

MTX *User-Club Deutschland*

Info 39
15.10.1990

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. 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 hier 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! Wir verschicken nur gegen Vorkasse!

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	Christian Löhrmann	Hans Gras
Sonnenau 2	Grevenbleck 24	Statenhoek 49
2000 Hamburg 76	3005 Hemmingen 1	NL 1506 VM Zaandam
(040) 200 87 04	(0511) 41 78 77	(0031-75) 17 49 91

Telefon-Sprechzeiten

Herbert zur Nedden: Do 18 - 22 Uhr, Sa 9 - 14 Uhr

Inhaltsverzeichnis

C l u b	
Korrektur & Nachtrag	Seite 2
Fragen & Antworten	Seite 2
BootEPROM (unter Fragen & Antworten)	Seite 2
L e s e r b r i e f	
Hartmut Traber, 5270	Seite 3
Hans Gras, NL-1506	Seite 4
Wolf-Dieter Zimmermann, 4330	Seite 5
S u p e r C a l c	
Balkendiagramme	Seite 6
Z C P R 3.3	
Programmier-Projekte verwalten	Seite 10
ZGREP	Seite 12
Aliase	Seite 12
C P / M	
Dateiextensions	Seite 13
H a r d w a r e	
Wie baue ich mir einen Port	Seite 15
Resetfest	Seite 19
S o f t w a r e	
RAM 6.1	Seite 20
Loaddir	Seite 22
K o m i k	
Minirechner	Seite 23

Preis für dieses Info: DM 8.60

Redaktionsschluß für's nächste Info: 01.12.90

RAM 6.1 (Herbert zur Nedden, 2000)
 Nun ist's doch soweit. RAM 6.1 ist zu haben. Der Hauptgrund für die Version 6.1 ist, daß es allmählich unübersichtlich mit den Patches wird und vor allem Programme abfragen können sollten, daß die laufende Version die Fehler nicht mehr hat. Diverse RAM 6.x-Dienstprogramme sind geZCPPR 3.3t, d.h. sie können ggf. das Fehlerflag setzen und INST61 kann die Installationswerte aus RAM 6.0 in RAM 6.1 übernehmen. Außerdem habe ich aus P2DOS 3.3 einige Korrekturen des P2DOS und ein paar schöne Ideen übernommen. Wenn Du weiterhin mit RAM 6.0 arbeiten willst, bitte UNBEDINGT folgendes tun:

```
GET 100 RAM60.COM;POKE 3F42 D1;SAVE 298 RAM60N.COM S
DIFF RAM60.COM RAM60N.COM /V
```

Der einzige Unterschied darf dieses Byte sein. In RAM60.COM war es ein C1.

Info auf Diskette (Herbert zur Nedden, 2000)
 In diesem Info werfe ich die Frage auf, ob das Info und dann natürlich auch die Angebotslisten zukünftig auf Diskette statt auf Papier herausgegeben werden sollen. Dann kann aber vermutlich keiner mehr nur die Angebotslisten abonnieren, sondern muß das ganze Info beziehen! Der Preis pro Info auf Diskette wäre dann vermutlich DM 5-6 oder es käme gar gleich auf eine CLUB-PD. Was sagst Du dazu?

Kontostand (Herbert zur Nedden, 2000)
 Bitte prüfe den Deinen!

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, Sonnenau 2, 2000 Hamburg 76, 040 - 2008704:

--> Interessiert Dich einer der von mir angebotenen Posten: Mach ein Angebot!
Meine hier genannten Preise sind nicht unbedingt unumstößlich!
(zum Teil NEUE PREISE)

- Infos 11-36, gebraucht VB DM 75.- frei Haus
- Rikadenki Plotter RY21, VB DM 1500,-
Flachbettplotter, DIN A4, Aufnahme für Rotringstifte, incl. Handbuch und Schaltplan, 8085 CPU (Z80-aufwärtskompatibel), Centronics-Schnittstelle. Positioniergenauigkeit: 0,1 mm, Zeichengeschwindigkeit 200 mm/s, 19 Kommandos wie u.a. Kreise, Kreisbögen, Rechtecke, verschiedene gestrichelte Linien, Textausgabe in 4 Richtungen und verschiedenen Größen, absolute und relative Koordinaten, Markierungen auf Linien und Graphen. Kann in Textmodus gesetzt werden, um als Drucker zu arbeiten.
- Ich vermittele jederzeit gebrauchte/neue Geräte und Teile der selben.
- Ich habe RAMs für 768kB-Erweiterung!
- Disketten-Laufwerke: (gebraucht und getestet)
 - EPSON SD 560: 5 1/4", 80-Spur, HD-Lw., d.h. Formate 1A, 1B, aber nicht 03, 07, 09, 0A usw. DM 80.-
- NewWord Version 2.16, im Tausch gegen Original-NW-Diskette: DM 50.-
- NewWord Version 2.02 DEUTSCH, im Tausch gegen Original-NW-Diskette: DM 50.-
- Original Microsoft-BASIC-Lizenz incl., M80/L80 Assembler/Linker, Handbuch, auf dem MTX lauffähig: DM 200.- (Microsoft liefert dieses Teil nicht mehr aus!)
- NonDoc-Editor für CP/M incl. Source, zeilen- und bildschirmorientiert, Macro-Fähig, mit Handbuch im Ringordner: VB DM 50.-
- Z80ASM (Club-PD Z80-Ass.): Listing und Handbuch, ca. 1.4 cm DIN A4: VB DM 5.-
- Extended FDX-BASIC von Andreas Viebke mit div. Tools und Handbuch: DM 20.-
- Einbau-Drehspulmeßgerät 0-50uA: DM 10,-
- Bücher
 - Programmierung des Z80, Rodney Zaks, 606 Seiten: DM 45.-
 - 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.-
- 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.-
 - TTL-IC's: 74LS175, 74LS368, 74LS173, 74LS158, 74LS258 je DM 0.50;
74LS10, 74LS11, 74LS21 je DM 0.30
 - Platinenstecker für Erweiterungen links am MTX-Grundgerät. Natürlich mit dem Gegenstück zu der Kerbe an Pin 5. je DM 4.-
 - Original-Memotech-Spielcassetten: Toado, Kilopede, Knuckles, Draughts, Reversi, Snappo, Blobbo, Utilities, Demo, StarCommand je DM 4.-;
 - 10 Disketten FUJI HD 5 1/4", gebraucht & o.k. DM 35.-, Originalverpackt: DM 60.-

Hilmar Pyttel, Universitätsstr. 38a, 4630 Bochum 1, 0234 - 331770

(Tel. tagsüber: 0234 - 319116) MTX, 768k, FDX, 2 Lw., TP 200 (schwarz), DMX 80, RAM 4.5. DM 1200.-

Peter Lang, Banderbacher Str. 24, 8502 Zirndorf, 0911 - 609595

2x EPPRON SD 521 (40 Spur) je DM 50.-, 2x TEAC FD55 GFR (80 Spur+HD) je DM 100.-, 1 FDX-Gehäuse (hinten Ausbrüche für div. Buchsen, oben angesenkte Löcher für Hauptplatten-Befestigung) DM 30.-, 2x orig. MTX-Tastatur (komplett, ohne Hauptplatte) je DM 30.-, 1x orig. FDX-Schaltnetzteil DM 50.-. Alles VB, plus P&V.

Holger Hansen, Scharnhorststr. 2, 3300 Braunschweig, 0531 - 795792

8 EPROMs 27256 + Netzteil für EPROM-Floppy: DM 35.-

K A U F

Claudio Romanazzi, Quaet Faselm Str. 12a, 3070 Nienburg, 05021 - 13750:

EDICTA Grafikkarte gesucht

Liebe Leserin, lieber Leser,

kürzlich hörte ich im Radio, daß schon in 60 Tagen Weihnachten sei. Mein Gott, wie doch die Zeit vergeht. Info 38 ist auch schon ein paar Tage alt. Also wurde es Zeit, Info 39 in Angriff zu nehmen.

Mittlerweile ist RAM 6.1 fertig. Bei der Suche nach dem Doppelstrich-Cursor haben einige von Euch versucht, mir zu helfen, und mir Disketten geschickt, von denen ich booten konnte, um nach einigen genannten Eingaben eben diesen Doppelstrich-Cursor zu sehen; nur er erschien beim besten Willen nicht. Konnte ja auch garnicht, da meine 80-Zeichen-Karte auf HGR aufgerüstet ist. Bei dieser Aufrüstung wurde die Portdekodierung dieser Karte etwas verfeinert. Der Controller liegt bei Memotechs Karte auf den Ports 38-3B, wobei 38 und 3A sowie 39 und 3B identisch sind - allerdings wird er nur über 38-39 angesprochen; daß die anderen beiden Ports dazu parallel liegen, ist Geiz in der Hardware. Nach der HGRisierung liegt der Controller nur noch auf den Ports 38 und 39, während 3A für HGR genutzt wird. Und beim Verlassen des KLIKK gibt RAM 6.0 etwas auf 3A aus, was für HGR-Karten O.K., bei Memotech-Karten jedoch weniger passend ist und den berühmten berüchtigten Doppelstrich-Cursor hervorrufen kann. Anscheinend genügt es nicht einmal, die ganze betroffene Software bei Problemen dem zu schicken, der helfen soll - der Rechner muß auch noch mit! Naja, so krass ist es wohl doch nicht; i.a. genügt immer noch eine bootfähige Diskette.

Wie ich im letzten Info erwähnte, hoffe ich ab Anfang 1991 unter einer neuen Anschrift (Alte Landstr. 21, 2071 Siek) zu wohnen. Der Grund dafür ist, daß ich zusammen mit meiner Frau ein Häusle baue. Ich hoffe, daß darunter meine Zeit für Euch und den MTX User-Club Deutschland nicht allzusehr leidet. Allerdings wird es immer wieder mal vorkommen, daß zu meinen Telefon-Sprechzeiten nicht oder nur teilweise erreichbar bin. Da ich nicht weiß, wie stark ich für den Hausbau in den nächsten Monaten eingespannt bin und wieviele Beiträge kommen, also nicht garantieren kann, ob es vor Weihnachten ein Info erscheint, hier schon mal vorsorglich: Frohe Weihnachten und ein Gutes Neues Jahr!

Wieder haben einige Mitglieder den MTX User-Club Deutschland verlassen - und zwar einfach durch Vernachlässigen des Kontostandes! Und auch von diesem Info werden wieder ein knappes Dutzend Mitglieder verschont, da das Konto nicht ausreicht. Ich muß schon sagen, daß es doch wohl nicht zu viel verlangt ist, wenn Du a) Deinen Kontostand prüfst, bevor Du den Umschlag aufreißt um das Info zu lesen und dabei dem Papierkorb überantwortest und b) wenn Du mir kurz mitteilen würdest, ob Du den MTX User-Club Deutschland verlassen willst. Viele Info- und Softwareautoren und ich geben sich Mühe, um das Info zu gestalten und den MTX User-Club Deutschland am Leben zu halten. Da ist es doch nicht zu viel verlangt, wenn Du die beiden o.g. Punkte berherzigst, oder?

