

MTX *User-Club Deutschland*

Info 46
02.06.1992

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 **Selbstgeschriebene**): Tausch von kurzen und einfachen Routinen. Gute Programme (mit Dokumentation) können über den Club an Mitglieder verkauft oder auf Public-Domain werden. Wer solche Programme an uns schickt erhält ggf. Verbesserungshinweise und eine Besprechung im Info.

Mitglied kann jede(r) werden! Keine Beitragsgebühr! Anmeldung kostet DM 2.-.

Verpflichtungen: Einsendung unseres Anmeldeformulars.

Bitte: Einsendung von Tips & Tricks, Fragen, Antworten, Routinen, Programmen, Beiträgen und Anregungen zum Info, Hinweisen auf preiswerte Hardware und Software, und was so zusammenkommt und andere interessieren könnte.

Club-Info, unser Blatt, verschicken wir ca. 8-wöchentlich. Inhalt ist alles was uns über den MTX/FDX (ohne Copyright) in die Hände fällt. Es kostet nicht über DM 12.- je Exemplar. Jeder kann dazu Beiträge liefern und gratis Kleinanzeigen veröffentlichen.

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

Geld/Konto: Für jedes Mitglied führt Herbert zur Nedden 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!

Einzahlungen bitte auf's Club-Konto: (oder V-Scheck)
(Absender! incl Name und Anschrift bitte nicht vergessen!)
Postgiroamt Hamburg, BLZ 200 100 20,
Herbert zur Nedden, Sonderkonto C, Nr. 3480 00-200

Kontaktadressen:

Herbert zur Nedden
Alte Landstraße 21
2071 Siek
(04107) 99 00

Hans Gras
Statenhoek 49
NL 1506 VM Zaandam
(0031-75) 17 49 91

Telefon-Sprechzeiten

Herbert zur Nedden: Do 18 - 21 Uhr, Sa 9 - 12 Uhr
(Etwas klingeln lassen oder nochmal versuchen!)

Inhaltsverzeichnis

C l u b:	
Fragen/Antworten	Seite 2
Korrektur/Nachtrag	Seite 2
Philosophisches	Seite 20
W o r d s t a r:	
Rekursive Macros	Seite 3
H a r d w a r e:	
8 MHz	Seite 4
MTX-Tasten selbst reparieren	Seite 7
Edicta-Grafikkarte	Seite 8
Die Festplatte	Seite 10
L e s e r b r i e f:	
Hartmut Traber, 5270	Seite 6
T u r b o - P a s c a l:	
Bug und Patches	Seite 9
S o f t w a r e:	
RAM 6.2 (b)	Seite 11
Diverses	Seite 22
d B A S E:	
Ein paar Tips	Seite 12
K o m i k:	
Bart	Seite 2
Anlage:	
Gesamtinhaltsverzeichnis der Infos 1 bis 46	

Preis für dieses Info: DM *12.-*

Neue KLIICK (Herbert zur Nedden, 2071)
 KLIICK.021 ist fertig: DiJey 3.03, LOAD 6.2 (kann mehrere KLIXer auf einmal laden), QL42 (mit schnellerem Un-LZH), LoadDir 3, Turbo-Patches 1.00 (machen Turbo Pascal ZCPR 3.3-fähig).

Clubtreffen (Herbert zur Nedden, 2071)
 Ist dieses Jahr mangels Interesse ausgefallen!

Kontostand (Herbert zur Nedden, 2071)
 Eine rote Markierung auf dem Umschlag bedeutet, daß er zu niedrig ist.

Anzeigetexte samt Absender bitte schriftlich an Herbert zur Nedden!

V E R K A U F (Preise sind i.a. ohne Porto & Verpackung)

Herbert zur Nedden, Alte Landstr. 21, 2071 Siek, 04107 - 9900:

- > Interessiert Dich einer der von mir angebotenen Posten: Mach ein Angebot!
Meine hier genannten Preise sind nicht unbedingt unumstößlich!
(Neue/geänderte Posten haben einen * statt des - vorne weg)
- Ich habe noch RAMs für 768kB-Erweiterung!
- Bücher
 - Programmierung des Z80, Rodney Zaks, 606 Seiten: DM 45.-
 - CP/M-Handbuch, Rodney Zaks, 309 Seiten: DM 15.-
 - Mikroprozessor Interface-Techniken, A. Lesea/R. Zaks, 425 Seiten: DM 30.-
 - Operationsverstärker Anwendung, 164 Seiten, DM 10.-
 - ECA-Tabelle ttl-IC's , endet im Bereich der 74xx400-er: DM 20.-
 - ECA-Tabelle dat 1: Transistoren A..BUY: DM 5.-
 - ECA-Tabelle tht: Thyristoren, Triacs, ...: DM 5.-
- Einbau-Drehspulmeßgerät 0-50uA: DM 10,-
- Solange der Vorrat reicht:
 - MTX-Tasten je DM 1.-, Tastenkappen je DM -.50
 - EPROMS 2564 für je DM 15.-
 - Dynamische RAMs 4116 (VRAMs) 8 Stück: DM 25.-
 - Dynamische RAMs 3732 (32k x 1Bit) 8 Stück: DM 1.50
 - Statische RAM's 2k x 8 Bit (6116): je DM 2.-
 - Original-Memotech-Spielecassetten: Toado, Kilopede, Knuckles, Draughts, Reversi, Snappo, Blobbo, Utilities, Demo, StarCommand je DM 4.-;
 - 10 Disketten FUJI HD 5 1/4", Originalverpackt: DM 60.-
- Für den ECB-Bus:
 - ECB-Adapter, alte Bauform ohne Terminierung, zwei Steckplätze, komplett aufgebaut: DM 10.-
 - ECB-Adapter, neue Bauform komplett aufgebaut mit zwei Steckplätzen und Terminierung: DM 50.-
 - ECB-Adapter, neue Bauform, ein Steckplatz ohne Terminierung (d.h. die eine Hälfte der Platine), komplett aufgebaut: DM 30.-
 - c't-Festplatten-Controller (ECB-Karte) incl. der Unterlagen aus c't dazu, komplett aufgebaut: DM 200.-
 - SRAM/EPROM-Floppy von c't bestückt, aber ohne Speicher, auf Wunsch auch komplett ausgerüstet, Preis auf Anfrage.

K A U F

Hartmut Traber, Hohbeulstr. 8, 5270 Gummersbach, 02261 - 65399:

Suche gegen gutes Geld: Aachener Eprom-Floppy, Bestückung egal bzw. nur Preisfrage.

Moin, moin!

Leider scheinen die Aktivitäten bezüglich unserer schönen schwarzen Kiste immer weniger zu werden! Auch ist die Flut der Anregungen und Beiträge für's Info nicht besonders. Ich hatte schon überlegt, daß ich das Gesamtinhaltsverzeichnis der Infos 1 bis 46 nicht in diesem Info beilege, aber dann kamen in letzter Sekunde doch noch einige Artikel, so daß das Info dick genug ist, das GI zu verkraften.

Trotz meines neuen Rechners habe ich natürlich vor, dem MTX User-Club Deutschland weiter treu zu bleiben! Immerhin hänge auch ich am MTX und werde das gute Stück noch lange nicht einmotten.

RAM 6.2b ist verfügbar. Sie korrigiert ein paar kleine harmlose Fehler in RAM 6.2 und bietet schnellere Festplattenzugriffe. Solltest Du eine ältere Version von RAM 6.2 haben, kannst Du diese Version gegen eine Leerdiskette im Format 09 und Rückporto von mir erhalten.

Ich habe vor, eine Wanderdiskette ins Leben zu rufen, die die Runde machen soll, um so Informationen weiterzugeben. (Auf die Idee sind Holger Göbel, Olaf Krumnow, Herbert Oppmann, Claudio Romanazzi, Hartmut Traber und ich kürzlich gekommen.) Kommt die Scheibe bei Dir an, kannst Du Dir davon kopieren, was Du magst. Die Dinge, die Du das letzte Mal draufgepackt hast solltest Du löschen und möglichst durch neues ersetzen. Dazu wird für jeden ein User-Bereich vorgesehen. Ach ja, und dann muß die Scheibe an den nächsten weiter gehen. Kommt die Diskette bei mir vorbei, dann schaue ich nach, was davon ins Info und was auf Public-Domain kann. Einen Haken hat die Chose: Die Diskette wird eine HD-Scheibe (Format 1B oder 1C) sein, da die anderen Formate zu klein sind. Interessenten melden sich bitte bei mir mit u.g. Abschnitt bis zum 15. Juli 1992.

Eus

Herbert zur Nedden

----- bitte hier durchschneiden -----

Wanderdiskette

Ich möchte an der Wanderdiskette teilnehmen und verpflichte mich, zum Inhalt beizutragen sowie die Diskette jeweils drei Tage nach Erhalt weiterzusenden.

Name: _____ Vorname: _____

Straße: _____ Wohnort: _____

Diskette: 5 1/4", Format 1B _____ 3 1/2", Format 1C _____

Datum: _____ Unterschrift: _____

Einsenden an Herbert zur Nedden, Alte Landstr. 21, 2071 Siek

C l u b: Fragen/Antworten Korrektur/Nachtrag Komik

Fragen und Antworten

F: Andreas Fischer, CH-4303

Wer hat Erfahrungen mit dem Einsatz MTX <-> Palmtop von HP? Die serielle Schnittstelle mit dem CONTACT.COM funktioniert problemlos.

Falls jemand bereits Software für den direkten Aufruf des Laufwerks und des Druckers an der parallelen Schnittstelle hat, wäre ich daran interessiert.

Korrektur und Nachtrag

RSINIT.KLX

(Herbert Oppmann, 8520)

Dieses KLIX-Overlay birgt leider eine derbe Tücke: Die Angaben für gerade und ungerade Parität sind im Menü vertauscht! Willst Du gerade, so gib ungerade ein und umgekehrt.

BACKITUP

(Herbert Oppmann, 8520)

Diess Programm nicht mit RAM 6.2 verwenden! Es patcht im nicht vorhandenen P2DOS herum!

