

MTX *User-Club Deutschland*

Info 16
20.09.1986

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

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

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

Verpflichtungen: Einsendung unseres Anmeldeformulars.

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

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

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

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

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

Kontaktadressen: (nach PLZ geordnet)

Herbert Herberg	Frank Bueschler	Christian Löhrmann	Detlef Harms	Thomas Pflaum
Sonnenau 2	Am Ochsenzoll 3	Grevenbleck 24	Feldstr. 26	Leipziger Platz 1
2000 Hamburg 76	2000 Norderstedt	3005 Hemmingen 1	4902 Bad Salzufflen	8500 Nürnberg 20
(040) 200 87 04	(040) 527 75 81	(0511) 41 78 77	(05222) 84 0 82	(0911) 51 35 21

Telefonzeiten

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

Inhaltsverzeichnis**C L U B:**

Dies und Das von Herbert Herberg	Seite 1
Korrektur & Nachtrag	Seite 2
Wer tut Was	Seite 3
Kleinanzeigen	Seite 3
Fragen	Seite 4
Neue Software	Seite 4

C P / M:

SUB	Seite 5
Dateigröße bei verschiedenen Formaten	Seite 17

C P U:

Interrupts der Z80	Seite 6
--------------------	---------

B I O S:

Carreras: Korrektur zu Info 15	Seite 10
Noch'n Filter	Seite 11

N e w W o r d s:

Dezimaltabulator	Seite 17
------------------	----------

H A R D W A R E:

Druckerumschaltung	Seite 18
Carrera-Interface	Seite 19
Eine Anzeige	Seite 21
32k-Karte	Seite 21
A/D-Wandler mit Multiplexer	Seite 23
ECB-Bus	Seite 24
ECB-Ports	Seite 26
EDICTA-Grafikkarte	Seite 26
RAM-Floppies	Seite 26

B A S I C:

Korrektur der VAL-Routine aus Info 15	Seite 27
Fractals	Seite 28
Das Spiel Light Riders	Seite 30

S u p e r C a l c:

Kurs Teil 2 von Wolfgang Gieger	Seite 34
---------------------------------	----------

L e s e r b r i e f e:

Seite 38

T i p s:

Seite 41

Preis für dieses Info: DM 9,20

C L U B: Dies und Das von Herbert Herberg

Liebe MTX-lerinnen und MTX-ler!

Der **ECB-Bus** läuft endlich tatsächlich besser. Dazu könnt Ihr jetzt selbstverständlich die Lösung/Ursachen weiter hinten in diesem Info lesen!

Ich bin Eigentümer einer **SDX-Anlage** als Zweitgerät. Sowie ich das Kistchen habe werde ich XBASIC von Andreas Viebke darauf testen, und mich dann an 512kB dafür machen --> RAM 3.4 ff.

Bestellaktion: jetzt endlich ist es soweit. Den entsprechenden Zettel findet Ihr im Info. Bislang habe ich die Erfahrung gemacht, daß ein zwei-Blatt-Bestellzettel nicht klappt (häufig fehlte ein Zettel, zum Teil halt der mit dem Absender). Daher dieses Mal nur zum Ankreuzen! Falls irgendjemand jetzt ein Teil noch gebraucht abgeben möchte, bitte auf der unteren Hälfte der Rückseite vermerken! Genauso könnt Ihr natürlich dort angeben, ob ihr ggf. an gebrauchtem (falls vorhanden) zu entsprechend niedrigerem Preis interessiert seid. Aber bitte nicht auf die Vorderseite schreiben, das macht mir das Durchsehen, ... komplizierter. Bestellungen (außer den bis heute eingegangenen und bestätigten telefonischen) nur auf dem Bestellzettel!! Die Bezahlung kann durch Überweisung (sofort) oder beigefügtem Verrechnungsscheck an mich erfolgen. **WICHTIG:** Diesmal ist die Frist wie angekündigt sehr kurz!

Abonnement von CLUB.nnn - Public-Domain Disketten: Warum nicht, einen Abonnenten habe ich schon. Problematisch wird es nur bei Bitten 'Alle CP/M-PDs'. Wann ist eine CP/M, wann was? Willst Du also ein selektives Abonnement, bitte deutlich formulieren was bzw. was nicht!

Anbei findet Ihr eine 'VisitenKarte' (eigentlich ein Blättchen), mit meiner Olympia-Carrera geschrieben, auf dem u.a. meine Sprechzeiten stehen. Vielleicht nützt es etwas, diesen Wisch in Nähe des MTX unterzubringen, damit Ihr wißt wann Ihr mich wo erreichen könnt. Aber - es kommt hin und wieder vor, daß auch mal verhindert bin, dann bitte den nächsten Termin versuchen.

Thomas Mäurer schlägt vor, die Kosten für Datenträger + Porto + Verpackung auf DM 6.- zu senken, da es ja billige Cassetten für DM 1.- zu kaufen gibt. Im Prinzip gebe ich Ihm bei dieser Rechnung recht, nur verwende ich nicht diese billigen Cassetten Dank der Erfahrung, daß dann ein anderer Recorder Probleme haben kann, und ich keinem zumuten möchte sich damit zu ärgern, bzw. die Programme ein zweites Mal zu überspielen. Außerdem verschickt z.B. Christian eine ganze Menge von Programmen als Public-Domain BASIC-?.001 auf einer Cassette, und daß erfordert längere solche. Im Einzelfall könnt Ihr ja mit dem Versender verhandeln. Bei Disketten verwende ich keine billigen, sondern gute Double Side, Double Density - auch Dank entsprechender Erfahrungen!

Gerade rief Hans Gras aus Holland an, und teilte mir mit, daß er einige Patches zu der neuesten BASIC-Version von Andreas Viebke hat. Diese werden Euch nicht vorenthalten! Kommen in Info 17.

Ich erhielt wieder den Vorschlag, das Info lochen zu lassen. Das habe ich einmal in der Druckerei angesprochen - sogar schon einmal bestellt. Seinerzeit scheiterte das an der defekten Bohrmaschine, aber jeden Fall verögert das die Auslieferung und hebt den Preis an! Daher müßt Ihr auch weiterhin lochen.

Imp an Hacky

Eure

Herbert Herberg

C L U B: Korrektur & Nachtrag**Korrektur & Nachtrag**

Info 15, Seite 12: Bios-Filter für Carrera von Herbert Herberg und

Info 15, Seite 9: VAL für (FDX)BASIC von Andreas Viebke

weiter hinten die Korrekturen.

Info 15, Seite 31: Bdos Err On SDX:

ACHTUNG das ist die SDX, also die kleine Kiste! Die große schwatte Schachtel mit dem 60-pol Flachbandkabel ist die FDX

CLUB.003 - Public-Domain des Club's:

1. WINDOW.PAS Zeile 5: (**I B:KASTEN2.INC*)
Zeile 6: (**I B:WINDOW.INC*)
2. PLIST.PAS Hinter dem ersten ClrScr im Hauptprogramm einfügen:
Zeile 70: PL:=72;
3. HARDBO.PAS Dies ist ein Unterprogramm, also ohne Hauptprogramm!