Ich wurde vor einiger Zeit mal gefragt: 'Könnte das Info nicht auf Diskette erscheinen?' Seinerzeit stand das nicht zur Debatte, da Nur-MTX-Besitzer durchaus stark im MTX User-Club Deutschland vertreten waren. Mittlerweile glaube ich, daß jeder im Club über ein (oder auch mehrere) Diskettenlaufwerke verfügt. Daher meine ich, daß wir gemeinsam überlegen sollte, ob - oder ob nicht. Es ist klar, daß sich Informationen auf Papier leichter lesen und ablegen lassen, aber auch ein Drucker dürfte jeder haben, oder?

Falls ja, könnte ich keine Artikel mehr auf Papier gebrauchen - sondern nur auf Diskette (Diskformat/Größe ist egal - außer 1C/1D auf 5 1/4"). Das Info auf Diskette würde ca. DM 5-6 kosten. Der Versand würde im Format 03 auf 5 1/4" erfolgen; auf Wunsch selbstverständlich ggf. auf 3 1/2" (das müßte ich aber rechtzeitig erfahren - die etwas höheren Diskettenkosten ...). Wenn Du mir formatierte Disketten mit Aufkleber mit deinem Namen schickst, sparst Du Geld.

Laß mich bitte Deine Meinung noch dieses Jahr wissen! Ich möchte keinem das Info auf diesem Wege verleiden - nur woher soll ich erfahren, ob ich das tue, wenn Du mir das nicht vorher sagst!

C l u b: Korrektur & Nachtrag / Fragen & Antworten**Korrektur & Nachtrag**Info 38, Seite 13, Übersicht über die KLIX-Overlays: (Hans Gras)

CRT40.KLX ist ein Bildschirm-Treiber, der den 40-Zeichen Schirm unterstützt. CTR40 ist also so etwas wie der Televideo- oder ADM 31-Treiber von RAM 4.x/RAM 6.x. Aber CTR40 läuft nur unter RAM 4.x mit ZCPR2 und dem dazu mitgelieferten speziellen SYSIO.IO-Treiber. Ein kleiner Teil des CRT40 wird zwischen FREE und TOAM gespeichert. Vielleicht kann ein angepaßter CRT40 als KLIX-Overlay unter RAM 6.x so wie die bereits vermerkten TVI- und ADM-Treiber funktionieren. Es ist mir klar das er nicht ein internes Teil von RAM 6.x wird; aber kann er nicht als Erweiterung seine Dienste leisten? Ich stelle mir folgendes vor: der (neue, von jemand zu ändernde) CTR40, wird mit Hilfe von LOAD6x.COM ins KLICK geladen und initialisiert. Die Initialisierung setzt, ggf. über eine UTILITY-Funktion, zwei Einsprünge. Einen für die Ausgabe von Daten auf den Schirm und einer für einen 1-Sekunden-Interrupt für's Cursor-Blinken. Bei der UnInitialisierung entsprechendes. Das Umleiten lief bei der alten Version über den ZCPR2 I/O-Treiber und war nicht komfortabel. Bei der neuen Version müßte über ESC-'A'...'3' und ESC-'A'...'0' umgeschaltet werden (wobei ... eine nicht belegte Bildschirm-Sequenz ist). Vielleicht kann TVI/ADM auch umgeschaltet werden; z.B. ESC-'A'...'1' ist TVI, '2' ist ADM, '3' ist CRT40, '4' ist z.B. ANSI. Alles andere ist wieder der MTX 6.0. Wenn ausreichendes Interesse besteht und eine Anpassung an RAM 6.x, ggf. mit Erweiterung von RAM 6.x (einfach) machbar ist, dann habe ich vor, mir Gedanken zu machen, CTR40 entsprechend anzupassen.

Anm.d.HzN: Grundsätzlich müßte es möglich sein, einen 40-Zeichen-Treiber für RAM 6.x auf der Basis von Hans' CTR40 als KLIX-Overlay zu erstellen. Dabei sollte meines Erachtens überlegt werden, ob es erforderlich ist, daß ein solcher Treiber so leistungsfähig wie der MTX 6.0-Treiber ist. Sicherlich wäre so ein Teil eine feine Sache für Debugging-Zwecke oder für Anwendungen, die die HGR-Grafik auf der 80-Zeichen-Karte für Grafik nutzen und auf dem anderen Schirm diverse Informationen als Text bereitstellen wollen. Bei diesen beiden Anwendungsbeispielen hingegen müßte der CTR40-Treiber einen eigenen Einsprung erhalten, damit die beiden Bildschirme parallel genutzt werden können.

Fragen & Antworten

F: (Holger Bauer, 7101)

Ich habe mir kürzlich auf einer Auktion einen Panasonic KX-P-1091-Drucker ersteigert. Leider ohne Handbuch! Ich bekomme aus dem Teil nur Pica-Schrift raus. Elite bleibt einfach aus. Wer kann mir (gegen Kostenerstattung) mit einem Handbuch oder einer Kopie davon helfen?
Außerdem reagiert der Drucker einfach nicht auf die NewWord .CW-Befehle.

F: (diverse)

Woran erkenne ich, ob ich ein Boot-EPROM von Michael Keßler in der FDX habe, also kein 60k-CP/M unter RAM 6.x laufen lassen kann?

A: (Herbert zur Nedden, 2000)

Wenn Du Zweifel hast, ist es kein Keßler-EPROM!

Michael ist ein Fan von Fenstern und Rahmen. Und das ist bei seinem Boot-EPROM nicht zu übersehen! Meldet sich das Boot-EPROM aber unauffällig ohne Fenster, so ist ein Original-Memotech oder ein Preusing-EPROM, also 60k-CP/M-tauglich unter RAM 6.x.

L e s e r b r i e f: Hartmut Traber, 5270

Hartmut Traber

02.09.90
Hohbeulstr. 8
5270 Gummersbach
Tel.: 02261/65399 ab 16.00
02261/80200 bis 15.45

Lieber Herbert,

ich hoffe, es ist doch noch nicht zu spät für das nächste Info.

Leserbrief:

Gleich nach dem Erhalt des letzten bestellte ich bei Erik d'Hondt das neue Grafik-Prom und die zugehörige Software.
Das "Zeug" kam prompt und ist

absolute Spitze!

wie man sieht.

Für einen lächerlichen Betrag erhält man außerordentliche zusätzliche Möglichkeiten bei der Textgestaltung.

Der Wermutstropfen:

nur unter Newword.

Ich glaube aber, daß Erik es bald geschafft haben wird, auch den Drukkertreibern von Wordstar die gleichen Möglichkeiten zu eröffnen, für Wordstar selber läuft es schon.

Um es kurz zu halten: Diese Hard- und Software ist ein Muß für jeden MTX'ler.

Aber Vorsicht: Wer das Boot-Eprom von M. Keßler benutzt, wird feststellen, daß dieses auf das Grafik-Prom zugreift und nach dessen Entfernung bzw. Austausch Bildschirm-Müll produziert.

Auch für dieses Problem hat Erik eine Lösung!

Vermutlich steht was darüber an anderer Stelle in diesem Info.

Eine Anmerkung möchte ich in diesem Zusammenhang noch loswerden: Es kam vor Zeiten mal eine leise Kritik im Info: Doc's in Holländisch oder Belgisch (ich weiß es nicht mehr).

Für solche Kritik habe ich nichts übrig, Fortschritt in Belgisch ist ebenso Fortschritt wie in Englisch, und mit Englisch lebt doch wohl jeder?

In diesem Sinne:

Danke Erik!

Hartmut Traber

Anm.d.HzN: Ich habe diesen Text mit WordStar ausgedruckt, allerdings nicht mir der Idee von Erik sondern mit der für die Rahmengrafikzeichen bekannten Lösung von Michael Keßler. Ich wollte diese Seite nun wirklich nicht erst mal unter NewWord drucken.

L e s e r b r i e f: Hans Gras, NL-1506

Bester Herbert:

Dein Hinweis für den Patch der StepRates des Festplatten-Laufwerks hat geklappt. 200uS läuft seit einiger Zeit ohne Probleme; mal sehen ob es schneller geht.... 3 Wochen später: keine Geräusch- oder Geschwindigkeits-Verbesserungen.

Laut meiner Untersuchungen hat Kaypro 10 Format (#C5) 128 Dir-Einträge, statt der in FORMATE.R6 angegebenen 64. Das bedeutet: ändern oder ein neues Format benutzen. Rcheck6 testet nur tracks 0...78 dieses Formats statt 0...79. Zur Untersuchung: In FORMATE.R6 haben Formate 60, 61, 62 und 84 alle 128 byte pro Sektor und sind Double Density. Ist das etwas neues, oder ist etwas falsch definiert?

Anm.d.HzN: Die Formatdefinitionen habe ich aus denen von Bernd Preusing übernommen, dessen Meinung ich mich anschließe: Toll, wenn überhaupt etwas stimmt! Korrekturen nehme ich gerne entgegen und gebe sie im Info weiter.

Ich habe auch etwas zur CLOCK zu melden: Wenn über das Programmchen UHR.COM, die Uhr auf "Keine UHR" umgeschaltet wird, funktioniert alles normal: Zeit steht. Normalerweise wird dann das Datum bzw. die Zeit über DATE.COM neu gesetzt. Klappt auch. Aber wenn zum Ende wieder auf die Hardware Uhr zurückgeschaltet wird, bleibt das alte Datum (von vor "Keine Uhr" gesetzt), aber wir haben die Zeit der Hardware Uhr. Eine RESET hilft meistens, ist aber ein wenig unschön. Bitte beim zurücksetzen der Hardware Uhr auch wieder Datum und Zeit der Hardware Uhr übernehmen.

Anm.d.HzN: Der Uhren-Interrupt berechnet das CP/M-Datum (Tage seit dem 1.1.78) nur dann neu, wenn er merkt, daß dafür ein Anlaß besteht. Darum bleibt das alte Datum bestehen - das spart nämlich einiges an Zeit und macht den Rechner schneller - ab RAM 6.1 wird die Uhr sogar nur alle Minute ausgelesen! Mit UHR wird dem Rechner gesagt, er habe keine Uhr - der Uhreninterrupt erhält aber kein neues Datum, tut also nichts. Wenn Du das aber willst, solltest Du einfach mit einem Alias mittels DATE das Datum neu stellen. Folgendes Alias zeigt Datum/Zeit an

```
d echo $DD.$DM.$DY $DH:$DN:$DS
```

Wir haben jetzt ein gutes System für Date/Time Stamping. Bitte auf allen (Club-) Disketten die Zeiteinträge eintragen. Auf einer Floppy sind ausreichende DIR-Einträge vorhanden. Wir benutzen doch .LBR wenn es knapp auf einer Floppy wird. Oder? Auch auf einer FDX ohne P2DOS usw. sind die Floppies ohne viele Probleme benutzbar.