INST62

(Herbert Oppmann, 8520)

Bei der Wahl der Speicheraufteilung auf Bank 0 (Punkt A des Hauptmenüs) fehlt im Text "... Speichermodell brauchst Du ..." die "2".

W o r d s t a r: Rekursive Macros**Rekursive Macros unter Wordstar**

(Tim Slater, SCUG)

Anm.d.HzN: Folgenden Artikel habe ich der Clubzeitung der Schneider/Amstrad User Group entnommen. Er stammt von Tim Slater.

Im Handbuch für WordStar 4 (CP/M) wird im Abschnitt über die Belegung der Tasten mit Makrobefehlen (heißt bei WS "Kurzschrift") die Verschachtelung solcher Zeichen- und/oder Befehlsfolgen beschrieben. Das heißt, daß eine solche Kette eine andere aufruft. (Übrigens, eine Anwendung, die häufiger nützlich sein dürfte als das gegebene Beispiel einer mehrzeiligen Anschrift, ist das Zusammenbauen eines Macros, das zu lang für eine einzelne Definition ist.) Ganz am Ende, auf Seite 71, wird erwähnt, daß eine Aufgabe ständig wiederholt werden kann, indem die Befehlsfolge sich am Schluß selbst wieder aufruft.

Hier zwei Beispiele aus meiner Praxis, wozu das nützlich sein kann:

Wenn eine Textdatei für den Satz aufgearbeitet wird, müssen die (Schreibmaschinen-) Anführungszeichen des ASCII-Code ("...") durch richtige An- und Abführungszeichen ersetzt werden (>>...<<). Um im Rechnersatz (sowie manchen anderen Textprogrammen) gibt es auch, im Gegensatz zu WordStar, unterschiedliche Befehle zum Ein- und Ausschalten von Unterstreichen, Kursivschrift, Fettschrift usw. Leider gibt es im WordStar nicht die Möglichkeit, einen Globalaustausch automatisch bei jedem zweiten Vorkommen vorzunehmen [*weiß jemand von einem Textprogramm mit dieser Fähigkeit*]. Aber mit einem "rekursiven" (wiederholenden) Macro läßt sich das so machen: (in Klammern stehen Erläuterungen)

- ESC-x definieren als <CTRL-L>, <CTRL-L> (zweimal Suche wiederholen), <CTRL-G> (Zeichen am Cursor löschen), xyz (Zeichenkette für Abführungszeichen, Unterstreichen aus, u.a.), <ESC> <x> (Macro ruft sich selbst wieder auf).
- Am Anfang des Textes mit <CTRL-QF> die gewünschte Zeichenfolge ("", ^S u.a.) suchen, Cursor einige Buchstaben zurücksetzen und <ESC> <x> eingeben. (Oft wird am Ende des Textes ein "x" gesetzt, da WordStar, wenn es mit <CTRL-L> nichts mehr findet auf ein ESC wartet, was dann auch kommt - und zwar als erstes Zeichen von <ESC> <x>. Das ESC "schluckt" WordStar, das "x" landet im Text).
- Damit werden alle zweiten Vorkommen der gesuchten Zeichenfolge ersetzt. Es bleiben nur "Gänsefüßchen" als Anführungszeichen bzw. ^S als Unterstreichen ein übrig. Diese können mit <CTRL-QA> global ausgetauscht werden.

Wenn man länger mit einem BBS (Bulletin-Board System, d.h. Mailbox) herumschaut, und as DFÜ-Programm alles speichert, enthält die rotokoll-Datei oft viele unterschiedliche Meldungen des Systems von der art wie "Press RET or ..." mit einer Leerzeile davor und dahinter, die man gerne automatisch löschen würde. Das geht so:

- ESC-z definieren als <CTRL-L> (Suche wiederholen), <CTRL-Y>, <CTRL-Y>, <CTRL-Y> (drei Zeilen löschen), <ESC> <z> (Macro ruft sich selbst wieder auf).
- Am Anfang des Textes mit <CTRL-QF> nach "^NPress RET" suchen (Cursor steht dann in der vorhergehenden Leerzeile). Einige Buchstaben zurücksetzeun und <ESC> <z> eingeben.

Anm.d.HzN: Vermutlich muß Du statt ^N ein ^J in o.g. Suchstring eingeben.

Hardware: 8 MHz

Meine Geschichte über die 8MHz-Aufrüstung

(Hartmut Traber, 5270)

Nachdem mein Rechner peu à peu über Jahre hinweg jetzt die von mir gewünschte Konstellation erreicht hatte (zähl ich gleich auf), blieb als letzter Schritt noch die Aufrüstung auf 8 oder mehr MHz.

Im MTX 500 ist die Haupt-, Speichererweiterungs- und die RS 232-Karte, in der FDX ein tolles Netzteil für alles, die Hard-Disk, zwei 80-Spur 1,76 MB-Foppies, und im Platinenkäfig von unten nach oben die Interfaceplatine, der Floppy-Controller, 80-Zeichen-Karte, eine Memotech-1 M-Sidisc, 2 c't-SRAM-Discs 512 K und ganz oben der HD-Controller.

Die c't-Karten und der HD-Controller stecken natürlich in Gerhard's Adapterplatinen, die zusätzlich bisher nur die Uhr enthalten, mehr ist ja noch möglich!

Links neben der FDX steht mein DMX 80, links auf der FDX der Monitor, daneben im Extra-Gehäuse noch zwei Floppies. Rechts neben der FDX hat mein Modem Platz gefunden.

Die Ausgangssituation war folgende:

Vor Jahren hatte ich die einfache Änderung auf 6 MHz durchgeführt, die auf dem Mist der Aachener gewachsen war, d. h. 6MHz-Quarz, 8 MHz Z80-CPU, 6 MHz CTC und DART, bzw. SIO. Die vorgeschlagene Änderung auf der 80-Z-Karte hatte ich auch durchgeführt. Heilfroh war ich bis dahin, daß die Investitionen für alles, was über die Grundausrüstung hinausging, nicht verloren waren, sondern alles einwandfrei lief.

Als Hilfsmittel für die Aufrüstung sollte mir das "Kochbuch" von Holger reichen (dachte ich!), und im Hinterkopf hatte ich die spleenige Idee, ohne Holgers Platine auskommen zu können?

Damit es kurz wird, aber dennoch informativ bleibt, beziehe ich mich jetzt immer auf Holger Göbels Kochbuch und dessen Punkte.

A konnte ich übergehen.

B Das Schlimmste war ja Punkt *5. Das hat Herbert auf dem Clubtreffen letztes Jahr Samstag/Sonntag 24.00/00.00 erschlagen, indem er mir Sockel einlötete. Den Einsatz eines Sockels für *6 habe ich ihm gnädig erlassen, da bei mir ein HC 04 drin war und ich..., s.o.! Noch am Sonntagnachmittag des Clubtreffens nach der eiligen Heimfahrt (Herbert, jetzt weißt Du, warum ich weg wollte, nein, nicht wegen Steuererhöhung auf Sprit oder meiner besseren 60%), habe ich die schnellen 4164-er (=Speicherbausteine) eingesetzt.

*7 habe ich nicht gemacht, da softwaremäßig in RAMxx installierbar.

*8 mußte ich wieder einlöten.

Dann habe ich *1 erledigt (über Schalter, auf der anderen Seite des Schalters war ein 4 MHz-Quarz) und hoffnungsvoll eingeschaltet.

Mit 4 MHz ging, mit 8 MHz nicht. Eigentlich hätte es tun sollen, denke ich, zumindest im MTX-Modus. Nein, auch MTX lief nur mit 4 MHz.

Dann forderten mich die tollen 60 %, außerdem war schönes Wetter geworden, und ich vertagte die Sache, nahm mir aber schon vor, am nächsten Tag einen Frequenzzähler mit nach Hause zu nehmen.

Den ich dann auch einschaltete:

Da habe ich die dSH (dunkle Stunde Hartmut's) stark empfunden:

Mein Rechner, das gute Stück, lief mit 4 MHz-Quarz mit 3,7 MHz, auch mit 6 MHz-Quarz. Ich habe jahrelang nach Einbau der Aachener Lösung ca. 0,3 Mhz an Geschwindigkeit verschenkt (gegenüber 4 MHz-Quarz, da ich aber dachte, er läuft mit 6 MHz: ich lag am Boden!).

Mit 8 MHz lief er aber auch nicht, mein Zähler zeigte 6 bis fast 10 MHz an, daher habe ich erstmal

*2 geprüft und mußte es ändern. ---> das hat es gebracht!

Zum diesem Zeitpunkt lief mein Rechner im Grundgeräte-Modus mit 8 MHz.

Hardware: 8 MHz

Dies habe ich mit dem einfachen Basic-Programm

```
10 For x=1 to 100
20 print x, sqr(x)
30 next
```

gestoppt, bei 8 Mhz ca. 7 Sekunden.

Soweit, sogut (nichtgut, denn RAM lief nicht, dachte ich).

Aber immerhin, 32 KB sind in der Speichererweiterung beim MTX 500, d. h. die müßte laufen (?).

Ja, die lief, im Nachhinein betrachtet, auch ohne C2 (Kochbuch).

Aber zu diesem Zeitpunkt war ich schon von meinen Ideen abgerückt, es ohne Holgers 8 Mhz-Platine versuchen zu wollen.

Bestellt, erhalten, HC 04 auf der Grund-Platine ausgelötet (*6), verdrahtet, alles läuft mit 4Mhz.

(Für das Auslöten HC 04 brauchte ich 'ne neue Zange...., wer kennt das nicht).

Ich danke Holger und Gerhard für die schnelle Zusendung der Platine, die im übrigen paßt, und paßt und paßt, auch nach mehrmaligem Stecken!).

Die ausführlichen Anweisungen zur 8 MHz-Platine (preiswert, um nicht zu sagen billig, das wäre für diese spitzenmäßige Ausführung eine Abwertung), die Ergänzungen und Wiederholungen aus dem Info sind, möchte ich jetzt nicht ansprechen, die Platine ist Spitze, die Anleitung auch, wie man noch sehen wird...

