

# MTX *User-Club Deutschland*

Info 15  
01. 08. 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 Grevensbleck 24 3005 Hemmingen 1 (0511) 41 78 77	Detlef Harms Feldstr. 26 4902 Bad Salzufflen (05222) 84 0 82	Thomas Pflaum Leipziger Platz 1 8500 Nürnberg 20 (0911) 51 35 21
---	---	--	---	---

**Telefonzeiten**

Herbert Herberg: Do 16 - 22 Uhr, Sa 13 - 16 Uhr

Inhaltsverzeichnis

**C L U B:**

Dies und Das von Herbert Herberg	Seite 1
Wer tut Was	Seite 2
Antworten	Seite 2
Kleinanzeigen	Seite 3
Assembler-Kurs	Seite 4
Club-Treffen	Seite 4

**BASIC:**

Apfelmännchen von Thomas Pflaum	Seite 5
VAL und richtige Bool'sche Ausdrücke eingebaut	Seite 9
Hardcopy für Seikosha	Seite 11

**BIOS:**

Zeichenumsetzer für merkwürdige Typenradmaschinen	Seite 12
---	----------

**CP/M:**

Beschreibung DDTZ, der dem DDT ähnlich ist!	Seite 15
Formate mittels COMFAT und FMAT - statt RAM 3.4	Seite 28

**TURBO:**

GotoXY, Write schneller	Seite 18
Error 93, 96	Seite 18
Hallo Hacker!	Seite 18
Probleme mit Windows und RunTimeErrors	Seite 47

**SuperCalc:**

Der Kurs von Wolfgang Gieger	Seite 19
------------------------------	----------

**C:**

C versus PASCAL	Seite 25
Trigonometrische Funktionen	Seite 26

**MSDOS:**

MSDOS: hääää?	Seite 27
---------------	----------

**Hardware:**

CoProzessoren	Seite 31
SDX und Bdos Err Bad Sector	Seite 31
Schaltplan 2-Kanal A/D-Wandler	Seite 38
Schaltplan Baudraten-Generator / Verstärker	Seite 40
Programmierung des SerialInputOutput (SIO)	Seite 41

**Leserbriefe:**

Softwareversand	Seite 48
-----------------	----------

Preis für dieses Info: DM 11.-

C L U B: Dies und Das von Herbert Herberg

Liebe MTX-lerinnen und MTX-ler!

Nochmals vielen Dank für Eure Mühe bei den **Emblemen**. Interessanterweise stimmten fast alle von Euch für die Vorschläge 2 und 9. Das Nummer 9 gewonnen hat ist sicherlich zu übersehen gewesen, aber da Emblem Nr. 2 knapp (3 Stimmen) drunter lag, finde ich auch dieser Vorschlag verdient eine Belohnung: 4 Public-Domain Disketten nach Wahl, die ich nach Erhalt des zugehörigen Druckprogrammes (bitte in maschinenlesbarer Form!) schicken werde.

Wie war das, Memotech **pleite** ?? Nun hier der Stand der Dinge:

1. Memotech Ltd, Whitney: pleite (kein Wunder bei SDX) und aufgekauft.
2. Memotech Computers Ltd heißt (genauer hieß) der neue Laden.
3. Auch pleite! Aber in Oxford ist eine Memotech-Vertretung um Garantiesprüchen genügen zu können.
4. In Rom, Italien gibt es nun Memotech International, die in den PC- und Grafik-Computermarkt einsteigen wollen.

Aber wozu brauchen wir Memotech ? Soft- und Hardware kriegen wir auch selbst auf die Beine, oder ? Wenn auch wir etwas Zeit brauchen, nicht immer alles perfekt ist, und auch billiger.

Und was wird aus denen, die eine **FDX** haben wollen ? Nun Euch kann ich beruhigen, da in unseren Kleinanzeigen einige stehen. Wer übrigens eine FDX von mir oder über mich bezieht, kann sicher sein, daß die Kiste von mir gründlich überholt ist, und nach dem Einschalten sofort bootet!

'**ECB-Bus** - oder Hardware ist doch nicht so einfach', das große Thema des Clubtreffens: Bei mir laufen auf dem ECB-Bus einige Karten. Aber einige machen Ärger. Dazu könnt Ihr etwas im nächsten *Info* (Nr. 16) lesen.

Im letzten **CHIP-Heft** war ein Artikel über Billig-Computer. Diese haben oft Probleme, da viele Steckkontakte mit verzinnten Kontakten ausgerüstet sind - die kennen wir ja zur Genüge, oder ? Da in CHIP keine Lösungen angeboten waren, diesen Mängeln beizukommen, habe ich die beiden im Club entstandenen Behebungs-Varianten der CHIP-Redaktion geschrieben: TUNER 600 und fest verlöten.

Ich bin übrigens an einer **SDX-Anlage** als Zweitgerät interessiert! Zum einen kann ich mich dann mit der SDX mal genauer befassen (Speichererweiterung, RAM3, ...??) und die Zweitanlage ist nicht so riesig. Ich habe hier eine FDX stehen, die ich gegen die SDX tauschen könnte. Das 80-Spur-Laufwerk der SDX könnte ich leicht in die FDX einbauen, damit der von Euch, der mit mir tauscht die alten Disketten weiterhin lesen kann. Über die Modalitäten und den Preis können wir reden/verhandeln.

Im nächsten Info werden wir wieder eine Bestellaktion starten. Dort werden wir Laufwerke, Disketten, Akustikkoppler, Farbbänder, u.s.w. anbieten. Das 'u.s.w.' könnt Ihr mitbestimmen. Bitte denkt schon jetzt mal über Eure Interessen/Finanzen/Wünsche nach, da wir dieses Mal nicht so lange Bedenkzeit geben wollen, damit sich die Sache nicht unnötig in die Länge zieht, und die Preise nur verändern. Wie das genau ablaufen wird steht auch im Info 16. Wenn Ihr aber jetzt schon wißt, was Euch interessiert schreibt uns bitte möglichst rasch, damit wir uns um günstige Quellen kümmern können.

Mit eXtra-gruß aus hamburg



C L U B: Wer tut Was / Antworten**Wer tut Was**

Allround	H. Herberg, C. Löhrmann, F. Bueschler
(FDX-)BASIC	A. Viebke, H. Oppmann, V. Griener, C. Romanazzi
CP/M System	B. Preusing, 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, H. Herberg
Reparatur	U. Hönisch, H. Herberg

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

**Antworten**Verstecktes Find & Replace (Michael Möwe, 2000)

Der Text von Hagen Wenzek stand etwas verstümmelt im Info, aber ich konnte dennoch daraus erlesen, was gemeint war. Suchen und ersetzen im Hintergrund, das geht schneller, das macht Spaß; die Frage ist nur noch, wie funktioniert es ?

Ganz einfach:

1. ^Q^A eingeben
2. zu suchende Zeichenfolge eingeben
3. neue Zeichenfolge eingeben
4. Suchparameter eingeben, mit <RETURN> abschließen
5. nach 4. die Space-Taste drücken (einmal kurz reicht) und schon sucht NewWord im Hintergrund und meldet sich erst wieder, wenn alles erledigt ist!!!!

Dazu die Suchparameter: (Herbert Herberg, 2000)

W = nur Ganze Worte, G = Ganzer Text, R = Rest des Textes,  
 N = Ersetze ohne zu fragen ob, U = ignoriere Groß/Kleinschreibung  
 B = Suche Rückwärts, ? = Fragezeichen im Suchstring sind Joker  
 Z.B.: Im Rest des Textes alle ersetzen: Suchparameter **RN**.

Für Christian Meervissen: (Claudio Romanazzi, 3070)

Der Zeiger auf den aktuellen Screen liegt in #ff5b + c!!!!  
 Im Übrigen frage ich mich, für was es die Spezialistenrubrik 'Wer tut Was' Info gibt. Da genügt doch ein kurzer Anruf!!!

VS 4 - Hohe Farbauflösung (Herbert Herberg, 2000)

Leider ist der VS 4 nicht gerade das NonPlusUltra. Die kleinste Fläche, die mit zwei verschiedenen Farben (Vordergrund und Hintergrund) versehen werden kann ist 1 Pixel hoch, aber 4 Pixel breit. Dazu müssen Vorder- und Hintergrundfarbe mit COLOUR 2, Hintergrund und COLOUR 3, Vordergrund gesetzt werden. Mit ATTR 2,1 (UnPlot) malt PLOT mit Hintergrundfarbe; nach ATTR 2,0 liefert PLOT die Vordergrundfarbe.

C L U B: KleinanzeigenK O N T A K T E

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

Herbert Herberg, Sonnenau 2, 2000 Hamburg 76, 040 - 2008704 hat Apple-Communication-Software: Software für Rechnerkopplung Computer mit einem Apple. Das sind zwei Disketten (1x MTX, 1x Apple), die ich ggf. verleihe, da ich die Apple nicht kopieren kann.

Herbert Herberg, Sonnenau 2, 2000 Hamburg 76, 040 - 2008704 ist interessiert eine FDX gegen eine SDX zu tauschen. Finanzielles ist Verhandlungssache!

V E R K A U F

Herbert Herberg, Sonnenau 2, 2000 Hamburg 76, 040 - 2008704 hat FDX und MTX mit/ohne Monitor, Drucker, ... zu verkaufen. Preis ist Verhandlungssache! **Ich habe diverse FDX-Stationen!!!** Ich habe auch verschiedene Laufwerke zu verkaufen (in Kommission).

Was ich weitergebe ist überprüft, FDX bootet dann einwandfrei!

Stefan Trucksäß, Zazenhäuser Str. 14, 7000 Stuttgart. 0711 - 8701251 verkauft FDX, MTX 500, 2. Laufwerk (TEAC), TP 200, DMX 80, Nw-Instal-Handbuch für DM 2000.-

Peter Restle, Strittmatterstr. 2, 7798 Pfullendorf verkauft MTX 512, FDX, TP 200, TURBO-Pascal, dBASE II für DM 1700.-

Reinhard Schletter, Kafkastr. 13, 4600 Dortmund 14, 0231 - 239927 verkauft MTX 512, RS 232, FDX, 2. Laufwerk, CP 80 Chinwa-Drucker für DM 2500.-

Andreas Meusch, Am Gonsenheimer Spieß 6 /EG, 6500 Mainz verkauft MTX 500, FDX, 2. Laufw., DMX 80, Software für DM 2200.-

05222 - 81678 verkauft MTX 512, FDX für DM 1500.-

Dieter Schwarz, Steinhauerstr. 5, 7558 Bischweier, 07222 - 47949 verkauft zwei EPSON SD 512 Laufwerke für jd DM 250.-

Wilfried Curthy, Schwindstr. 2, 8000 München 40, 089 - 257630 verkauft MTX 500, FDX, 2. Laufw., TP 200 für DM 1500.- und Drucker Panasonic KXP 1091 für DM 650.-

Oskar Karl, Gonsenheimer Str. 15, 6500 Mainz 21, 06131 - 476328 verkauft MTX 512, 8 Monate jung, div. Cassettenprogramme. Preis VHS. Infos komplett zum halben Preis, evtl. DMX 80 und TP 200.

Götz Neumann, Am Steinicht 22, 8630 Coburg, 09561 - 66887 verkauft Infos 1-14 für DM 80.- (incl. RDM-Unterlagen), MTX-Kompendium für DM 20.- (3. Auflage), Software: CLUB.001, BASIC-G/S/U.001, div. Spiele (Lunar, 4 Gewinnt, ..) für DM 30.- und Rodnay Zaks: Programmierung des Z80 für DM 30.-. Alle Preise incl. P&V.

A N K A U F

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

Rainer Tewes, Nordstr. 188, 4690 Herne 1, 02323 = 83520 sucht billige FDX (DM 850.-)

C L U B: Assembler-Kurs / Club-Treffen**Assembler-Kurs** (Kurt-Bernd Rohloff, Kafkastr. 14, 8000 München 83)

Wie ihr Euch sicher erinnert, wurde vor einiger Zeit im Info der Wunsch laut, einen Assemblerkurs zu veröffentlichen. Nach Abschluß der Vorarbeiten dazu zeichnet sich ab, daß dies mit erheblichem Aufwand verbunden ist. Für die Autoren und mich ist nun die Frage, ob sich dieser wirklich lohnt. Daher meine Bitte: Wer an einem Assemblerkurs ernsthaft interessiert ist, möge mir bis zum Erscheinen des nächsten Infos (am besten jedoch gleich) eine Postkarte schicken. Es genügt, wenn (außer dem Absender) lediglich "ASM-KURS" auf der Karte steht. Wenn ich deutlich weniger als 50 Karten bekomme, ist die Fortführung des Projektes sehr fraglich.

**Dies und das**

(Michael Möwe, 2000)

" Programmers are always surrounded by complexity; we cannot avoid it. If our basic tool, the language in which we design and code our programs, is also complicated, the language becomes part of the problem rather than part of its solution."

C.A.R. Hoare

Deutlicher kann man wohl kaum sagen, welchen Stellenwert man BASIC, FORTRAN oder ähnlichen Sprachen zuweisen muß.

**Anm.d.HH.** Hier die Übersetzung:

" Programmierer sind immer von Komplexität umgeben; das können wir nicht verhindern. Wenn unser elementarstes Werkzeug, die Sprache in der wir unsere Programme schreiben ebenfalls kompliziert ist, wird die Programmiersprache zu einem Teil des Problems anstelle ein Teil der Lösung."

**Kurz zum Clubtreffen**

(Herbert Herberg, 2000)

Leider kam das Treffen nur langsam in Schwung, da kaum Fragen in großer Runde gestellt wurden.

Ein zentrales Thema war der ECB-Bus, auf dem bei mir eine Grafik-Karte von EDIC-TA ihre ersten Gehversuche machte (Auflösung 1024 x 1024 Pixel). Aber leider läuft der ECB noch nicht 100%-ig. Wenn, dann erfährst Du es als erster!

Die Vorversion von RAM4 mit dem erweiterbaren KLICK-Menü war auch zu sehen! Ich kann zwar kaum warten, bis es fertig ist, aber Bernd sagt Weihnachten!

Das Wichtigste jedoch erscheint mir, daß wieder mal einige von uns sich trafen, wie dies ja auch anderswo im Club geschieht.

**EXIT PROGRAMM.**

Grafik: Apfelmännchen von Thomas Pflaum

Hallo Leute!

Nürnberg im Sommer 86

(ca.0.7 mittler 1/2W.Zeiten n.Tb)

Na wie läuft die Maschine heute?

Es ist 20:27 und sie hat seit 9 auf Arbeit gewartet, hochgekommen (auf RAM31) schon/erst beim 3 Reset (meist beim 1. oder 2. Lesen abgestützt) und warm.

Ja die RAM! Wer kennt sie nicht? Beim letzten Treffen in München war sogar eine Kombination von 2.5 M, zu sehen; nämlich: 512 auf 32K-Karte (Klub) als F; plus eine auf 2 M aufgerüstete FDX-RAM-Karte von Fa. Harter (für unter 900.- als G; Und alles unter Berndts hervorragender RAM 3.x-Treibersoftware incl. Klick.. Bei mir ist F; seit ein paar Tagen noch das zuverlässigste Laufwerk, und Angst fährt mit wenn eine Kopfbewegung erfolgt... Habe ich beim Einbau des Zeichensatz PROMs zuviel Staub aufbewirbelt? , falsche Stepratzen eingestellt? oder hinne Disketten? oder Wackelkontakte oder Hallos oder nicht bloß eines davon??

Aber sie läuft, schön warm geworden, und ich schreibe Euch (in Turbo) und hoffe, daß es Euren Kisten gut oder besser besser geht... Wenn nicht, ist die Zweitmaschine eine Alternative zum Umsteigen und meine hat im letzten Monat das 2-seitige Fractal gerechnet; sie ist ein Zweithänder (500er+2x40) zu 1400.- Inzwischen hab ich auch einen Koppler, also wer ein Fractal will... ein kleines läßt sich bei 300 Baud in lumpigen 23 min übertragen, das kostet 8.- Aber wer glaubt ich blockier mir die Leitung dafür, piepst wohl eher doppel-dumplex..

Mal sehen, ob es noch eine dritte oder vierte Anlage gibt, wenn Vobis den restlichen Memotech-Bestand wirklich demnächst verschleudert; mehr dazu und was wir als Club organisieren können von HaHa...

Das Fractal ist ein Apfelmännchen-Ausschnitt, entsprechend Map 36 ( Peitgen/86) nachgerechnet in 24 VS4-großen Files (a 50K). Gedruckt wurden nur Punkte mit  $32 < k < 64$  oder  $128 < k < kmax$ .