An alle Besitzer von MONI 2.00a

Leider hat sich ein kleiner Fehler in die Assemblereingabe von MONI in der Version 2.00a (21.04.86) eingeschlichen, der bewirkt, daß die Befehle BIT, SET, RES für die Indexregister nicht richtig assembliert werden. Dieser Fehler läßt sich glücklicherweise leicht beheben. Im folgenden werden die Korrekturabläufe (sie sind mit MONI2 selbst durchführbar) für jede der drei Dateien (MONI2, MONI2A, MONI2B) beschrieben. Deine Eingaben sind unterstrichen, der / bedeutet <RET>.

Für alle: Die entsprechende Version mit MONI2 laden. Dann den Modus 1 anwählen.

Für MONI2: Kommando>A Von>FDD/
OFDD CALL B922/
OFEO LD A,OCB/
OFE2 CALL B82E/
OFE5 <BRK>

Für MONI2A: Kommando>A Von>FDD/
OFDD CALL A922/
OFEO LD A,OCB/
OFE2 CALL AB2E/
OFE5 <BRK>

Für MONI2B: Kommando>A Von>F80/
OF80 CALL 7223/
OF83 LD A,OCB/
OF85 CALL 712F/
OF88 <BRK>

Für alle: Kommando>Q
Kommando>W Von>0100 Bis>* Name>*

Jetzt funktioniert die Eingabe korrekt. Übrigens, wer Schwierigkeiten hat, das Inhaltsverzeichnis von MONI2.DOC auszudrucken, muß beim Drucken mit Seite 0 anfangen (Starting Page? 0/).

Olaf Krumnow

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

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

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

Kleinanzeigen

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

- vermittelt jederzeit gebrauchte/neue Geräte und Teile der selben. Außerdem weiß ich i.a. was es wo am billigsten gibt.
- Ich habe Apple-Communication-Software: Software für Rechnerkopplung Computer mit einem Apple. Das sind zwei Disketten (1x MTX, 1x Apple), die ich ggf. verleihe, da ich die Apple nicht kopieren kann.
- Ich habe FDX und MTX mit/ohne Monitor, Drucker, ... zu verkaufen. Preis ist Verhandlungssache! **Ich habe diverse FDX-Stationen!!!** Ich habe auch verschiedene Laufwerke und MTX 512 zu verkaufen (in Kommission).
Was ich weitergebe ist überprüft, FDX bootet dann einwandfrei!
- Ich verkaufe ein 8"-Laufwerk incl. Gehäuse, Trafo, Handbuch (3cm Papier A4): DM 75.-

V E R K A U F

Detlef Woineck, Heeperstr. 1, 4902 Bad Salzufen, 05208 - 7538
Altinfos 1 - 15 für DM 125.-, Druckerkabel Memotech DM 35.-.

Holger Reiss, 040 - 6071175

MTX 512 Preis VHS, Info 1 - 12 für 50% NP.

Jürgen Schletter, Stollengäßle 10, 7265 Neubulach, 07053 - 7401

FDX, 2. Lw, RS 232 für DM 1600.- (VB)

Wolfgang Wortmann, Auf dem Unterfeld 34, 6450 Hanau, 06181 - 26144

MTX 512, FDX, 2. Lw, DMX 80, TP 200 DM 2000.-, Hisoft-Pascal-Karte DM 100.- und TURBO 3.0 incl. Utilities DM 200.-

Götz Neumann, Am Steinicht 22, 8630 Coburg, 09561 - 66877

DMX 80 für DM 350.-

Uwe Freikamp, Wilhelmstr. 48, 4100 Duisburg 14, 02135 - 47520 (ab 20h)

MTX 500 + ca. 80 Programme + Dokumentation für DM 400.-

C L U B: Fragen / Neue Software

Fragen

Wolfgang Gieger, J.-F.-Kennedy-Allee 35, 3180 Wolfsburg entdeckte folgendes in SuperCalc:

Um Formatbögen von einem Fremden z.B. bedienen zu lassen, kann man das ja in einer automatischen Datei ablaufen lassen und entweder mit SC ... oder mit der Eingabe /X und Dateiname ablaufen lassen (mit der Dateiextension .XQT). Soweit so gut, aber um den (auto) Ablauf im Programm zu unterbrechen ist die Eingabe des Zeichens & notwendig. Um den Ablauf fortzusetzen ist es nötig nochmals & einzugeben. Alles funktioniert - außer dem Start/Stop-Befehl &.

Software

```

PPPP  RRRR   OOO   GGG   RRRR   A   M   M   M   M   EEEEE
P  P  R  R  O  O  G     R  R   A  A   MM  MM  MM  MM  E
PPPP  RRRR   O  O  G  GG  RRRR   AAAAA  M  M  M  M  M  M  EEEE
P     R  R   O  O  G  G  R  R   A  A   M   M   M   M   E
P     R  R   OOO   GGG   R  R   A  A   M   M   M   M   EEEEE
    
```

by dieter ritter charlottenstr.8 8800 ansbach

FRACTL: Ein wirklich komfortables Fractl-Rechenprogramm (30k), das die faszinierenden Bilder mittels Assembler berechnet und daher unschlagbar schnell ist. Über ein Menü können bequem Bildaufbereitung, Bildverwaltung, neue Berechnung und Hardcopy verwaltet werden. Es benötigt 310k auf Disk und verwaltet darauf fünf Bilder, die jeweils in vielen Effektstufen ausgegeben werden können. Vom "Mutterfractl" aus werden über bestimmbare Quadrate die gewünschten Ausschnitte gezeichnet. Das Programmpaket enthält: FRACTL/ein Installationsprogramm/fünf Fractls/und als Bonbon eine mathematische Einführung in die Fractlsystematik. DM 25.-

DEFENDER: Space Shuttle startet von einer Startrampe und wehrt Angreifer ab. Durch die Verwendung von Assembler, 32 Sprites und Sound braucht dieses Spiel keinen Vergleich mit professionellen Spielen zu scheuen. Den Hintergrund bilden eine bewegte Bergkette, ein vorbeiziehender Sternenhimmel sowie Statusanzeigen. Die Raumschiffe greifen aktiv an; Richtungsumkehr, Schutzschild etc. lassen dieses Spiel spannend werden. (20k) DM 15.-

FOURIERENTWICKLUNG: Wen's interessiert, wie eine Rechteckspannung durch Überlagerung von Sinuskurven approximiert wird. Mit graphischem Gag! DM 5.-

CP/M: SUB

(Anton Reiser, 8977)

Seit kurzem stolzer Besitzer einer RAM-Disk, ärgerte mich, daß beim Kopieren der Programme aus einer SUB-Datei jedesmal PIP neu geladen wurde, obwohl PIP nach dem ersten Aufruf bereits im Speicher steht. Das kostet Zeit, und beim System hochfahren kann es nicht schnell genug gehen. Abhilfe ist auf zwei Arten möglich:

Methode 1

Wir haben von Memotech das spezielle Dienstprogramm ENTER, welches in SUB-Dateien eine Konsoleingabe simuliert. Also als erste Zeile in der SUB-Datei ENTER mit der gewünschten Eingabe schreiben. Nach jedem Befehl ein großes 'ö' (gleicher ASCII-Code wie 'Backslash'), welches ein Carriage Return simuliert. Beim letzten Befehl wird es automatisch gesetzt, beim Aufruf von PIP als Programm muß aber ein ^M kommen (entspricht auch CR) um aus dem Programm wieder ins Betriebssystem zu kommen.