Mit dieser Platine lief mein Rechner immer noch (man beachte dies) mit 8 MHz, im Grundgeräte-Modus, aber nicht unter RAM, vermeintlich.

Zeitaufwendige Tests folgten dann mit dem Timing der Speichererweiterungskarte, völlig überflüssig, wie sich zeigte!

Mein Anzeigeeinstrument war ja der Bildschirm, der wird von der 80-Zeichen-Karte gespeist, und da lag dann auch das Problem.

Wie kam ich darauf?

Ich habe mit 8 MHz gebootet, alle notwendigen Befehle eingegeben, unabhängig von der wilden Bildschirmdarstellung, ein Directory aufgerufen, umgeschaltet auf andere Laufwerke, alles gemacht wie gewohnt, dann auf 4MHz umgeschaltet:

---> NICHTS!

Dasselbe wieder versucht, als letzten Schritt Shift-Esc..., umgeschaltet:

---> der Klick-Bildschirm war da!!!, und ich konnte weiterarbeiten!!!

Was lehrte mich das?

Die 8 MHz liefen! Aber...

Die 80-Zeichen-Karte rettete keinen Bildschirm, genausowenig wie sie einen aufbaute. Im Info nachgeschaut, um keinen Fehler zu machen, 100 nF-C's nachgelötet, kein Erfolg, bis jetzt.

Auch eine andere 80-Zeichen-Karte von HzN schaffte keine Abhilfe - die Karte lief zwar etwas besser aber der Bildschirm-Müll blieb.

Als ich eines Tages auf die glorreiche Idee kam, den M6845P der Karte zu kühlen, aber kein Kältespray hatte, nahm ich einfach Feuerzeug-Gas (Vorsicht!), das kühlt auch, und siehe da, alle Zeichen standen eine Weile an der richtigen Stelle. Holger sah das Problem voraus und schickte mir sogar einen Kühlkörper für dieses Teil mit. Das brachte es aber nicht, auch nicht die empfohlene Änderung der Luftführung in der FDX.

Nach langem Hin und Her bestellte ich mir einen neuen Controller: M6845B statt des originalen MC6845P. In der Rechnung stand: CRT-Controller 2 MHz. Mit diesem Teil lief meine 80-Z-Karte auf Anhieb mit 8 MHz, und das bei offenem Gehäuse und ohne zusätzliche Kühlung, nunmehr über Stunden und bei voller Bestückung des Bus (Busses?).

Leserbrief: Hartmut Traber, 5270

Mein Hauptspeicher auf der MTX-Hauptplatine enthält z. Zt. 150-ns-RAM's. Einen weiteren Satz 100-ns-RAM's habe ich noch und habe die eben auch probiert, mit keinem anderen Ergebnis, d. h. alles paletti. Das Booten vom c't-Laufwerk I: (512K) geht jetzt so schnell, daß meine Harddisc noch nicht bereit ist, wenn ich im Booter flott auf sie zugreife, sie braucht einige Zeit, um auf Drehzahl zu kommen. Das wird kein Problem darstellen, sobald ich I: erstmal wieder mit den richtigen Klickern geladen habe. Allerdings kommt keine Fehlermeldung, wenn die Harddisc noch nicht bereit ist! Dann steht der Rechner.

Jetzt bleibt mir noch übrig, RAM62 ans Laufen zu bringen. Im Gegensatz zu RAM61 läuft dies noch nicht, weder unter 4 noch unter 8 MHz. Es ist mir richtig peinlich, daß ich noch keinen Kommentar zu dieser erneuten Spitzenleistung von Dir und Olaf zu Papier oder Diskette bringen konnte, aber die anderen Dinge gingen vor.

Es kommt natürlich der Wunsch auf, auch von der Harddisc zu booten. Im Boot-Eprom ist sicher Platz auch für ausgefallendste Boot-Laufwerke und Kombinationen? Mein Wunsch wäre: Boot I:51, J:30, B:1C, B:03, B:07, B:09

Anm.d.HzN: RAM 6.2 läuft mittlerweile auch bei Hartmut - es hatte sich noch ein kleiner Fehler in seiner Hardware eingeschlichen: ein Wackelkontakt.

Die Kiste läuft ...

(Hartmut Traber, 5270)

Moin,Moin, liebe Leute,

mein Rechner stand nach der doch mühevollen Umrüstung auf 8 MHz einige Monate ziemlich still, der Ersatzrechner, nicht aufgerüstet, konnte nur RAM 45. Damit bin ich der Zeit hinterher, bemühe mich ums Aufholen in Sachen RAM 62b. Nicht daß ich die zwischenzeitlich eingetrudelten Disketten nicht hätte lesen können: ah-bah, das war nicht das Problem. Aber die neuen Spitzenprodukte, als da sind: LOADDIR.COM, MOUNTLIB.KLX mit der entsprechenden Umgebung und der hervorragenden Doku, überwiegend oder ausschließlich in den *.INS-Dateien untergebracht, konnte ich erst jetzt sichten und laufen lassen. **HERVORRAGEND!**

Auch MMENU6 hat mich beeindruckt:

Ich hatte Gelegenheit, die Maus in Aktion zu sehen: --> meine nächste Erwerbung! Damit sind nicht nur die obigen Errzeugnisse von Claudio, sondern auch die (?) Abfallprodukte von HzN in Sachen Maus von mir gewürdigt. Sie werden künftig genutzt! Und das soll Anerkennung und Dank ausdrücken!

Michael Keßler schrieb mir mal:

"Dein Einzeiler im Info....." (betraf mal ein Programm von ihm), "hat mich veranlaßt...." usw.

Es wäre nett von Euch allen, doch auch mal einen "Einzeiler" loszulassen. So wie ich Herbert zur Nedden kenne, ist es ihm wurschtegal, in welcher Form sowas bei ihm eintrudelt.

Es (der "Einzeiler" nämlich) würde aber für uns alle Dein Interesse am Club und am Meinungsaustausch ausdrücken. Und wie schon mehrfach gesagt: Kein Thema ist zu blöd für unser Info! (Ich möchte hier in Bezug auf "blöd" nicht mißverstanden werden! So "blöd" ich mich manchmal in den letzten 7-8 Jahren angestellt habe, das sage ich Dir jetzt nicht (denk Dir was)!

Hardware: MTX-Tasten selbst reparieren**MTX-Tasten selbst reparieren**

(Jan Bredereke, 2000)

Mit der Zeit läßt selbst die beste Tastatur nach: In den letzten zwei Jahren habe ich auf dem Memotech sowohl meine Studienarbeit als auch meine Diplomarbeit geschrieben und dabei etliche Millionen Anschläge produziert. So wurde schließlich bei den häufig gebrauchten Tasten (zuerst SHIFT & Co.) der Kontakt schlecht, sie prellten und gaben nur noch unsicheren Kontakt. Besonders bei SHIFT + Funktionstaste kann das verheerend werden.

So habe ich zuerst bei Herbert Ersatztasten bestellt und eingebaut. Als ich aber schließlich mehr neue Tasten brauchte, als ich gerade vor mir liegen hatte, sah ich sie mir genauer an und stellte fest, daß sie ganz leicht selbst zu reparieren sind. Nur Aus- und Einbau machen noch etwas Mühe, aber die bleibt auch beim schlichten Auswechseln.

Zuerst zum Aus- und Einbau:

Nötig sind ein Elektronik-LötKolben, Lötzinn, ein kleiner Schraubenzieher, Löt-sauglitze und/oder eine Löt-saugpumpe (ein Teil für DM 5,-- reicht völlig).

Rechner aufschrauben, Oberteil abnehmen (dabei das Tastaturkabel zur Hauptplatine nicht abreißen!), mit Durchgangspieper die wackeligen Tasten herausfinden (das Wackeln ist sehr gut zu hören!) und markieren. Betreffende Tastenkappen mit den Fingern abziehen (evtl. vom Rand her herarbeiten, das geht leichter). Tasten auslöten, Lötzinn ganz entfernen, so daß die Taste nur noch von ihren Rastzungen auf der Außenseite (ehemals unter der Tastenkappe) gehalten wird. Mit dem Schraubenzieher beide Zungen leicht eindrücken, Taste herausnehmen.

Einbau: Entsprechend umgekehrt.

Wie funktioniert nun die Taste?

Die beiden Metallkontaktstifte, an denen sie angelötet wird, reichen fast durch die gesamte Taste hindurch. Hier war der Hersteller sehr trickreich: Sie sind gleichzeitig die Schrauben, die die Taste zusammenhalten! Mit einer kleinen Zange kann man beide linksherum leicht herausdrehen und sich das Innere der Taste ansehen. Wird die Taste gedrückt, dann bewegt sich ein Plastikstab nach unten und drückt ein kleines Metallfederblech zur Seite gegen einen der Kontaktstifte. Der andere Stift ist ohnehin dauerhaft mit dem Blech verbunden, so daß der Kontakt dann geschlossen ist. Bei jedem Schalten entsteht aber ein winziger Funken, der auf die Dauer den Stift (aber nicht das Blech) zum Oxidieren bringt, er wird schwarz.

Meine Lösung daher: Ich drehe nur diesen einen Stift heraus (die Taste bleibt dabei beisammen), reinige ihn mit feinem Schmirgelpapier und drehe ihn wieder hinein. Um den richtigen Stift zu erwischen, habe ich mir folgende Eselsbrücke gebaut: Blickt man auf die Unterseite der Taste mit den beiden Kontakten, so sind diese etwas zu einem Rand hin versetzt. Ich stelle mir vor, daß es die "Augen" eines runden Gesichtes seien, wobei die "Augen" oberhalb der Mitte angeordnet sind. Der betreffende Stift ist das "linke Auge" (also auf der rechten Seite, wenn die Stifte "oben" sind).