Um die in einem Byte gespeicherte Information auf einen Druckpunkt abzubilden sind Graustufen und Farben sehr hilfreich; beides taugt aber leider nicht für Photokopien.

TIP: Wer einen FX-80 hat, kann einen prima Farbdrucker draus machen, wenn er farbige Farbbänder verwendet. Bei Compana in Nürnberg gibt's immerhin blau und rotbraun; das ist schon ein sehr beeindruckender Unterschied!!

ACHTUNG: Wer Interesse an einer Sammelbestellung von FX-80 Farbbändern in sinnvollen Farben (gelb/cyan/magenta..) hat soll sich bei mir melden!! Unter 30 Stück wird keine Farbe geliefert; das Stück käme je nach Beteiligung auf 15.- bis 25.- Schreibt mir bitte, ob Ihr Euch an einem 100.-Paket beteiligen würdet.

Die Fractale in den bisherigen Infos waren Ausschnitte aus Julia-Mengen von  $x*x+c$ , also die Darstellung der Formel in der x-Ebene mit  $x_{n+1}=x_n*x_n+c_0$ ; dagegen erhält man die bekannte Apfelmännchenfigur (nach ihren Entdecker (März 1980) Mandelbrot-Menge M genannt) durch die Entwicklung in der c-Ebene mit  $c_{n+1}=x_0*x_0+c_n$ . Diese Bilder sind stets Ausschnitte aus der Grundfigur, die durch Variation von  $c_0$  nur geringfügig verzerrt wird.

Die bei variierendem  $c_0$  entstehenden Julia-Mengen sind hingegen höchst vielfältige punktsymmetrische Gebilde. Bei ihnen sind denn auch durch tiefere Ausschnitte keine neuen Strukturen zu finden (finde ich).

Grafik: Apfelmännchen von Thomas Pflaum

Wer mehr wissen will findet in

H.-D. Peitgen , P.H. Richter  
 The Beauty Of Fractals 1986, Springer Verlag

ein Standardwerk mit prächtigen Farbbildern. Die allein sind die 80.- wert; die Texte verlangen vom Leser Mathematik- und Englischkenntnisse. Besser als der erste Eindruck bei mir ist aber wohl auch

K.-H. Becker, M. Dörfler  
 Coputergrafische Experimente mit Pascal, 1986 Vieweg-Verlag

Für 42.- erhält man zwar nur popelige Grafiken und der Text ist betulich bis wirr, aber es ist insgesamt recht vielseitig und Frucht echten Fractalismus. So deckt es auch andere Fractalsysteme ab, worunter eine Klasse von Phänomenen zu verstehen ist, die 1. selbstähnlich, 2. rekursiv und 3. gebrochen-dimensional sind. Vielleicht folgt eines aus den anderen aber welches?

Fractale Effekte können bei Zufallsfunktionen auftreten oder bei einfachen geometrischen Vorgängen wie der z.B. der Schneeflocke, die entsteht wenn man jede Seite eines Ur- und jeden folgenden Dreiecks durch ein mittendrauf abwechselnd und innen und außen liegendes Dreieck mit 1/3 Seitenlänge ersetzt.

Auch Physiker finden sie in Strömungen und Magnetfedern und die restliche Natur schaut in der Computersimulation am echtesten aus, wenn sie mit einprogrammiert wurden.

Das Bild wurde auf vier Seiten gedruckt und je zwei zusammen verkleinert. Bei der Datenmenge läuft da ohne F; nichts mehr, zumal das Druckprogramm die Dateien am liebsten paarweise und von hinten verarbeitet...

Wer nachrechnen will hier die technischen Daten:

$x = -0.74985... - 0.74410 + 0.10568*i... 0.10928*i$

$c = 0 + 0*i$  (wie üblich) Auflösung: 1536\*786 Rechentiefe  $k_{max} = 1000$

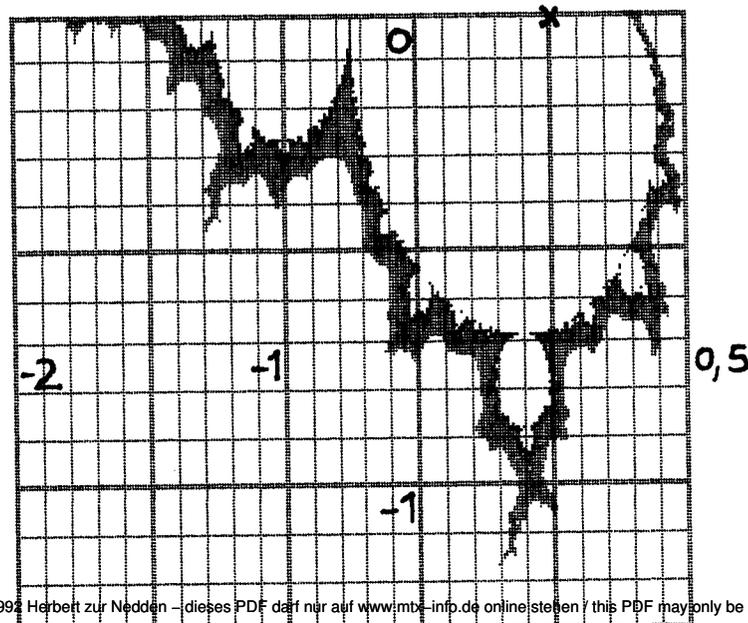
Zum Speichern wird das Ergebnis auf ein Byte herunterdividiert, aber es sind immer noch vier Disketten!

So, es ist 22:55 und alle drei Zwischensicherungen sind heil geschrieben worden; na also Alter!

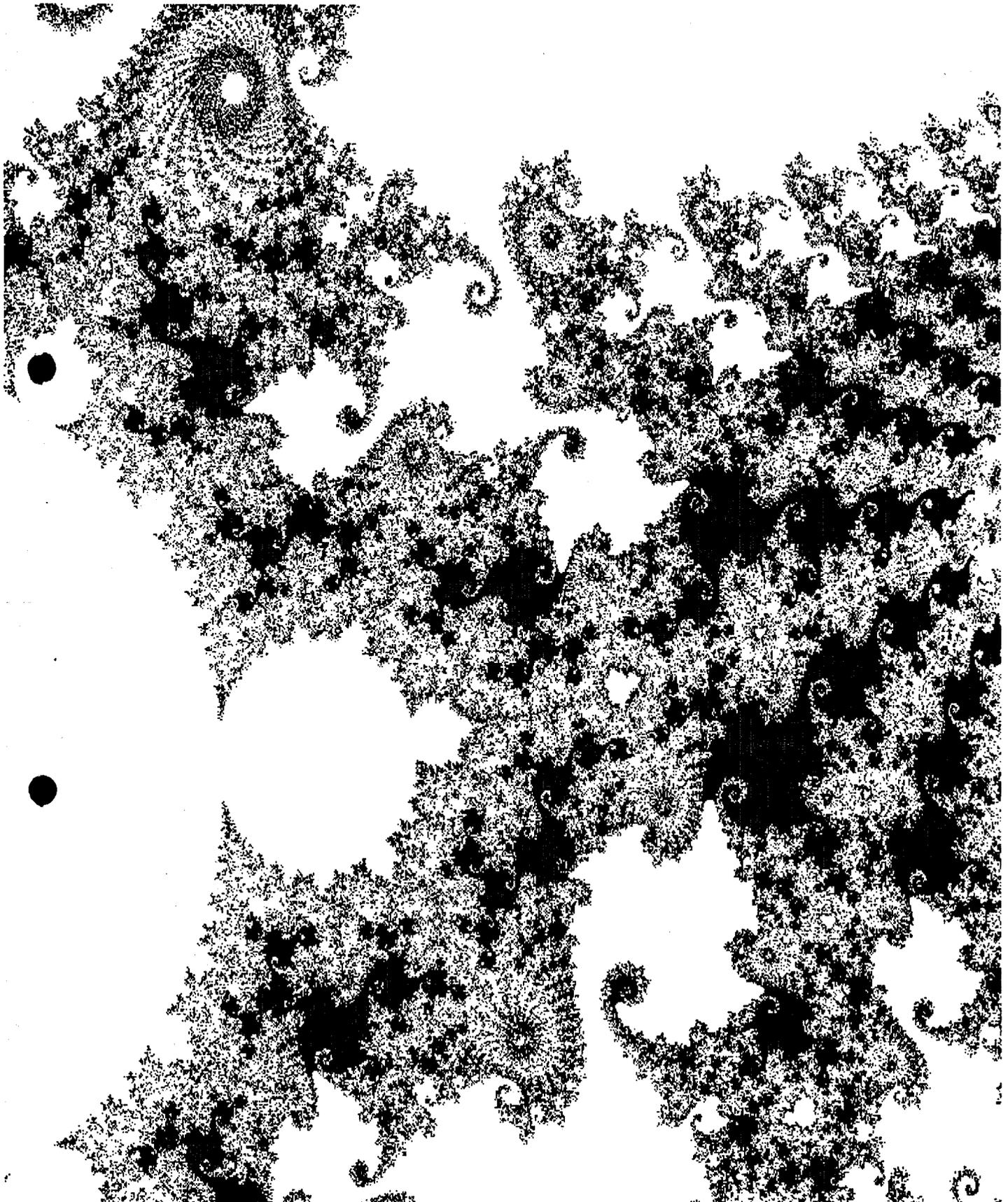
Macht 's gut, beim nächsten Mal gibt's dann ein fractales Turbo-Musikprojekt.

Thomas Pflaum

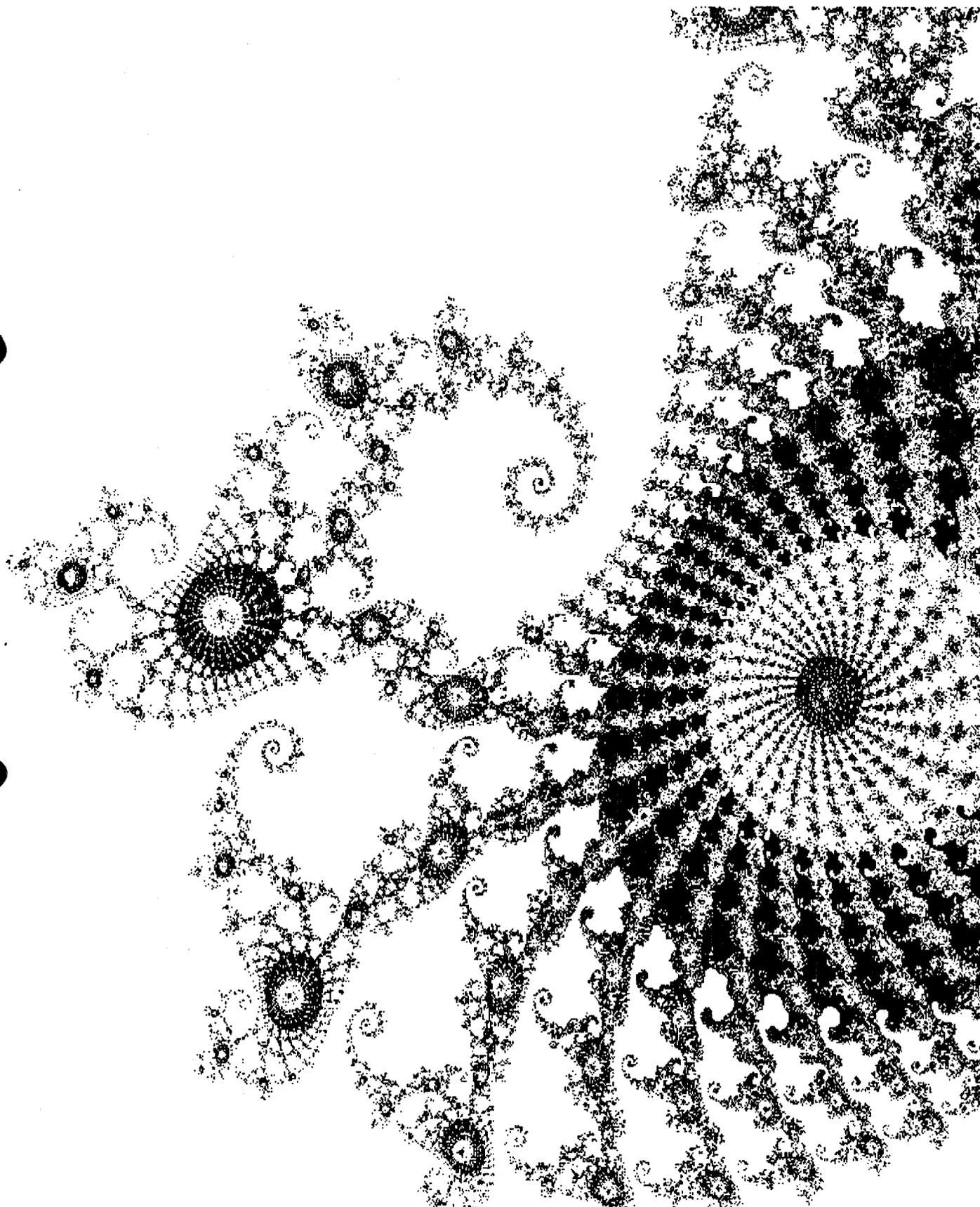
→  
 die  
 untere  
 Hälfte  
 von  
 M



Grafik: Apfelmännchen von Thomas Pflaum



Grafik: Apfelmännchen von Thomas Pflaum



BASIC: VAL und richtige Bool'sche Ausdrücke eingebautEIN BESSERES "VAL" FÜR'S MTX- UND FDX-BASIC,  
UND ENDLICH RICHTIGE BOOLE'SCHE AUSDRÜCKE FÜR'S FDX-BASIC

Andreas Viebke

Wer von Euch BASIC-Freaks Sinclairs ZX-Spectrum kennt, weiß die dort vorhandene gute VAL-Funktion zu schätzen. Man ist nicht gleich aufgeschmissen, wenn man mal einen arithmetischen Ausdruck, der sich in einem String befindet, ausrechnen lassen will. Da kann man z.B.

```
PRINT VAL("SIN(PI-X)")
```

eingeben, und tatsächlich kommt da auch was Richtiges raus. Mit dem MTX dagegen kann man so nicht rechnen. VAL kann nur Zahlen behandeln. Das ist schade, denn oft will man ja eine Funktion über die Tastatur per INPUT-Statement holen. Doch nicht verzagen - mit unserem BASIC geht's auch!

Erst mal für die FDXler: Die Routine für VAL liegt bei #5F9E (s. Sprungtabelle von mir) und ist 40 Bytes lang. Die rechtsstehende ist um 5 Bytes kürzer, kann also ohne Mühe dorthin verschoben werden. Und wenn Ihr das gemacht habt, indem Ihr sie abtippt und dann RUN eingibt, kann FDX-BASIC auch sowas:

```
A$="SQR(B*B+C*C-2*B*C*COS(AA))"
PRINT VAL(A$)
```

10 CODE	VAL für FDX
8007	LD HL,VAL
800A	LD DE,#5F9E
800D	LD BC,#28
8010	LDIR
8012	RET
8013 VAL:	RST 28
8014	DB #B9
8015	LD (HL),#FF
8017	INC HL
8018	LD (#FA81),HL
801B	PUSH DE
801C	EX DE,HL
801D	PUSH HL
801E	CALL #58BB
8021	POP HL
8022	PUSH HL
8023	CALL #599E
8026	POP HL
8027	PUSH HL
8028	RST 28
8029	DB #B6
802A	POP DE
802B	JR Z,SYNOK
802D	RST 28
802E	DB #31
802F SYNOK:	RST 28
8030	DB #80
8031	POP HL
8032	LD (#FA81),HL
8035	RET
8036	DB 0,0,0,0,0
803B	RET

Symbols:

Aufpassen muß man allerdings. Es wird zwar ein Syntax-Check durchgeführt, doch sind nicht alle Fehler Runtime-Fehler, sodaß (bei fehlenden Klammern z.B.) die Zeile, in der der Fehler auftrat, nicht in die Editier-Zeile gegeben wird. Aber was soll's?

Jetzt zu den MTXlern: Da muß eine andere Methode verwendet werden: Ihr tippt das Progrämmchen ein und müßt bei den Labels FROM und TO Variablen einsetzen. Bei TO steht "A=". A ist die Variable, der das Ergebnis zugewiesen werden soll. Und bei FROM steht 1,"\$",#FF. Die 1 steht für A - das BASIC benutzt die beiden oberen Bits des ersten Buchstabens des Variablen-Namens zur Unterscheidung zwischen String- und numerischen Variablen und zwischen einfachen Variablen und Feldern. A\$ ist hier also die String-Variable, die das Argument von VAL ist. Ihr könnt dieses A\$

BASIC: VAL und richtige Bool'sche Ausdrücke eingebaut

natürlich ändern in z.B. EXTRA\$. Dann muß bei FROM 5,"XTRA\$",&#xFF stehen. Um einen String 'auszurechnen', gibt man einfach folgendes ein:

```
LET A$="PI*A^2"
GOSUB 10: PRINT A
```

In meinem Beispiel findet man das Ergebnis in A. Das Listing findet Ihr auf der rechten Seite. Na dann tippt mal schön - viel ist's ja nicht.