Anm.d.HzN: Im Prinzip hast Du recht. Nur habe ich es immer wieder mal bei den Disketten im Format 03 erlebt, daß die DIR-Einträge nicht ausreichen, wenn ich Zeiteinträge verwende (48, d.h. 64 minus 1/4 für Zeiteinträge, sind gelegentlich zu wenig). Da ich davon die Nase voll hatte, habe ich die Zeiteinträge verbannt; und mit Libraries (.LBR) arbeite ich auch nicht immer. Außerdem kopiert MFT die Einträge nicht mit (DIJEY kann das mittlerweile; demnächst auf KLINK.012) und ich habe nicht auf all meinen Disketten Zeiteinträge. Ich habe schon genug mit dem Kopieren und Zusammenstellen zu der PDs tun, so daß ich mich nicht auch noch mit Zeiteinträgen herumschlagen will. Obendrein erhalte ich einige Programme ohne Einträge - was soll ich dann verwenden?

Nur das alte SUB macht Ärger. Eigentlich nicht SUB.COM, aber das SUB-Handling des Digital Research CCP, wenn die \$\$\$SUB-Datei auf einem Sektor steht wo auch ein P2DOS-Stamp ist. Warum es nicht klappt, ist mir nicht klar. Ich habe es entdeckt beim Fertigen einer 3.5", Typ #07 Bootdiskette mit RAM 1 (von BP). Dieses System benutze ich weil CONTACT.COM einige Feinheiten gegenüber M1.COM hat, und CONTACT nicht mit RAM 3.x, RAM 4.x oder RAM 6.x zusammen läuft, wegen der eigenen Interrupts.

L e s e r b r i e f: Wolf-Dieter Zimmermann, 4330

Hallo, Herbert

dank Deiner Tips und dem Hinweis eines meiner Kollegen aus eine Münchner Firma habe ich nun die beiden Formate von Bernd Preusing in die CPM-Emulationssoftware (Anm.d.HzN: für MsDos-Geräte) eingebaut. Falls also noch jemand vom Club in die gleiche unangenehme Lage wie ich kommen sollte und plötzlich nicht mehr auf den Memotech Rechner zugreifen kann, jedoch die Disketten im 09 oder 0A Format (das 03-Format stimmt irgendwo nicht, deshalb habe ich es nicht implementiert. Wenn Du mir alle für den Club interessanten Formate in der untenstehenden Form lieferst, will ich die Datei gerne ergänzen) vorliegen hat, kannst Du ihn gerne an mich verweisen.

Ich lege Dir außerdem die nicht indizierte Liste von CPM-Formaten bei. Vielleicht nützt Dir das Wissen bei einer Anfrage.

Es bleibt mir abschließend nur eines:

Dem Memotech-Club Deutschland und seinen emsigen Programmierern ein ganz herzliches Dankeschön zu sagen. Ich weiß eines, ohne den Club und ohne die Hilfe, die ich in den verschiedensten Problemlagen (angefangen bei Boot-Problemen, dabei aber noch lange nicht aufgehört) erfahren habe, wäre ich seit langem ausgestiegen. Da mir die Fähigkeiten zum Elektronikingenieur völlig fehlen, auch der Gebrauch eines Lötkolbens ziemlich fremd ist, war ich oftmals auf Hilfe von außen angewiesen. Daß ich also so lange dem Club "treu" geblieben bin, hängt nahezu ausschließlich an anderen Clubmitgliedern und auch daran, daß durch das Info mir vieles vertraut wurde, was mir aus den "Lochkartenzeiten", in denen ich erstmals mit Rechnern zu tun bekam, nicht bekannt war.

Laßt es euch weiterhin gut ergehen.

Wolf-Dieter

Anm.d.HzN: Ich habe die o.g. Liste der Wolf-Dieter verfügbaren CP/M-Formate auf Diskette vorliegen. Ich packe sie auf KLINK.012. Evtl. kannst Du anhand dieser Informationen das eine oder andere Problem beim Lesen von Disketten anderer Hersteller auf dem MTX unter RAM x.x lösen oder Fehler in den Formatdefinitionen in FORMATE.DAT bzw. FORMATE.R6 aufdecken und mir mitteilen. Da Wolf-Dieter den MTX User-Club Deutschland verlassen hat, hier seine Anschrift für den Fall, daß Du Dich wegen dem Lesen von CP/M-Disketten auf einem DoMessDos wenden willst: Steinkuhle 17, 4330 Mülheim/Ruhr, (0208) 434726. Ich habe ihm trotz seines o.g. Angebots keine Formatliste geschickt, da ich nicht weiß, was Dich als Format für MsDos interessiert.

Eine Formatbeschreibung, wie Wolf-Dieter sie benötigt hat folgenden Aufbau (Erläuterungen von mir in *kursiv*): (Du solltest sie ihm am besten auf einer MsDos-Diskette zuschicken, was ja mit RAM x.x kein Problem ist, natürlich ohne meine u.g. *Kommentare*.)

```
BEGIN BP09 BPreusing Memotech-Format 09 Kurzname und Name des Formats
DENSITY MFM,LOW Schreibdichte
CYLINDERS 40 phys. Spuren
SIDES 2 phys. Seiten
SECTORS 5,1024 Anzahl phys. Sektoren je phys. Spur
(diese Zeile ist nur mein Kommentar:) und Länge phys. Sektoren
SIDE1 0 1,4,2,5,3 phys. Seitennummer und Verteilung
SIDE2 1 1,4,2,5,3 der Sektoren auf den Seiten
ORDER SIDES entspricht unseren Track-Typ
BSH 4 BLM 15 EXM 1 DSM 194 DRM 63 ALO 128 AL1 0 OFS 2
(diese Zeile ist nur mein Kommentar:) Einige Felder des DPB
END
```

S u p e r C a l c: Balkendiagramme

Balkendiagramme mit SuperCalc

(Kurt-Bernd Rohloff, 8000)

Balkendiagramme werden in der beschreibenden Statistik gern benutzt, um die Verteilung einer (Zufalls-) Größe X anschaulich darzustellen. Dazu teilt man den Bereich der möglichen (oder zu erwartenden) Werte von X in kleine Intervalle, sog. Klassen, ein. Diese Einteilung ist letztlich willkürlich und erfordert ein bißchen Geschick, da sie das Aussehen der Verteilung erheblich beeinflussen kann. Liegt nun von der Größe X eine konkrete Stichprobe in Form einer Folge von Zahlen x_1, x_2, \dots, x_n vor, so ermittelt man nun, wie viele Werte davon in die erste, zweite usw. Klasse fallen. Diese Anzahlen bilden die Verteilung dieser Stichprobe. Im folgenden werde ich ein Spreadsheet beschreiben, das solche Verteilungen auf sehr einfache Weise erstellt. Die Größe X ist dabei, dies sei nur am Rande erwähnt, die Wertentwicklung von Rentenfonds (in Prozent) in den letzten fünf Jahren. Das Spreadsheet sieht folgendermaßen aus:

```

1  A 11  B 11  C 11  D 11  E 11  F 1
11Häufigkeitsverteilung          DEM01  nonext col
21Rentenfonds
311--6-11---9---11---9---11---9---11-----25-----1
41  Schalter:          1
51  Wert              1000
61  Klasseneinteilung:
71  Anfang            10
81  Breite             5
91  Anzahl            10
101
111          von      bis      Summe  Anzahl
121  -----
131          unter    10       1       1
141          10       15       2       1 *
151          15       20       4       2 ***
161          20       25       8       4 *****
171          25       30      27      19 *****
181          30       35      45      18 *****
191          35       40      46       1 *
201          40       45      47       1 *
211          45       50      49       2 ***
221          50       55      49       0
231          55       60      50       1 *
241          über     60      50       0
251  -----
261          Maximum          19
271          Summe            50
281          Skalierungsfaktor  1.263158
    
```

Beschreibung:

Das Spreadsheet muß im Modus NONEXT und COLUMNWISE betrieben werden (/GN und /GC). Außerdem wird es zunächst auf MANUAL gestellt (/GM, das ist beim Eintippen eines Spreadsheets immer ratsam). In der Zeile 3 sind die Breiten der einzelnen Spalten zu sehen. Die Spalte F wird natürlich im "*" Format dargestellt (/FCF,*,25). Das Spreadsheet wird wieder durch einen Schalter in C4 gesteuert. Ist C4=0, so wird das Spreadsheet initialisiert. In C5 werden später die einzelnen Werte (x_1, x_2 usw.) eingegeben. Mit C7 und C8 wird die Klasseneinteilung festgelegt. (Was zu tun ist, wenn man noch mehr oder weniger Klassen braucht, beschreibe ich später.) Alle anderen Zellen sollten geschützt werden! Die Zelle C9 gibt die Anzahl der Klassen an. In den Zeilen 14 bis 23 ist die Klasseneinteilung dargestellt. Dabei gilt immer der "von" Wert einschließlich, der "bis"

S u p e r C a l c: Balkendiagramme

Wert ausschließlich. Die Spalte "Summe" überspringe ich erstmal. In der Spalte "Anzahl" wird nun festgehalten, wieviele Werte in die entsprechende Klasse fielen. In obigem Beispiel fielen also 4 Werte in den Bereich 20 (inkl.) bis 25. (Die einzelnen Daten gebe ich hier nicht wieder.) Diese Zahlen werden in der Spalte F nochmal im "*" Format angezeigt, wobei der größte Wert automatisch die Spalte F voll ausfüllt, dafür sorgt nämlich der Skalierungsfaktor in Zelle E28. In der Spalte "Summe" werden die Zahlen aus "Anzahl" aufkumuliert. Tatsächlich ist es einfacher, zuerst diese kumulierten Werte zu ermitteln und dann daraus durch Differenzbildung die einzelnen "Anzahl"en (man braucht die kumulierten Werte auch zur Ermittlung des Medians). In "Maximum" (E26) ist die größte Anzahl festgehalten. In "Summe" ihre Summe, was immer mit dem Wert in D24 übereinstimmen sollte. Eine Sonderrolle spielen die Zeilen 13 und 24, die die x-Werte auffangen, die aus der gewählten Klasseneinteilung herausfallen. Sie werden nicht als Sternchen dargestellt.