So, das war bereits alles. Ich hoffe, Herbert ist mir nicht allzu böse, daß ich ihm sein Geschäft mit gebrauchten Tasten kaputt mache, aber Ihr haltet es sicherlich genauso für sehr beruhigend wie ich, wenn man jederzeit, unabhängig und vor allem schnell müde Tasten wieder auf Vordermann bringen kann.

Anm.d.HzN: Ich bin Jan natürlich nicht böse!

Hardware: Edicta-Grafikkarte**RAMs für die Edicta-Karte**

(Herbert zur Nedden, 2071)

Die Edicta-Karte benötigt pro Farb-Ebene zwei RAMs vom Typ 85103. Da diese Bauteile schwer zu bekommen sind, habe ich für je ein Paar dieser Teile einen Ersatz entwickelt, der auch unter dem Namen "Turm" bekannt ist, da er aus vier übereinandergestapelten RAMs vom Typ 41464 besteht. Daß das die Bauhöhe der Platine nicht unwesentlich vergrößert und nicht besonders angenehm zu löten ist, läßt sich sicherlich denken.

Gerhard Witzel hatte daher vor einiger Zeit mal ein Platinchen (die Antiturm-Platine) entwickelt, welches meinen Turm durch eine flache Konstruktion ablösen konnte - dafür aber verhinderte, daß mehr als eine Farb-Ebene auf die Karte kam, was eh der Normalfall ist.

Nun hat sich Gerhard erneut dieser Platine angenommen und ein kleines Ersatzplatinchen für einen 85103 auf der Basis von zwei 41646 gebastelt. Diese Platine ist so gestaltet, daß sie die 85103 ersetzen kann und insbesondere eine Aufrüstung aller vier Farbebenen der Grafikkarte ermöglicht. Dabei setzt er modernste Technik ein: ein NAND-Gatter in SMD-Ausführung.

Hier die Preise (stehen etwas kompakter auch in der Angebotsliste):

Je Platine:

	Platine blank	DM 3.-
	Platine mit Brücken und CMOS SMD NAND-Gatter	DM 4.50
	Dto. mit 22-poliger Steckerleiste	DM 9.-
	Dto. mit zwei RAMs 41646 (bei moment. IC-Preis)	DM 25.-

P&V: DM 2.-. Ab 4 Platinen je DM 1.50, ab 6 je DM 1.-

Auf Wunsch verwendet Gerhard auch Teile aus einer vorhandenen Edicta-RAM-Konstruktion - ermöglicht durch seinen aufladungsfreien Lötplatz. Preis dann je nach Materialbedarf. Bei seiner Antiturm-Platine sind z.B. RAMs und die Steckerleiste i.a. noch zu retten: DM 8.- pro Platine fertig. Tausch gegen ein CAD-Programm für die Editca ist auch denkbar. Auf Wunsch liefert Gerhard auch komplett-Bausätze.

Turbo-Pascal: Bug und Patches**Kleiner Bug im Turbo-Pascal-Compiler**

(Jan Bredereke, 2000)

Neulich bin ich auf einen kleinen Fehler im Turbo-Pascal-Compiler gestoßen. Für korrekte Programme hat er keine Bedeutung, aber er verhindert die Erkennung von bestimmten Programmierfehlern. Beispiel: Ich schreibe folgende Testroutine:

```
Function test_a(ch : Char) : Boolean;
Begin
  test_a := ch = 'a';
End;
```

Da ich auch eine entsprechende Testroutine "test_z" benötige, erhalte ich durch Kopieren und Ändern:

```
Function test_z(ch : Char) : Boolean;
Begin
  test_a := ch = 'z';
End;
```

Der Compiler übersetzt dies anstandslos, aber das Ergebnis tut nicht das, was es soll!

Natürlich hast Du in diesem einfachen Beispiel sicherlich längst gesehen, daß ich vergessen habe, die Zuweisung für das Funktionsergebnis von "test_a" in "test_z" zu ändern. Aber der Compiler hat es nicht gemerkt, obwohl es gar keine Variable "test_a" gibt.

Der Grund für dieses Verhalten erklärt sich so: Turbo-Pascal ordnet jeder Funktion eine Variable mit dem gleichen Namen zu, in der das Funktionsergebnis bis zur Rückkehr zum Aufrufer zwischengespeichert wird. So ist die Zuweisung für das Funktionsergebnis als ganz normale Variablenzuweisung implementiert. Weiterhin ist der Gültigkeitsbereich für den Funktionsnamen der umgebende Block ab dem Definitionspunkt der Funktion. So ist bei der zweiten Zuweisung auf "test_a" dieser Bezeichner definiert, allerdings als Funktion. Aber bei der Zuweisung kann er keine Funktion bezeichnen, und so wird er fehlerhafterweise als Variable verwendet, und der zugewiesene Wert wird in dem Zwischenspeicher abgelegt. Dort ist er dann allerdings nicht mehr zugreifbar, sofern die Funktion "test_a" vor ihrer eigenen Rückkehr immer selbst einen neuen Rückkehrwert dort ablegt.

Vom Ausnutzen dieser neuen Variablen ist dringend abzuraten, man sollte im Gegenteil bei solchen Kopieraktionen wie der oben beschriebenen immer verflüchtigt gut aufpassen, denn bei komplizierten Funktionen vergißt man sehr leicht einmal eine Änderung.

Anm.d.HzN: Da lobe ich mir Programmiersprachen, die die Rückgabe von Funktionswerten über Befehle wie RETURN(...) o.ä. lösen!

Patches

(Herbert zur Nedden, 2071)

Herbert Oppmann hat eine ganze Reihe von Patches für die Laufzeitbibliothek von Turbo-Pascal erstellt, die Du auf KLICK.021 finden kannst. Dazu gehören u.a., daß Turbo-Programme damit echte ZCPR 3.3-Programme werden, User-Angaben in Dateinamen abkönnen uvm.

Mehr zu den Patches wird es sicherlich im nächsten Info zu lesen geben.

Hardware: Die FestplatteTiming ist alles

(Herbert Oppmann, 8520)

Einige Tests mit der Harddisk (Timing, falls nicht extra vermerkt, bei 8 MHz)

Bei 3600 U/min = 60 U/sec dauert eine Umdrehung 16,7 msec.

Bei 17 Sektoren a 512 Byte dauert ein Sektor 0,98 msec.

Bei Interleave 1:1 könnte mit 510 KByte/sec gelesen werden.

Ergebnisse Hardware:

- Zeit zwischen Select und Aufforderung zum Senden des ersten Befehlsbytes beim ersten Mal ca. 10 usec, alle weiteren Male ca. 100 usec
- die Aufforderungen zum Senden weiterer Befehlsbytes erfolgen so schnell, daß kein Maschinenbefehl dazwischenpaßt! --> Ausgabe durch OTIR problemlos
- Zeit zwischen Befehlsausgabe und erstem Datenbyte (Sektor auf gleichem Cylinder wie der vorherige) ist ca. 17 msec, also etwa eine Umdrehung.
- die Aufforderungen zum Lesen weiterer Datenbytes erfolgen so schnell, daß kein Maschinenbefehl dazwischenpaßt! --> Lesen mittels INIR problemlos
- Zeit zwischen letztem Datenbyte und Status ist ca. 170 usec
- beim Lesen mehrerer Sektoren in einem Rutsch beträgt die Zeit zwischen den Sektoren (auf gleichem Cylinder) zwischen ca. 2 und 3,6 msec bei Interleave 4.
- minimale Einlesezeit (INIR, 512 x 21 Takte) liegt bei 1,344 msec.
- Das Blocking/Deblocking erfordert einen Kopiervorgang, der meistens ein Interbanktransfer ist. Die in RAM 6.1 hierfür zuständige Routine benötigt $80 + n \times 42$ Takte, für einen Sektor also 5456 Takte oder 0,682 msec. Eine von mir entwickelte Routine schafft hier $330 + n/16 \times 329$ Takte, für einen Sektor also 2962 Takte oder 0,37 msec. Ab ca. 750 Byte arbeitet diese Routine sogar schneller als LDIR!
- Interleave 1:1 kann ich nicht durchhalten. Interleave 1:2 geht mit 8 MHz, Interleave 1:3 mit 4 MHz am besten (nacktes Lesen). Diese Werte sind interessant, wenn mit einem Befehl gleich mehr als ein Sektor eingelesen wird. In der jetzigen Patch-Version müssen die im Voraus eingelesenen Sektoren bei Bedarf in den Sektorpuffer kopiert werden. Das frißt einen Großteil des Zeitgewinns wieder auf, so daß es sich im Endeffekt nicht lohnt. Bei einer Verankerung im RAM würde es sich aber vermutlich lohnen. Beim Lesen von einzelnen Sektoren spielt der Interleave kaum eine Rolle.

Ergebnisse BIOS-Schnittstelle:

- Lesen Interleave 4, Cache an (Lw J)
aus Sektorpuffer: 1 msec, mit meinem schnellerem Interbanktransfer: 0,64 msec
erster Zugriff: 50-110 msec, danach 11-15 msec
Problem: diese Verbesserung wirkt sich auf der Festplatte kaum aus, da die Sektoren in festen Zeitabständen auftauchen und ich daher das, was ich beim Lesen aus dem Sektorpuffer spare, beim Lesen des nächsten phys. Sektors wieder durch längeres Warten verliere. Erst wenn ich so viel einspare, daß ich einen kleineren Interleave-Faktor nehmen kann, wirkt sich die Einsparung aus.
- Lesen, Interleave 2, kein Cache (Lw M)
Die Lösung mit einem Sektor im Voraus als Puffer sieht gut aus. Viel besser als das schnellere Kopieren. Wenn nun noch das doppelte Kopieren wegfallen würde, ...
Aus ReadAhead-Buffer: ca. 4,6 msec

Ergebnisse high level: (Angaben in %)

Kopierprogramm (acopy m0:*. * a0: /v)		Assembler (m90 fast/af)
User:	3,1	User: 25,9
BDOS:	32,8 1)	BDOS: 11,1
ConOut:	2,7	ConOut: 1,3
ReadSec:	55,8 2)	ReadSec: 42,0
WriteSec:	5,5 3)	WriteSec: 19,7

1) incl. andere BIOS-Aufrufe (z.B. SetTrack), 2) incl. Dir-Lesen A:, 3) nur A:

Software: RAM 6.2 (b)

RAM 6.2 (b)

(Herbert zur Nedden, 2071)

Die schnelle(re) Platte

Dank einiger Hinweise von Herbert Oppmann (s.o.) konnte ich die Festplattenzugriffe in RAM 6.2 beschleunigen. Er hat nämlich u.a. folgendes gestellt:

Das einlesen eines Sektors lief in der folgenden Form ab:

```

DI
Nxt: CALL Rdy      ----> Rdy: INC  C
INI                Wt:  IN   A,(C)
DEC  DE           AND  01
LD   A,D          JR   Z,Wt
OR   E            DEC  C
JR   NZ,Nxt       RET
EI

```

Die Wt-Schleife wird jedoch bei ihm definitiv IMMER NUR GENAU 1 MAL durchlaufen! Nach einigen Tests kam heraus: die ganze Abfragerei, ob der Controller fertig ist, bremst nur gewaltig runter! Denn der ist definitiv schneller fertig, als die Z80 mit 8 MHz abholen kann.