Jetzt zu Mr. Boole: In vielen BASICs produziert das Statement

```
IF A
```

einen Wahrheitswert - Es ist in diesem Statement ein Vergleich mit den Interpreter-eigenen Wahrheitswerten impliziert, also z.B.

```
IF A=-1 oder IF A>0
```

Entsprechendes gilt für Unwahrheiten wie

```
IF NOT A, das z.B. IF A<=0
```

entsprechen könnte. Unser BASIC produziert aber nur einen Wahrheitswert, wenn man explizit vergleicht, d.h. einen der Vergleichsoperatoren benutzt, wie z.B. in

```
IF A>=3
```

Das ist ab heute anders, denn durch einen winzigen (einen sogenannten Minimal-) Patch sind auf einmal alle gängigen Formen des Vergleichs erlaubt. Wer von Euch FDXlern also noch Zeile 20 abtippt und diese neue Funktion mit RUN aktiviert, darf in Zukunft auch

```
IF A AND NOT B oder IF NOT A
oder IF A>=7 AND B OR C usw.,
```

inclusive Klammern und allem, was das Herz begehrt, benutzen. Das is doch was, oder? Das ist auch in NEWCOM und BASIC 1.1 drin. Klaro.

10 CODE VAL für MTX

```
8007 JP VAL
800A FROM: DB 1,"$",&#xFF
800D TO: DB "A="
800F SPC: DS #80
808F VAL: LD HL,&#xFD81
8092 RES 2,(HL)
8094 LD DE,FROM
8097 RST 28
8098 DB #BB
8099 RST 28
809A DB #B9
809B LD (HL),&#xFF
809D INC BC
809E EX DE,HL
809F LD (&#xFA81),HL
80A2 LD DE,SPC
80A5 LDIR
80A7 LD HL,TO
80AA PUSH HL
80AB CALL #38BB
80AE POP HL
80AF CALL #399E
80B2 LD HL,SPC
80B5 RST 28
80B6 DB #B6
80B7 JR Z,SYNOK
80B9 RST 28
80BA DB #31
80BB SYNOK: LD DE,TO
80BE JP #29DA
80C1 RET
```

Symbols:

20 CODE Boole für FDX

```
8007 LD HL,PATCH
800A LD DE,&#x0269
800D LD BC,&#x0009
8010 LDIR
8012 LD DE,&#x5D4C
8015 LD BC,&#x0007
8018 LDIR
801A RET
801B PATCH: CALL #5BEA
801E JP NZ,&#x5C38
8021 RST 28
8022 DB #B6
8023 RST 20
8024 JP NZ,&#x5C40
8027 JP #0269
802A NOP
802B RET
```

Symbols:

BASIC: Hardcopy für Seikosha

Die total-MC-Grafik-Copy-routine für Seikosha (Claudio Romanazzi, 3070)

```

push bc           Rette die Register
push de
push hl
xor a
ld d,a           de = Punktpos. im VRAM
ld e,a           d = y, e = x Koordinate
ld h,a           h = Zwischenvariable
ld c,#1c         c = 192 Grafikzeilen dividiert
                  durch die 7ermatrix des Seikosha

inc a
ld (#fd75),a     Ausgabe auf Drucker umleiten
loop1: call newline neue Zeile
loop2: ld d,h      alten y-Wert wieder holen
        ld b,7     Seikosha hat eine 7-er Matrix
loop1: push bc     bc retten
        push hl    hl retten
        call #34b3 Romroutine, cy gesetzt, wenn Punkt
                  gesetzt
        rr l       carry nach rechts reinschieben
                  l = copybyte
inc d            y = y+1
pop bc          Zähler wiederherstellen
djnz loop1      nochmal, wenn noch nicht 7x
ld a,c          a = restlicher Zeilenzähler/7
cp l            sind bereits 27 abgehandelt?
ld a,l          a = copybyte
jr nz,ausgabe   nein -> Ausgabe
and #0f         ja, dann oberes Nibble ausblenden
ausgabe: call docopy
inc e           e = e+1 (x-Koordinate)
jr nz,loop2     nochmal, wenn noch nicht 256 x
                  ausgeführt
ld h,d          nächste 7ergruppe anpeilen
dec c           restlicher Zeilenzähler/7 -1
jr nz,loop3     nochmal, wenn noch nicht 28x
call newline    danach ist a = 0
ld (#fd75),a    Ausgabe auf Screen leiten
pop hl          Register wiederherstellen
pop de
pop bc
ret

newline: rst 10   universale Ausgaberroutine
db 130,20,10    Drucker initialisieren und LF
xor a           a = 0
ld (pos),a      Druckerpositionsangaben für Grafik
                  initialisieren (Lowbyte)
ld (pos1),a     + Highbyte
ret

docopy: rra      Copybyte zum 8. mal nach rechts
                  rollen
or #80         Bit 7 muß gesetzt werden bei
                  Seikosha
ld (char),a    in reservierten Platz laden

```

BASIC: Seikosha-Hardcopy / BIOS: Carrera

```

rst 10
db 134,8,27,16      gib 6 Zeichen aus
pos1: db 0
pos:   db 0
char:  db 0
      ld a,(pos)      pos = pos+1
      inc a
      ld (pos),a
      ret nz          zurück, wenn noch nicht #ff über-
                      schritten
      inc a           a = 1
      ld (pos1),a     das Highbyte kann nicht >1 sein
      ret

```

Diese Routine ist ziemlich rausgepopelt aus meinem Malprog. Ich habe sie noch etwas gekürzt und vereinfacht auf das reine Copy. Da ich jedoch meinen Seikosha (GP 100-A) nicht mehr habe, kann ich sie nicht ausprobieren. Falls Rolf Schmidt oder jemand anderes nicht damit zurechtkommt, anrufen oder schreiben. Soweit dazu.

; **Bios-Overlay: Filter in Druckerausgabe** (Herbert Herberg, 2000)

; Beispiel:

; Olympia Carrera: Umsetzen von ">" und "<"

; Dieses Programm ist für all diejenigen interessant, deren Drucker  
; das eine oder das andere Zeichen nicht so ausgeben, wie es sein  
; soll, da diese nicht den ASCII-Zeichensatz haben.  
; Dieses betrifft z.B. einige Typenradschreibmaschinen von der Firma  
; Olympia, wie die Carrera (die ich habe) und die Compact 2.  
; Bei diesen beiden Maschinen liegen die Zeichen ">" und "<" auf  
; einer Sonderzeichenebene, die zuvor ein-, und ggf. nachher ausgeschaltet  
; werden muß.

; Dieses Programm erweitert das BIOS (also das System) um eine Routine,  
; die einzelne Zeichen durch eine beliebig lange Folge anderer Zeichen  
; ersetzt.

; **WICHTIG:** Diese Routine darf nur ein mal aufgerufen werden, da sie nicht  
; prüft, ob sie schon im Speicher ist, und nicht unbedingt ein  
; zweifaches Ersetzen sinnvoll ist. Außerdem kostet sie Platz!

; Um die Routine zu deaktivieren muß RESET gedrückt werden!

; Vorgehen: - Zuerst wird geprüft, ob Platz für die Routine vorhanden ist  
; - Wenn Platz vorhanden ist, werden die Zugriffe auf den Drucker  
; auf die richtigen Werte gesetzt, die ja im BIOS stehen, und  
; der Sprung zur Druckerausgabe im BIOS auf unsere Routine um-  
; gelenkt.  
; - Schließlich wird die Routine ins BIOS verschoben

BIOS: Carrera

```

ORG      100H
FREE    EQU    40H      ;Anfang des Platzes im BIOS
TOAM    EQU    46H      ;Ende des Platzes im BIOS

```

```

; Prüfen, ob Platz vorhanden ist
; -----

```

```

LD      HL,(TOAM)      ;Ende
LD      DE,(FREE)     ;Anfang
XOR     A              ;Lösche Carry
SBC    HL,DE          ;Differenz = Platz
LD      DE,ENDE-START ;Benötigter Platz
;Vergleiche jetzt vorhandenen mit benötigtem Platz
LD      A,H
CP      D              ;CP H mit D (also obere Bytes)
JP      C,FULL         ;H < D : kein Platz
JP      NZ,ROOM        ;H > D : Platz
LD      A,L
CP      E              ;CP L mit E (also untere Bytes)
JP      NC,ROOM        ;L <= E : Platz
FULL:   LD      DE,ERRM ;Gib Fehlermeldung aus
LD      C,9            ;BDOS-Funktion 9: String-Ausgabe
JP      5              ;Zurück ins System CP/M
ERRM:   DEFB        "Kein Platz für Treiber",10,13,"$"

```

```

; Sprünge korrigieren
; -----

```

```

; Im BIOS steht ein JP LSTOUT, also ein Sprung auf die Routine für
; Druckerausgabe. Diese muß nun umelenkt werden über unsere Zwischen-
; Routine, die einige Zeichen durch andere ersetzt. Dazu muß in dem
; BIOS JP LSTOUT die Adresse durch die Adresse unserer Routine ersetzt
; werden, und die Adresse LSTOUT in unserer Routine eingefügt werden,
; damit die umgesetzten Zeichen dorthin kommen, wohin sie sollen.

```

```

ROOM:   DI              ;Keine Interrupts, da sonst der ggf.
                       ;arbeitende Druckerpuffer von RAM3
                       ;stören würde.
LD      DE,(FREE)     ;Overlay kommt an (FREE)+1, d.h.
INC     DE              ;diese Adresse muß in BIOS-Tabelle
LD      BC,12         ;Da an 0 ein JP WARMBOOT steht, also
                       ;die Adresse des 2. BIOS-Sprunges, ist
                       ;LSTOUT von da ab der 4. Einsprung
                       ;BC = 4*3, d.h. Abstand von LSTOUT zu
                       ;WBOOT (je Sprung 3 Bytes: JP Adresse)
LD      HL,(0001H)    ;Adresse WARMBOOT
ADD     HL,BC          ;Nun in HL Adresse von LSTOUT
INC     HL              ;Dort steht JP, also positioniere auf
                       ;eigentliche Adresse dahinter
LD      C,(HL)        ;Adresse in DE, also die oben einge-
LD      (HL),E        ;tragene ins BIOS schreiben, und die
INC     HL              ;aus dem BIOS ins BC-Register laden
LD      B,(HL)
LD      (HL),D
LD      H,B
LD      L,C            ;Jetzt steht in HL die Adresse des Drucker-
                       ;Treibers

```

BIOS: Carrera

```

; Bis hierher wurde also in die BIOS-Tabelle die Adresse unserer Zwi-
; schenroutine eingetragen, und die Adresse von LSTOUT, die ja in der
; BIOS-Tabelle stand ist nun in HL.
; Die Zwischenroutine enthält einen CALL- auf die Druckerausgabe
; LSTOUT. Dieser hat den Label OUT. Dort muß die Adresse des BIOS-
; Treibers dafür eingetragen werden, den wir ja oben durch den Sprung
; auf unsere Routine ersetzt haben.
; Im Original steht dort die Adresse 0. Dieses Eintragen folgt hier:
      LD      (OUT+1),HL      ;Eintragen der LSTOUT-Adresse in den
                              ;CALL-Befehl unserer Routine. +1, da
                              ;das erste Byte das CALL ist.

```

```

; Verschieben unserer Routine
; -----

```

```

      LD      DE,(FREE)      ;Zieladresse
      INC     DE              ;(FREE)+1
      LD      HL,START       ;Dort steht sie noch
      LD      BC,ENDE-START  ;Länge
      LDIR                     ;Verschieben
      INC     DE              ;Einen weiter, und FREE neu schreiben,
      LD      (FREE),DE      ;damit die Routine nicht gelöscht wird
      EI                          ;Interrupts wieder einschalten, sonst
                              ;ist unter RAM3 die Tastatur tot.
      JP      0              ;Warmstart

```

```

; OVERLAY, also die Routine
; -----

```

```

;In C steht das Zeichen, welches gedruckt werden soll
START:  LD      A,C          ;Zeichen ins A-Register
        LD      HL,NORM     ;Wenn keine Umsetzung erforderlich ist
                              ;wird der String von NORM gedruckt.
        LD      (HL),A      ;Dort steht das Zeichen.
CHR1:   CP      "<"         ;Ist es ein "<"
        JR      NZ,CHR2     ;Nein, dann Weiter
        LD      HL,CODE1    ;Pointer für Ersatzfolge
CHR2:   CP      ">"         ;Ist es ein ">"
        JR      NZ,PRINT    ;Nein, also normale Ausgabe
        LD      HL,CODE2    ;Pointer für Ersatzfolge
PRINT:  LD      A,(HL)      ;Hole Zeichen
        OR      A           ;Ist es eine Null, d.h. letztes?
        RET     Z           ;Ja, dann zurück
        LD      C,A        ;Nach C
        PUSH   HL          ;Rette Pointer
OUT:    CALL   0            ;Hier Ausgabe auf Drucker
        POP    HL          ;Hole Pointer zurück
        INC    HL          ;Nächstes Zeichen
        JR     PRINT       ;Weiter

NORM:   DEFB    0,0        ;Für Normale Ausgabe
CODE1:  DEFB    27,64H,0   ;">" --> ESC, 64H
CODE2:  DEFB    27,62H,0   ;"<" --> ESC, 62H
                              ;ACHTUNG: Am Ende eine Null, als
                              ;           Ende-Markierung.
ENDE:   END

```

CP/M: Beschreibung DDTZ, der dem DDT ähnlich ist!**Anleitung zum DDTZ**

(Dieter Mueller, 1987)

DDTZ = Development Debugging Tool for Z80 CPU

Zusammenfassung der Befehle:

```

> @          examine [and substitute]
              the displacement registers @
              Assemble
> A [Startadr]
> B          display all Breakpoints
              B breakp [breakp..] [:count] set Breakpoints
              BX          clear all Breakpoints
              BX adr [adr..] clear Breakpoints
>>C[N][J] [befehlsanzahl] trace over calls [no list]
              [jumps only]
              C[N][J] W ausdruck ..while..
              C[N][J] U ausdruck ..until..
>>D [startadr] [endadr] display memory in hex and ascii
> F          specify filename and commandline
> G [startadr] [,breakp..] go [to start] [temporary breakpoints]
> H          display high and maximal size of file
> H ausdruck compute hex and other expressions
> H ausdruck ausdruck hex and other sum and difference
> I [port]   input a byte from port
> L [startadr] [endadr] list disassembled code
> M[V] startadr endadr zieladr Move memory [and verify]
> O[byte] [port] output a byte to port
> Q[J] startadr endadr byte(s) query (search) memory for byte string
              [justified]
> R [displacement] Read binary or hex file
              [add displacement]
> S startadr substitute memory
> T[N][J] [befehlsanzahl] trace [no list] [jumps only]
> T[N][J] W ausdruck ..while..
> T[N][J] U ausdruck ..until..
> V startadr endadr zieladr verify (compare) two memory areas
> W startadr endadr write a file to disk
> X          examine all CPU registers
> X register examine [and substitute] a register
> Y          examine all Y variables
> Yziffer 0 bis 9 examine [and substitute] an Y var.
> Z startadr endadr bytes zap (fill) memory with a byte string

```

Bei Befehlen, die eine Start- und eine Endadresse erfordern, kann anstelle der Endadresse auch der Buchstabe 'S', gefolgt von einer Laengenangabe verwendet werden.

z.B.: D 1000 S 100 oder D1000S100 ist dasselbe  
wie : D 1000 10FF oder D1000,10FF

Registernamen:

A, F, B, C, D, E, H, L, BC, DE, HL,  
A', F', B', C', D', E', H', L', BC', DE', HL',  
IX oder X, IY oder Y, SP oder S, PC oder P, I

CP/M: Beschreibung DDTZ, der dem DDT ähnlich ist!Flags des F- und des F'-Registers:

S	Sign	Vorzeichen
Z	Zero	Null
H	Half carry	Uebertrag von bit 3 nach bit 4
V	oVerflow / parity	Ueberlauf / gerade Paritaet
N	Negation	NEG, DEC, SUB o.ae. ausgefuehrt
C	Carry	Uebertrag von bit 7
E	interrupt Enabled	ein flag des F-Registers behandelt.