Bedienung:

Das Balkendiagramm wird dann folgendermaßen erstellt:

- 1) Spreadsheet initialisieren (C4 auf 0 und !)
- 2) Spreadsheet "scharf" machen (C4 auf 1)
- 3) Zellcursor auf C5 stellen
- 4) Automatische Berechnung einschalten (/GA)
- 5) Werte eintippen
- 6) Manuelle Berechnung einschalten (/GM)
- 7) einen Wert ≥ 1000 (in diesem Beispiel) eintippen

Anmerkung:

Werte eintippen ist natürlich mühsam, insbesondere wenn sich die gewählte Klasseneinteilung nachher als ungünstig herausgestellt hat. Besser ist es, die Werte vorher in eine Datei zu schreiben, eine Zahl je Zeile. Dies kann von dBASE oder durch ein Pascal Programm oder auch wieder durch SC geschehen. Meistens werden die Werte ohnehin aus einem anderen Spreadsheet stammen. Man speichert sie dann in eine Datei mit /ODvon:bis,Dwerte.XQT ab, aber bitte ohne Border! Die Zahlen müssen dazu natürlich in einer Spalte untereinander stehen. Leere Zellen dazwischen stören nicht, aber Texte! Im Schritt 5 liest man dann die Werte als Programm ein: /Xwerte. Hat man beim Abspeichern nicht die Endung XQT gewählt, so muß man jetzt die Endung explizit mit angeben.

Arbeitsweise:

Um zu verstehen, wie das Spreadsheet arbeitet, müssen wir uns die dahinter stehenden Formeln anschauen. Ich habe deshalb das Spreadsheet noch einmal mit allen Formeln ausgegeben. Das sieht dann so aus (mit anderer Spaltenbreite):

```

1 A 11 B 11 C 11 D 11 E 11 F 1
11Häufigkeitsverteilung DEMO1 nonext col
21Rentenfonds
31
41 Schalter: 1
51 Wert 1000
61 Klasseneinteilung:
71 Anfang 10
81 Breite 5
91 Anzahl COUNT(B12:B25)
101

```

S u p e r C a l c : Balkendiagramme

	A	B	C	D	E	F
111	von	bis	Summe		Anzahl	
121	-----					
131	unter C7		IF(C4,IF(C5<C13,D13+1,D13))		D13-D12	
141	C13	B14+C8	IF(C4,IF(C5<C14,D14+1,D14))		D14-D13	E14*E28+.5
151	C14	B15+C8	IF(C4,IF(C5<C15,D15+1,D15))		D15-D14	E15*E28+.5
161	C15	B16+C8	IF(C4,IF(C5<C16,D16+1,D16))		D16-D15	E16*E28+.5
171	C16	B17+C8	IF(C4,IF(C5<C17,D17+1,D17))		D17-D16	E17*E28+.5
181	C17	B18+C8	IF(C4,IF(C5<C18,D18+1,D18))		D18-D17	E18*E28+.5
191	C18	B19+C8	IF(C4,IF(C5<C19,D19+1,D19))		D19-D18	E19*E28+.5
201	C19	B20+C8	IF(C4,IF(C5<C20,D20+1,D20))		D20-D19	E20*E28+.5
211	C20	B21+C8	IF(C4,IF(C5<C21,D21+1,D21))		D21-D20	E21*E28+.5
221	C21	B22+C8	IF(C4,IF(C5<C22,D22+1,D22))		D22-D21	E22*E28+.5
231	C22	B23+C8	IF(C4,IF(C5<C23,D23+1,D23))		D23-D22	E23*E28+.5
241	über C7+C8*C9		IF(C4,IF(C5<1000,D24+1,D24))		D24-D23	
251	-----					
261		Maximum			MAX(E12:E25)	
271		Summe			SUM(E12:E25)	
281		Skalierungsfaktor			IF(C4,24/E26)	

Die Spalten "von" und "bis" dürften wohl noch durchschaubar sein. Interessanter ist die Spalte "Summe". Dort wird, wenn das Spreadsheet "scharf" ist, der aktuelle Wert inkrementiert, wenn der momentane x-Wert in C5 kleiner als der "bis" Wert in der aktuellen Zeile ist, sonst bleibt die Zelle unverändert. Beispielsweise wird in D19 jedesmal um 1 erhöht, wenn der eingegebene x-Wert kleiner als 40 ist, also beispielsweise für 7, 12 oder 38. Es spielt hier also keine Rolle, ob der Wert größer oder gleich 35 ist. Genauso wurde aber auch in D18 verfahren, nur daß hier die obere Grenze bei 35 liegt. Die x-Werte, die größer oder gleich 35, aber kleiner als 40 sind, erhält man daher einfach aus der Differenz von D19 und D18, die in E19 eingetragen ist. Diese Zahlen werden nun noch mit dem Skalierungsfaktor in E28 multipliziert, um auf die Anzahl der Sternchen in Spalte F zu kommen. Die Addition von 0.5 rührt daher, daß SC bei der Anzeige der Sternchen aus dem Zahlenwert keine Rundung vornimmt, sondern den Nachkommaanteil einfach abschneidet. Damit eine Rundung erfolgt, muß also vorher zu dem Zahlenwert noch 0.5 addiert werden.

Anmerkung:

Die Formeln in Spalte D sind wie folgt verkürzt worden: Wenn in einer Bedingung kein Vergleich angegeben wird, vergleicht SC automatisch mit Null. Die Bedingung 'IF(C4,...)' ist also gleichbedeutend mit 'IF(C4<>0,...)'. Außerdem kann man den ELSE Teil einer Bedingung weglassen, SC nimmt dann automatisch Null an. Die Formel in E28 lautet also ganz ausführlich 'IF(C4<>0,24/E26,0)'.

Die letzte Formel in Spalte D fällt etwas aus der Reihe, denn dort wird C5 nicht mit, wie zu erwarten, C24 verglichen, sondern mit 1000. Dieser Wert wurde willkürlich festgesetzt unter der Annahme, daß kein x-Wert jemals 1000 erreicht oder überschreitet. Diesen Wert (oder einen größeren) können wir dann in C5 eintragen, wenn das Diagramm fertig ist. Das Ausschalten der automatischen Berechnung allein reicht nicht aus, denn SC berechnet das Spreadsheet auf alle Fälle jedesmal beim Laden. Daher darf in C5 nicht einfach der letzte x-Wert stehenbleiben! Der Skalierungsfaktor in E28 ergibt sich aus der Überlegung, daß der maximale "Anzahl" Wert, aus E26 ersichtlich, zu der Zahl 24 (genau gesagt 24.5) führen soll. Ja, man muß hier 24 nehmen, wenn die Spalte F 25 Stellen breit ist, weil, wie bereits erwähnt, SC immer die erste Stelle frei läßt.

Nun will ich noch die Frage erörtern, wie man die Anzahl der Klassen verändern kann. Das Einfügen bzw. das Löschen einer Zeile ist ein bißchen vertrackt, da die Zeilen ja miteinander verkettet sind. Ich habe daher zwei Programme geschrieben, die diese Aufgabe erledigen.

S u p e r C a l c: Balkendiagramme

Einfügen einer Zeile: Das Spreadsheet muß initialisiert werden. Den Zellcursor in Spalte B irgendwo in die Mitte, also zwischen Zeile 14 und 23 stellen. Dann das folgende Programm ausführen (/XHISTINS wenn es in der Datei HISTINS.XQT steht, bitte die Kommas genau beachten):

```

/IR
"Eine Zeile in ein Histogramm einfuegen 19.08.90
"anfangen in Spalte B, NONEXT Modus!!!
/CB14,
>
/CC14,,AYN
>
/CD14,,ANNYYY
>
/CE14,
>
/CF14,,AYN
<<<<V
/U
/CB14,
>>>
/U
/CE14,
/GR
!
/GC

```

Löschen einer Zeile: Wieder Spreadsheet initialisieren. Wo der Zellcursor steht, ist jetzt egal. Dann folgendes Programm ausführen:

```

=B14
V
/U
"Zeile 14 aus einem Histogramm ausfügen 20.08.90
/CB14,,N
/U14
/DR14
=E14
/U
^
/C,E14
/GR
!
/GC

```

Z C P R 3.3: Programmier-Projekte verwalten**Programmier-Projekte verwalten**

(Olaf Krumnow, 2050)

Vorweg:

Folgende Programme, die alle in der CLUB.9xx-Serie zu finden sind, benutze ich:

AC	ist ACOPY Version 2.2	<i>Kopierprogramm</i>
%	ist RESOLVE Version 1.4	<i>ZCPR3-Variablen-Umsetzer</i>
LOCNDO	in der Version 1.2	<i>Suche und tue dann was</i>
SHV	in der Version 1.3	<i>SHELL-Variablen-Bearbeiter</i>
REG	in der Version 1.3	<i>ZCPR3-Register-Bearbeiter</i>

Alle anderen sind CCP- oder FCP-Kommandos, bzw. Aliase, die beschrieben werden.

Statt immer über das Programmieren an sich zu schreiben, will ich hier mal aufzeigen, wie ich mir das Leben etwas leichter mache. Dabei greife ich natürlich voll auf ZCPR 3.3 und seine Utilities zurück. Kurz zur Situation: Wenn ich programmiere, tue ich das aus Gründen der Geschwindigkeit auf meiner RAM-Floppy. Natürlich ist die beim Einschalten des Rechners leer. Den Stand der Arbeit habe ich auf einer Diskette im Laufwerk D:. Da sich dort meistens mehrere Projekte tummeln, an denen ich mehr oder minder aktiv arbeite, habe ich jedem dieser Projekte einen einzelnen User zugeteilt.

Ursprünglich habe ich mit folgenden Aliasen gearbeitet:

BAK era \$tp1*.bak
löscht Backups (d.h. .BAK-Dateien) im angegebenen Directory

HOL if \$tf1=;ac \$1*.* \$hb *.lbr /d;else;ac \$1 \$hb *.lbr /d;fi
*kopiert Dateien in das aktuelle Directory und setzt das Archiv-Flag bei diesen Dateien. Libraries werden vom Kopieren ausgeschlossen, da ich darin i.a. abgeschlossene Versionen habe, die nicht mehr bearbeitet werden. Zwei Aufrufmöglichkeiten: HOL DU: oder HOL DU:filemask
Im ersten Fall werden alle Dateien aus dem Directory kopiert, im zweiten Fall nur die auf beim Aufruf angegebenen filemask passenden Dateien.*

BACK bak;ac *.* \$1 /nos;echo backup komplett!
Zunächst lösche ich alle BAK-Dateien, dann werden alle veränderten Dateien in das angegebene Directory kopiert.

ALL era *.*
löscht alle Dateien im aktuellen Directory.

Ich will jetzt mein letztes Programm LBRCHECK (ist auf der PD-Diskette CLUB.919) bearbeiten. Dazu schaue ich nach, in welchem Userbereich auf Laufwerk D: das gewünschte ist; in diesem Fall User 11. Dann

HOL D11: *kopiert alle Dateien auf die RAM-Floppy.*

Jetzt kann ich beliebig drauflos programmieren. Immer mal zwischendurch, und natürlich zum Schluß, gebe ich ein

BACK D11: *sichere veränderte Dateien*

um die veränderten Dateien zurückzuschreiben. Will ich nicht mehr an diesem Projekt weiterarbeiten, lösche ich mit

ALL

alle Dateien und habe wieder eine saubere RAM-Floppy.