Nur beim ersten Byte muß ich warten (logisch):

```

DI
CALL Rdy
INIR      2 x 256 = 512 Byte Sektorgroesse
INIR
EI

```

Ab Version 6.2b von RAM ist es installierbar, wie Du Deine Platte bedient wissen möchtest: mit INIR, OTIR oder dem alten Algorithmus.

Weiterhin hat mir Herbert eine etwas schneller Routine für den Transport eines Sektors aus dem Puffer an die DMA-Adresse und zurück geschickt. Wenn ich Platz im Common finde, baue ich die vielleicht auch noch ein --> RAM 6.2c oder 6.2d.

Booten von der Festplatte

RAM 6.2 kannst Du ja im Prinzip ohne Betriebssystem booten. Dazu war bei RAM 6.2 ja das RAM62LDR mit dabei. Hans Gras hat ein Boot-EPROM kreiert, welches ihm genau das ermöglicht: es bootet RAM 6.2 direkt von der Festplatte, indem es RAM 6.2 samt RAM62LDR von den Systemspuren in den Speicher lädt und aufruft.

Alternativ kann man natürlich den Inhalt des Gras-Boot-EPROMs auch als kleines Programm umschreiben, welches beim Booten von Diskette oder SRAM-Floppy geladen wird. Das hat den Vorteil, daß Du das Boot-EPROM nicht auszutauschen brauchst.

AT-Bus-Platte

Ich habe vor, demnächst (wann immer das auch sein mag), eine AT-Bus-Platte (auch IDE-Platte genannt) an den Memotech anzuschließen. Wenn's funktioniert, baue ich den Treiber in RAM 6.2c oder 6.2d oder ... ein oder gebe ihn als Zusatz-KLIX-Overlay heraus.

d B A S E: Ein paar Tips**dBASE - ein paar Tips**

(Herbert zur Nedden, 2071)

Die hier stehenden Tips sind nicht alle auf meinem Mist gewachsen - eigentlich kommt von mir selbst hier kaum etwas. Ich habe sie hie und da erhalten und mich drangesetzt, sie mal zusammenzupacken. Alle Tips hier sind Ideen/Ansätze, natürlich ohne Pistole - ääh Gewähr!

Beschleunigen von dBASE-Programmen

dBASE ist ein Interpreter, d.h. jedesmal, wenn es eine Zeile ausführt, muß es diese erst analysieren. Damit ist eigentlich schon klar, was Ihr tun könnt, damit die dBASE-Programme (d.h. die *.CMD-Dinger) schneller werden:

1. Jedes dBASE-Schlüsselwort, d.h. die Befehle wie auch die Optionen, können auf die ersten vier Buchstaben abgekürzt werden. Statt STORE kannst Du auch STOR schreiben, statt BOTTOM tut's auch BOTT.
2. Variablennamen, die Du ja selbst vergibst, solltest Du zwar sprechend aber auch möglichst kurz gestalten. Warum muß das Feld der Datenbank Nachname heißen? Reicht NName oder NN nicht aus?
3. Kommentare kosten nur Zeit! Äh nein, das ist Quatsch - sie sind für Deine Arbeit natürlich sehr sinnvoll. Aber ggf. kannst Du Dir ja ein kleines Turbo-Pascal-Programmchen schreiben, welches im Kern folgendes tut:

```
WHILE not eof(Eingabe) do
  begin
    ReadLn(Eingabe,Zeile);
    if (copy(Zeile,1,4) <> 'NOTE' ) and (copy(Zeile,1,1) <> '*') then
      Writeln(Ausgabe,Zeile);
  end;
```

Und schon sind die Kommentare raus. Die Ausgabe-Datei nutzt Du für dBASE, die Eingabe hebst Du Dir als Source auf.

Mit diesen drei Ansätzen kannst Du dBASE-Programme sicherlich beschleunigen, wenn Du sie nicht eh schon beachtet hast.

Unnötige Eingaben vermeiden

Nehmen wir mal an, wir wollen eine Adreßdatenbank aufbauen, in der vorwiegend deutsche Adressen stehen. Das Wörtchen 'vorwiegend' sagt schon fast alles: Wir brauchen eine Länderkennung, die bei den meisten auf Deutschland verweist. Wie sparen wir uns unnötige Arbeit?

So wie in der Mitglieder-Datenbank des MTX User-Club Deutschlands: Bei Deutschen Adressen der alten Bundesländer beginnt die Ortsangabe direkt mit der Postleitzahl; bei allen anderen steht der Ländercode davor. Sollte für die Länderangabe ein eigenes Feld in der Datenbank vorgesehen werden, kann man hier die selbe Logik anwenden: Leer = 'Deutschland, alte Bundesländer'.

Einen anderen Weg der Arbeitersparnis geht das Info-Inhaltsverzeichnis des MTX User-Club Deutschlands: Die Eingabe z.B. des Autors kann natürlich durch Eingabe eben dieses Namens erfolgen. Einige Namen tauchen jedoch recht häufig auf; bei diesen wird einfach eine Abkürzung wie z.B. OK, CR oder HzN eingegeben; bevor der Satz gespeichert wird, werden diese Abkürzungen in die entsprechenden Namen automatisch umgesetzt.

d B A S E: Ein paar TipsEine oder mehrere Datenbanken:

Habe ich je Datensatz viele Informationen, benötige aber bei den meisten Operationen nur wenige Felder daraus, dann sollte ich die Daten in mehrere Datenbanken aufteilen, damit ich meist nur mit einer kleinen Datenbank arbeiten muß.

Bleibt die Frage, wie ich die Zusammengehörigkeit unterschiedlicher Sätze der verschiedenen Datenbanken hinbekomme. Hier drei Möglichkeiten:

1. Meine kleine Arbeits-Datenbank ist ein Extrakt der Großen.
2. In einem Feld der Arbeits-Datenbank steht ein Zeiger auf ein Index-Feld der Großen
3. Ich habe in beiden das selbe eindeutige Index-Feld.

Abfragen

Nehmen wir wieder mal eine Adreßdatenbank mit einem zweistelligen Feld für den Ländercode. Wir wollen wissen, ob das Land deutschsprachig ist, d.h. ob es sich um Deutschland (D), Österreich (A) oder die Schweiz (CH) handelt. Da natürlich in den FNL (Fünf neue Länder) auch deutsch gesprochen, diese aber mit Ländercode vor der Postleitzahl versehen werden, wird auch das 0 abgefragt.

Es geht ganz einfach:

```
IF lnd = ' ' .or. lnd = 'D' .or. lnd = 'O' .or. lnd = 'A' .or. lnd = 'CH'
```

oder lesbarer und kompakter:

```
IF lnd$' D O A CH'
```

oder auch:

```
IF $(lnd,' D O A CH') > 0
```

Schleifen

Da dBASE keinen allzu großen Speicherplatz für Programme hat, sollten Schleifen kurz gehalten werden, damit dBASE nicht während einer Schleife die CMD-Datei laufend nachladen muß.

Werte, die in einer Schleife gebraucht, aber nie innerhalb verändert werden, sollten natürlich außerhalb einmal errechnet werden!

Index-Tips

Indiziere nie auf ein numerisches Feld, sondern ggf. auf den STR(...) davon.

Index-Felder kurz halten! Ist die Datenbank nach dem Index sortiert, geht's mit dem Indizieren und dem Suchen am schnellsten.

d B A S E: Ein paar TipsSonderfälle

Viele Programme wären schnell und kurz, wären da nicht die paar Sonderfälle, die es auch zu behandeln gilt. Hier bietet es sich an, die Behandlung der Sonderfälle in getrennte CMD-Dateien auszulagern und im Haupt-Programm nur die Sonderfall-Abfrage und den entsprechenden DO-Befehl zu haben, wenn die Sonderfälle selten sind. Dadurch ist das Haupt-Programm kompakter und da dBASE ein Interpreter ist das Programm folglich bei den nicht-Sonderfällen schneller.

Statt:

```
IF Sonderfall
  STOR ....
  STOR ....
  STOR ....
  STOR ....
ENDIF
```

lieber:

```
IF Sonderfall
  DO ....
ENDIF
```

Ein paar kleine Beschleuniger:

Im folgenden werde ich für div. Teilaufgaben in dBASE eine langsamere und eine schnellere Lösung für die gleiche Aufgabe aufzeigen.