Variablen und Konstanten:

L	Low	Benutzerspeicher-Anfang. 100H fest.
H	High	Hoechste Adresse des letzten gelesenen Files.
M	Max	Hoechste Adresse aller gelesenen Files
T	Top	Oberes Ende des Benutzerspeichers.
@		Displacement Register.
\$		beim Assemblieren Anfangadresse des Befehls sonst = PC.
YØ bis 9		frei verwendbare Variablen.
^Register		Inhalt eines Registers.
(adresse)		Inhalt eines Speicherplatzes (byte)
(adresse).		Inhalt eines Speicherplatzes (wort)

Aufruf des DDTZ:

**DDTZ**  
oder **DDTZ d: filename.ext**

Bei beiden Aufrufen laedt das CP/M den DDTZ bei Adresse 0100H beginnend in den Speicher. Das DDTZ verlagert sich danach direkt vor das BDOS und ueberschreibt dabei den CCP. Der Adressteil des Befehls JP BDOS auf Adresse 0005H wird erniedrigt, um dem Anwenderprogramm die durch den DDTZ verringerte Speichergroesse mitzuteilen. Der urspruenglich belegte Speicher wird daraufhin mit Nullen geloescht. Wurde im Aufruf ein Filename angegeben, so laedt das DDTZ das File in den Speicher. Nach Ausgabe des Promp-Zeichens '>' ist das DDTZ betriebsbereit.

Allgemeines zum Gebrauch des DDTZ:Ausdruecke:

In allen Situationen, in denen Zahlen eingegeben werden koennen, ist auch die Eingabe von Ausdruecken erlaubt.

Ausdruecke haben die Form

**Arithmetischer Ausdruck** oder  
**Arithmetischer Ausdruck Relationsoperator Arithmetischer Ausdruck**

Die Relationsoperatoren sind

= <> < > <= >=

Der Ausdruck hat den Wert -1, wenn die Relation wahr ist, sonst 0.

z.B.: der Ausdruck 1+2=3 hat den Wert -1, und 'A' > 'B' den Wert 0.

Ein Arithmetischer Ausdruck hat die allgemeine Form:

**Faktor Arithmetikoperator Faktor Arith... Faktor**

CP/M: Beschreibung DDTZ, der dem DDT ähnlich ist!Die Arithmetikoperatoren sind:

+ - \* / %            Plus, Minus, Multiplikation, Division, Modulo  
& ! #                bitweise AND, OR, XOR

Ein Faktor hat die Form:

[ausdruck]	geklammerter Ausdruck
+faktor	
-faktor	Negation
~faktor	bitweise NOT
'a'	Wert eines ASCII-Zeichens
'a'.	mit gesetztem Bit 7
'ab'	Wert zweier ASCII-Zeichen
'ab'.	das Niederwertige mit gesetztem Bit 7
(ausdruck)	ein Byte im Speicher
(ausdruck).	ein Wort im Speicher
variable	eine der oben aufgezählten Variablen
hhhh[H]	Hex-Zahl
dddd.	Dezimalzahl
bbbbbbb"bbbbbbb"	Binaerzahl (beliebig mit '"' unterteilt)

Enthaelt ein Ausdruck einen Relationsoperator, so werden erst die beiden arithmetischen Ausdruecke berechnet und dann die Relation. Ein arithmetischer Ausdruck wird von links nach rechts berechnet, also ohne Beachtung 'Punkt vor Strich', die Reihenfolge kann jedoch mit Klammern [ und ] geaendert werden. Ausdruecke duerfen keine Leerzeichen enthalten.

Befehle:

Befehle beginnen mit einem der Zeichen '@' bis 'Z', eventuell gefolgt von weiteren optionalen Buchstaben, sowie Argumenten. Wird eine optionale Anfangsadresse weggelassen, so wird die Adresse benutzt, bei der ein Befehl zuletzt beendet wurde. Wird eine optionale Endadresse weggelassen, so wird der Befehl (z.B. L oder D) so lange ausgefuehrt, bis etwa ein Bildschirm vollgeschrieben ist. Ein weggelassenes optionales Argument, hinter dem weitere Argumente angegeben werden sollen, muss durch ein Komma ersetzt werden. Zwischen zwei Argumenten muss ein Leerzeichen, ein Tab (^I) oder ein Komma stehen.

TURBO: GotoXY, Write schneller / Error 93, 96 / Hallo Hacker!

Benutzt man anstatt GOTOXY, WRITE und WRITELN in TurboPascal 3.0 bei der Stringausgabe auf den Bildschirm die Prozeduren LOCATE, PRINT und PRINTELN so erreicht man eine beachtliche Geschwindigkeitssteigerung, die sogar noch geringfügig schneller ist als die von TurboPascal 2.0.

```
z.B: 10000 mal GOTOXY(10,10); ca 19,7 sek
      10000 mal LOCATE(10,10); ca 8,9 sek

z.B 1000 mal GOTOXY(10,10);
      WRITE('12345678901234567890'); ca 15,3 sek
      1000 mal LOCATE(10,10);
      PRINT('12345678901234567890'); ca 8,9 sek

z.B 1000 mal WRITELN('12345678901234567890'); ca 16,4 sek
      1000 mal PRINTELN('12345678901234567890'); ca 10,1 sek

z.B 100 mal WRITE * 80 Zeichenstring * ca 5,7 sek
      100 mal PRINT * 80 Zeichenstring * ca 3,5 sek
```

```
procedure locate(x,y:byte);
begin
  bios(3,3);bios(3,x+32);bios(3,y+32);
end;

type x=string$80;
procedure print(a:x);
var i:byte;
begin
  for i:=1 to ord(a) do bios(3,ord(a[i]));
end;

procedure println(a:x);
var i:byte;
begin
  for i:=1 to ord(a) do bios(3,ord(a[i]));
  bios(3,13);bios(3,10);
end;
```

Der Fehlermeldungsfile TURBO.MSG kann, wenn man Spaß daran hat, um zwei Meldungen erweitert werden:

```
Error 93 : "Illegal Compiler Option"
Error 96 : "Nested Include"
```

Und noch eine Möglichkeit von Turbo den Hacker tagelang den Fehler suchen zu lassen.

```
program guck;

var a:real; (* Versuchen Sie mal mit a:integer *)

begin
  a:=0;
  if a=(-a) then
    writeln ('Stimmt a = -a = ',a)
  else
    writeln ('a <> -a aber es ist a=',a,' und -a=',-a);
  readln;
```

(Volkmar Döring,  
5300)

SuperCalc: Der Kurs von Wolfgang GiegerKursus SC.001Wolfgang Gieger, 3180

Dieser Kurs setzt sich aus mehreren Folgen zusammen. Er beinhaltet eine Führung zum Umgang mit Supercalc. In kleinen Schritten wird erklärt, wie mit der Software umzugehen ist. Es ist für alle Clubmitglieder, die noch nie, oder nur selten mit Supercalc gearbeitet haben. Dafür ist dieser Kurs besonders gut geeignet. Vielleicht kommen dann in Zukunft auch hier mehr Beiträge. Auch für die Anwendung der RAM3-Disc besonders zu empfehlen, da viele Operationen dann nicht mehr von dem Laufwerk durchgeführt werden. Zeit kostet ja viel Geld.

So dann noch der Hinweis für alle die den Ärger mit der englischen Sprache umgehen möchten : "Supercalc hat ein ausführliches Hilfsmenue das nach jeder Eingabe mit SHIFT und dem ? Fragezeichen erreichbar ist, aber nur in Englisch."

Da es aber den sogenannten Firmenschutz gibt ist eine REINE Übersetzung leider nicht erlaubt.

Aber das spielt jetzt auch keine Rolle mehr. Ich habe 80% eigene Anleitungen in dem Hilfsmenue in deutscher Fassung untergebracht und somit von einen Anteil von nur 20% der REINEN englischen Übersetzung verwendet und das ist laut Softwaregesetzen erlaubt.

Wer also den Kurs nicht bis zum Ende erwarten kann kommt mit diesem Hilfsmittel sehr schnell und gut durchs Supercalc. Dieses Hilfsmittel umfast circa 80 bis 90 Bidschirmseiten die der gröÙe des Monitorbildes entspricht, das sind ungefähr 50 seiten die eines Handbuches. Bei Zusendung einer Leerdiskette und Rückporto und einen Superkleinen Betrag von 10.- DM kann die deutsche Fassung von mir erworben werden und der Preis für teure Bücher entfällt.

So das Kurz zur Einleitung dieses Kurses!

Nun aber zur Praxis. Legen Sie eine Arbeitskopie von Supercalc in das Laufwerk >A< und schauen mit >DIR< nach der Direktorie. Es befindet sich eine Datei Namens > SC.COM < auf der Diskette. Wir geben hinter dem > A < und dem blinkenden Cursor nur > SC < ein gefolgt von >RET< und warten dann guter Dinge.

Supercalc ab hier kürze ich es in > SC. < ab, damit nicht soviel Papier verschwendet wird, SC. meldet sich mit Version 1,12 und erwartet eine Eingabe.

Geben wir > RET < ein, dann bekommen wir sofort den Arbeitsbogen, geben wir aber ein Fragezeichen ein, dann meldet sich das Hilfsmenue mit weiteren Informationen.

Wir geben Ret ein, weil die Hilfsmenues ja in Englisch sind und wir sie hier noch nicht brauchen, außerdem ist zu jedem der Menues im Kurs eine Erklarung dabei. Wir gehen also Schrittweise vor.

Nach der Taste RET bekommen wir folgenden Bildschirm Aufbau und der nennt sich FORMATBOGEN.

SuperCalc: Der Kurs von Wolfgang Gieger

```

!      A      !      B      !      C      !      D      !      E      !      F      !
1!<          >
2!
3!
4! <---- Es erscheinen noch 16 Zahlen und der Bogen geht bis zur
5!      20. Zeile.
6!
7!
8!
9!

```

```

A1> Width:9  Memory:22  Last Col/Row:A1
1>

```

Sie sehen also in der oberen Zeile des Monitors die Buchstaben von A-H getrennt durch Ausrufezeichen in einen Abstand von 9 Einzelzeichen. Da es 8 Buchstaben sind kommen wir auf eine Breite von 72 Zeichen, eben  $8 \times 9 = 72$ . Unter dem Buchstaben > A < ist mit dem Pfeil links und rechts genau 9 Zeichen der Cursor gekennzeichnet, und der steht in der ersten Zeile nach der EINS mit dem AUSTRUFEZEICHEN.

Das bedeutet die Position > A1 < in der steht jetzt also der Cursor. Weiter unten wird links mit > A1 < die Bewegung des Cursors und die Position angezeigt.

Daneben die Zeichenbreite, also die Weite mit 9, daneben Memory:22 das heißt Sie haben eine Speicherkapazität von 22 Kbyte im Arbeitsbogen und daneben mit > COL/ROW < die Columnen und Zeilenanzeige mit der Zellenangabe > A1 < und darunter die Statuszeile in der Sie Ihre Eingaben vornehmen können. Diese umfassende Erklärung ist nötig, da wir im weiteren Verlauf nur noch diese Bezeichnung benutzen.

Um Text zu schreiben muß ein < " > fübchen vorangestellt werden.

```

1>"Dieser Satz wird dann genauso in die Zelle,
   in der der Cursor steht, übernommen

```

Nun wollen wir ein paar Eingaben in den Bogen vornehmen, um einige Unterschiede bei den Texten und Zahlen und die Wirkungsweise der Zeilen, zu studieren.

Wir geben also ein: Ab hier den Text abschreiben : Taste

```

Cursor in Zelle > A1 < bringen 1>"Hier ist die Zelle A1 mit Text >RET<

```

```

Cursor in Zelle > A2 < bringen 1>"und hier folgt die zweite >RET<

```

```

Cursor in Zelle > A3 < bringen 1>"Zeile und nun folgen Zahlen >RET<

```

```

Cursor in Zelle > C5 < bringen 1>123456789 >RET<

```

```

Cursor in Zelle > C3 < bringen 1>777.7777 >RET<

```

SuperCalc: Der Kurs von Wolfgang Gieger

Hier passiert in der Zelle >C3< die uebernahme der Zahl 777.777 und der vorher ab Zelle >C3< stehende Text verschwindet, aber keine Angst er geht nicht verloren, sondern bleibt in voller Laenge erhalten nicht sichtbar, weil die Zelle C3 mit einem Inhalt belegt worden ist.

Wir gehen mit dem Cursor nach Zelle A3 und sehen ueber der Statuszeile hinter -----> Last Col/Row: den immer noch alten geschriebenen Text Durch Löschen der Zelle C3 ist dann wieder alles so wie vorher.

Eingabe -----> /B und C3 und >RET<

Nun ist die Zelle C3 gelöscht und der alte Text erscheint wieder. So das nur zur Wirkungsweise von Text und Zahlen mit unbestimmter Laenge in den Zellen.

Da das aber nicht immer der Sinn ist koennen wir natuerlich den Formatbogen aendern und ihn auf den jeweiligeigen Zweck anpassen. Bevor wir aber dorthin gehen kommt ersteinmal eine weitere Menue-Erklaerung um zu wissen wie man an weitere Hilfen und Umgehungsweise im SC. benutzen kann.

Hier nun eine weitere Hilfe von SC. Durch die Eingabe des / Striches erscheint ueber der Statuszeile eine Leiste mit Buchstaben von B-Z und dem ? Zeichen.

Wenn wir nun das Fragezeichen eingeben : /? erscheint die nun folgende Auflistung mit der Bedeutung der einzelnen Buchstaben, aber nur in Englisch, hier aber schon für den deutschen Gebrauch:

\*\*\*\*\*

B(lank)-----> löschen von Zellen, Zeilen, Spalten und Blöcken  
 C(opy)-----> copieren von Zellen, Zeilen, Spalten und Blöcken  
 D(elete)-----> löschen von Zeilen oder Spalten  
 E(dit)-----> Editieren (Neueingabe) aller Zelleninhalte  
 F(ormat)-----> Bereichsdarstellung von Zeilen und Spalten  
 G(lobal)-----> Globale Darstellung oder Caculations-Option  
 I(nsert)-----> Einfügen einer neuen Zeile oder Spalte  
 L(oad)-----> Daten von der Diskette einladen (File-Namen)  
 M(ove)-----> Verschieben von ganzen Zeilen und Spalten  
 O(utput)-----> Ausgang Printer, Disk und Optuzionen  
 P(rotect)----> Schützen von Zellen, Zeilen, Spalten und Blöcken  
 Q(uit)-----> Verlassen von S.Calc (abspeichern nicht Vergessen)  
 B(lank)-----> löschen von Zellen, Zeilen, Spalten und Blöcken  
 C(opy)-----> copieren von Zellen, Zeilen, Spalten und Blöcken  
 D(elete)-----> löschen von Zeilen oder Spalten  
 E(dit)-----> Editieren (Neueingabe) aller Zelleninhalte  
 F(ormat)-----> Bereichsdarstellung von Zeilen und Spalten  
 G(lobal)-----> Globale Darstellung oder Caculations-Option  
 I(nsert)-----> Einfügen einer neuen Zeile oder Spalte  
 L(oad)-----> Daten von der Diskette einladen (File-Namen)  
 M(ove)-----> Verschieben von ganzen Zeilen und Spalten  
 O(utput)-----> Ausgang Printer, Disk und Optuzionen  
 P(rotect)----> Schützen von Zellen, Zeilen, Spalten und Blöcken  
 Q(uit)-----> Verlassen von S.Calc (abspeichern nicht Vergessen)  
 R(eplicate)--> Wiederholen gleicher Eingaben, bei Zeilen u. Spalten  
 S(ave)-----> Das Wichtigste überhaupt, Daten abspeichern!!!!!!!  
 T(itle)-----> Fixieren von Titelzeilen in Zelle, Zeile und Spalte  
 U(nprotect)--> Aufhebung des schutzes von Protect, weiter oben (P)  
 W(indow)-----> Teilen des Bildschirmformates, Arbeitsformat  
 X(eXecute)--> Automatische Datendatei, Programmablauf in "XQT"  
 Z(ap)-----> Löschen des ganzes Arbeitsblattes von Supercalc

SuperCalc: Der Kurs von Wolfgang Gieger

Im verlauf des Kurses werden dann die einzelnen Operationen der Buchstabenleiste von B-Z erklart und dann ist jeder in der Lage effektiv mit Supercalc zu arbeiten.

Wir kommen jetzt zum FORMATIEREN des Arbeitsbogens:

Alle folgenden Eingaben betreffen nur die Darstellungen am Bildschirm oder im Ausdruck !!!