Z C P R 3.3: Programmier-Projekte verwalten

Alles sehr schön und gut, aber nicht befriedigend.

Denn häufig kommt es vor, daß ich die Diskette in Laufwerk D: wechsele, aus welchem Grund auch immer. Mache ich jetzt ein Backup und habe die falsche Diskette im Laufwerk, landet das Backup auf einer völlig falschen Diskette. Ärgerlich!

Ferner habe ich etliche Projekte auf meiner Arbeitsdiskette, so daß ich immer erst umständlich suchen muß, wo das gewünschte Projekt untergebracht ist.

Also muß eine Lösung her, die mir das Suchen abnimmt und auch dafür sorgt, daß Backups auf der richtigen Diskette landen. Dafür braucht die Arbeitsdiskette ein wenig Vorbereitung. Ich lege auf jedem benutzten User eine OkB-Datei an, die einen Gartenzaun # und den Namen des Projektes beinhaltet; für obiges Projekt ist das #LBR (SAVE 0 D11:#LBR). Das dient zur Identifizierung des Projektes.

Ferner kommen folgende Aliase hinzu:

```
PROJEKT locndo d:#$1 pro1 $$
PRO1 shv pro$hb $1;hol $tp1
PROBACK % if ex %pro$hb;% back %pro$hb;reg se 0 q;else;    <<geht noch weiter!
        echo ^gfalsche diskette!;reg se 255 q
PROENDE proback;if ~er;shv pro$hb;all;fi
```

Beginne ich ein Projekt, so gebe ich

PROJEKT LBR

ein. LOCNDO ist ein kleines Programm, welches die angegebene Datei auf allen Usern sucht und dann eine Kommandozeile erzeugen kann. In dieser Kommandozeile steht jetzt Disk, User und Programmname mit drin. Die erzeugte Programmzeile sieht dann so aus:

PRO1 D11:#LBR

Das Alias PRO1 schreibt jetzt diese komplette Angabe in eine Shell-Variable, deren Name aus PRO und dem aktuellen Laufwerk und User besteht, z.B. PROH0 (meine RAM-Floppy ist H:). Anschließend werden mit dem Alias HOL alle Dateien auf die RAM-Floppy kopiert. Jetzt arbeite ich an meinem Projekt. Backups mache ich mit

PROBACK

welches jetzt keinerlei Parameter benötigt, da diese Informationen ja alle in der Shell-Variablen PROH0 stehen. Also wird nachgeguckt, ob die Datei #LBR im vermuteten Directory (nämlich D11:) steht, ob also die richtige Diskette im Laufwerk ist. Wenn ja, werden mit BACK alle veränderten Dateien auf der Diskette gesichert und das Error-Flag gelöscht. Liegt die falsche Diskette im Laufwerk, so wird eine Warnung ausgegeben und das Error-Flag gesetzt. Bin ich fertig, so beende ich die Arbeit mit

PROENDE

welches zunächst PROBACK aufruft. Tritt dort kein Fehler auf (if ~er), so wird die Shell-Variable PROH0 entfernt und anschließend mit ALL alle Dateien gelöscht. Ansonsten (falsche Diskette) wird natürlich nichts gelöscht. Fertig!

Ich kann mit diesen Aliasen auch mehrere Projekte gleichzeitig auf der RAM-Floppy haben, natürlich auf unterschiedlichen Usern, ohne daß Verwirrung auftritt. Habe ich z.B. noch ein Projekt auf H1:, so legt PRO1 eine weitere Shell-Variable an, diesmal mit Namen PROH1. Das funktioniert notfalls auch bei mehreren RAM-Floppies. Die Aliase arbeiten immer mit dem Projekt im aktuellen Directory der RAM-Floppy.

Z C P R 3.3: ZGREP / Aliase**ZGREP**

(Herbert zur Nedden, 2000)

In dem ganzen Wust der Z3COMs (das ist eine Sammlung aller .COMs der ZCPDR 3.3-Welt - sind auf CLUB.914-917) stolperte ich über ZGREP. Das ist ein Programm, mit dem ich Zeichenfolgen in einer oder mehreren Dateien suchen kann. Soweit kann FILZ noch mithalten, aber ZGREP kann nach ganz besonderen Strings suchen. Ich will an dieser Stelle nicht die ZGREP-Doku komplett abschreiben - sie ist in den Z3HELPS (CLUB.911-913) unter Z zu finden. Hier nur ein paar Varianten der Stringangabe mit Erläuterungen.

Syntax:

<code>\</code>	nächstes Zeichen unverändert, d.h. nicht als Steuerzeichen (ö)
<code>^B</code>	klar, Ctrl-B; geht natürlich auch mit anderen Buchstaben
<code>!</code>	Anfang der Zeile, wenn es am Anfang des Strings steht (ö)
<code>\$</code>	Rest der Zeile, wenn es am Ende des Strings steht
<code>_</code>	Leerzeichen oder Tab
<code>:a</code>	Buchstabe
<code>:c</code>	Irgendein Ctrl- außer Tab
<code>:d</code>	Ziffer
<code>:n</code>	Buchstabe oder Zahl
<code>:p</code>	Trennzeichen: !"#\$%&'()*+,-./:;<=>?@ÄÖÜ^_`äöüß
<code>[xyz]</code>	einer aus der Menge, also x, y oder z (ÄxyzÜ)
<code>[!xyz]</code>	keiner aus der Menge, also nicht x, y oder z (Ä!xyzÜ)
<code>[a-d]</code>	einer aus der Menge, also a, b, c oder d (Äa-dÜ)
<code>-</code>	hinter etwas: 0x oder 1x
<code>+</code>	hinter etwas: mind. 1x
<code>*</code>	hinter etwas: 0x oder öfter

Beispiel:

```
:a+:n*\: *_push$
```

Alle Zeilen, die mit einem Assembler-Label (d.h. Buchstaben gefolgt von Buchstaben/Zahlen und einem Doppelpunkt) und dann einen PUSH-Befehl enthalten.

Aliase

(Herbert zur Nedden, 2000)

Es kommt häufiger vor, daß ich testen will, was unter SUB, ZEX oder gar als Alias so abläuft - insbesondere, wenn in der Zeile Parameter stehen, und ich wissen will, was sie so bewirken. Dazu habe ich mir das Alias

```
# if in S%oll ich? %<$*;$*;fi
```

erstellt. Wenn ich nun vor ein auszuführendes Kommando den Gartenzaun und ein Leerzeichen stelle, so wird das Kommando erst mal angezeigt, und ich kann entscheiden, ob es ausgeführt werden soll.

```
DIR
```

wird sofort ausgeführt

```
# DIR
```

es erscheint 'Soll ich? DIR' und ich kann durch Eingabe von N die Ausführung von DIR verhindern; mit Y bestätigen. Statt DIR können da auch andere Kommandos stehen.

Hier ein weiterer Tester. Ist das ZCPR 3.3-Fehlerflag gesetzt, erscheint 'ERR' auf dem Bildschirm, anderenfalls 'OK'.

```
## if er;echo ERR;else;echo OK;fi
```

WordStar kann man ja beim Aufruf nicht nur wie NewWord den Dateinamen, sondern auch noch die Angaben 'd' = Document, 'n' = NonDocument, 'p' = Print und 'px' = Print und Exit mitgeben. Daher kann ich mein auf NonDocument installiertes WordStar durch ein Alias ersetzen:

```
wn ws $1 n
```

rufe WordStar auf mit dem angegebenen Dateinamen und gehe in den NonDocument-Modus.

C P / M: Dateiextensions**Dateiextensions**

(Herbert zur Nedden, 2000)

Durch den Einstieg in ZCPR 3.3 wurde die Flut der Datei-Extensions (also der drei Zeichen hinter dem Punkt im Dateinamen) noch größer. Daher ist es vielleicht an der Zeit, sich darüber Gedanken zu machen. Hier findest Du eine Liste gängiger Extensions, die nicht den Anspruch auf Vollständigkeit erhebt - aber andererseits kurz davor sein sollte.

Dabei bitte ich zu beachten, daß Z80-Sources nicht mit .MAC sondern mit .Z80 als Extension versehen wurden. Das liegt daran, daß das der Standard ist - auch wenn Microsofts M80 es anders sieht.

Dateien, die Texte enthalten

.DOC	Dokument (NewWord oder Wordstar)
.DOK	Schreibfehler, soll .DOC lauten
.FOR	Kurzinfo: 'What for'
.HIS	Historie
.HLP	Help-Datei (anzeigbar mit Programm HELP o.ä.)
.IDX	Stichwortverzeichnis zu einem Dokument
.NOT	Notizen (Notes)
.NW	NewWord-Dokument (vorsorglich, eigentlich unüblich wie NewW.)
.TOC	Inhaltsverzeichnis zu einem Dokument
.WS	Wordstar-Dokument

Dateien, die Programm-Sources enthalten

.APL	APL-Source
.ASM	8080-Source
.BAS	BASIC-Source
.C	C-Source
.INC	Turbo-Pascal-Include
.F80	Fortran80-Source
.LIB	Assembler-Definitions-Include (maclib)
.MAC	8080- oder Z80-Source (M80, RMAC)
.MSG	MsgToMac-Source
.PAS	Turbo-Pascal-Source
.PLI	PL/I-Source
.Z80	Z80-Source
.180	HD64180-Source
.280	Z280-Source

Dateien, die Programme oder ähnliches enthalten

.CHN	Turbo-Pascal Chains
.CIM	Nicht-Standard-COM, häufig Typ 3-Programm
.CMD	dBASE-Kommandodateien oder ALIAS.CMD
.COM	Programme (sog. Transiente Programme/Befehle)
.CPR	Menüs für MENU von ZCPR2
.KLX	KLIX-Overlays
.MNU	Menüs für MENU von ZCPR 3.3
.OVL	Overlays von Programmen
.OVR	Overlays von Programmen
.SUB	Submit-Scripts
.VMN	Menüs für VMENU
.ZEX	ZEX-Scripts
.30M	Typ 3-Programme
.40M	Typ 4-Programme (laufen nicht bei uns)
.###	### numerisch: Overlays von Turbo-Pascal-Pprogrammen

C P / M: DateiextensionsDateien, die Dateien enthalten

.ARC	Archiv
.ARK	Archiv
.LBR	Libraries

Dateien, die Daten enthalten

.DBF	dBASE-Datenbank
.NDX	dBASE-Index

Dateien, die Listings enthalten

.LST	Listing
.PRN	Assembler-Listing
.SYM	Symboldatei (Assembler)

Dateien, die Patchanweisungen enthalten

.INS	TheInst-Installationsdatei
.PAT	ThePatch-Patchdatei
.WSP	Patchdatei für Wordstar-Patcher

ZCPR 3.3-Pakete

.ENV	Environment
.FCP	Flow Command Package
.IOP	I/O-Package
.NDR	Named Directories
.RCP	Resident Command Package

Dateien mit Disketteninhalts-Informationen

.CAT	Katalog (meist MAST.CAT)
.DIR	Directory
.###	### numerisch: Disketten-Name/Kurzdoku, falls Dateiname mit - beginnt

Sonstige Dateien

.HEX	Intel-Hex-Datei
.MSR	Microsoft-.REL mit 6-Zeichen-Globals
.REL	REL-Datei, d.h. die relozierbaren Code für einen Linker
.SLR	SLR-.REL mit 16-Zeichen-Globals

Komprimierte Dateien

.?Q?	geSQUEEZete Dateien
.?Y?	LZH-geCRUNCHte Dateien
.?Z?	geCRUNCHte Dateien

Für Anmerkungen sind Olaf Krumnow und ich dankbar!