Beispiel:

```
enter f:=a:nw.comöf:=a:nw.ovröf:=a:nwmsgs.ovröf:=a:config.com^M
ram 7 0
config f:50
syscopy f:
pip
f:
b:coldboot f:
```

Methode 2

Wir erzeugen mit Hilfe des resistenten Befehls SAVE einen Directory-Eintrag mit der Programmlänge Null:

```
SAVE 0 CONT.COM
```

Nach dem Aufruf des Programms CONT wird CP/M dieses laden versuchen und zur Speicheradresse 100 hex springen. Da CONT.COM keinen Inhalt hat, wird das noch im Speicher stehende Programm erneut gestartet. (Quelle: Alan R. Miller, Programmieren mit CP/M)

Mit CONT können auch Parameter übergeben werden, was wir in der SUB-Datei nutzen.

Beispiel:

```
ram 7 0
config f:50
syscopy f:
pip f:=a:nw.com
cont f:=a:nw.ovr
cont f:=a:nwmsgs.ovr
cont f:=a:config.com
f:
b:coldboot f:
```

Bemerkungen zu CONT.COM (Herbert Herberg, 2000)

Die Verwendung eines solchen leeren .COM (s.o. Methode 2) ist nicht ganz ohne Tücken:

1. Viele Programme wie PIP lassen sich so problemlos wieder starten.
2. Es gibt jedoch auch Programme, zu denen WUSCH/FAST bzw. MDC/SDC gehören, die sich so nicht aufrufen lassen.
3. REZILOG startet sich durch Aufruf von CONT nicht neu, sondern man landet in dem Zustand, in dem es verlassen wurde.
4. Bei DDT und MONI stehen nicht diese .COMs an 100 hex, sondern dort kommt das geladene Programm an (oder wo die Daten sind, wie sonst auch debuggen. Mit CONT rufe ich also das geladene Programm auf!

CPU: Interrupts der Z80**Z80 - Interrupts - Eine Einführung**

(Herbert Herberg, 2000)

Interrupt = englisch für Unterbrechung!

Es gibt zwei Arten von Interrupts, die an verschiedenen Pins der CPU ausgelöst werden: maskierbare (d.h. solche, die ignorierbar sind) und den nicht-maskierbaren. Wird ein Interrupt am Pin 17 der CPU ausgelöst, dann reagiert die CPU immer auf dieses Interrupt (es sei denn die CPU ist ausgeschaltet). Ob die CPU auf einen Interrupt am Pin 16 reagiert hängt davon ab, ob diese Interrupts mit EI (Enable Interrupt) zugelassen werden, oder mit DI (Disable Interrupt) unterdrückt werden. Nachdem ein Interrupt durch ein Low-Pegel am entsprechenden Pin der CPU ausgelöst wird, führt die CPU ggf. die u.g. Aktion nach Beendigung des Befehls, der gerade ausgeführt wird, aus.

Wozu Interrupts ? Nun da gibt es zwei Anwendungen:

1. irgendetwas soll in regelmäßigen Zeitabständen im Hintergrund ablaufen (Uhr,... oder gar Bernd Peusings Tastaturabfrage mit Puffer)
2. die Ursache kommt unregelmäßig und selten, und eine Abfrageroutine würde das System unnötig belasten (z.B. Power Fail).

Was macht die Z80 wenn ein Interrupt ausgelöst wird?

Das steht je nach Interrupt unten, aber i.a. wird ein für den Interrupt bereitgestelltes Unterprogramm, die Interrupt-Routine, aufgerufen. Eine solche Interrupt-Routine sollte die Register, die sie verändert sichern, damit die CPU nach Beendigung der Interrupt-Routine mit ihrer ursprünglichen Arbeit fortfahren kann.

Im folgenden ist mit dem Wort Befehl ein Z80-Assemblerbefehl gemeint, und die Zahlen sind alle in Hex (daher dahinter ein H).

Nun zu den verschiedenen Arten von Interrupts.

1. NMI = Nicht Maskierbarer Interrupt

Es passiert:

1. Die CPU merkt sich, ob die maskierbaren Interrupts zugelassen sind, oder unterdrückt werden sollen.
2. Sie unterdrückt vorerst maskierbare Interrupts (gibt sich also den Befehl DI)
3. Sie macht einen CALL 0066H.

Die an 0066H stehende Interrupt-Routine kann auf drei Weisen beendet werden:

- Durch den Befehl RETN. Dieser bewirkt ein RETURN, läßt also die CPU dort weitermachen, wo sie vor dem Interrupt war. Aber zusätzlich wird der oben gemerkte Zustand 'Interrupts unterdrücken oder nicht' wieder hergestellt.
- Mit RET. Halt nur ein RETURN, Interrupts werden dann weiter unterdrückt.
- Anders: Vielleicht soll mit der laufenden Arbeit nicht weitergemacht werden, sondern z.B. das System gebootet werden.

CPU: Interrupts der Z802. INT = Interrupt (maskierbar)

Maskierbare Interrupts, die durch ein Low am Pin 16 der CPU ausgelöst werden, können per Software zugelassen bzw. ignoriert werden. Der Befehl EI bewirkt, daß die CPU auf solche Interrupts reagiert, der Befehl DI deaktiviert diese Interrupts.

Für diese Interrupts gibt es drei verschiedene Betriebs-Modi, die mit jeweils dem Befehl IMO, IM1 bzw. IM2 angewählt werden können.

3. IMO = Modus 0

Dieser Modus wird bei RESET automatisch gesetzt oder kann per Software mit dem Befehl IMO eingeschaltet werden. Ein Interrupt in diesem Modus bewirkt: (falls nicht mittels dem Befehl DI ausgeschaltet!)

1. Die CPU holt sich den nächsten Befehl, der ein Ein-Byte-Befehl sein muß, vom unterbrechenden Baustein (s.u.). Sinnvollerweise sollte dies ein RST-Befehl sein. Das sind ein Byte lange CALL-Befehle auf eine der Adressen 0000H, 000BH, 0010H, ..., 003BH: RST 00H, RST 0BH, ..., RST 3BH.

Wird ein solcher RST-Befehl an die CPU vom unterbrechenden Baustein übergeben, so wird also die Interrupt-Routine als Unterprogramm aufgerufen. Dieses Unterprogramm sollte an Stelle von RET mit RETI abgeschlossen werden. Dieses RETI wird von verschiedenen Z80-Bausteinen (CTC, PIO, ...) erkannt, so daß diese daran erkennen, daß der Interrupt bearbeitet wurde - das muß dem Baustein dann also nicht mehr mitgeteilt werden. Z.B. der CTC merkt, wenn einer interruptet - wartet also darauf, daß irgendwann die CPU ein RETI ausführt, sonst: (Tastatur evtl. tot unter RAM 3.x).

