwas hoch (z.B. verglichen mit dem Preis des RAM-Programms!), bestellt habe ich aber trotzdem. Erhalten habe ich dann eine Diskette mit einigen Dateien, von denen einige durch ihre Endung ".DOC" Hilfe bei der Benutzung versprochen.

Es war aber einfach nicht zu ersehen, wie man nun wirklich einen Text auf den Drucker bekommt! Ein DOC erklärte die Benutzung des Zeicheneditors unter "FDXB 40" ausreichend gut. Aber sonst? Der DOC-File für das Programm "NLQ-DRU.DIS" enthielt viel "Internes", aber **eben nicht das Entscheidende: Das Programm läuft unter FDXB!!** Die Endung ".DIS" läßt das nicht vermuten, in der Beschreibung im Info 11 steht es ebensowenig. Ich hatte ganz klar ein ".COM"-Programm erwartet und keines das unter Basic läuft! Das zur Dokumentation, nun zum Gebrauch. Man braucht also immer auch FDXB, aber das Programm kümmert sich auch nicht um Laufwerke: Es ließt **immer** die Directory von LW B: und bietet nur diese Dateien zum Druck an (selbst wenn die NLQ-Diskette in LW C: liegt). Drückt es dann, läßt es sich auch nicht mehr stoppen und zu allem Überfluß findet man im Ausdruck seine Dotkommandos wieder, wie z.B. ".op". Weitere Versuche habe ich dann eingestellt.

Auf der Diskette befindet sich ein Programm "NLQ2.DIS", welches überhaupt nicht erwähnt wird. Auf meinen 2. Brief antwortete Volker dann endlich mit einer Postkarte, die das FDXB-Geheimnis lüftet und verrät, daß NLQ2.DIS eine "etwas erweiterte Version von NLQ-DRU.DIS" (Zitat) ist, ohne daß die Form der Erweiterung verraten wird.

**Fazit:** Das Programm ist unausgereift, schlecht dokumentiert und sein Geld nicht wert.

Leserbriefe**Lobeshymne :** (Cristof Niehörster)

Ich möchte an die Clubredaktion ein sehr großes Lob aufgrund der Fülle und der Bandbreite der Informationen, die in den Info's stehen, aussprechen. Ich kann die Info's zwar erst ab Nr.10 beurteilen, glaube aber, daß dies eine weitverbreitete Meinung ist.

Gut finden würde ich es nur noch, wenn etwas mehr Hardware-Stories veröffentlicht werden würden (das richtet sich jedoch wieder mehr an die überaktiven Hardware-Freaks, zu denen ich nicht gerade gehöre).

**Ortogra wie ?**

(Michael Möwe)

Herberts Beitrag zum Thema Ortograwie hat mir sehr gefallen, endlich weiß ich, wie ich meine Artiggel fürs Info schreiben musz .

Vielen Dank Hörbat.

**Niveau**

(Michael Möwe)

Es ist immer wieder das gleiche Problem, vielen Lesern ist das Niveau des Infos zu hoch. Die Leute, die die Beiträge zum Info schreiben, wissen dies zwar schon, aber nicht wo die Probleme liegen. Ein weiteres Manko ist auch, daß nur (?) die FDX-Besitzer Beiträge zum Info schreiben und naturgemäß anders gelagerte Sorgen als die nicht-FDX-Besitzer haben. Ich muß gestehen, daß ich mit Bernd Preusings Diskettenstory auch zuerst nichts anfangen konnte, aber nach einer gewissen Zeit war mein Interesse da und ich hatte wenigstens die Möglichkeit mir das alles noch einmal in Ruhe durchzulesen (die Sache mit den 'verdeckten' Sektoren habe ich unter anderem auch durch die Disketten-Story in den Griff bekommen, hätte ich gewußt, daß es jemanden interessiert, wäre ein Programm in Turbo schon früher dagewesen). Daraus läßt sich ersehen, daß das Niveau zwar für den Einzelnen momentan zu hoch sein kann, aber für später hat man ein Nachschlagewerk von unschätzbarem Wert; besonders jetzt, wo ein vernünftiges Inhaltsverzeichnis in Arbeit ist.

P.S.: Auf Fragen der nicht so hoch fliegenden Leute ist, soweit ich weiß, noch immer mit bestem Wissen und Gewissen geantwortet worden, also können auch Fragen das Info mitgestalten.

Guido Ehlers  
Mainzerstr. 61  
5401 Rhens

den 6. 4. '86

Tach,

ich gehöre auch zu denen, die sich darauf beschränken erstmal 'nen paar Programme abzutippen und das nachzumachen, was die Wissenden ( H. H., F. B. etc. ) vormachen. Daher bin ich froh das es den Club gibt, und verschlinge jedes Info, die einfach ein Lob verdienen.

Ich bin mir sicher, daß auch Ihr mit der Arbeit von den Organisatoren und Leitern des Clubs zufrieden seit. Weil das so ist, möchte ich einen Vorschlag machen.

Als der Club damals gegründet wurde, hat man bestimmt nicht damit gerechnet, das er in relativ kurzer Zeit derartige Dimensionen annehmen würde. Mit mittlerweile fast 400 Mitgliedern, je mehr desto lieber, ist aber auch der Aufwand in der Verwaltung gestiegen. Damit meine ich nicht das Verwalten der einzelnen Konten etc. . Man stelle sich nur mal vor, wie das ist, wenn mehr als 300 Info's in der Druckerei abgeholt werden, in Briefumschläge gepackt und zur Post gebracht werden müssen. Man kann sich wohl leicht ausrechnen wieviel Arbeit das ist. Aber es sind ja nicht die Info's allein, obwohl die ja auch erstmal geschrieben sein wollen, da sind ja noch der Telefondienst und all die anderen Sachen, die wir nicht mitkriegen.

Ich bin daher der Meinung man sollte, entgegen der bisherigen Regelung, denen die Arbeit leisten auch eine finanzielle Entschädigung zukommen lassen. Wie das möglicherweise geregelt werden kann, darüber sollte man Reden, wenn klar ist, was Ihr davon haltet.

Dies soll ein Anstoß sein, ich hoffe, daß recht viele sich dazu äußern. Das einige nicht meiner Meinung sein werden, ist mir klar. Dennoch bin ich der Meinung, das man darüber einmal reden sollte.

Soviel für jetzt aus Rhens, tschö,