Hardware: Wie baue ich mir einen Port**Wie baue ich mir einen Port**

(Herbert zur Nedden, 2000)

Kürzlich wurde ich gefragt, wie a) Port 7 funktioniert und wie man b) einen neuen Port lötet. Ich will versuchen, daß mal zu erläutern.

Allgemeines:

Im Gegensatz zu vielen anderen CPUs unterscheidet die Z80 per Hardware zwischen Ports und Speicher.

Bei Speicherzugriffen liefert die Z80 folgende Signale:

\overline{MREQ} = Low, \overline{RD} = Low bei Lesezugriffen, \overline{WR} = Low bei Schreibzugriffen,
A₀-A₁₅ = Adresse, D₀-D₇ = Daten.

Bei normalen Portzugriffen liefert die Z80 folgende Signale:

\overline{IOREQ} = Low, \overline{RD} = Low bei Lesezugriffen, \overline{WR} = Low bei Schreibzugriffen,
A₀-A₇ = Adresse, D₀-D₇ = Daten *und* \overline{MT} = High.

Ist beim Portzugriff jedoch \overline{MT} = Low, so handelt es sich um einen Interrupt-Acknowledge-Zyklus. Dazu findest Du Einiges in den Infos 16-6, 8-11, 9-38, 34-3 und einigen weiteren.

(Kennt eine CPU von Haus aus keine Ports, dann müssen bestimmte Speicherstellen dazu mißbraucht werden, d.h. dort an Stelle der Speicher halt Ports hingelötet werden.)

Zur Hardware:

Da \overline{RD} und \overline{WR} nicht gleichzeitig auf Low liegen, kannst Du Dich i.a. entscheiden, welches dieser beiden Signale Du verwenden willst.

Ein Port besteht aus zwei Komponenten: Dem Adreßdecoder, der prüft, ob der jeweilige Port gemeint ist und dem 'eentlichen' Port, nämlich dem Teil, welches dann die Daten entgegennimmt bzw. liefert.

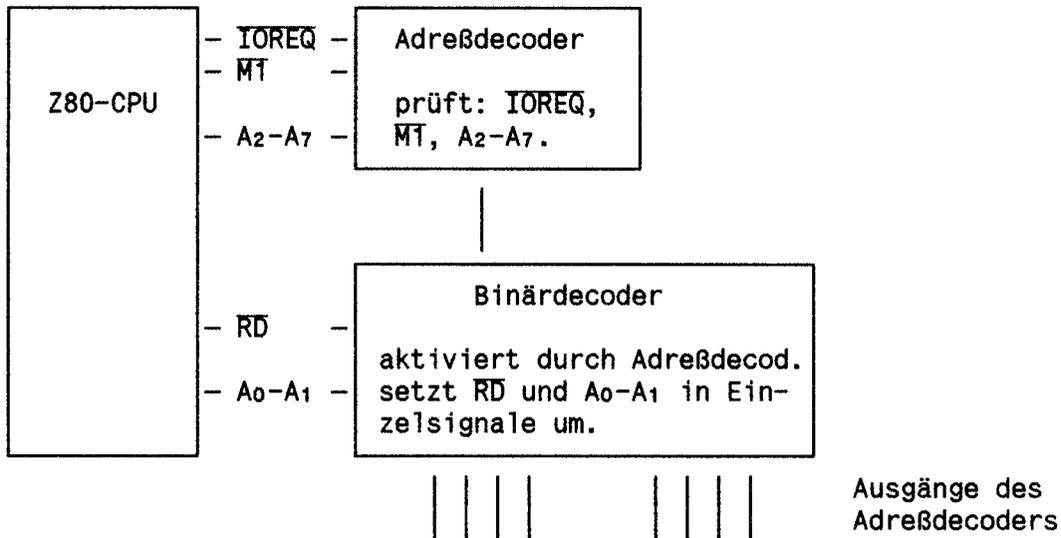
Der Adreßdecoder muß folglich prüfen, ob \overline{IOREQ} = Low, \overline{MT} = High, A₀-A₇ stimmen (d.h. die richtige Adresse anliegt) und je nach Sachlage, ob \overline{WR} oder \overline{RD} den richtigen Pegel aufweisen (Z.B. \overline{WR} = Low und \overline{RD} = High für einen Ausgabeport).

Üblicherweise liefert der Adreßdecoder ein Ausgangs-Signal, welches Low ist, wenn der Port angesprochen wird. Mit diesem Signal wird dann der 'eentlichen' Port dazu überredet, sich um D₀-D₇ zu kümmern.

Will ich gleich mehrere Ports stricken, so kann das sogar mit einem Mehrfach-Adreßdecoder erreichen. Sollen es z.B. 4 Ports sein, nehme ich einen normalen Adreßdecoder, der jedoch die Adreßleitungen A₀-A₁ ignoriert. Damit spricht er auf vier Ports an, da ihm ja A₀-A₁ egal sind. Diese beiden Adreßleitungen gebe ich dann auf einen Binärdecoder-IC, welches durch den Adreßdecoder aktiviert wird und dann je nachdem, was an seinen Eingängen anliegt einen von seinen vier Ausgängen auf Low legt. Damit hat also der Mehrfach-Adreßdecoder (bestehend aus dem Adreßdecoder und dem nachgeschalteten Binärdecoder) vier Ausgänge.

Wenn ich will, kann ich beim o.g. Mehrfach-Adreßdecoder auch \overline{RD} bzw. \overline{WR} statt an den Adreßdecoder an den Binärdecoder legen. Dann hängt der auf Low gelegte Ausgang auch davon ab, ob ich Schreiben oder Lesen will.

H a r d w a r e: Wie baue ich mir einen Port

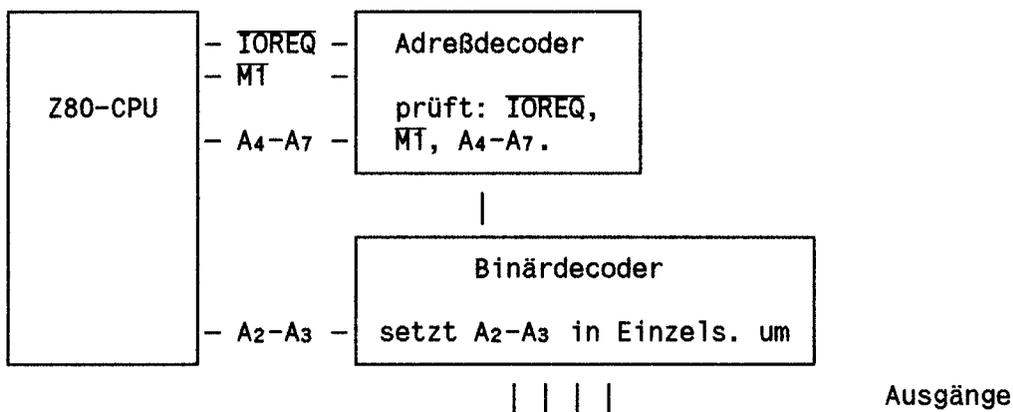


Der Adreßdecoder liefert eine Freigabe für Portzugriffe auf die Adressen, für die A2-A7 den von ihm geprüften Wert haben und für die A0-A1 egal sind. Die Adresse, für die A0-A1 = Low sind ist die sog. Basis-Adresse: die kleinste Adresse, auf die der Adreßdecoder anspringt.

Der Binärdecoder wird vom Adreßdecoder aktiviert. Je nachdem, was er als Signale A0-A1 und RD erhält, legt er dann einen seiner acht Ausgänge auf Low. Der erste entspricht dem Lesezugriff auf den Port, für den A0 = A1 = Low, also der kleinsten Adresse, der nächste ist der Lesezugriff auf den Port mit A0 = High und A1 = Low, usw. bis zum letzten Ausgang für den Schreibzugriff mit A0 = A1 = High.

Ist der Adreßdecoder so gelötet, daß er auf die Adressen E4h-E7h (hex!) reagiert, so entsprechen die acht Ausgänge des Binärdecoders den vier Ports E4h, E5h, E6h, E7h, wobei für jeden dieser Ports getrennte Ausgänge Schreiben und Lesen mit Low-Pegel anziehen. D.h. wenn ich einen dieser Ports schreibend oder lesend zugreife, wird der (und nur der) entsprechende Ausgang des Binärdecoders auf Low gelegt.