Suchen von Datensätzen:

```
GO TOP
DO WHILE .NOT. EOF
  SKIP
  IF ein gewünschtes
    DO was-auch-immer
  ENDIF
ENDDO
```

get schneller so:

```
GO TOP
LOCATE FOR gewünschte
DO WHILE .NOT. EOF
  DO was-auch-immer
  CONTINUE
ENDDO
```

Schneller als:

```
STOR 0 TO cnt
DO WHILE cnt < 12
  DISP
  STOR cnt+1 to cnt
  SKIP
ENDDO
```

ist:

```
LIST NEXT 12
```

GET mir paar Daten:

```
STOR ' ' TO mfeld
§ 1,2 SAY 'neue daten ' GET mfeld
READ
REPL feld WITH mfeld
```

Deutlich schneller:

```
§ 1,2 SAY 'neue daten ' GET feld
READ
```

Komprimieren:

```
PACK
```

Zügiger geht es so:

```
COPY TO tempfile
USE
DELE database
RENA tempfile TO database
```

d B A S E: Ein paar Tips

WAIT in einer Schleife läßt dBASE
laufend in seinem *.OVR lesen:

```
DO WHIL t
  WAIT
ENDD
```

Disketten-schonender und schneller ist:

```
DO WHIL t
  STOR ' ' TO mx
  SET colo OFF
  § 1,2 SAY 'Waiting ' GET mx
  READ
  SET colo ON
ENDD
```

Update einer Datenbank aus einer anderen

Üblicherweise sieht das Spiel hierfür so aus, daß die eigentliche Datenbank als Primär- und die Update-Info als Sekundär-DB betrieben werden:

```
SELE PRIM
USE Bestand
SELE SECO
USE Update
INDE ON Suchmerkmal TO Temp
SELE PRIM
DO WHIL .not. EOF
  STOR Suchmerkmal TO MSuch
  SELE SECO
  FIND &MSuch
  IF ##0
    STOR Feld TO MFeld
    SELE PRIM
    REPL Feld WITH MFeld
  ENDI
  SKIP
ENDD
SELE PRIM
USE
SELE SECO
USE
```

(Heißt Record-Nummer <> 0)

Wie wäre es mit: (BestNDX ist Bestand-Index auf Suchmerkmal)

```
USE Update
SORT ON Suchmerkmal TO UpdateNeu
USE Bestand INDE BestNDX
UPDA ON Suchmerkmal FROM UpdateNeu REPL Feld
USE
```

Beschränkungen des UPDATE-Befehls:

1. Die FROM-DB (UpdateNeu) muß sortiert sein. Die aktive DB (Bestand) muß sortiert oder mit Index sein.
2. Als Modifikations-Optionen sind nur ein einfaches REPL oder ADD erlaubt!
3. WHILE-Option ist zulässig.

d B A S E: Ein paar TipsZählen so und so

Beispiel: Du willst wissen, wieviele große, mittlere und kleine Dosen auf Lager sind:

Langsam:

```

SELE PRIM
USE Dosen
SELE SECO
USE Anzahl
SELE PRIM
COUN FOR Groesse='Gross' TO Gross:num
SELE SECO
APPE BLAN
REPL Groesse WITH 'Gross',anzahl WITH Gross:num
SELE PRIM
COUN FOR Groesse='Mittel' TO Mittel:num
SELE SECO
APPE BLAN
REPL Groesse WITH 'Mittel',anzahl WITH Mittel:num
SELE PRIM
COUN FOR Groesse='Klein' TO Klein:num
SELE SECO
APPE BLAN
REPL Groesse WITH 'Klein',anzahl WITH Klein:num
SELE PRIM

```

Hierbei wird die Pimär-DB drei mal durchforstet - für jede Doesngröße einmal!

Schneller geht es so:

```

USE Dosen
INDE ON Groesse TO HilfNDX
GO TOP
STOR Groesse to m:Groesse
STOR m:Groesse+':num' TO Size:num      speichert 'Gross:num' in Size:num
DO WHIL .not. EOF
    COUN WHIL Groesse=m:Groesse TO &Size:num Hier kommt die Speed her!!
    STOR Groesse to m:Groesse
    STOR m:Groesse+':num' TO Size:num
ENDD
USE Anzahl
IF 0=test(Gross:num)
    APPE BLAN
    REPL Groesse WITH 'Gross', Anzahll WITH Gross:num
ENDI
IF 0=test(Mittel:num)
    APPE BLAN
    REPL Groesse WITH 'Mittel', Anzahll WITH Mittel:num
ENDI
IF 0=test(Klein:num)
    APPE BLAN
    REPL Groesse WITH 'Klein', Anzahll WITH Klein:num
ENDI

```

d B A S E: Ein paar Tips

Flexibler geht's es auch - dann ist sogar die Info über die möglichen Größenangaben der Dosen unnötig:

```

USE Dosen
INDE ON Groesse TO HilfNDX
GO TOP
STOR Groesse to m:Groesse
STOR m:Groesse+':num' TO Size:num
SET ALTE TO Hilf.
SET ALTE ON
DO WHIL .not. EOF
  COUN WHIL Groesse=m:Groesse TO &Size:num
  ? Groesse, &Size:num
  STOR Groesse to m:Groesse
  STOR m:Groesse+':num' TO Size:num
ENDD
USE Anzahl
SET ALTE OFF      Dieser Befehl stoppt das Schreiben in's alternate File
SET ALTE TO      Das tut erst dieser
APPE FROM Hilf. SDF
USE

```

WHILE versus DO WHILE

Das While im Count-Befehl ist nicht das Do-While! Es schränkt die Ausführung des Befehls, in dem es steht auf bestimmte Datensätze ein - funktioniert jedoch nur dann so, wie Du es vermutest, wenn die Daten nach dem relevanten Feld sortiert oder indiziert vorliegen und auf den erste passenden Datensatz mit LOCA, FIND oder GO positioniert wurde! Wenn der Befehl bis zum Ende der Datenbank laufen kann, dann sollte noch ein '.AND .NOT. EOF' in die Bedingungen aufgenommen werden! Sonst loopt die Kiste.

Hier die Befehle, die geWHILEt werden können:

```

APPE FROM Datenbank WHIL Bedingungen
COPY TO Datenbank WHIL Bedingungen
DELE WHIL Bedingungen
LIST Feld WHIL Bedingungen
RECA WHIL Bedingungen
REPL Feld WITH Inhalt WHIL Bedingungen
SUM Feld TO Wohin WHIL Bedingungen
TOTA Feld TO Feld WHIL Bedingungen

```

Langsam:

```

STOR 'Dave' TO MName
USE Datenbank
COUN FOR Name=MName TO Zahl

```

Besser:

```

STOR 'Dave' TO MName
STOR 0 TO Zahl
USE Datenbank
FIND &MName
DO WHIL Name=MName
  STOR Zahl+1 TO Zahl
SKIP
ENDD

```

Noch schneller:

```

STOR 'Dave' TO MName
STOR 0 TO zahl
USE Datenbank
FIND &MName
COUN WHIL Name=MName

```

d B A S E: Ein paar TipsSei nett zu intelligenten Anwendern ?

Auf eine Eingabe wird oft so gewartet:

```

STOR 'F' TO was
DO WHIL was='F'
  § 1,2 SAY "Watt nu?" GET was PICT '!'
  READ
  IF .not. was$'AB'
    STOR 'F' TO was
  LOOP
ENDI
ENDD

```

dBASE liest diesen Loop 2 Mal, selbst wenn das erste Mal eine richtige Antwort eingegeben wurde, denn angekommen beim ENDD geht dBASE wieder zurück zum DO WHIL und testet die Bedingung. Eine Alternative:

```

STOR 'F' TO was
§ 1,2 SAY "Watt nu?" GET was PICT '!'
READ
IF .not. was$'AB'
  STOR 'Watt nu?' TO frage
  DO Fragnach
ENDI

```

Ist die Antwort gleich richtig, geht's so schneller, anderenfalls wird es für Fehleingebende dafür langsamer, da Fragnach.CMD erst mal geladen werden muß. Dort drin kann man dann mit dem o.g. WHILE-Loop der Antworten harren.

Anm.d.HzN: Das finde ich einen genialen Weg, Zeit zu sparen! Ist doch die Eingabe von der Tastatur so unheimlich schnell im Vergleich zum Rechner!

True ist .not. False

dBASE kann logische Zuweisungen:

```

STOR 1 TO eins
STOR 2 TO zwei
STOR eins=zwei TO wahr
? wahr
.F.

```

Übrigens ist die =-Abfrage ca. 22% schneller als die <>- bzw. #-Abfrage.

Übrigens ist die Abfrage auf True/False langsamer als die auf einen Wert.

Slow-Motion:
DO WHIL var=t

Faster:
DO WHIL var='X'

d B A S E: Ein paar TipsSTOR TO Variable

Gemütlich:

```
STOR ' ' TO antwort1
STOR ' ' TO antwort2
STOR ' ' TO antwort3
```

Doppelt so schnell:

```
STOR ' ' TO antwort1,antwort2,antwort2
```

und noch schneller (2x):

```
STOR ' ' to a,a2,a3
```

Mehrfachzuweisungen zusammenfassen und kurze Variablennamen bringen's!

eVALuation

& ist schneller als VAL:

```
STOR '12345.678' TO Buchstaben
STOR VAL(Buchstaben) TO Zahl
    ergibt 12345
STOR &Buchstaben TO Zahl
    ergibt aber 12345.678 inklusive Dezimalstellen
```

Folgendes geht nicht mit &:

```
STOR VAL($(Buchstaben,1,3)) TO Teil
    ergibt 123
STOR VAL($(Buchstaben,6,3)) TO Teil
    ergibt .67
```

IF CASE oder wie oder was

Case ist schneller (ab 3 Möglichkeiten) als IF/ENDI-Sammlungen, da die Bedingungsprüfung beim ersten Treffer aufhört.

Anm.d.HzN: Übrigens gibt es auch noch Friede ... äh Else!

Datenbank-Feld versus Memory-Variable

dBASE sucht, wenn es eine Variable in einem 'STOR Variable TO ...'-Befehl findet erst einmal in der aktuellen Datenbank nach, ob dort ein Feld Variable existiert. Erst, dann wird nach Memory-Variablen gesucht.

Bei kleinen Anwendungen, sprich wenn noch etwas Speicherplatz für eine 2. Datenbank da ist, so daß sie ganz in den Speicher paßt, dann ist es schneller, die paar Variablen, die man braucht als Felder dieser 2. Datenbank zu definieren und direkt anzusprechen statt Memory-Variablen zu verwenden.