Das Standartformat von Supercalc ist folgendermaßen definiert:

- ```

-----> Zahlen werden ganzzahlig und rechtsbündig ausgegeben
-----> Texte werden linksbündig ausgegeben
-----> Die Spaltenbreite beträgt neun Stellen

```

Wenn Ihnen diese Standart-Formatierung nicht gefällt, können Sie für den gesamten Arbeitsbogen oder für Teilbereiche Änderungen veranlassen

Bei jeder Eingabe des "/" und eines Buchstaben von A-Z können Sie mit der Eingabe der Shifttaste und des Fragezeichens ein bestimmtes Hilfsmenue für den entsprechenden Buchstaben anfordern. Dies erledigt Supercalc automatisch. Das weiterblättern im Menue um weitere Bildschirmseiten aufzulisten erfolgt solange mit "RET" bis der Ruecksprung zur letzten Eingabe erfolgt.

Das noch als kurzer Hinweis fuer den Aufruf der Operationen, die Sie versuchsweise einmal durchspielen sollten.

Bereich designators.

- ```

A   Bereich -->   ist eine Zeile, Spalte, Zelle oder Block
A   Zeile   -->   ist eine Nummer von 1 bis 254
A   Spalte  -->   ist ein Buchstabe von A bis BK
A   Zelle   -->   ist eine Zahl und ein Buchstabe z.B. "J10"
A   Block   -->   sind 2 Zellenangaben, getrennt durch (:)

```

Das sind die Aufteilungen eines Arbeitsbogen bei Supercalc. Bei der Eingabe, ALL, ist der Hauptbereich A1:<Letzte Spalte/Zeile>

```

=====
= 1. Arbeitsbogen                                     =
= -----                                           =
=           Spalten :   A bis BK (64)                =
=           Zeilen  :   1   bis   254                =
=           Zellen  :   Definiert durch Spalte und Zeile =
=           Blöcke  :   Definiert Durch erste und letzte Zelle =
=
= 2. Statuszeile   > A1      :   Sprungrichtung des Kusors =
= -----                                           =
=           A1      :   Inhalt der Zelle              =
=           Width:9  Zellenweite neun Zeilen        =
=           Memory:22 Speicherkapazitaet 22 Kbyte    =
=           Last Col/Row :inhaltsangabe der Zelle auf dem der Cursor steht=
=====

```

SuperCalc: Der Kurs von Wolfgang Gieger

Hier noch einmal die einzelnen aufgelisteten Befehlsablaeufe bei Supercalc.

Nun die Eingabe : /F führt zur Anzeige:

Enter Level : G(lobal), C(olumn), R(ow) , E(ntry) , or D(efine)  
Bereich eingeben: G(lobal), C(Spalte),R(Zeile),E(ingabe),oder D(efinit

Hiermit wird Ihnen die Auswahl angeboten, ob Sie für den gesamten Arbeitsbogen, für einzelne Spalten oder Zeilen oder für einen bestimmten Bereich die Formatierung aendern wollen, oder ob Sie bestimmte Anwender-Formate definieren wollen.

Achtung!!!!

Bei unterschiedlichen Formateingaben gilt folgende Prioritaet:

1. Zelle
2. Zeile
3. Spalte
4. Total

Das bedeutet, daß immer die kleinere Einheit nicht von einer Aenderung der naechst groeßeren Einheit betroffen wird, solange es sich um dieselbe Detailformation handelt.

Haben Sie z.B. fuer eine Zelle definiert, daß dort Exponentialschreibweise verwendet wird, so wird diese auch erhalten bleiben, wenn Sie fuer die Spalte eine andere Zahlendarstellung verlangen.

Eingabe: >C< führt zur Anzeige:  
Enter Column Letter  
Eingabe Spaltenbuchstabe

Wir geben also einen Buchstaben ein z.B. das A und Return

Eingabe: >R< führt zur Anzeige:  
Enter Row Nummer  
Eingabe Zeilennummer

Wir geben hier eine Zahl ein z.B. 13 und Return

Eingabe: >E< führt zur Anzeige:  
Enter Range  
Eingabe Bereich

Hier geben wir die Ziel-Zelle im Bereich ein z.B. D12.

Eingabe: >G< führt zur Anzeige:  
Enter Range  
Eingabe Bereich

Hier geben wir den Zielbereich ein z.B. f54 und alle Spalten und Zeilen werden bis zu der Zelle F54 gleich formatiert, eben Global.

SuperCalc: Der Kurs von Wolfgang Gieger

Bei all diesen Eingaben führt das zur Weiteren Anzeige:

---

Define Formats (I,G,E,\$,R,L,TR,TL,\*,Column Width)  
 Formatbestimmung (I,G,E,\$,R,L,TR,TL,\*,Spaltenbreite)

Mit diesen Auswahlmoeglichkeiten können wir weitere Format-Einstellungen bestimmen.

Es sind die wie folgt:

---

- I -----> Zahlen werden als ganze Zahlen dargestellt, dabei wird auf und abgerundet
- G -----> Solange die Stellenzahl ausreicht, werden Zahlen ganzzahlig dargestellt. Reicht der Platz nicht, folgt diese >>>>>> Darstellung.
- E -----> Die Zahlendarstellung erfolgt grundsaeztlich exponential als Dezimalwert einer Zehnerpotens. 1543 wird dargestellt als 1.543e3.
- \$ -----> Die Darstellung erfolgt im Waehrungsformat mit Angabe von zwei Dezimalstellen. Ganze Werte werden mit .00 ausgegeben Es wird auf hunderstel gerundet.
- R oder L ---> Die Zahlen werden innerhalb der Zellen an den rechten oder linken Rand gerueckt.
- TR oder TL --> Texteingaben werden innerhalb der Zellen an den rechten oder linken Rand gerueckt.
- \* -----> Die Zahlenwerte werden graphisch mit \*\*\*\*\* dargestellt.  
 Eine ANPASSUNG an die Zellenbreite ist notwendig, Standard ist (9).
- D -----> Hiermit wird die Standart-Darstellung aufgerufen.

---

Sie haben nun genug Material zum arbeiten, probieren Sie alle moeglichkeiten durch und formatieren ihren Arbeitsbogen in allen Variationen mit verschiedenen breiten.

Sehr wichtig dabei ist, immer einen Block und Stift zur Hand haben!!!!

Fortsetzung folgt in: Kurs-02.cal      Viel Spass!

C: C versus PASCAL**'C', ein Abriß**

(Michael Möwe, 2000)

Man hört das eine oder andere von 'C', aber was es damit auf sich hat, erfährt man erst, wenn man 'C' gelernt hat. Um dem ein wenig abzuhelpfen, eine kurze Exkursion in 'C'.

Grundsätzlich gilt, wer PASCAL kann, kann auch 'C', die zur Verfügung stehenden Sprachkonstrukte sind einander (sehr) ähnlich.

<u>PASCAL</u>	<u>'C'</u>
For ...	for ...
Repeat ... Until	do ... while
While ...	while ...
If ... Then ... Else	if () ... else ...
Case ...	switch ...

und so weiter ...

Soweit sehen sich die beiden Sprachen also wirklich ziemlich ähnlich, und sie verhalten sich auch so. Warum dann 'C', ist die naheliegende Frage? Da müßte man eigentlich K & R fragen, denn bis auf zwei Fälle, wüßte ich keine überzeugende Antwort. Doch zuerst möchte ich noch ein paar kleine Programmchen als Beispiele herhalten lassen.

<u>PASCAL</u>	<u>'C'</u>
For I:=1 to 10 do Write(I);	for(i=1;i<=10;i++) printf("%d",i);
I := 1; Repeat Write(I); I := I + 1; Until I>10;	i = 1; do { printf("%d",i); i++; } while (i<=10);
If (A = B) Then Write('True') Else Write('False');	if (a == b) printf("True"); else printf("False");

Wie man sieht, können PASCAL-Programme (fast) 1 zu 1 in 'C'-Programme übersetzt werden.

Ein Grund, warum K & R 'C' erfunden haben, ist sicherlich deren Schreibfaulheit, was ich anhand eines Programmchens, das einen ASCII-String in eine Integer-Zahl umwandelt, demonstrieren möchte:

```
for(x=0;(*s>='0') && (*s<='9');x = x * 10 + *(s++) - '0');
```

x enthält die Integer-Zahl

s ist ein Zeiger auf den String, der umgewandelt werden soll.

Kürzer geht's wohl nicht mehr.

Für diejenigen die neugierig geworden sind, ein Literaturhinweis:

Kernighan & Ritchie  
'Programmieren in C'  
Hanser Verlag  
ISBN 3-446-13878-1

C: Trigonometrische Funktionen

**Trigonometrische Funktionen unter (in) 'C'** (Michael Möwe, 2000)  
 'C' ist eine 'schöne' Sprache, jedoch bietet sie nicht alle Features, die z.B. Pascal kennt; unter anderem nicht die trigonometrischen Funktionen (sin(), cos(), ...).

Um diesem Manko abzuhelpfen hier ein Beispiel, wie man diese Funktionen selber schreiben kann.

Zuerst einmal die mathematische Grundlage, die Funktion sin(x) kann wie folgt dargestellt werden, wobei beachtet werden muß, daß die Reihenentwicklung der Sinus-Funktion nur eine Näherung an die 'wirkliche' Sinus-Funktion darstellt, was aber bei ausreichender Genauigkeit vernachlässigt werden darf.

$$\sin(x) = x - x^3/3! + x^5/5! - x^7/7! + - \dots$$

Bis hierhin läßt sich alles in einer mathematischen Formelsammlung (z.B. 'Bartsch') nachlesen, wie macht man daraus jetzt eine 'C'-Funktion ?

Ein Weg wäre:

```

1.
/*
   fak(x) berechnet die Fakultät von x. Aus Gründen der Programm-
   Systematik wird mit float-Zahlen gerechnet.
*/
float fak(x)
float x;
{
   float f;

   f = 1.0;
   for(;x>0;f*=x--);
   return(f);
}/* fak */

/*
   sin(x) berechnet den 'Sinus von x'.
*/
float sin(x)
float x;
{
   return(x-x^3/fak(3)+x^5/fak(5)-x^7/fak(7));
}/* sin(x) */

```

Diese Lösung für den Sinus kann ohne Probleme auf andere Funktionen der Trigonometrie angewendet werden.

Dennoch gibt es 2 Dinge, die ich noch etwas 'besser' lösen will.

C: Trigonometrische Funktionen / MSDOS: hääää?

**a. Berechnung der Fakultät**

Diese Berechnung über fak(x) dauert unnötig lange so, daß es besser wäre vorab eine Tabelle mit Fakultätswerten zu erstellen.

```
static float _fak[] = { 1.0, /* 0! */
                       1.0, /* 1! */
                       2.0, /* 2! */
                       6.0, /* 3! */
                       24.0, /* 4! */
                       120.0 /* 5! */
                       }; /* _fak */
```

Diese Tabelle kann (nat)rlich) nach Belieben erweitert werden.

**b. Berechnung des Sinus**

Die Berechnung eines Polynoms über das 'Horner Schema' beschleunigt diese beträchtlich.

$$dx^3 + cx^2 + bx + a = ((d * x + c) * x + b) * x + a$$

Durch die geringere Anzahl von Multiplikationen (Zeitfresser) ist dieser Algorithmus schneller als die herkömmliche Art der Berechnung.

**2.**

Jetzt also der 'neue' Sinus.

```
float sin(x)
float x;
{
  static float q; /* Zwischenspeicher für x^2 */

  q = x * x;
  return(((((-q/_fak[7]+_fak[5])*q-_fak[3])*q+1)*x));
} /* sin(x) */
```

Ich bin begeistert, ich hoffe Ihr auch.

P.S.: das ganze ist ohne Gewähr, ich hab's nicht getestet.

P.P.S.: Gruß an Manfred Flume

**MS-DOS-verdorben ?**

(Michael Möwe)

Betrifft: Info 14, Seite 30.

Ich denke, daß der Uebergang von ü nach ue, eher durch den ständigen Gebrauch von Mailboxen als durch die Benutzung von MS-DOS begründet ist, denn MS-DOS-Rechner bieten im allgemeinen die deutschen Sonderzeichen.

P.S.: Ich schreibe Uebergang mit Ue, da das Ü so beschissen aussieht.

**Anm.d.HH:** Auf dem Drucker ist das Ü wirklich nicht schön, jedoch auf dem Bildschirm mit meinem neuen Zeichensatz solltest Du davon begeistert sein, oder ?

CP/M: Formate mittels COMPAT und FMAT - statt RAM 3.4**FMAT.COM und COMPAT.COM / Zusammenarbeit mit 80-Spur-Laufwerke**

(Rainer Dwczorz, 5100)

Die Programme COMPAT und FMAT sollen MTX-Besitzern, die sich ein zweites Laufwerk angeschafft haben, die Möglichkeit bieten, Disketten mit 'Fremdformaten' zu lesen oder zu beschreiben und auch neue Disketten in fremden Formaten zu formatieren. Insgesamt enthalten COMPAT und FMAT jeweils acht identische Menüs, aus denen man sich das gewünschte Format aussuchen kann. In Laufwerk C: kann nach der Auswahl im Programm FMAT dann die Diskette formatiert werden; im Programm COMPAT kann sowohl das Laufwerk A/B: wie das Laufwerk C: unter dem neu gewählten Format genutzt und die mit FMAT formatierte Diskette beschrieben werden.

So sollte es sein. Hat man sich aber, so wie ich, als Zweitlaufwerk ein 80-Spur-Laufwerk zugelegt laufen beide Programme nicht, d.h. sie laufen schon, aber machen nicht das, was sie versprechen. Erstens kann man nur sechs der acht vorhandenen Menüs auf den Bildschirm bekommen (nur 40-Spur-Formate), zweitens kann in Laufwerk C: die Diskette mit FMAT nur falsch formatiert werden und drittens ist COMPAT nicht in der Lage auf einem der Laufwerke ein fremdes Format zu lesen oder zu schreiben.

Es sollen zwar nicht alle in den Menüs angegebenen Formate korrekt sein, doch gleich auch auf die richtigen Formatierungsmöglichkeiten zu verzichten, wollte ich auch nicht. Deshalb habe ich mir die beiden Programme etwas näher angesehen und war bislang bei FMAT erfolgreich. Der Anfang dieses Programms sieht so aus:

```

JP      L1000          ;1000H
DEFB   ODH,OAH
DEFB   'Program: FORMAT - disk initializer'
DEFB   ODH,OAH
DEFB   'Author: Lawrence E. Hughes'
DEFB   ODH,OAH
DEFB   'Copyright (c) 1984, Mycroft '
DEFB   'Labs, Inc.'
DEFB   ODH,OAH
DEFB   'All rights reserved'
DEFB   ODH,OAH

```

Dieser Text erscheint im Programmcode immer wieder, aber nie während des Programmablaufes. So zum Beispiel bei B0238: und B0239:.

```

DEFB   ' '
DEFB   ' '
DEFB   'Lawren'
B0238: DEFB   'P'
B0239: DEFB   'B'
DEFB   ' '
DEFB   'E'
DEFB   '. Hughes'
DEFB   ODH,OAH
DEFB   'Copyright (c) 1984, Mycroft '
DEFB   'Labs, Inc.'
DEFB   ODH,OAH
DEFB   'All rights reserved'
DEFB   ODH,OAH

```

CP/M: Formate mittels COMPAT und FMAT - statt RAM 3.4

Eigentlich dergleiche Text, nur sind hier ein 'P' und ein 'B' eingefügt. Die Anwesenheit des 'P' wird später vom Programm überprüft. Bei L1000: geht's folgendermaßen weiter:

```

      LD      SF,L1000
!-->  LD      A,(B0238)
      AND    10H
L1008: JF      Z,HOP1
      LD      A,'C'          ;43H
      LD      (DRVOUT),A    ;Drive in die Message schreiben
HOP1:  CALL   L0200
      LD      A,(B005D)
      CP      ' '          ;20H
      JP      Z,SHOWM1

```

Ändert man nun das 'P' = 50h in D0h hat man auf alle FMAT-Menüs Zugriff, von 1 bis 8, und auch die Formatierung der Diskette in C: läuft, für 40-Spur und 80-Spur-Formate. Überprüft habe ich das für EPSON QX10, Kaypro II (40-Spuren) und DEC Rainbow (80-Spuren). Soweit zu FMAT.COM.