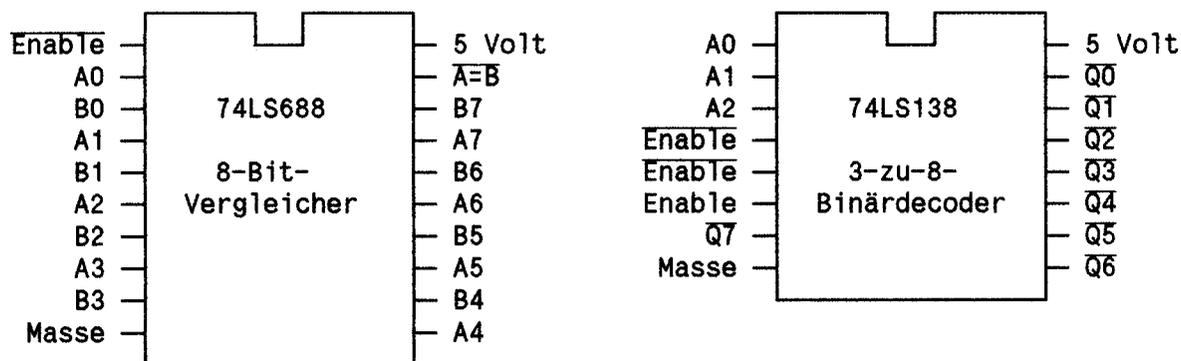
Port-Bausteine wie der Z80-PIO oder Z80-DART belegen übrigens i.a. mehr als eine Adresse. Sie haben dann i.a. Pins, an die A0 und ggf. auch A1 angeschlossen werden sollten. Außerdem kümmern sie sich i.a. auch gleich um die Zugriffsrichtung, d.h. um RD bzw. WR. Dann dürfen diese Adreßleitungen, RD und WR natürlich nicht vom Adreßdecoder verwendet werden - das macht dann der Portbaustein selbst. D.h. er enthält einen Teil des Adreßdecoders in sich. Will ich mehrere (z.B. vier) dieser komplexen Bausteine auf eine Platine stecken, dann kann ich das o.g. Bild so anpassen (falls A0 und A1 an den Portbaustein kommen):



H a r d w a r e: Wie baue ich mir einen Port

Bleibt noch die Frage, mit welchen ICs Du ans Ziel kommen kannst. Zeitgemäß sind natürlich PALs, d.h. solche ICs, in die Logik-Funktionen hineingebrannt werden können. Aber wer hat schon einen PAL-Programmierer. Also nehmen wir handelsübliche ICs:

Als Vergleichler, d.h. für den Adreßdecoder ist der 74LS688 sehr praktisch, da er acht Signale vergleichen kann. Als Binärdecoder dient der 74LS138, der aus einer 3-Bit Information 8 Einzelsignale macht.



Beide ICs liefern nur dann Informationen, wenn die Eingänge $\overline{\text{Enable}}$ auf Low und Enable auf High liegen (soweit jeweils vorhanden) – sonst liegen alle Ausgänge auf High, d.h. sie halten die Klappe. Stimmen die Enables, dann

- liefert der 74LS688 an seinem Ausgang $\overline{\text{A=B}}$ genau dann Low, wenn alle seine Vergleichs-Eingangspaare jeweils gleich sind, d.h. an seinen Eingängen A0=B0 , A1=B1 , ..., A7=B7 ist und
- liefert der 74LS138 an dem $\overline{\text{Qn}}$ -Ausgang ein Low, der den an seinen Eingängen A0-A2 anliegenden Signalen entspricht. Sind zum Beispiel A0-A2 alle Low, ist $\overline{\text{Q0}} = \text{Low}$; sind $\text{A0-A1} = \text{Low}$ und $\text{A2} = \text{High}$, ist $\overline{\text{Q4}} = \text{Low}$. Alle nicht gemeinten $\overline{\text{Qn}}$ sind High.

Mit diesen beiden Bausteinen kann eine Portdekodierung für die Ports 18h bis 1Bh (d.h. Basisadresse 18h) zum Schreiben und Lesen z.B. wie folgt gelötet werden:

$\overline{\text{IOREQ}}$ an A0	vom LS688	Masse an B0	vom LS688
$\overline{\text{MT}}$ an A1	vom LS688	5 Volt an B1	vom LS688
A2-A7 an A2-A7	vom LS688	Masse an B7-B5, B2	vom LS688
		5 Volt an B4-B3	vom LS688

$\overline{\text{RD}}$ an A2	vom LS138	$\overline{\text{A=B}}$ des LS688 an einen $\overline{\text{Enable}}$	vom LS138
A0-A1 an A0-A1	vom LS138	Masse an den anderen $\overline{\text{Enable}}$	vom LS138
		5 Volt an Enable	vom LS138

Alternativ kann natürlich auch $\overline{\text{IOREQ}}$ an den anderen $\overline{\text{Enable}}$ vom LS138 und $\overline{\text{MT}}$ an Enable vom LS138 und A0-A1, B0-B1 des LS688 alle an Masse gelegt werden.

Die Ausgänge des LS138 haben folgende Belegung:

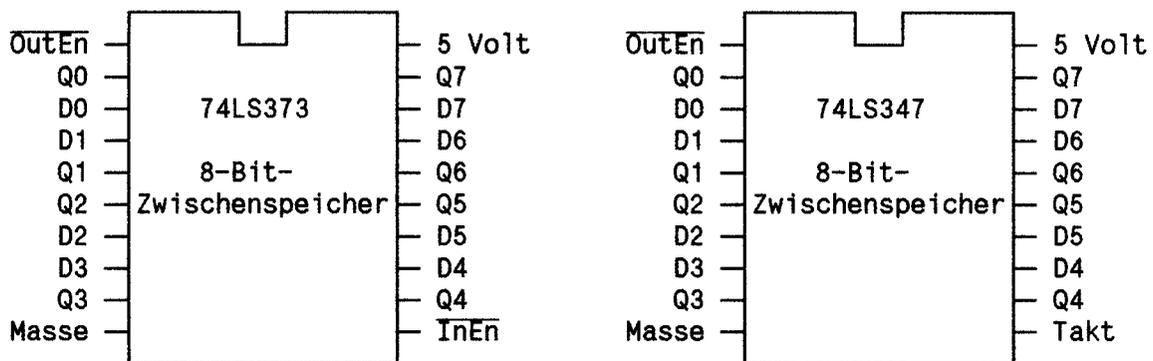
$\overline{\text{Q0}}$: 18h lesen,	$\overline{\text{Q1}}$: 19h lesen,	$\overline{\text{Q2}}$: 1Ah lesen,	$\overline{\text{Q3}}$: 1Bh lesen	und
$\overline{\text{Q4}}$: 18h schreiben,	$\overline{\text{Q5}}$: 19h schreiben,	$\overline{\text{Q6}}$: 1Ah schreiben,	$\overline{\text{Q7}}$: 1Bh schreiben	

Auf Erweiterungsplatinen werden für o.g. Beispiel allerdings B2-B7 des LS688 nicht an 5 Volt bzw. Masse gelötet, sondern über einen 4,7 kOhm-Widerstand an 5 Volt gelegt und mit Jumpfern versehen, die, falls gesetzt den jeweiligen Pin an Masse legen. Damit kann die Basis-Adresse ohne Löten individuell gewählt werden.

Hardware: Wie baue ich mir einen Port

Bis jetzt ging es eigentlich nur um den Adreßdecoder. Doch, wohin oder woher mit den Daten. Dafür fehlt der eigentliche Portbaustein. Im einfachsten Fall ist das einfach ein IC, welches durch Anlegen eines Low-Pegels (der vom Adreßdecoder geliefert wird) Daten vom Datenbus nach außen weiterreicht oder auch von außen auf den Datenbus legt. Üblicherweise wird man dafür aber gleich ein IC nehmen, welches die Daten auch noch speichern kann, damit Daten, die ich z.B. auf einen Port ausgabe auch dort verweilen.

Für Port 7 hat Memotech je einen 74LS373 und einen 74LS374 verwendet:



Beide der o.g. Zwischenspeicher übernehmen unter bestimmten Bedingungen die 8-Bit-Information von ihren Eingängen D0-D7 in ihren internen Speicher, wo sie so lange bleibt, bis sie auf gleichem Wege durch eine neue überschrieben oder der Strom abgeschaltet wird.

Ist der Eingang InEn des 74LS373 = Low, dann übernimmt der 74LS373 die Informationen von seinen Eingängen D0-D7 in seinen internen Speicher. D.h. je nach dem Pegel, der an diesem Pin anliegt, werden die Daten übernommen oder nicht. Der 74LS374 hingegen übernimmt die Daten von seinen Eingängen D0-D7 wenn er an seinem Takt-Eingang eine steigende Flanke erhält. Der Adreßdecoder liefert ja einen Low-Impuls beim Portzugriff (Impuls, da der Portzugriff der CPU nicht lange dauert), der also auch eine steigende Flanke hat!

Ist der Pin OutEn des 74LS373 oder 74LS374 = Low, liegt der Inhalt des internen Speichers an den Ausgängen Q0-Q7 an; anderenfalls sind die Ausgänge offen.

Wenn Du Dir nun mal das MTX-BASIC-Handbuch schnappst, wirst Du hinten den Schaltplan des MTX finden - und dort auch den Port 7. Die Signale IN7 und OUT7 kommen vom Adreßdecoder. OutEn wird dort als OC, InEn als G und der Takt durch ein kleines Dreieck gekennzeichnet.

Bustreiber:

Eine Z80CPU hat auch nur begrenzte Kräfte. Daher darfst Du nicht beliebig viele Ports einfach an den Datenbus D0-D7 der CPU anschließen. Memotech hat dafür einen sog. Bustreiber verwendet. Das ist einfach ein IC (üblicherweise ein 74LS245), der eine 8-Bit-Information (also 8 Signale) von seiner einen Seite an die andere weitergibt. Dabei hat so ein IC i.a. einen Eingang zur Richtungssteuerung, da es die Daten sowohl in die eine als auch andere Richtung weiterleiten kann. Hänge ich z.B. an den Datenbus der CPU einen Bustreiber und meine Ports auf dessen andere Seite, wird die CPU nur mit einem IC belastet. Bustreiber hingegen können meist 10-20 ICs ab.

Wenn Du nur einen Port löten willst, mach das einfach auf der Hauptplatine und nimm D0-D7 nicht direkt von der Z80 sondern von den Datenbusanschlüssen einer der ICs von Port 7 ab.

Software: RAM 6.1**RAM 6.1**

(Herbert zur Nedden, 2000)

RAM 6.1 ist ein Update für RAM 6.0. Zweck dieses Updates ist es, ein paar kleine Macken und Unzulänglichkeiten von RAM 6.0 zu beseitigen. Die mit den Updates gemachten Korrekturen von RAM 6.0 -> RAM 6.0a -> RAM 6.0b sind mit enthalten.