Achtung: Die Z80-Peripheriebausteine liefern nicht immer das Byte, welches sie bekamen zurück - sondern addieren etwas hinzu -> IM 2.

4. IM1 = Modus 1

Dieser Modus ist ähnlich dem Modus 0 (s.o.) nur, daß die CPU nicht den nächsten Befehl vom unterbrechenden Baustein holt, sondern sofort den Befehl RST 3BH ausführt. Rest wie oben. (Einschalten mit IM 1.)

5. IM2 = Modus 2

Dieser Modus (einzuschalten mit dem Befehl IM2) ist der mächtigste der Z80. RAM3 von Bernd Preusing benutzt ihn für die interruptgesteuerte Tastatur.

Was tut die CPU in diesem Fall:

1. DI (unterdrücken maskierbarer Interrupts)
2. Sie holt sich vom unterbrechenden Baustein die unteren 8 Bit (also das untere Byte), und aus ihrem I-Register die oberen 8 Bit eines 16-Bit-Pointers.
3. Sie holt sich von der Adresse, auf die der Pointer zeigt die Adresse der Interrupt-Routine und ruft diese mittels CALL auf.

Hier muß nun der Programmierer zum einen das I-Register und den betroffenen Baustein mit dem richtigen Byte versorgen, und zum anderen sicherstellen, daß die Adresse der Interrupt-Routine auch dort im Speicher steht, wo der Pointer hinzeigt wird.

Falls nach dem Ausführen der Interrupt-Routine die CPU an alter Stelle weitermachen soll muß der RETI-Befehl dieses Unterprogramm beenden.

CPU: Interrupts der Z80

Den Sinn dieses anscheinend komplizierten Verfahrens versteht man besser, wenn man sich die Funktionsweise der Z80-Peripheriebausteine (DART, CTC, PIO, ...) betrachtet. Diese Bausteine können i.a. aus vier Gründen einen Interrupt generieren. Bei der Initialisierung des Bausteins wird ihm ein Byte gegeben zu dem dann von eben diesem Baustein im Interrupt-Fall je nach Ursache 0, 2, 4 oder 6 (also zwei mal 0, 1, 2 bzw. 3) addiert werden. Ergänzt man dieses Byte mit dem I-Register der CPU zu einem 16Bit-Pointer (also einer Speicheradresse) so erkennt man, daß dadurch die Interrupts eines Bausteins auf ein Doppelbyte aus einer Tabelle zu vier Werten ausgewählt. Diese Tabelle sähe dann in Assembler etwa so aus:

```
TABELLE: DW  ADR1  ; Adresse der Interrupt-Routine No. 1
          DW  ADR2  ; Adresse der Interrupt-Routine No. 2
          DW  ADR3  ; Adresse der Interrupt-Routine No. 3
          DW  ADR4  ; Adresse der Interrupt-Routine No. 4
```

Dabei muß die Adresse TABELLE gerade (durch 2 Teilbar) sein.

Initialisierung: LD HL,TABELLE

```
LD  A,H          ;Oberes Byte nach A
```

```
LD  I,A          ;Und nun ins I-Register
```

Jetzt muß das L-Register dem Baustein noch übergeben werden.

Aber da gibt's noch eine Falle. L muß kleiner gleich 256-8 sein, denn sonst wäre z.B. L+6 größer als 256, und diese 256 sind dann ein Übertrag (ein Byte faßt nur 0 - 255) in das obere Byte. Dieses jedoch wird aus dem I-Register geholt, bekommt diesen Überlauf nicht mit. Also muß beim Anlegen der o.g. TABELLE schon darauf geachtet werden, daß diese nicht sonstwo liegt. Am sichersten ist es, diese Tabelle an eine freie Adresse zu verschieben, die in der Form ??00H aufgebaut ist, also z.B. nach 8000H. Dann gibt's keinerlei Probleme. Und den Befehl IM2 nicht vergessen.

Sollen zwei Bausteine Interrupts generieren wäre es etwa so:

```
TABELLE: DW  ADR1  ; Adresse Int-Routine No. 1 Baustein 1
          ...
          DW  ADR4  ; Adresse Int-Routine No. 4 Baustein 1
          DW  ADR5  ; Adresse Int-Routine No. 1 Baustein 2
          ...
          DW  ADR8  ; Adresse Int-Routine No. 4 Baustein 2
```

und

```
LD  HL,TABELLE
```

```
LD  A,H
```

```
LD  I,A
```

nun L an den 1. Baustein und

```
LD  A,B
```

```
ADD A,L          ;L plus 8
```

nun A an den 2. Baustein. Hier natürlich L <= 256 - 16 (s.o.)

Wie Ihr in Bernd Preusings Artikel im Info 17 sehen können werdet, gibt es auch Bausteine wie den DART, der sogar 8 Interrupt-Gründe hat, also auch mit 8 Interrupt-Routinen versehen werden muß.

Achtung: Da RAM3 diesen Interrupt-Modus zusammen mit einem Kanal des Z80-CTC benutzt solltet Ihr Euch Bernd Preusings Artikel hierzu im Info 17 ansehen! - nein besser noch durchlesen!

CPU: Interrupts der Z806. Unterbrechender Baustein

Oben habe ich mehrfach erwähnt, daß sich die CPU von dem aufrufenden Baustein einen Wert holt. Nun fragt sich sicherlich jeder, wo denn das Problem liegt, da die CPU doch einfach einen Port-Zugriff auf den Baustein machen muß, und schon ist die Sache unter Dach und Fach. Das würde die CPU auch gerne so einfach tun, aber was nun, wenn zwei verschiedene (oder gar mehr) Bausteine die Übeltäter sein können. Woher weiß die CPU welcher den Interrupt ausgelöst hat. Ja, ich sehe schon wieder jemanden denken: einfach diesen Baustein antesten (d.h. ein Port-Zugriff auf ihn) - aber wenn die CPU nicht weiß wer, wen ansprechen? Es ginge natürlich einfach alle in Frage kommenden Bausteine zu untersuchen, aber dann müßte die CPU irgendwo ein Programm haben, aus dem hervorgeht, welche Port-Adresse anzusprechen sind, und woran dann die CPU erkennt was los ist.

Mein-Gott-ist-das-umständlich-!-!

Also hatten die Z80-Entwickler (eigentlich die der BOBO) eine einfache Idee: Der Baustein, der den Interrupt ausgelöst hat weiß ja ICH-WAR-ES. Also muß die CPU einfach mal auf ein mal alle gleichzeitig ansprechen, und die Bausteine sind verpflichtet nichts von sich zu geben - bis auf unseren speziellen Freund, den Auslöser. Genial, nicht ?

Und so wird es gemacht. Im folgenden könnt Ihr sehen, was die CPU auf ihre Daten-, Adreß- und Steuerleitungen ausgibt bzw. einliest:

1. Normaler Port-Zugriff lesen

- Adreßleitungen Bit 0 - 7: Port-Adresse
 - Steuerleitungen MREQ (Memory-Request = Speicherzugriff) 5 V
 - IOREQ (I/O-Request = Portzugriff) 0 V
 - M1 (Machine-Zyklus = Befehllese) 5 V
 - RD (Read = Lesen) 0 V
 - Datenleitungen CPU liest was drauf kommt (vom Port)
- In diesem Fall ist also IOREQ aktiv (weil 0 V), aber M1 nicht, da es sich nicht um das Lesen eines Befehls handelt.