COMPAT sieht ähnlich aus:

```

L0100: JF      L1800
      DEFB   0DH,0AH
      DEFB   'Program: COMPAT - Direct '
      DEFB   'Disk Interchange Utility'
      DEFB   0DH,0AH
      DEFB   'Author: Lawrence E. Hughes'
      DEFB   0DH,0AH
      DEFB   'Copyright (c) 1984, by Mycroft '
      DEFB   'Labs Inc'
      DEFB   0DH,0AH
      DEFB   'All Rights Reserved'

```

Und auch in diesem Programm kommt die Kombination 'P','B' vor:

```

      DEFB   3,0C3H
      DEFB   '2'
      DEFB   2
B0369: DEFB   'P'
B036A: DEFB   'B'
      DEFB   1,0CH,0,0,0,0,0,0,0,0
      DEFB   2AH,0EH,9DH,1,13H,0DH,0EDH
      DEFB   0CH,0,0,0,0,0,0,0,0,0

```

CP/M: Formate mittels COMPAT und FMAT - statt RAM 3.4

und wird im Programmlauf ausgewertet:

```

L1B00: LD      SP,0200H
        LD      A,(B036A)      ;'B' nach Register A und dann
        LD      (B4242),A      ;nach 4242 und
        LD      (B4243),A      ;nach 4243
        LD      A,'B'          ;42H
        LD      (B4245),A      ;erst 'B' nach 4245
        LD      A,1
        LD      (B4246),A      ;und 1 nach 4246
! --> LD      A,(B0369)      ;ist das richtig ?
        AND     10H
        JF      Z,HOP1
        LD      A,'C'          ;43H
        LD      (B4245),A      ;jetzt doch 'C' nach 4245 und ...
        LD      (DRMG1),A      ;'C' in das Hauptmenue eintragen
        LD      (DRMG2),A      ;'C' in das Hauptmenue eintragen
        LD      (DRMG3),A      ;'C' in das Hauptmenue eintragen
        LD      A,2
        LD      (B4246),A      ;...und 2 nach 4246
HOP1:  CALL    INSTLD

```

Nur leider bringt eine Änderung wie bei FMAT, also 'P'=50h ersetzen durch D0h, nicht den gewünschten Effekt. Zwar gelangt man jetzt in alle acht Menüs, aber das Lesen der mit dem geänderten FMAT formatierten Disketten klappt immer noch nicht.

Das ist der Stand der Dinge. Wenn jemand schon weiter ist und die beiden Programme schon hingebogen hat, sollte er sich bitte bei mir melden. (Rainer Dwczorz, 0241/48876, 5100 Aachen, Suermondtplatz 11)

(Disassembliert habe ich FMAT und COMPAT mit REZILOG, tolles Ding.)

**Anm.d.HH** Leider sehe ich nicht ganz den Sinn von COMPAT/FMAT ein (ich habe diese Programme schon einmal gesehen). Zum einen sind sie sehr langsam, insbesondere beim Warm Boot (also ^C), kennen nicht einmal das Format MTX 07, also unser 80-Spur-Format, und bieten nur wenige Formate, insbesondere keinerlei RAM-Discs. Mit COMPAT/FMAT kennt der MTX nur die Laufwerke B: und C:, kann nur C: formatieren (es sei dann man patcht, aber mehr als zwei Laufwerke sind nicht drin!), und mit einem einzelnen Laufwerk ist man aufgeschmissen! Und zu allem Überfluß gibt's ein wesentlich billigeres Programm im Club, welches ab einem Gesamtspeicherplatz von 128k im MTX incl. Speichererweiterung weit mehr Formate zur Verfügung stellt, aber ohne jegliche Einschränkungen: **RAM 3.4** von Bernd Preusing!

Hardware: CoProzessoren / SDX und Bdos Err Bad Sector**68000, 68020, 8086 oder 80286 am MTX ?** (Michael Möwe, 2000)

Wie soll das realisiert werden, wo liegt der Nährwert ?

Eine Möglichkeit eine Verbindung des MTX mit einem dieser Mikro-Professoren herzustellen wäre die, sich einen SAM-68K zu kaufen und den MTX als 'intelligentes Terminal' zu benutzen, oder eine ähnliche Lösung. Weder CP/M noch ein unter CP/M geschriebenes Programm unterstützt Co-Prozessoren, dieses Unterfangen gleicht dem Versuch aus einem Teekessel einen Helikopter zu machen.

Wenn es nur darum geht diese Prozessoren 'näher kennenzulernen', so bietet das Ingenieur-Büro Wilke diverse Cross-Assembler und Runtime-Simulatoren für diese Prozessoren -unter CP/M- an (z.B. OPAL, siehe auch: c't 7/86, Seite 104).

Dabei möchte ich hervorheben, daß man sich mit den Prozessoren 8086, 80186 und 80286 nur dann auseinandersetzen sollte, wenn man unbedingt muß. Diese Prozessoren besitzen einen linearen Adressraum von nicht mehr als 64k-Byte, und das kann unsere alte Z80 auch noch. Was diese CPU's sonst noch auszeichnet ist eine kranke Architektur und die sich daraus ergebende Schwierigkeit sie vernünftig zu programmieren. Ich glaube schon, daß ich weiss wovon ich rede, denn leider muß ich so eine Krücke in Assembler programmieren (vorher hab' ich ein wenig mit dem 68010 gespielt, das hat wenigstens Spaß gebracht).

Wer Interesse an zukunftsweisenden Prozessoren hat, dem sei mitgeteilt, daß es ein 'Transputer-Evaluation-Board' gibt, das über 2 serielle Schnittstellen betrieben werden kann. Der Preis dieses 'Boards' dürfte aber wahrscheinlich den unseres guten alten MTX + FDX übertreffen.

Nähere Informationen über Transputer und OCCAM:

ASTEK Elektronik Vertriebs GmbH,

Gottlieb-Daimler-Str. 7, 2358 Kaltenkirchen, Tel.: 04191 / 87 11

c't Hefte 1/85, 7 & 8/86

**BDOS Err: Bad Sector - ein Muß bei der SDX ?** (Andreas Illgner, 3000)

Man ist ja von Memotech einiges gewohnt, und das Kürzel dSM ist ja mittlerweile wohl jedem MTX'ler bekannt, aber bei der Entwicklung des SDX muß es wohl bei Memotech eine ganze Woche dunkel gewesen sein.

Du wirst jetzt fragen warum so böse Worte. Nun wie in INFO 13 zu lesen war und wie vielleicht einige Besitzer des SDX auch festgestellt haben funktionieren die Programme FRMSDX und FORMAT zum Formatieren von Disketten nicht immer einwandfrei. Ferner trat bei mir in einigen Programmen z.B. Newword häufig folender Fehler auf:

Bei häufigen Diskettenzugriffen (z.B. lange Texte im Newword durchblättern) wurden zum Teil Sektoren "zerstört" und beim Versuch sie zu lesen erfolgte die Meldung 'Bdos Error on X: Bad Sector'.

Da ein vernünftiges Arbeiten zeitweise nicht mehr möglich war machte ich mich auf die Suche nach dem Fehler. Mir fiel auf, daß er meistens auftrat wenn der Motor im Laufwerk eingeschaltet wird. Das Laufwerk schien zu langsam zu sein. Das ist aber im Normalfall nicht möglich, denn das Laufwerk meldet dem Floppycontroller über die READY-Leitung wenn es bereit ist zum Lesen oder Schreiben von Daten. Also untersuchte ich die Leitung zwischen Laufwerk und Controller auf schlechte Kontakte.

Hardware: SDX und Bdos Err Bad Sector

Dabei stellte sich heraus, daß das READY-Signal überhaupt nicht angeschlossen ist. Bei Memotech ist man offensichtlich der Meinung, auf die Abfrage verzichten zu können. Stattdessen geht man einen recht ungewöhnlichen Weg:

Da der Controller an seinem READY-Eingang das READY-Signal braucht erzeugt man es softwaremäßig und gibt es über ein Port (Bit 5 Control Byte) aus. Das BIOS prüft ob sich der Controller mit READY meldet. Ist das nicht der Fall, so wird gleichzeitig 'Motor ein' und 'Motor ready' ausgegeben und eine Pauseschleife aufgerufen. Nach der Pause wird der Controller erneut abgefragt und der meldet sich jetzt natürlich, unabhängig davon ob das Laufwerk fertig ist oder nicht, mit Ready.

Das Laufwerk benötigt laut Datenblatt bis zu 500 msec bis der Motor seine Geschwindigkeit erreicht hat. Das SDX wartet aber nur knapp 170 msec und fährt dann mit den nächsten Befehlen fort. Das führt z.B. bei den Formatierprogrammen dazu, daß die ersten Sektoren nicht korrekt geschrieben werden, weil der Motor noch zu langsam läuft.

Nun kann es sein, daß einige Laufwerke tatsächlich in dieser kurzen Zeit hochlaufen. Die glücklichen Besitzer eines solchen Laufwerks könnten jetzt eigentlich weiter blättern. Aber vielleicht tritt der Fehler später (z.B. beim Anschluß eines zweiten Laufwerks) doch noch auf. Das kann bei dem etwas schwächlichen Netzteil durchaus passieren.

Es gibt nun zwei Möglichkeiten das Problem zu erledigen:

- Version 1

Man kann einfach die Pause verlängern und dem Laufwerk mehr Zeit zum hochlaufen lassen. Das war auch meine erste Lösung. Der Vorteil liegt darin, das man keine Hardwareänderungen vornehmen muß. Der Nachteil liegt in der zum Teil wesentlich längeren Zeit, die für Diskettenoperationen benötigt wird. Außerdem läßt sich der Fehler damit nicht völlig ausschließen, da es trotzdem (sehr selten) vorkommen kann, daß das Laufwerk noch nicht zum schreiben oder lesen bereit ist.

- Version 2

Die zweite wesentlich elegantere Lösung liegt darin, daß man den Controller mit dem READY-Signal verbindet. Dabei ist es aber notwendig, mit einem Lötkolben zu arbeiten. Ferner ist es sinnvoll, eine neue Routine zu schreiben, die das READY-Signal abfragt.

Da ich weiß, daß es für manchen nichts schlimmeres gibt als die schwarze Kiste aufzuschrauben, ich aber mehr zu der 'eleganten' Lösung tendiere, habe ich mich entschlossen beide Varianten ausführlich darzustellen.

Doch zuvor will ich die Änderungen in den Formatierprogrammen beschreiben. Diese Änderungen sind für beide Versionen notwendig. Auch wer bisher keine Schwierigkeiten hatte und ohne BIOS Änderung weiterarbeitet sollte diese Korrekturen durchführen, da sie Fehler im Programm beheben, die mit dem Umbau zum Teil nichts zu tun haben. Die Darstellung in gewohnter Form mit '/=RET' und alle Zahlen in Hexadezimalform.

Hardware: SDX und Bdos Err Bad Sector

1. für FRMSDX.COM

```
DDT FRMSDX.COM/ S58B/ 6E/ ./ S590/ 6E/ ./ S5CF/ 00/ 10/ ./ ^C
SAVE 5 FRMSDXN.COM
```

2. für FORMAT.COM

```
DDT FORMAT.COM/ S57E/ 6E/ ./ S583/ 6E/ ./ S5C2/ 00/ 10/ ./ ^C
SAVE 5 FORMATN.COM
```

Beide Programme enthalten übrigens noch einen weiteren Fehler, der sich jedoch nur durch einen neuen Assembler-Lauf beheben läßt. Dieser Fehler tritt immer auf, wenn man mit F.COM die Funktionstasten belegt hat. Versucht man, nachdem FORMAT oder FRMSDX gelaufen sind die Tastenbelegung zu ändern so führt das zum Absturz des Rechners.

Doch nun wieder zum SDX selber:

Zuerst die einfachere Version. Dafür benötigen wir ein Programm das bei jedem Kaltstart automatisch aufgerufen wird. Man könnte auch ein extra Programm schreiben. Da es sich jedoch nur um zwei Befehle handelt, die eingebaut werden müssen heißt das mit Kanonen auf Spatzen schießen.

Die beiden Befehle sind:

MVI A,10H	bzw.	LD A,10H
STA OFE56H		LD (OFE56H),A
im 8080-Code		im Z80-Code

Mein Vorschlag wäre der Einbau in CONFIG. Das geht folgendermaßen:

1. Mit **DDT CONFIG.COM/** das Programm in den Rechner laden.
2. Mit **A100/ CALL 8A5//** (das sind 2 Return) wird der erste Befehl (bei CONFIG ist das der Befehl LXI H,0000) durch den Unterprogrammaufruf CALL 08A5 ersetzt. Bei anderen Programmen kann der Befehl auch zwei oder sogar drei Befehle überschreiben.
3. Jetzt werden die beiden obenstehenden Befehle wie folgt ab 08A5 (da ist das Config zuende) in den Speicher geschrieben:  
**ABA5/ MVI A,10/ STA OFE56/**
4. Jetzt mit **LXI H,0/** den bei 100 überschriebenen Befehl hier einfügen und das "Unterprogramm mit **RET/** beenden. Nun mit einem weiteren / die Befehlseingabe beenden und das DDT mit ^C verlassen.
5. Zuletzt mit **SAVE 8 CONFIGP.COM** das geänderte Programm wieder auf die Diskette schreiben.

Noch einmal die Eingabe zusammenhängend:

```
DDT CONFIG.COM/ A100/ CALL8A5// ABA5/ MVI A,10/ STA OFE56/ LXI H,0//
^C
SAVE 8 CONFIGP.COM
```

Das war für die erste Version schon alles. Die zweite Version ist da schon etwas komplizierter. **Allen die jetzt resignierend aufhören zu lesen, sei die Lektüre des letzten Abschnitts empfohlen.**

Doch beginnen wir mit der Hardwareänderung. Dazu sind die folgenden Teile notwendig:

- etwa 10 cm isolierten Draht
- etwa 2cm Isolierschlauch oder etwas Isolierband
- 2 Widerstände, einen mit 220 Ohm und einen mit 1 Kiloohm

Hardware: SDX und Bdos Err Bad Sector

Zuerst müßt ihr die Controller-platine 'enthüllen'. Das Gehäuse sollte wohl jeder aufkriegen. Merkt euch gleich wie das Kabel zum Laufwerk angeschlossen ist damit beim Zusammenbau kein Fehler gemacht wird (bei mir rote Markierung auf der linken Seite).

Die Platine selbst wird von zwei Schrauben an der Oberseite gehalten (siehe Bild 1). Um Verwechslungen mit den Bezeichnungen oben, unten, rechts ,usw. zu vermeiden habe ich in den Bildern 1 und 2 die Lage der Platine festgelegt.

Fangen wir auf der Bauteileseite (Bild 1) an:

Man kann leider nicht einfach den READY-Anschluß von Laufwerk mit dem Controller verbinden, weil der Ausgang am Laufwerk invertiert ist. Es muß also ein Inverter zwischengeschaltet werden. Glücklicherweise ist ein solcher Inverter auf der Platine noch ungenutzt und kann hierfür verwendet werden. Es handelt sich hierbei um das IC 8 in der linken oberen Ecke. Für Fachleute: es handelt sich um einen 7406 mit open-collector Ausgang. Deshalb sind Pull-up-Widerstände erforderlich.

Die Widerstände werden wie in Bild 3 dargestellt angelötet. Der linke Anschluß für beide Widerstände ist das Loch oberhalb des IC's. Das ist die 5 V Versorgungsspannung. Der rechte Anschluß des 220 Ohm Widerstandes (das ist der mit der Farbfolge rot-rot-braun) wird an Pin 5 von IC 8 angelötet (Pin 1 ist rechts unten). Der 1 Kiloohm Widerstand (braun-schwarz-rot) wird mit der rechten Seite an Pin 6 angeschlossen. Man sollte darauf achten, daß keine Kurzschlüsse entstehen. Daher habe ich über die Anschlüsse der Widerstände ein Stück Isolierschlauch gezogen. Es geht aber auch wenn man die Anschlüsse mit Isolierband umwickelt. Wichtig ist auch, daß der linke Anschluß der Widerstände isoliert ist, da beim Zusammenbau der Metallwinkel der die Rückwand bildet leicht einen Kurzschluß verursacht. Ferner sollte man beim Anlöten der Widerstände an das IC ein Lötzeit von 5 Sekunden nicht überschreiten um Hitzeschäden am und im IC zu vermeiden.

Jetzt gehen wir ziemlich genau in die Mitte der Platine nach IC 4. Das ist das große 40 polige IC. Wir müssen hier die alte Ready-Leitung, die vom PROM kommt unterbrechen. Das ist in Bild 5 dargestellt. Wichtig ist hierbei, daß die Leiterbahn unterhalb des LötAuges unterbrochen wird. Die Verbindung vom Lötauge nach Pin 6 von IC 2 muß erhalten bleiben. An das Lötauge wird später an der Unterseite das neue READY-Signal angeschlossen. Ferner sollte man darauf achten, daß man nur die eine Leiterbahn unterbricht und nicht einmal über die gesamte Platine kratzt. **Anm.d.HH.** Das geht gut mit einem scharfen Messer.