Club: Philosophisches**Unsere Maschine - und der Archimedes**

(Holger Göbel, 8630; 09561/15131)

Ende Mai hat Claudio Romanazzi ein paar Clubmitglieder zu sich eingeladen. HzN hatte seinen neuen Rechner (ist diese Bezeichnung Blasphemie?), den Archimedes, dabei. Man war beeindruckt von der Rechenleistung, die im Multitasking wahre Kunststücke vollbrachte, von der Benutzeroberfläche, von der Architektur des Rechners und seines RISC-Prozessors, von den schier unerschöpflichen Möglichkeiten seiner Programme, v.a. des Desktop-Publishing, und das alles für einen noch annehmbaren Preis.

Als ich mit Herbert Oppmann dort hinfuhr, hatte ich noch etwas Bedenken, ob ich nicht der Versuchung erliegen könnte, meine geliebte schwarze Kiste in die Ecke zu stellen und überzuwechseln. Komischerweise aber wich diese Befürchtung während dieses Treffens und v.a. auch in der Reflektion danach immer mehr der Überzeugung, daß das nicht (so schnell?) geschehen wird.

Warum?

Gewiß nicht aus Trotz, aus dem Gefühl: 'jetzt erst recht'. Ich möchte versuchen, ein paar Gedanken dazu zusammenzutragen:

1. Ich brauche, und ich meine wirklich 'brauche'!, meinen MTX nur aus ein paar Gründen:
 - a) Zur Bearbeitung von Texten - da genügt mir eigentlich Newword oder Wordstar, v.a., weil Letzteres jetzt auch noch rechnen kann. Und zur schöneren Präsentation von z.B. Einladungskarten usw. nehme ich Bradford. Claudio will übrigens immer noch ein Textverarbeitungsprogramm schreiben, das auch Graphik einbindet.
 - b) Zur Verwaltung von Daten - ich habe mir eine Literatursammlung meiner Fachzeitschriften unter dBase erstellt, die ich nach Suchbegriffen bequem durchstöbern kann. Der Komfort dazu ist vollkommen ausreichend (trotz einer Datenmenge von einigen 100 kB).
 - c) Kleinere Supercalc-Rechnungen sind schon nicht mehr so unbedingt nötig - meine Haushaltsrechnung hab ich wieder aufgegeben, weil sie mir doch keinen Pfennig mehr Geld bescherten konnte.
 - d) zum Programmieren: Dieser Punkt ist zugegebenermaßen der mir Wichtigste. Selbst Programme zu entwerfen, nächtelang zu grübeln, zu knobeln, stolz zu sein, wenn es läuft, frustriert, wenn dann doch wieder nicht. Und der Ansporn, seine Programme auf einem vergleichsweise altertümlichen Rechner mit ansprechendem Outfit und angemessener Rechengeschwindigkeit zu versehen, mag noch größer sein als bei einem Computer, der das schon alles hat.
2. Nur ein Stichwort: Pioniergeist! Wie hat mein Rechner ausgesehen, als ich ihn 1985 kaufte, und wie jetzt - dank der Hilfe unseres Clubs. Noch ein Wort dazu von Claudio: 'Ich schau mir den Archimedes ganz gerne an, weil ich mir Anregungen holen will, was ich davon alles auf unserem Rechner (vielleicht sogar noch besser) realisiere.'
3. Auf der Heimfahrt mit Herbert O. habe ich Dinge über unseren Rechner und sein System erfahren, die mich zum Staunen brachten. HzN - vielleicht weißt Du schon zuviel und suchst Dir deshalb Neuland. Ich jedenfalls - und sicher viele andere - sind noch längst nicht an die Grenzen unseres Systems vorgestoßen. Wenn ich nur an die Z280-Karte denke, die ein paar Clubmitglieder aufbauen wollen!

Club: Philosophisches

4. Jeder sollte sich im Klaren sein, daß im Moment des Kaufes eines neuen Rechners dieser schon wieder veraltet ist. So ist das eben: der Eingriff in die Geschichte hat etwas Vergängliches an sich.
5. Die Zeit am Computer müssen sich viele von uns irgendwie stehlen - nicht nur sich stehlen, sondern auch von z.B. der Familie stehlen. Das Einarbeiten in ein neues System braucht wieder unverhältnismäßig viel Zeit. Der Preis (in DM) ist irgendwo auch ein Stehlen von anderen Projekten, die man sich (und der Familie) gönnen will.
6. Ohne sentimental zu werden - ich hänge an meinem MTX. Es ist z.B. schön, wenn Kollegen weder mit 'Memotech' noch mit 'CP/M' etwas anfangen können. Mein Memotech und ich sind eben etwas besonderes.
7. Mir würde schon noch einiges einfallen, aber jetzt Schluß!

Sicherlich führen jetzt manche ganz vehemente Argumente für einen neuen, leistungsfähigern Rechner an. Schön - heute und wahrscheinlich noch für einige Zeit jedenfalls können sie mich nicht so recht überzeugen.

Anm.d.HzN: Zu Holgers o.g. Punkten kann ich mir einige ergänzende Anmerkungen nicht verkneifen:

1. Da gebe ich HG vollkommen Recht: Wirklich brauchen tue ich vermutlich lediglich einen Rechner, der eine einigermaßen anständige Textverarbeitung bietet - und durch den Club noch etwas Datenbank.
Für mich ist der Rechner ein reines Hobby - und da ...
2. Auch beim Archimedes kann man sich mit Pioniergeist noch reichlich austoben, da es, obschon er auch in England mittlerweile recht verbreitet ist, noch an einiger Software mangelt - insbesondere an anständigen Libraries zur bewuemen Programmierung.
3. Über die Z280 haben Olaf und ich auch eine Weile nachgedacht und das Thema dann doch irgendwann fallengelassen - mit einem weinenden und einem lachenden Auge.
Die beiden Hauptgründe für unsere Entscheidung waren zum einen die mittlerweile recht inhomogene Hardware des MTX (CPU auf Hauptplatine ja/nein, ECB auf Buskabel, in FDX oder garnicht usw.) sowie der Umstand, daß ein Umstieg auf die Z280 bedeutet hätte, daß die CPU, wenn es Sinn machen soll im 16-Bit-Modus laufen und ihre MMU nutzen muß, also folglich auch RAM 6.2 weitgehend neu geschrieben werden müßte. Der Kram mit Port 0 usw. hätte weg gemußt, damit es sich wirklich lohnt. (Claudio kann davon sicherlich so langsam ein Lied singen). Das käme weitgehend einem Neubeginn nahe ...
4. Klar, daß jeder Rechner schon mit seiner Auslieferung veraltet, aber wir hoffen, daß der zur Zeit sehr moderne Archimedes noch eine Weile aktuell bleibt.
5. Wem sagst Du das? Das ist sicherlich eine recht drastische Konsequenz des neuen Rechners - zumal Olaf und ich auch noch einen Club für den Archimedes gegründet haben. Tel Aviv, wie der Franzose sagt.
6. Ich hänge auch am MTX!!! Der wird auch weiterhin bei mir stehen und für einige Dinge genutzt werden. Irgendwann werden wir es auch geschafft haben, daß wir den MTX vom Archimedes aus bedienen und dessen Bildschirminhalt auf dem Archi als Fenster sehen können - damit endlich die zweite Tastatur und Glotze vom Tisch kann.
7. Uff...

S o f t w a r e: Diverses

Verschiedenes

(Holger Göbel, 8630; 09561/15131)

Für ein großes Thema reicht dieser Club-Beitrag nicht, deswegen möchte ich ihn 'Verschiedenes' nennen. Fangen wir an:

I. Unsere Maschine - und der Archimedes

Anm.d.HzN: Diesen Teil habe ich als eigenen Artikel ausgelagert (s.o.), da er, im Gegensatz zu den restlichen Punkten nicht so ganz unter die Überschrift 'Software' paßte.

II. Noch einmal: Turbo Pascal unter CP/M und MsDos

Eine Macke habe ich noch festgestellt beim Übertragen eines unter CP/M-Turbo-Pascal 3.0 geschriebenen Programms auf einen DoMessDos-Rechner:

Lokale Variablen einer Prozedur bleiben unter CP/M nach Verlassen der Prozedur unangetastet. Beim nochmaligen Aufruf der Prozedur kann man sich darauf verlassen, daß der alte Wert noch darinnen steht.

Nicht so unter MsDos: Hier werden die lokalen Variablen offensichtlich nach Verlassen der Prozedur freigegeben. Man kann sich nur behelfen, indem man die interessierenden Variablen global deklariert.

Eh ich's vergesse: Meine Routinen zum Berechnen osternabhängiger Feiertage (INFO 42 und PD CLUB.057) sind leider etwas fehlerhaft. Profi Tilmann Reh hat mich zudem noch darauf aufmerksam gemacht, daß sie fürchterlich umständlich programmiert sind. Sollte sie tatsächlich jemand mal brauchen, kann er die verbesserten Versionen von mir bekommen.

III. Programm-Kritiken

Die Fülle der Programme auf den PD-Disketten ist immens. Sie auch nur annähernd auszuleuchten, würde eine lange Zeit verbrauchen. Nur ein paar möchte ich ansprechen:

1. MMenu6 von Claudio Romanazzi

Das ist ja eigentlich kein Programm, sondern eine Shell, eine Benutzeroberfläche also, von der ich die verschiedensten Aufrufe machen kann, ohne auf Systemebene heruntersteigen zu müssen, und das noch ganz auf meine Bedürfnisse zugeschnitten. Wer 'Windows' o.ä. kennt, weiß auch das Prinzip. Damit man eine Vorstellung davon bekommt, habe ich einmal 3 Menüs, so wie ich sie benutze, als Kopie beigelegt: (Anm.d.HzN: damit leichter auszudrucken gab ich sie ein.)

Nach dem Booten befinde ich mich im Basis-Menü.