2. Interrupt-Zugriff (Byte vom aufrufenden Baustein holen)

- Steuerleitungen MREQ (Memory-Request = Speicherzugriff) 5 V
 - IOREQ (I/O-Request = Portzugriff) 0 V
 - M1 (Machine-Zyklus = Befehllese) 0 V
 - RD (Read = Lesen) 0 V
 - Datenleitungen CPU liest was drauf kommt (vom Port)
- In diesem Fall sind also IOREQ und M1 aktiv, da nun die CPU ja einen Befehl (Modus 0) liest, bzw. eine halbe Adresse (Modus 2), was zwar kein Befehl ist, liest, aber vom Baustein das gleiche Verhalten erwartet: Ein Byte zu liefern, wenn er den Interrupt ausgelöst hat.

Für eventuelle Interrupt-Prioritäten muß die Hardware sorgen. Wer sich hier vertiefen möchte, sollte mal unter **Daysy-Chain** in Z80-Bücher schauen.

BIOS: Carrera**Korrektur zu Info 15, Seite 12 ff**

(Herbert Herberg, 2000)

In Info 15, Seite 12 ff habe ich leider eine kleine Routine vergessen. Hab Dank Andreas Viebke für Deinen Hinweis darauf.

Die von Andreas Viebke in seinem Leserbrief in diesem Info erwähnte Originalfassung des Filters stammt hingegen nicht, wie er vermutete von Uwe Grass, sondern ebenfalls von mir, ist aber im Gegensatz zu der, die ich veröffentlicht habe, komplizierter zu ändern. Darum habe ich ja diese neue zusammengestirckt.

Also nun erst mal die Korrektur der meinigen Routine:

Folgendes kommt direkt vor '; Verschieben unserer Routine'

```

; Korrigieren der Adressen der Ersatz-Zeichenfolgen
; -----
; Die Werte der Labels NORM, CODE1, CODE2 und ggf. weiterer im OVERLAY
; ändern sich ja, da das Overlay verschoben wird. Wenn nicht in den
; drei Befehlen LD HL,NORM, LD HL,CODE1 und LD HL,CODE2 die richtigen
; Adressen, d.h. die wo die Strings hingeschoben werden, eingesetzt
; werden, dann holt sich das Programm die Werte ja von falscher Stelle
;
LD HL,(FREE)
INC HL ;dort kommt die Routine hin
LD DE,START ;dort ist sie jetzt
XOR A ;lösche Carry-Flag
SBC HL,DE ;HL = Abstand von Ursprungslage zum Ziel
EX DE,HL ;nach DE
SETNRM: LD HL,NORM ;Der Wert
ADD HL,DE ;addiere den Versatz
LD (START+2),HL ;dort steht der Befehl LD HL,NORM
SETCD1: LD HL,CODE1 ;Der Wert
ADD HL,DE ;addiere auch hier den Versatz
LD (CHR1+5),HL ;dort steht LD HL,CODE1
SETCD2: LD HL,CODE2 ;s.o.
ADD HL,DE
LD (CHR2+5),HL
;hier folgen ggf. weitere CODE-Anpassungen falls noch ein CHR3
;mit entsprechendem CODE3 vorkommen soll.

; Verschieben unserer Routine
; -----
u.s.w.

```

Kommentar von Bernd Preusing

(Herbert Herberg, 2000)

Bernd mente zu mir am Telefon, daß es sinnvoller ist, solche Filter wie oben statt hinter FREE vor TOAM zu schieben, da sonst ein erneutes CONFIG (bzw. CFIG3) zwar funktioniert, wohl aber unnötig Platz braucht (dieses nutzt nämlich den Platz ab FREE). Dazu müßte das Ziel statt mittels LD DE,(FREE) / INC DE aus TOAM berechnet werden: (/ = RET)

LD HL,(TOAM) / LD DE,ENDE-START / SBC HL,DE / DEC HL / EX DE,HL. Nun steht in DE das Ziel. Außerdem wird dann nicht FREE korrigiert durch LD (FREE),DE nach dem Verschieben, sondern TOAM durch LD (TOAM),DE vor dem Verschieben.

BIOS: Noch'n Filter

OLYMPIA: Noch ein Filter und ein Dump-Modus

Andreas Viebke

Nach der Lektüre von Herberts Filter-Programm aus Info 15 bin ich zu dem Schluß gekommen, daß dieses Programm nicht das tun kann, was es soll (siehe auch den diesbezüglichen Leserbrief von mir). Deshalb habe ich selbst so einen Filter geschrieben, für die Olympia electronic compact 2 (ec2), die ich besitze. Es hat den Vorzug, daß es außer "<" und ">" auch noch "^" verarbeiten und TABs (Tabulatorzeichen) in eine entsprechende Anzahl von Leerzeichen umwandeln kann. Die ec2 ist nämlich zu doof dazu.

Dann kann mein Programm das von ihm installierte Overlay auch wieder "uninstallieren", den von ihm beanspruchten Speicherplatz wieder freigeben usw., d.h., ich brauche nicht auf die Reset-Tasten zu drücken, wenn ich zwischen DMX80 und ec2 wechseln will.

Ferner habe ich einen Dump-Modus eingebaut: Ein Overlay, das jedes ankommende Zeichen in zweistellige Hex-Zahlen umwandelt und diese, gefolgt von einem Leerzeichen, ausgibt. Ein Zeilenvorschub bewirkt die Ausgabe von RET und LF. Dieser Modus ist ganz nützlich, wenn man mal wissen möchte, welche Steuersequenzen ein Anwenderprogramm zum Drucker schickt. Da kann man dann wieder neue Overlays schreiben... Der Dump-Modus ist nicht nur für die ec2 geeignet, sie funktioniert auch mit einem DMX80. Wenn man also nur diese Eigenschaft nutzen möchte, dann können der ec2-"Treiber" und seine Installationsroutine einfach weggelassen werden.

Wie hält man dies alles auseinander? Ganz einfach: Es gibt drei Formen des OLYMPIA-Kommandos:

- (1) OLYMPIA
- (2) OLYMPIA DUMP
- (3) OLYMPIA OFF

(1) installiert den Filter, (2) die Dump-Routine und (3) schaltet ab, was gerade installiert ist, wenn was installiert ist.

Die Besitzer anderer Maschinen von Olympia oder Druckern, die nicht 100%ig dem ASCII gehorchen, können den Teil DRIVER sicherlich leicht ihren Bedürfnissen anpassen.