Das war auf der Bauteileseite alles. Wir drehen nun die Platine so um, daß sie so liegt wie in Bild 2.

Mit dem isolierten Draht werden jetzt zwei Verbindungen hergestellt. Die erste geht von Pin 1 des Laufwerksteckers (Nr steht auf der Bauteileseite) nach Pin 5 von IC 8 (Pin 1 ist jetzt links unten). Das ist in Bild 4 dargestellt. Die zweite Verbindung geht von Pin 6 (IC 8) nach dem schon mehrfach erwähnten Lötauge. Wo sich das Lötauge auf der Unterseite genau befindet, kann man aus Bild 6 ersehen. Es ist mit dem Controller (IC 4) Pin 32 verbunden. Man sollte den Draht aber nicht direkt am Controller anlöten, da das IC ziemlich empfindlich und relativ teuer ist.

**Anm.d.HH.** Beim Löten solltet Ihr einen feinen LötKolben verwenden, und auf keinen Fall mit Lötzinn arbeiten, da das spritzt!!

Hardware: SDX und Bdos Err Bad Sector

Nun könnt Ihr Euch an den Zusammenbau des Gehäuses machen. Dabei sind zwei entscheidende Dinge zu beachten. Zum einen darf, wie schon erwähnt, der Metallwinkel keinen Kurzschluß mit den eingebauten Widerständen verursachen. Zum zweiten sollte man beim festziehen der beiden Schrauben, die die Platine auf der Grundplatte festhalten, mit äußerster Vorsicht vorgehen. Zieht man sie nämlich zu fest an, so kann es sehr schnell zum Platinenbruch kommen.

Nun müssen wir uns mit der erforderlichen Softwareänderung befassen. Es ist möglich nur die Pauseschleife zu verlängern wie schon beschrieben. Dann hätte man sich aber auch gleich den Umbau sparen können. Daher habe ich eine neue Routine geschrieben, die das READY-Signal abfragt. Die Beschreibung für den Einbau in ein Programm läßt sich aber sehr schlecht allgemein halten (für beliebige Programme). Ich beschreibe daher hier nur den Einbau in das CONFIG und gebe hinterher noch das Listing in Z80 Code an. Wem das beides nicht reicht den kann ich nur auf mein Angebot im letzten Abschnitt verweisen.

Hier nun die erforderlichen Eingaben:

```
DDT CONFIG.COM/ A100/ CALL 8A5// 8BA5/ 01/ 22/ 00/ 21/ B4/ 08/ 11/ 3A/
FE/ ED/ B0/ 21/ 00/ 00/ C9/ CD/ 60/ FE/ C8/ 3E/ 04/ 47/ CD/ 0B/ FE/
CD/ 4B/ FE/ C8/ 3E/ 09/ C9/ 01/ 00/ 10/ CD/ 1B/ FE/ CD/ 60/ FE/ C8/
0B/ 79/ B0/ 20/ F4/ 3C/ C9/ ./ ^C
SAVE 8 CONFIGR.COM
```

und hier das Listing der Routine:

```

; der erste Teil verschiebt die neue RADY-Routine ins Bios
LD      BC,0022H    ; das ist die Länge der Routine
LD      HL,ADR1    ; das ist die Adresse wo die Routine steht
LD      DE,0FE3AH  ; und hier soll die Routine hin
LDIR                   ; jetzt das ganze verschieben
RET     oder JP xxxx ; und jetzt zurück zum Hauptprogramm

; Hier fängt die neue Routine an
ADR1:   CALL     OFE60H
RET     Z
LD      A,04H
LD      B,A
CALL    OFE0BH
CALL    OFE4BH
RET     Z
LD      A,09H
RET
LD      BC,1000H
CALL    OFE1BH
CALL    OFE60H
RET     Z
DEC     BC
LD      A,C
OR      B
JR      NZ, OFE4EH
INC     A
RET
```

Hardware: SDX und Bdos Err Bad Sector

Noch einige Hinweise: Wer die Routine mit DDT ins CONFIG einbaut (siehe oben) sieht beim auflisten ein anderes Bild. Das liegt daran, daß das DDT im 8080 Code arbeitet und daher einige Befehle nicht kennt. Das H hinter den Zahlen steht für Hexadezimal.

Soweit mein Vorschlag für den Umbau. In der Zeit in der dieser Text Ich hoffe es hilft euch wenigsten etwas. Eine Berichtigung ist noch notwendig: Ich habe im letzten Info leichtsinnigerweise behauptet man könne mit dem SDX kein Singel-Density Formate lesen. Das war eine Falschmeldung. Ich war nur wiederum zu blind um eine Leiterbahn auf der Platine zu verfolgen. Also noch einmal: Auch Singel-Density Formate können gelesen und geschrieben werden.

Ich kann mir vorstellen, daß diejenigen von euch, die nicht viel (oder gar keine) Erfahrung im Umgang mit dem LötKolben haben vor dem Umbau zurückschrecken. Da ich aber auch den Frust kenne, wenn wiedereinmal ein mühsam eingetippter Text zerstört wird, mache ich den 'Bastelfeinden' unter euch folgende Angebote:

1. Ich bin bereit für 20.- DM incl. Porto und Verpackung den Umbau des SDX vorzunehmen. Schickt mir aber bitte nur die Controller-Platine und nicht auch noch das Gehäuse. Das spart Porto!!
2. Wer mir eine leere Diskette mit einem an sich adressierten und frankierten Rückumschlag schickt, erhält von mir kostenlos die korrigierten Programme FRMSDX.COM und FORMAT.COM. Auch der oben beschriebene Fehler mit den Funktionstasten ist darin behoben. Ferner füge ich ein geändertes CONFIG.COM bei, das die Warteschleife wie oben beschrieben verlängert, bzw. das die READY-Routine im Bios überschreibt (je nach Wunsch).
3. Für 5.- DM + frankiertem Rückumschlag baue ich die neue READY Routine oder auf Wunsch nur die "Pauseverlängerung" in ein mir auf Diskette geschicktes Programm ein. Wer mir ein Assembler-Lisning des Programms schickt braucht nur den Rückumschlag beilegen (natürlich auch frankiert) und spart die 5.- DM. Dabei wäre es sehr nützlich wenn ihr kurz beschreibt, was das Programm macht.

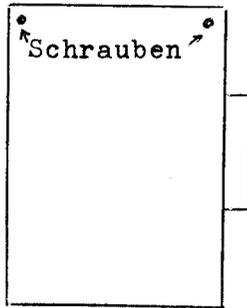
Bitte schickt mir nur einen funktionsfähigen Controller und keine kaputt gebastelten Geräte. Ich übernehme keine Garantien und keine Haftung für eventuell auftretende Schäden. Auch Terminzusagen kann ich nicht machen. Wenn ihr eure "Kiste" dringend braucht ruft mich am besten vorher an.

Das Geld überweist bitte auf mein Konto (Nr.:831395 Volksbank Hannover BLZ 251 900 01 und Absender angeben) und die Bestellung (oder den Controller) an Andreas Illgner, Große Düwelstr.15, 3000 Hannover 1

Hardware: SDX und Bdos Err Bad Sector

Bauteilseite

Lötseite



Steck-  
verbinder

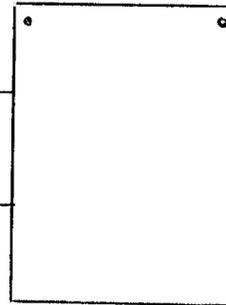


Bild 1

Bild 2

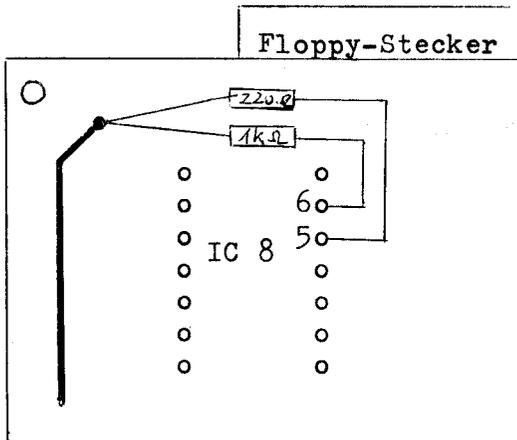


Bild 3

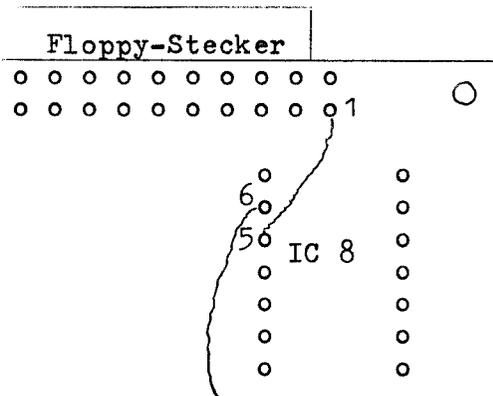


Bild 4

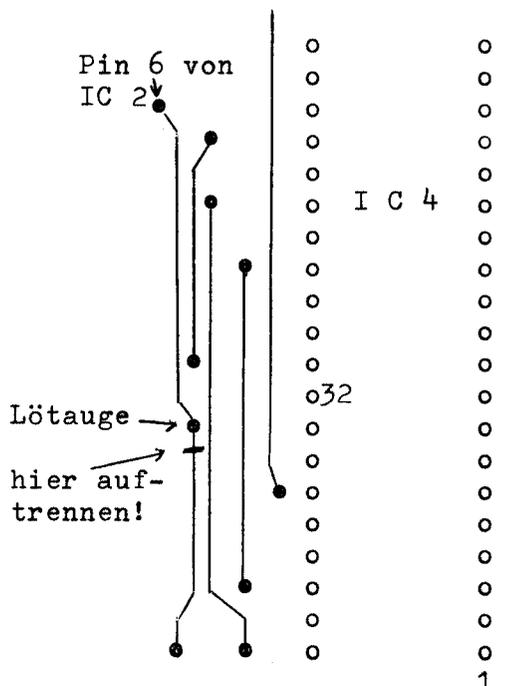


Bild 5

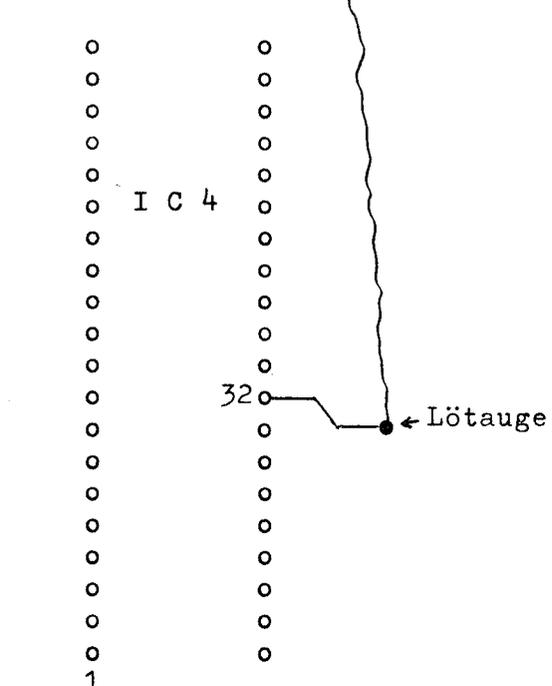
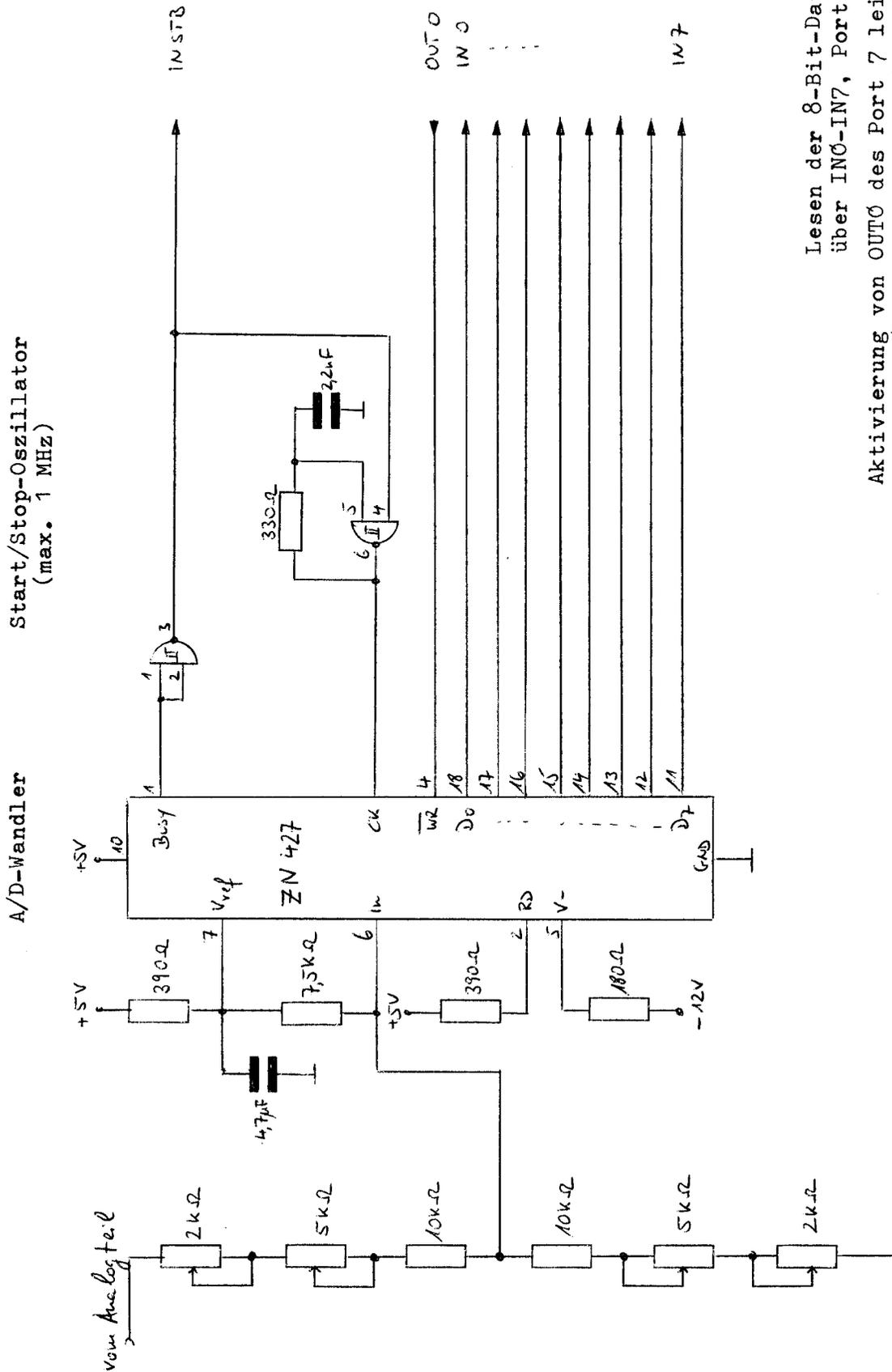


Bild 6

Hardware: Schaltplan 2-Kanal A/D-Wandler

(Manfred Flume, 5600)



Lesen der 8-Bit-Daten über IN0-IN7, Port 7

Aktivierung von OUT0 des Port 7 leitet eine A/D-Wandlung des anliegenden Analogsignals ein. Nach der Umwandlung kann dieses Signal mittels IN A,(7) (Assembler) bzw. A=INP(7) (Basic) eingelesen werden.