Hier die Neuerungen im Einzelnen gegenüber der Dokumentation von RAM 6.0:

RAM61.COM

1. Die Sekunden der Uhr werden immer per Software erhöht und dabei die anderen Uhren-Daten nicht verändert. Das geht deutlich schneller, als wenn immer die ganze Uhr durchgenudelt wird. Daher muß die richtige CPU-Taktfrequenz auch bei Verwendung der Hardware-Uhr installiert werden.
2. Im KLIICK-Bildschirm-Menü kann die Uhr gestellt werden. Sobald <RET> gedrückt wird, wird die neue Zeit eingestellt. Durch <ESC> kommt man aus dem Menü heraus, ohne die Uhr zu verstellen. Der Cursor kann nur mit den beiden entsprechenden Pfeilen nach rechts und links bewegt werden.
3. Im Environment-Descriptor ist nun für den Bildschirm die Seitenlänge um zwei größer als die Anzahl der Textzeilen. Das entspricht den ZCPR 3-Konventionen.
4. Im Environment-Descriptor ist nun für den ersten Drucker Seitenlänge 72 und die Anzahl der Textzeilen 66.
5. Der Einsprung Config pflegt die DriveMask im Environment-Descriptor, es sind also nur die Bits der configurierten Laufwerke gesetzt sind. Damit können ZCPR 3.3-Utilities abfragen, welche Laufwerke wirklich da sind.
6. KLIICK rettet nicht mehr den BDOS-Loginvektor, damit aus dem KLIICK ein Diskreset richtig funktioniert.
7. Zwei Utility-Funktionen kennt RAM 6.0 schon, die aber nicht dokumentiert wurden:
 - 83: DErrOff: Für einen Disk-Zugriff das Blinkfenster unterdrücken und den Fehlercode in A zurückliefern.
 - 84: UtilRdSc: Einen Sektor wie mit RdSc einlesen, jedoch ohne Blinkfenster. Der Fehlercode kommt in A zurück.
8. Drei neue Utility-Funktionen gibt's:
 - 35: HexOutWord: HL 4-stellig Hex ausgeben
 - 36: HexOutByte: L 2-stellig Hex ausgeben
 - 85: RetryCount: Retry-Counter für Disk-Zugriffe auf den Wert aus A setzen. Normalerweise ist dieser Wert 10; zum Testen kann 1 sinnvoll sein. (0 ist 256!)
9. Nach dem Booten von RAM 6.1 steht der Buchstabe des Bootlaufwerks im ZCPR 3.3-Register 0. Dieses kann damit in Boot-Aliasen o.ä. abgefragt werden.

INST61.CHN

Das Installationsprogramm wurde fehlerbereinigt und erhielt zusätzlich die Auswahl Import um die Installationswerte von einem RAM 6.x in das neue RAM 6.1 zu übernehmen, damit Du nicht alles von vorne installieren mußst.

P2DOS und ZCPR

1. Die BDOS-Funktion zum Umbenennen von Dateien läßt die Dateiattribute unverändert (löscht jedoch das Archiv-Bit) und akzeptiert auch das ? im FCB als Wildcard. Dies wird vom ZCPR-Befehl unterstützt.
2. Die BDOS-Funktion zum Ändern der Dateiattribute läßt den Dateinamen unverändert und akzeptiert auch das ? im FCB als Wildcard.
3. Die BDOS-Funktion zum Erzeugen einer neuen Datei löscht alle Dateiattribute.
4. BDOS prüft nun nicht nur ^S sondern ^P so häufig es geht, so daß der Drucker auch bei Consol-Ausgaben aktiviert werden kann.

CHAINDU

ist auf die EndAdresse 2800H kompiliert. Damit kann dieses Teil sicherlich unter ZEX laufen. Allerdings nützt das nichts, wenn das geladene .CHN zu groß ist.

Software: RAM 6.1**Utilities**

Neu hinzugekommen ist SYSCOMP6 zum Vergleichen der Systemspuren zweier Laufwerke. CFG6 stellt die Config-Codes der Laufwerke A:-P: in ZCPR 3.3-Register. Ausserdem setzen CFG6, FKEY6, FORM6, ERADIR6, RCHECK6, SYSCOMP6, SYSCOPY6 und WRSYS6 bei Auftreten eines Fehlers das ZCPR 3.3-Fehlerflag.

Bereinigte Fehler, die noch in RAM 6.0b bestanden:**RAM61**

1. Der Doppelstrich-Cursor ist weg. Die 80-Zeichen-Karte von Memotech ist ja in Sachen Portdekodierung eher faul gearbeitet - was aber reicht. Bei der Ausrüstung auf HGR (Grafik) wurde die Portdekodierung etwas verfeinert, um zwei neue Ports zu schaffen. Bei Memotech-Original sind Ports 38 und 3A identisch, bei HGR nicht mehr. Beim Wiederherstellen des Bildschirms wird von RAM 6.0 Port 3A als HGR-Port angesprochen - egal ob es eine HGR-Karte ist oder nicht. Das ist mit RAM 6.1 behoben - wenn Du es korrekt installierst.
2. Die Utility-Funktionen, die in DE einen Pointer auf einen Blocknamen erwarten, holen sich diesen Namen von der Bank, von dem aus Utility aufgerufen wird.
3. Beim Verlassen des KLICK wurden die Register BC und DE nicht richtig restauriert! Es ist schon erstaunlich, wie wenig Schaden das anrichtete!
4. Die Bildschirm-Sequenz Esc Ä B zum unveränderten Ausgeben eines Zeichens auf den Bildschirm funktioniert.
5. INITDIR6 hat den Cache überlistet! Es hat das Directory geblockt geschrieben, und nicht wie bei Directory-Schreibzugriffen so, daß der Sektor sofort auf Disk wandert. Daher stand der richtige Inhalt des letzten Directory-Sektors im Sektorpuffer des Disktreibers, aber nicht auf Disk. Danach liest INITDIR6 noch einmal den ersten Directory-Sektor, damit der Disk-Treiber den Sektor aus seinem Puffer wegschreibt, da er den Puffer für den neuen benötigt. Nun lieferte der Cache diesen Sektor und der letzte Directory-Sektor blieb im Puffer. Das ist auch behoben. Wenn bei einem Diskzugriff, den der Cache befriedigen könnte im Sektor-Puffer des Disktreibers noch etwas steht, was auf Disk gehört, so schweigt der Cache stille, damit der Sektor auch auf Disk kommt. Das bremst die Zugriffe nicht, da üblicherweise Directory-Zugriffe eh nur passieren, wenn der phys. Sektor auf der Disk gewechselt wird.

INST61

Bei der Option Lage des Bildes auf der Glotze werden nun alle Zeilen angezeigt - die letzte scrollt nicht mehr aus dem Bild.

FKEY6

Bei Laden von Funktionstasteblegungen wurde der F-Tasten-Zeiger nicht initialisiert. Daher wurden die Zeichen in den Tastaturpuffer gestellt, auf die dieser zufällig zeigte. FKEY6 behebt diesen Mangel nun.

SUB34

Läuft nun auch 2x hintereinander ohne ^C.

Noch eine bekannte Macke gibt's:

Wenn Du SC innerhalb eines ZEX-Scripts aufrufst, darfst Du im KLICK kein Programm aufrufen, welches auf BDOS zugreift - also auch nicht DiJey! Peter Würfel hat es entdeckt. Vorläufig solltest Du SC daher nicht aus einem ZEX-Script heraus aufrufen.

Copyright:

RAM 6.1 ist nicht Public-Domain, darf also nicht weitergegeben werden - auch nicht an registrierte Besitzer von RAM 6.0!

Software: Loaddir**Loaddir**

(Claudio Romanazzi, 3070)

'Auch das noch', wird man sagen, 'ein Utility, schon wieder!' Aber wie üblich steckt der Nutzen im Detail. Wer hätte nicht schon mal über die 'Unaussprechlichen', will heißen die aus diversen Abkürzungen, deren Sinn natürlich nur der Autor kennt, zusammengesetzten Titel von Dateien geschimpft. Bis man sowas auswendig gelernt hat, speziell wenn man das Werk nur selten braucht, dauert es seine Zeit und ist deswegen lästig.

Loaddir ist ein Programm, dessen 'Outfit' ich vom Datei-Manager des Turbo 5.5 von Domesdos abgeschaut habe. Bei Aufruf erscheint ein Fenster, in dem die Dateien des aktuellen Laufwerks/Users stehen. Man kann blättern, sowohl durch die verschiedenen Laufwerke, als auch durch die verschiedenen User. Dabei werden natürlich nur die vorhandenen Laufwerke abgesucht. Nicht vorhandene werden übersprungen. Die Anzeige erfolgt seitenweise. Hat man sich für ein Laufwerk/User entschieden, wählt man auf dem Bildschirm im Fenster eine Datei mittels Ansteuerung durch den Cursor aus. Es genügt dann ein Druck auf die Return-Taste und die gewählte Datei wird ausgeführt.

Das hört sich einfach an, ist aber etwas komplizierter. Einfach ist es bei Com-Dateien. Ein Com wird angeklickt und wird ausgeführt (übrigens egal auf welchem Laufwerk/User die Datei gefunden wurde, Loaddir kehrt immer zur aufrufenden Konfiguration zurück). Was ist aber mit den anderen. Hier fittet Loaddir nicely in (=paßt gut zu) unseren Kommando-Prozessoren, speziell zu XRUN. Zu diesem Behufe kann Loaddir mit speziellen Kommandos so eingerichtet werden, daß nur Coms, Docs, Lbrs, Macs und Pass angezeigt werden. Mit dieser Eigenschaft sind die Aufruf-Möglichkeiten vervielfacht.

Nebenbei werden noch verschiedene Diskettenparameter angezeigt, wie Laufwerk, Anzahl der Dateien auf dieser Diskette, freier Diskettenplatz, Benutzernummer, Anzahl der Dateien dieser Benutzernummer und Dateigröße.

Jetzt werden natürlich alle angerannt kommen und fragen, was, wann, wo und last not least wieviel. Doch all das kann ich nicht beantworten, denn Loaddir befindet sich in den Tiefen von Herberts Sammlung für die nächste PD-Diskette (?).

Anm.d.HzN: Loaddir macht in der Tat einen sehr guten Eindruck und wird KLICK.012 anreichern - wobei ich es persönlich etwas gewöhnungsbedürftig finde, daß in der Anzeige von Loaddir Laufwerk und User nicht beieinander stehen und vor allem statt User der Text 'Benutzer' steht. Das ist zwar richtig übersetzt, aber für mich ist User eine Nummer, die etwas mit dem Directory zu tun hat und Benutzer der, der die Tastatur maltretiert.

Ich hoffe, daß ich Claudio noch dazu überreden kann, die Liste der gültigen Laufwerke nicht aus dem Bootsektor zu holen, sondern entweder via BIOS-Drive-select oder Config-Einsprung von RAM 6.x zu fragen, ob das Laufwerk gültig ist. (Er war sofort bereit, diese Änderung zu machen!)

Meine ersten Versuche lieferten natürlich nur dümmliche Fehlermeldungen von XRUN - klar, wenn man sich nicht die Mühe nimmt, die Dokumentation genau zu lesen. Wähle ich z.B. LOADDIRD.DOC auf C0: aus, bin jedoch auf G0:, so stellt LOADDIR folgendes in die Kommandozeile: C0:;LOADDIR;G0: (genau so, wie Claudio es ja auch in der Dokumentation beschrieben hat!)

Und da mein XRUN nicht auf .DOC reagiert, kann auch nicht passieren. Die Extension der ge<RET>teten Datei sollte XRUN schon verstehen.

Als Ergänzung wäre es natürlich nett, wenn Loaddir gar folgendes in die Kommandozeile stellen würde: C0:;LOADDIR;G0:;LOADDIR damit Loaddir nach Ausführung des gewünschten Befehls gleich wieder anläuft. Z.B. könnte ja mit <RET> das Kommando ohne, mit <SHIFT-RET> das mit Loaddir genommen werden.

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