In der Dump-Routine habe ich den Befehl DAA verwendet. Das funktioniert zwar, ist aber nicht besonders elegant. Wer hat eine bessere Idee zur effektiveren Nutzung des DAA-Befehls? Hier ist OLYMPIA.COM, das man kostenlos von mir bekommen kann.

```
;OLYMPIA von AV. Ein kleines Programm, u.a für die Typenrad-
;schreibmaschine Olympia electronic compact 2, das entweder
;
;eine Routine (DRIVER, eigentlich kein Treiber, sondern eher
;Filter) installiert, die bestimmte an den Drucker gesandte
;Zeichen in andere Zeichen oder Sequenzen umwandelt
;
;oder den Drucker (bzw. die Schreibmaschine) in den Dump-
;Modus (DUMP) schaltet, in dem alle Zeichen als hexadezimale
;Codes, gefolgt von einem Leerzeichen, ausgegeben werden. In
;diesem Modus bewirkt ein Zeilenvorschub (0A) die Ausgabe
;einer CR-LF-Sequenz.
```

BIOS: Noch'n Filter

```

.Z80
ASEG
ORG      100H

FREE    EQU    40H
TOAM    EQU    46H
DUMMY   EQU    0
LSTOUT  EQU    DUMMY
FCB     EQU    5DH

;Prüfe, ob Treiber oder Dump-Routine schon im Speicher sitzt,
;entferne ihn/sie in diesem Fall. DRIVER und DUMP müssen mit
;einem Sprung zur alten Druckerausgabe, gefolgt von der Iden-
;tifikationssequenz OLYM, enden.

LD      DE,SIGNON      ;Begrüßung ausgeben
LD      C,9
CALL    5
LD      HL,(FREE)      ;Höchste verwendete Adresse +1
LD      DE,ID$+4      ;Ende der ID-Sequenz +1
LD      B,4            ;4 Zeichen
COMPARE:DEC HL        ;Vergleiche B Bytes
DEC     DE
LD      A,(DE)
CP      (HL)
JR      NZ,NOMATCH    ;Nicht gleich, Overlay nicht da
DJNZ    COMPARE       ;Gleich, nächstes Zeichen
DEC     HL             ;Adr. des alten LSTOUTs ist
LD      B,(HL)        ;unterhalb der ID-Sequenz
DEC     HL             ;BC damit laden.
LD      C,(HL)
CALL    GETLST        ;Zeiger auf LSTOUT in Sprungleiste
LD      (FREE),DE     ;nach HL, die Adresse von dort nach
LD      (HL),C        ;DE. Das ist der alte FREE-Zeiger
INC     HL             ;Altes FREE eintragen und altes
LD      (HL),B        ;LSTOUT, BC, bei (HL) eintragen

;Eventuell vorhandenes Overlay ist entfernt. Jetzt muß die
;Kommandozeile überprüft werden. Da nur ein kurzes Wort da
;stehen kann, kann man in CP/Ms erstem FCB (5DH ohne Lauf-
;werk) nachsehen. Dort kann entweder OFF, DUMP oder gar
;nichts stehen. Bei OFF geschieht nichts mehr, bei DUMP wird
;die Dump-Routine installiert, und steht gar nichts da, wird
;der Filter installiert.

NOMATCH:LD DE,OFF$    ;Adresse von OFF
LD      B,4            ;Zeichenanz. (incl. Leerzeichen.)
CALL    CHECK         ;Vergleiche
LD      DE,UNINST     ;Mitteilung im Falle eines
LD      C,9            ;erfolgreichen Vergleichs
JP      Z,5           ;Z heißt "Vergleich klappte"

LD      DE,DUMP$      ;Steht dort DUMP?
LD      B,5            ;Anzahl der Zeichen
CALL    CHECK         ;Vergleiche
JP      NZ,DRIVER     ;Nein, dann DRIVER installieren

```

BIOS: Noch'n Filter

;Hier beginnt die Installation der DUMP-Routine
 ;-----

```
LD      BC,ENDUMP-DUMP ;Länge der DUMP-Routine
CALL    SPACE?        ;Genug Platz im BIOS?
RET     C              ;C, wenn nicht
CALL    GETLST        ;HL=Zeiger auf LSTOUT in Sprungleiste
EX      DE,HL         ;DE=Adresse von dort
LD      (DL1+1),HL    ;DE bei Bezügen auf LSTOUT in DUMP-
LD      (DL2+1),HL    ;Routine eintragen
LD      (DL3+1),HL
LD      (DL4+1),HL
EX      DE,HL

PUSH    HL
LD      HL,(FREE)     ;In Dump selbst gibt es einen CALL,
INC      HL           ;der sich auf DUMP bezieht. Diese CALL-
INC      HL           ;Adresse liegt 2 Bytes über dem zu-
LD      (DS1+1),HL    ;künftigen Einsprung von LSTOUT, jetzt
LD      (DS2+1),HL    ;noch (FREE). Zweimal taucht er auf,
DEC      HL           ;bei DS1 und DS2. Zukünftige Adresse
DEC      HL           ;eintragen.
EX      DE,HL         ;DE = (FREE)
LD      HL,DUMP       ;HL = Start des Overlays
LD      BC,ENDUMP-DUMP ;BC = Länge des Overlays
LDIR    ;Verschieben
POP     HL            ;Immer noch Zeiger auf Adresse
LD      BC,(FREE)     ;des LSTOUT-Sprungs in Sprungleise
LD      (HL),C        ;(FREE) ist neuer Wert, der da
INC      HL           ;eingetragen werden muß
LD      (HL),B
LD      (FREE),DE     ;DE zeigt nach LDIR auf nächstes
LD      DE,CONDUMP    ;freies Byte im BIOS-Patch-Bereich
LD      C,9           ;Installation bestätigen
JP      5
```

;Hier beginnt die Installation des DRIVERS (des Filters)
 ;-----

```
DRIVER: LD      BC,END-START ;Sie ist analog zur Installation
CALL    SPACE?        ;der Dump-Routine. Auch hier gibt
RET     C              ;es einen Bezug auf eine Adresse
CALL    GETLST        ;innerhalb der Routine: PRPOS dient
EX      DE,HL         ;der Speicherung der Druckposition,
LD      (L1+1),HL    ;die für TAB wichtig ist.
LD      (L2+1),HL
LD      (L3+1),HL
LD      (L4+1),HL
LD      (L5+1),HL
LD      (L6+1),HL
EX      DE,HL

PUSH    HL
LD      HL,(FREE)     ;(FREE)+2 ist die zukünftige
INC      HL           ;Adresse von PRPOS, die bei S1+1
INC      HL           ;steht.
LD      (S1+1),HL
DEC      HL
DEC      HL
```

BIOS: Noch'n Filter

```

EX      DE,HL          ;DE = (FREE)
LD      HL,START      ;HL = Start des "Treibers"
LD      BC,END-START  ;BC = Länge des "Treibers"
LDIR                               ;Verschieben
POP     HL
LD      BC,(FREE)
LD      (HL),C
INC     HL
LD      (HL),B
LD      (FREE),DE
LD      DE,CONDRIV    ;Installation bestätigen
LD      C,9
JP      5