2-Kanal A/D-Wandler (Digitalteil)

Hardware: Schaltplan 2-Kanal A/D-Wandler

Umschalter  
Kanal 1/2

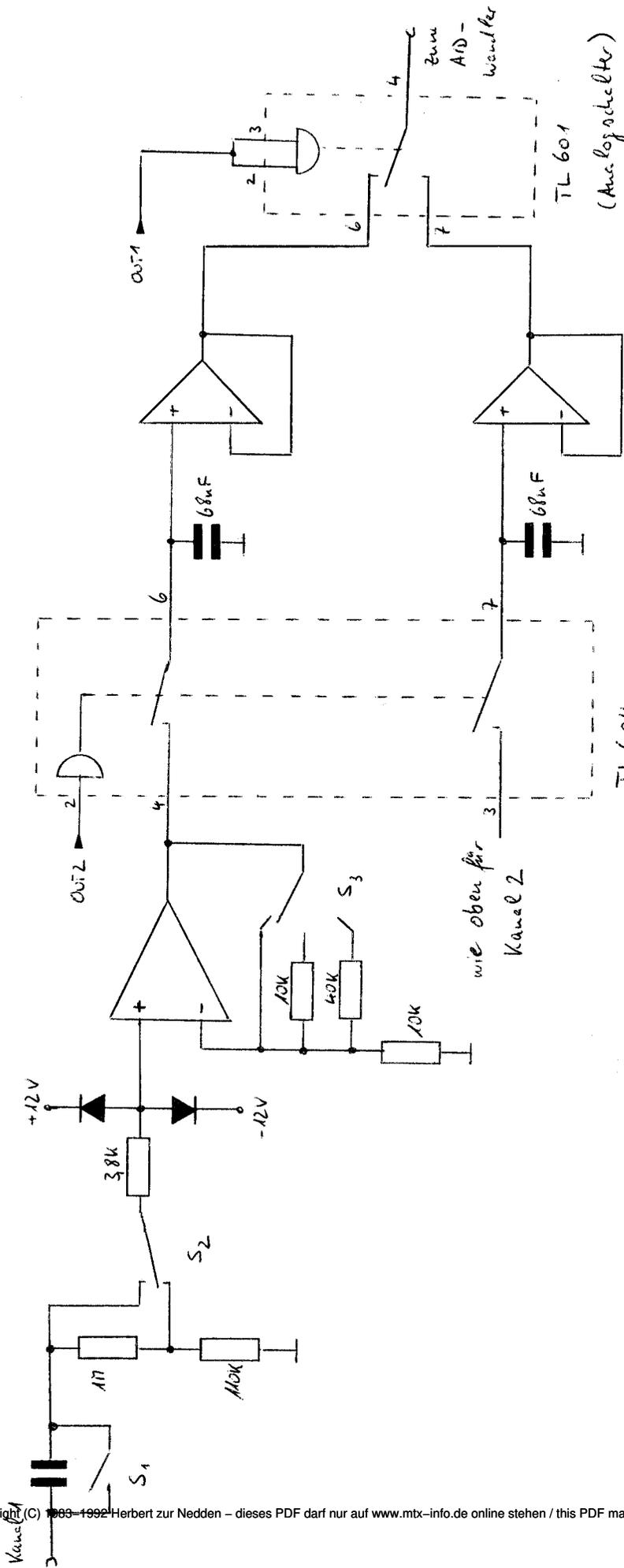
Sample and Hold  
Schaltung

Verstärker  
1,2,5 Teilung

Schutz-  
schaltung

Vorteiler  
(1:1, 10:1)

AC / DC-  
Schalter



TL604  
(Analogschalter)

TL601  
(Analogschalter)

Signal OUT2 von Port 7  
steuert Sample/Hold

Signal OUT1 von Port7  
wählt Kanal aus

wie oben für  
Kanal 2

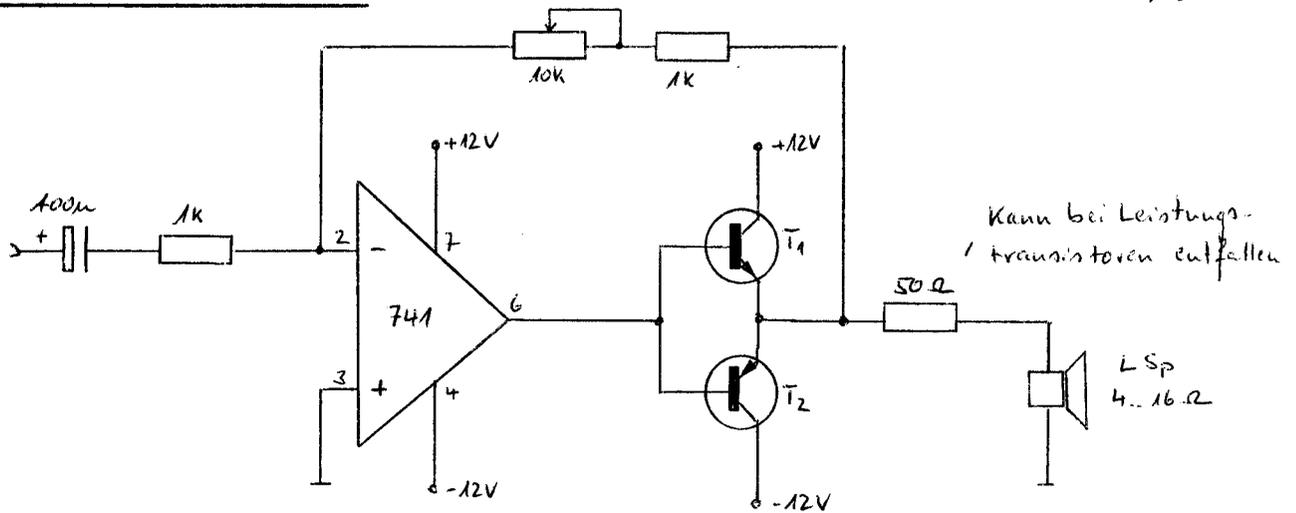
2-Kanal A/D-Wandler (Analogteil)

Vor dem IC TL604 ist nur Kanal 1 gezeichnet.  
Kanal 2 ist genauso aufzubauen.

Hardware: Schaltplan Baudraten-Generator / Verstärker

Einfacher Audio-Verstärker

(Manfred Flume, 5600)



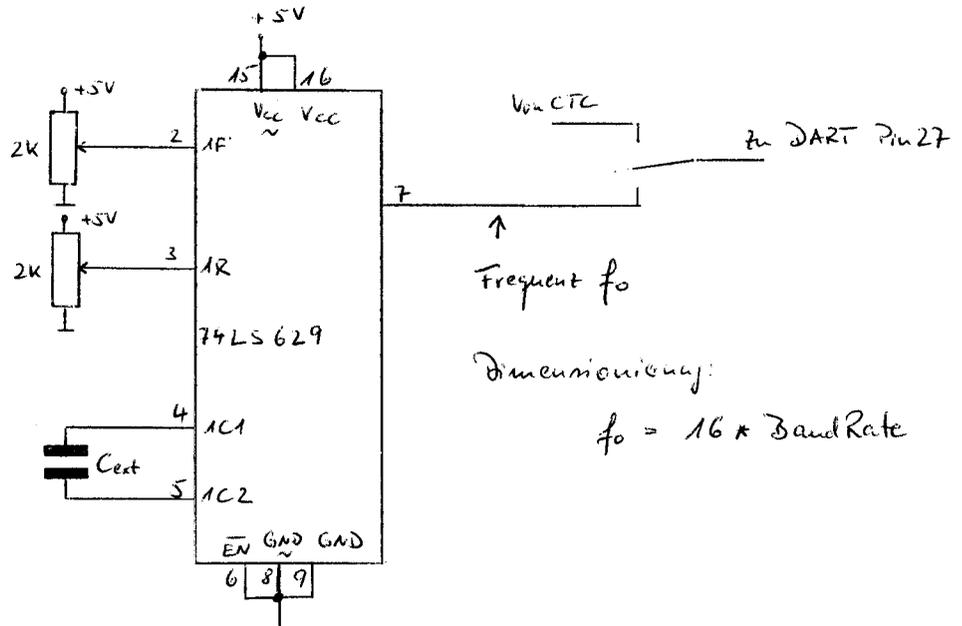
T<sub>1</sub> : BCY 58 (NPN) o.ä.  
T<sub>2</sub> : BCY 78 (PNP)

Baudraten-Generator für RS 232, Kanal B

(Manfred Flume, 5600)

Die Unterbrechung CTC - DART wird am besten an Pin 30 der RS 232 -Karte (oben hinten) für Kanal B der Schnittstelle vorgenommen.

Mit dieser Schaltung sind deutlich höhere Baudraten einstellbar, z.B. zur vernünftigen Ansteuerung der TERM 1 - Grafikkarte.



**WICHTIG:**

Datenblätter zu den Spezial-ICs, die für obige Schaltungen und den 2-Kanal

AD/Wandler benötigt werden gibt es bei Herbert Herberg, 2000 Hamburg 76

Sorry, aber diese Seite musste ich entfernen, weil ich für eine Veröffentlichung außer in der ehemaligen, gedruckten Form keine Freigabe habe.

Sorry, aber diese Seite musste ich entfernen, weil ich für eine Veröffentlichung außer in der ehemaligen, gedruckten Form keine Freigabe habe.

Sorry, aber diese Seite musste ich entfernen, weil ich für eine Veröffentlichung außer in der ehemaligen, gedruckten Form keine Freigabe habe.

Sorry, aber diese Seite musste ich entfernen, weil ich für eine Veröffentlichung außer in der ehemaligen, gedruckten Form keine Freigabe habe.

Sorry, aber diese Seite musste ich entfernen, weil ich für eine Veröffentlichung außer in der ehemaligen, gedruckten Form keine Freigabe habe.

Sorry, aber diese Seite musste ich entfernen, weil ich für eine Veröffentlichung außer in der ehemaligen, gedruckten Form keine Freigabe habe.

TURBO: Probleme mit Windows und RunTimeErrors**TURBO-WINDOWS und der Run-Time-Error** (Olaf Krumnow, 2000)

Seit RAM3 läßt unser Bildschirmtreiber Fensterdefinitionen zu, die z.B. von WINDW3 fleißig genutzt werden und auch ansonsten recht nützlich zur Anwendung in eigenen Programmen sind, weil sie oftmals den Bildschirmaufbau vereinfachen. Leider besteht bei der Programmentwicklung unter TURBO-Pascal ein ärgerliches Problem. Tritt nämlich ein Run-Time-Error (analog dazu User-Break oder I/O-Error) auf, so behält TURBO (bzw. der Bildschirmtreiber) strikt die letzte Fenstergröße bei. Wer mal versucht hat, mit einem kleineren Fenster Programmtext zu editieren, weiß was ich meine. Abhilfe schafft hier ein ^X, daß an den Bildschirmtreiber geschickt werden muß, aber wie??? Normalerweise geht das nur, indem TURBO verlassen wird, dann das ^X an den Bildschirmtreiber geschickt wird und man zuletzt wieder TURBO aufruft. Das ist natürlich ziemlich nervig. Bleibt wieder nur das Patchen. Also nichts wie ran: Mit MONI 2.00 ist die Sache ganz einfach, mit einer älteren Version oder gar DDT (VDEB läuft unter RAM3 ja nicht mehr) dauerts etwas länger: (wie gewohnt: der / bedeutet <RET>, Deine Eingaben sind in fett gedruckt, die unterstrichenen Angaben sind für die Hex-Eingabe mit MONI 1.36 oder DDT. Alle Angaben für TURBO 3.0, zu TURBO 2.0 s.u.)

**MONI2 TURBO.COM/** (alle Angaben beziehen sich auf TURBO 3.0)  
**1A20D4/** (Modus 1 und Assemblereingabe ab 20D4H)  
**CALL xxyy** (xxyy ist ein unbenutzter Speicherbereich von  
CD/ yy/ xx/ mindestens 9 Bytes Länge, am besten zwischen 103H und 19FH. Hast Du TURBO bisher nicht gepatched, so kannst Du für xxyy ruhig 103 einsetzen. Wir setzen hier ein CALL auf diese Adresse ein. Der alte Befehl wird überschrieben und muß vom Unterprogramm nachgeholt werden)

**<BRK>**  
**Axxyy/** (xxyy ist die gleiche Adresse wie oben)  
**LD C,1B/** (CTRL-X nach C laden)  
OE/ 1B/  
**CALL F400/** (Bildschirmtreiber direkt aufrufen)  
CD/ 00/ F4/  
**LD A,(DB)/** (den Befehl nachholen, der bei 20D4 überschrieben wurde)  
3A/ DB/ 00/  
**RET/** (Rücksprung zum Aufrufer)  
C9/

**<BRK>**  
**OW\*TURBON.COM/** (unter neuem Namen abspeichern)

Das ist schon alles. Besitzer von älteren MONI-Versionen müssen die Assemblerbefehle als Hex-Zahlen mit dem C-Kommando in Modus 1 eingeben, DDT-Besitzer nehmen den S-Befehl. Dort müssen jeweils die unterstrichenen Werte eingegeben werden!

Was haben wir jetzt überhaupt gemacht? Nach jedem Fehler oder Programmabbruch (halt-Kommando) springt TURBO-Pascal früher oder später an die Adresse 20D4H. Dort steht nach dem Patchen ein Unterprogrammaufruf nach xxyy. Dieses Unterprogramm sendet ein CTRL-X an den Bildschirmtreiber, der daraufhin alle Fensterdefinitionen löscht, holt dann den überschriebenen Befehl nach und springt zurück nach 20D7H, wo TURBO dann normal weitermacht. Wir haben also danach den ganzen Bildschirm wieder und brauchen TURBO nicht zu verlassen.

TURBO: Probleme mit Windows und RunTimeErrors / Leserbrief

Jetzt kommt bestimmt wieder ein Dauer-Besserwisser (Hallooo MM!) und sagt: 'Schreibe Dir doch eine eigene Fehlerroutine (erst ab TURBO 3.0 möglich, ErrorPtr usw.), die das CTRL-X ausgibt und dann das Programm beendet'. Die Sache hat leider den Nachteil, daß TURBO dann nicht mehr selbständig die Fehlerstelle sucht, keine Fehlermeldung gibt usw., ist also auch nicht so ideal.

Was machen Leute mit TURBO 2.0 (oder gar 1.0)? Das ist relativ einfach: Ein kurzes Programm schreiben (nur **begin end.**), als COM übersetzen und mit MONI kurz verfolgen (T-Kommando!). Zunächst kommt ein JP 1FD9H, dann eine kurze Initialisierung aus 6 oder 7 Befehlen, dann ein absoluter Sprung an eine Adresse kleiner 2000H. An dieser Adresse muß dann der CALL xxyy (wie oben beschrieben) stehen. Auf keinen Fall vergessen, den überschriebenen Befehl im Unterprogramm nachzuholen!! Er kann anders als bei TURBO 3.0 sein!!

**Betrifft: Versand von Soft- und Hardware** (Olaf Krumnow, 2000)

Seit einiger Zeit verkaufe ich über den Club Software. Dadurch sind mir einige Dinge aufgefallen, die mir vorher nicht so klar waren, und denen, die kein Software verkaufen, auch nicht unbedingt klar sein dürften. Es ist für den Verkäufer und den Käufer sicherlich von Vorteil, wenn die Beträge direkt auf das Empfängerkonto überwiesen werden und nicht per V-Scheck bezahlt werden. Damit das deutlich wird, ein kleiner Zeitvergleich.

## Bezahlung per V-Scheck:

Brief mit Bestellung und V-Scheck	: 1-3	Tage (Verlustrisiko)
Einlösung des V-Schecks	: 10-14	Tage
<u>Absendung der Software</u>	<u>: 1-2</u>	<u>Tage</u>
Summe	: 12-19	Tage

## Bezahlung per Überweisung:

Überweisung mit Bestellung	: 2-4	Tage
<u>Absendung der Software</u>	<u>: 1-2</u>	<u>Tage</u>
Summe	: 3-6	Tage

Auf gut deutsch, wenn Ihr das Geld direkt überweist, habt Ihr Eure Software in weniger als einer Woche, mit V-Scheck kann es aber durchaus 3 Wochen dauern, und wer wartet schon gerne. (Daß die Softwareversender bei V-Schecks den Eingang des Geldes abwarten, ist doch verständlich, oder?). Dazu müssen die Verkäufer natürlich auch ihre Bankverbindung angeben, klaro? Dritte Möglichkeit wäre, den Betrag in bar oder in Briefmarken beizulegen (auch das Verlustrisiko). Das geht natürlich genauso schnell oder schneller als die Überweisung. Bei Briefmarken sollte dann aber bitte beachtet werden, daß wir mit den Dingen auch was anfangen können (von den SDM-Marken braucht man nunmal nicht so viele). Die günstigsten Beträge sind -.10, -.60, -.80, 1.10, weil diese doch recht häufig benutzt werden.

Ferner kommt es (leider) manchmal vor, daß einige Leute vergessen, den Betrag für D&P&V mitzubezahlen. Es ist ja leider nicht so, daß wir diese Sachen geschenkt bekommen. In dem Fall gibt es prinzipiell zwei Möglichkeiten: a) per Nachnahme das zusätzliche Geld eintreiben, was aber zusätzlich noch einiges Geld an die Post kostet, oder b) den Betrag über das Clubkonto verrechnen, was Herbert wieder zusätzliche Arbeit macht. Beide Arten sind nicht schön (ich verwende b), also bitte bei der Bestellung doch nochmal kontrollieren, ob dieser Betrag dabei ist.

Olaf Krumnow