Mit der Maus fahre ich die einzelnen Fenster an (keine Ikonen, ich kann ja schließlich lesen), klicke kurz mit der Maus, und schon macht der Rechner das, was ich ihm beige knuspert habe. Z.B.: Mal schnell alle Club-Beiträge, die ich geschrieben habe, aus der Library holen, Wordstar aufrufen, einen neuen schreiben, den wieder in die Library einpacken. Oder: Versicherungs-Schriftverkehr automatisieren. Oder: Mal schnell ins Assembler-Menü wechseln, mein Programmier-Objekt bearbeiten, assemblieren, linken, als Klix assemblieren, backupen usw. usf. All diese Befehle hintereinander von Systemebene einzugeben dauert lang und ist mitunter fehlerträchtig.

Software: Diverses

Ausgang

BASISMENÜ

Kommando

DU: =G00:

Loggen	WordStar	MTXEdit	dBase	Turbo	Assembler
LoadDir	Versich	Menu	Literatur	Dienstplan	
Kopieren	KVB	SysFiles			
Library	StMarien				
DiskMan	MTX-Club				
Show					

Demgegenüber steht natürlich die Arbeit, sich solche Menus zu erstellen. Diese ist zwar einmalig, aber nicht ganz so einfach. Vor allem diese umständlichen Bildschirm-Codes. Die Gebrauchsanleitung von Claudio ist umfangreich und beinhaltet das Wichtigste. Bis man alles versteht, muß man sich allerdings schon recht intensiv damit auseinandersetzen. Wenn man sich dann aber schöne Menus erstellt hat - dann geht sie ab, die Post.

Diskettenmanager CR 6.91

DU: =G00:

Config ?	Was darf es sein ?	Laufwerk ?
03 09 0A 1B	zurück	A B C D
1C E0	Format	E G H
	Rcheck	I J K L
	DU	M

S o f t w a r e: Diverses

Ich würde mir eine Erleichterung noch wünschen: Es wäre doch machbar, daß ein Hilfsprogramm diese Bildschirmgestaltung vereinfacht (wo steht schon wieder dieses Fenster? Mist, 2 nach links, 3 zu groß ...). Das könnte so ablaufen: Mit der Maus oder den Cursortasten Fenster auf dem Bildschirm hin- und herbewegen, bis sie passen und ablegen. Dann was hineinschreiben ... und das Hilfsprogramm macht zum Schluß eine entsprechende Textdatei daraus, die man in sein Menu-Masterfile einbindet. Unter den Zeilen mit dem Bildschirmaufbau sollen dann schon die Koordinaten der Fenster stehen, die aktiv werden, wenn der Mauszeiger sich dorthin verirrt hat und angeklickt wird.

Anm.d.HzN: MTX-Edit kann da etwas helfen: Mit ^KT einen Bildschirm anzeigen lassen und dann mit dem Cursor an die gewünschte Stelle laufen. Drückst Du nun die Leertaste, so stellt MTX-Edit automatisch eine entsprechende Cursor-Positionierungs-Bildschirmsequenz in den Text. Das ist zwar nicht ganz so einfach, wie Du es oben skizziertest, könnte aber sicherlich schon etwas helfen.

Kommando	Assembler G00:WYSIWIG.Z80 Turbo WYSFONT		
<u>Programm</u> lauf	<u>Entwick</u> lung	<u>Thema</u>	
WYSIWIG	Editieren	Wechseln	
WYSFONT	Turbo	S WYSIWYG.Z80	
Klix laden	Assemblieren	S *.REL	Backup
Klix löschen	Linken	S *.COM	
	Assemble+Link	S WYSFONT.PAS	
	Klix erzeugen	Sichern *.*	

Fazit: MMenu6 ist Spitze, die Erstellung der Menüs nicht ganz einfach.

2. Catw von Olaf Krumnow

Auch schon älter, dieses intelligente Suchprogramm, das in Katalogen (also Disketten-Inhaltsverzeichnissen) die Programme, die man doch schon irgendwo mal gesehen hat und jetzt unbedingt braucht, wiederfindet. Allein die Club-PDs haben schon über 3000 Dateien! Mein sehnlichster Wunsch: Catw soll auch in Libraries reinschauen können.

Tip.d.HzN: Nimm Mountlib Eigentlich sollte doch für den Katalog (alle PDs machen zusammen ca. 170kB aus) noch Platz auf einer Diskette sein ...

S o f t w a r e: Diverses

3. QCC von HzN

Ein Kopier-, Vergleichs-, Archivier- und-und-und-Programm, das fast keine Wünsche mehr offen läßt. Fast. Manches ist jedenfalls gewöhnungsbedürftig:

`qcc g:=k:test.com test.mac` läuft richtig, aber

`qcc g:=k: test.com test.mac` (man beachte das Leerzeichen) kopiert alle

Dateien von k: nach g: und kann ganz schön Unheil anrichten, wenn man es so installiert hat, daß es ohne Nachfragen überschreiben soll. Schlimm ist das Ganze noch, weil QCC nicht auf ^C reagiert!

Anm.d.HzN: Erst mit ^S die Bildschirmausgabe anhalten, dann funktioniertu auch ^C - das ist nämlich Sache des BDOS!

Bei der Mehrfach-Kopie zwischen verschiedenen Usern muß bei den Quellen jedesmal der User explizit angegeben werden, also so:

`qcc j10:=g1:test.com 1:test.mac 1:test.rel`, wenn das nicht geschieht, nimmt er bei der Quelle den User 0 (oder den eingeloggten?).

In der deutschen Version stehen im Hilfsmenu die Buchstaben für Quelle und Ziel vertauscht (also Z und Q).

Man kann zwar QCC dazu überreden, etwas mehr zu 'quatschen', im Quiet-Modus, den ich eingestellt habe, ist es mir allerdings doch zu still. Wenn QCC Dateien, die es kopieren soll, nicht findet, dann sollte es das schon mitteilen, und nicht so tun, als sei alles ok. Ein Tipfehler bei der Dateiangebe hat dazu geführt, daß die Programmierarbeit eines ganzen Abends futsch war (haha - sei nur schadenfroh, HzN).

HzN wird jetzt wieder sagen: ich hab doch den Quellcode rausgegeben, patch doch selbst. Aber: erstens kennt der Autor das Programm wohl am besten, zweitens ist es schon wichtig, daß die Pflege eines Programms in einer Hand bleibt.

4. Library-Programme

Der Aufruf von LPUT (und zwar der neuesten mir zugänglichen Version 16 von Club.901 und .801), führt bei mir zum Rechnerabsturz.

LBREXT (V.29 von Club.801 und Klick.010) beharrt auf der CRC-Prüfsumme, die NULU leider nicht ins Directory einträgt. Der Library-Checker LBRC von Olaf Krumnow kann da weiterhelfen (der kann nämlich die CRC-Summe nachträglich einsetzen).

Anm.d.HzN: Lput 1.6p von KLICK.010 sollte laufen.

LBREXT 3.0p von KLICK.011 hat keine Probleme, wenn eine Datei keine CRC-Prüfsumme hat - genau daß ist der Zweck der darin gemachten Änderung.

NULU kann gefährlich sein: Ich habe eine Library gekruncht und anschließend NULU mit ^C (wie man's eben von DU3 gewöhnt ist) verlassen, ohne vorher die Library zu schließen (mit -c oder -x). Die Library ist anschließend noch da und nimmt weiterhin gehörigen Diskettenplatz ein - nur lassen sich die Dateien in der Library nicht mehr ansprechen, denn NULU hat das Directory seiner Library gelöscht und nicht aktualisiert.

S o f t w a r e: Diverses

5. Einige Klicker

Ohne die könnte man nicht mehr leben. Toll ist z.B. Mountlib von Jan Bredereke, mit dessen Hilfe man eine Library wie ein R/O-Laufwerk ansprechen kann. Ich war te nur sehnsüchtig auf die angekündigte neue Version, die mehr als nur eine Library verwalten und auf verschiedene User legen kann.

Manche Klicker haben an der Stelle ClrOld im Klix-Header keine 0. Das führt dazu, daß beim nochmaligen Laden Schaden angerichtet wird: Der Maustreiber z.B. hängt sich noch einmal zwischen TOAM und FREE und verkleinert dadurch den ach so kargen Platz. Ein nochmaliges Laden von Dijey führt sogar zum Absturz.

6. E60 (oder MTX-Edit)

Manchmal stürzt das Teil beim Suchen ab. Wann genau, weiß ich noch nicht. Unvorteilhaft ist die Cursorsteuerung: Mal hoch- und wieder runtergeblättert kann zur Folge haben, daß der Cursor irgendwo in der Zeile steht, was (besonders bei Blindschreibern) ärgerlich ist.

Ansonsten: Ein flinker Text-Editor, der noch besser würde, wenn er die Fast-Screen-Routinen verwenden würde, und vor allem, wenn er die Folding-Fähigkeit erhielte, die HzN immer mal einbauen wollte. Was Folding ist, soll HzN erklären.

Anm.d.HzN: Also gut: Folding bedeutet, daß man eine oder mehrere Zeilen des Textes von der Bildschirmanzeige entfernen kann - sprich sie sind (fast) nicht mehr zu sehen, damit man so leicht auseinanderliegende Teile des Textes beagapfeln kann.

Jetzt muß ich aufhören. Nicht, weil es keine bemerkenswerten Programme mehr gäbe, sondern weil es jetzt zuviel wird. Aber: bei unserem kleinen Treffen bei Claudio hat sich wieder einmal gezeigt, daß viele von uns Programme gar nicht kennen, die andere mit Vorzug (und das mit Recht) einsetzen. Ich würde mir deswegen schon ein bißchen mehr Programmbesprechungen wünschen, damit man aufmerksam wird und all das auch nutzt, was unseren Rechner so ungemein vielseitig macht. Ich bin mir sicher, daß ich dann auch noch lange Zeit auf einen anderen, im ersten Moment besser anmutenden (was heißt schon: besser? Ist schneller schon gleich besser?) Computer verzichten kann.