```

```

;Hilfsroutinen
;-----

```

*;HL soll auf die Adresse des sechsten BDOS Jumps in der Sprung-
;leiste zeigen, bzw. auf die Adresse, die die Adresse enthält,
;zu der dieser Sprung führt. Der eigentliche LSTOUT-Vektor wird
;nach DE geladen. Die Adresse des sechsten BDOS-Sprunges kann
;folgendermaßen berechnet werden (s. auch die Beschreibung von
;Uwe Grass): Ein Sprung zum zweiten BDOS-Sprung findet man an
;der Adresse 0 (JP WBOOT). Addiert man zu der Adresse, zu der
;der Sprung führt, weitere vier Sprünge (4*3 Bytes), erhält man
;einen Zeiger auf den sechsten Sprung, den man nur noch um eins
;zu inkrementieren braucht, um ihn auf die Adresse zeigen zu
;lassen anstatt auf den Assembler-Sprungbefehl.*

```

GETLST: PUSH  BC
LD      HL,(1)
LD      BC,4*3+1
ADD     HL,BC
LD      E,(HL)
INC     HL
LD      D,(HL)
DEC     HL
POP     BC
RET

```

*;Prüfe, ob genügend Platz im BIOS-Patch-Bereich ist. Carry
;ist gesetzt, wenn kein Platz. Diese Unteroutine gibt auch
;gleich eine Fehlermeldung aus, wenn's sein muß. BC muß beim
;Eintritt die Länge des Overlays enthalten.*

```

SPACE?: LD      HL,(TOAM)      ;Obergrenze des verfügbaren Speichers
LD      DE,(FREE)            ;Adresse des ersten freien Bytes
AND     A                    ;Carry löschen
SBC     HL,DE                 ;Differenz ist Anzahl der freien
JR      C,PRERR               ;Bytes im Patch-Bereich
SBC     HL,BC                 ;Länge des Overlays abziehen
RET     NC                    ;Ok, wenn kein (negativer) Überlauf
PRERR:  LD      DE,ERROR      ;Gib Fehlermeldung aus
LD      C,9
CALL   5
SCF                               ;Zeige Fehler an
RET

```

BIOS: Noch'n Filter

;Hier wird die Kommandozeile, bzw. CP/Ms FCB mit dem String
;verglichen, auf den DE zeigt. B muß beim Eintritt die Länge
;des Strings enthalten.

```
CHECK: LD      HL,FCB
OFFFLP: LD     A,(DE)
        CP     (HL)
        INC   HL
        INC   DE
        RET   NZ           ;NZ, wenn Ungleichheit
        DJNZ  OFFFLP
        RET

SIGNON: DB    13,10,'*** Olympia ec2 adjustment 1986 by AV ***',13,10,'$'
ERROR:  DB    'NO ROOM FOR OLYMPIA OVERLAY',13,10,'$'
CONDUMP:DB    'OLYMPIA DUMP OVERLAY INSTALLED',13,10,'$'
CONDRIV:DB    'OLYMPIA DRIVER OVERLAY INSTALLED',13,10,'$'
UNINST: DB    'OLYMPIA OVERLAY REMOVED',13,10,'$'
OFF$:   DB    'OFF '
DUMP$:  DB    'DUMP '
ID$:    DB    'OLYM'
```

;Dies ist der "Treiber" für die Olympia ec2

;------

```
START: JR      SUBST           ;Überspringe Datenbyte
PRPOS: DB      0               ;zur Speicherung der TABS
SUBST: LD      A,C             ;Zeichen kommt in C
        CP      9               ;ist es ein TAB?
S1:   LD      HL,PRPOS         ;HL auf Druckposition
        JR      NZ,NOTAB       ;Springe, wenn kein TAB
        LD      A,(HL)         ;Hole Druckposition nach A
        LD      (HL),0         ;und setze sie auf 0
        AND     7               ;Berechne Anzahl der anstatt des
        NEG     ;TABS auszugebenen Leerzeichen
        ADD    A,8             ;(8-(PRPOS))
        LD      B,A            ;Nach B
TABLP: PUSH    BC              ;Sende B Leerzeichen zum Drucker
        LD      C,' '
L6:   CALL    LSTOUT
        POP     BC
        DJNZ   TABLP
        RET

NOTAB: CP      8               ;Kein TAB, sondern BS?
        JR      Z,DECBY1       ;Ja, dann dekrementiere Zähler
        CP      0DH            ;Ist es ein RET?
        JR      Z,SET0         ;Ja, dann setze Zähler auf 0
        CP      ' '            ;Ist das Zeichen nicht-druckbar?
        JR      C,PRINT        ;Ja, dann inkrementiere Zähler nicht
        INC    (HL)           ;Druckbar - inkrementiere Zähler
        JR      PRINT          ;Zur Ausgabe
SET0:  LD      (HL),1
DECBY1: DEC    (HL)
PRINT: CP      '<'             ;'<' und '>' herausfiltern, sie
        LD      C,2DH          ;erfordern beide die Umschaltung
        CP      ' >'          ;auf die Sonderzeichenebene, Ausgabe
        LD      C,2FH          ;des Ersatz-Zeichens und das Zurück-
        LD      C,2FH          ;schalten auf die Normal-Ebene
```

BIOS: Noch'n Filter

```

        JR      Z, CHRS
        CP      '^'          ;ist Zeichen ein '^'?
        LD      C, A
        JR      NZ, LASTCHR

L1:     CALL   LSTOUT        ;Ja, dann sende beide Akzente und
        LD      C, 60H      ;ein Leerzeichen (macht 'ne Schreib-
L2:     CALL   LSTOUT        ;maschine normalerweise ja nicht)
        LD      C, 20H
        JR      LASTCHR

CHRS:   PUSH   BC          ;Hier Umwandlung von '<' und '>'
        LD      C, 14      ;Rette Ersatz-Zeichen in C
L3:     CALL   LSTOUT        ;14 ist Code für Sonderzeichen
        POP    BC          ;Ersatzzeichen wiederholen
L4:     CALL   LSTOUT        ;Und ausgeben
        LD      C, 15      ;15 ist Code für normale Ebene

LASTCHR:
L5:     JP     LSTOUT        ;LSTOUT-Sprung und ID-Sequenz sind
        DB     'OLYM'      ;obligatorisch am Schluß

END:    ;Ende des Filter-Overlays

```

```

;Dies ist die DUMP-Routine - auch für andere Drucker
;-----

```

```

DUMP:   JR      INDUMP      ;Sub-Routine überspringen
ONEBYT: OR      A          ;Ausgabe des unteren Halb-Bytes
        DAA          ;von A.
        SUB     10H
        CCF
        ADC     A, 40H
        PUSH   BC
        LD      C, A
DL1:    CALL   LSTOUT
        POP    BC
        LD      A, C
        RET

INDUMP: LD      A, C      ;Oberes Halb-Byte zu unterem
        AND     0F0H      ;Halb-Byte machen
        RRCA
        RRCA
        RRCA
        RRCA
DS1:    CALL   ONEBYT      ;Und ausgeben
        AND     0FH      ;Unteres Halb-Byte isolieren
DS2:    CALL   ONEBYT      ;Und auch dieses ausgeben
        CP      10      ;War Zeichen ein Zeilenvorschub?
        LD      C, ' '    ;Wenn nicht, dann ist nächstes und
DL2:    JP     NZ, LSTOUT   ;letztes Zeichen ein Leerzeichen
        LD      C, 13      ;Ansonsten RET
DL3:    CALL   LSTOUT      ;ausgeben
        LD      C, 10      ;gefolgt von einem LF
DL4:    JP     LSTOUT      ;Alte LSTOUT-Adresse
        DB     'OLYM'      ;und die ID-Sequenz

ENDUMP:

```

NewWord: Dezimaltabulator**Newword, Dezimaltabulator**

(Uwe Grass, 3300)

Wenn es Euch schon immer geärgert hat, daß sich Newword beim Tabellieren von Dezimalzahlen am Punkt orientiert, so gibt es jetzt endlich eine Lösung. Man muß im Overlay NWMSG5.OVR ein wenig herumflicken (patchen). Die Stelle im alten NW ist CBF5, im neuen NW 018B. Es muß der Hex-Code 2D 3D 2E 15 in 2D 3D 2C 15 geändert werden. Zur Vorgehensweise (für Moni-Besitzer):

MONI NWMSG5.OVR/ (/ = RETURN, BRK = Break-Taste, ESC = Escape)

E018B/ , / BRK W * NWMSG5.OVR/ ESC (neues NW)

ECBF5/ , / BRK W * NWMSG5.OVR/ ESC (Altes NW)

ACHTUNG: bei obigem die Leerzeichen nicht mit eingeben!!!)

Das war schon alles.

UG

Anm.d.HH. Mit DDT geht's auch, aber nur mit einem 59k-System. Ein entsprechendes Patch-Programm von mir gibt's auf CLUB.012.

Dateigröße auf Diskette

(Herbert Herberg, 2000)

Warum belegen meine Dateien auf Format 03 nur 202kB, auf Format 13 hingegen 280kB?

Die Lösung ist recht einfach, aber nicht leicht zu formulieren:

Ein Directory-Eintrag enthält außer dem Dateinamen auch noch die Adresse der sogenannten Blöcke der Diskette, in denen die Datei untergebracht ist. Ggf. belegt eine Datei also mehrere Einträge, da sonst die Einträge zu groß wären, und kleine Dateien unnötig große Einträge belegen würden. Ach so, natürlich kann man die Einträge verschieden groß machen - aber wer verwaltet das, insbesondere bei Delete? Einfacher sind kleine Directory-Einträge, und ggf. mehrere für eine Datei. Eine Datei von einem Byte Länge belegt also einen ganzen Block, da es kleiner nicht unter CP/M geht (sonst wäre der Verwaltungsaufwand zu groß - alles langsamer). Man spricht übrigens von Kompromissen in diesem Zusammenhang, oder?

Nun deutlicher:

1. Directory-Eintrag zeigt auf Blöcke, die Datei belegt.
2. Datei belegt also auf Diskette immer ein vielfaches der Länge eines Blockes, unabhängig von der eigentlichen Größe!
3. Blöcke klein bedeutet, daß weniger Platz verschwendet wird, da der letzte Zipfel der Datei nicht einen großen Block belegt, aber auch größere oder mehr Directory-Einträge.
4. D.h. Kleine Blöcke -> bessere Platznutzung, größere/mehr Directory-Einträge, mehr Directoryuntersuchung für eine Datei nötig und größeres Directory (auf Diskette).
5. Großes Directory und lange Einträge machen Zugriff langsamer!
6. Lösung: Paßt mehr auf die Scheibe, mach Blöcke größer!

Fazit: Format 03: Blockgröße 2kB, Format 13: Blockgröße 4kB.

Es belegt eine Datei von 200 Bytes auf 03: 2kB, auf 13: 4kB. Klaro?

Wenn nicht nocheinmal lesen, dann meckern!

HARDWARE: Druckerumschaltung

Hardware, Umschaltung zwischen zwei Druckern (Uwe Grass, 3300)
 Eine Bastelei auch für Anfänger

Da ich seit längerer Zeit zwei Drucker besitze (Typenradrunder Olympia Carrera und Nadelrunder DMX 80), ergab es sich oft, daß ich die Drucker gewechselt habe. Das brachte jedesmal ein Umstöpseln des Centronics-Steckers mit sich. Da der Nadelrunder inzwischen recht gut im Regal eingebaut ist, war dies immer sehr umständlich. Wie ihr wohl bemerkt habt, ist dies bei mir jetzt schon Vergangenheit.

Wie habe ich das nun gemacht? Alle Signale auf zwei Drucker gleichzeitig zu geben ist nicht möglich. Jeder Drucker gibt nämlich Informationen an den Rechner zurück, der könnte nun aber immer nur auf ein Signal richtig reagieren. Also muß ein Umschalter her. Wenn man aber alle Signale umschalten will, dann braucht man schon einen 21-poligen Schalter. Das ist ziemlich aufwendig. Aber man muß ja nicht alle Signale umschalten, denn Schwierigkeiten bereiten nur wenige.

Da sind vor allem die vom Drucker kommenden Informationen, ACK (Acknowledge), Busy, PE (Paper End), SLCT (Select) und ERROR. Und siehe da, wir benötigen nur noch einen 6-poligen Umschalter.

Da ich den MTX aus der Tastatur in ein Extragehäuse ausgelagert habe, war es für mich klar, wo der Umschalter einzubauen war. Wer im Rechnergehäuse nicht genug Platz hat, der kann den Umschalter auch in eine eigene Schachtel stecken oder in ein eventuell vorhandenes Interface einsetzen (zur Carrera gibt es eins, dort ist in dem Gehäuse soooo viel Platz).

In die Schachtel hinein geht dann der Druckeranschluß vom Rechner, heraus kommen zwei Anschlüsse für je einen Drucker. Innen werden die Datenleitungen direkt auf beide Abgänge verteilt, ebenso STB (Strobe), Auto Feed XT, SG (Signal Ground), FG (Frame Ground, bei uns das gleiche wie SG), PRIME. Umgeschaltet werden die oben erwähnten Signale. Das ganze noch einmal in einem Diagramm:

Die Leitungsnummern geben den Pin des Steckers an, man beachte, daß beim MTX nur 34 Pole vorhanden sind, während beim Centronics 36 Pole gibt. D.h., daß die Kontakte 18 und 36 beim MTX einfach fehlen. Beachtet das beim Abzählen der Kontakte am Stecker!

Achtung: Auf dem Flachbandkabel liegt nicht Leitung 1 neben 2, da die gegenüberliegende Leitung, Nummer 19 dazwischen liegt.

<u>Leitung</u>	<u>Schalter / Direkte Verbindung</u>
1	STROBE ----- Schalter
2-9	DATEN 0 bis 7 ----- Direkte Verbindung (acht Leitungen)
10	ACK ----- Schalter
11	BUSY ----- Schalter
12	PE ----- Schalter
13	SLCT ----- Schalter
14-31	Verschiedene ----- Direkte Verbindung
32	ERROR ----- Schalter
33-35	Verschiedene ----- Direkte Verbindung
18 und 36 bleiben leer!	

HARDWARE: Druckerumschaltung / Carrera-Interface

Es könnte sein, daß es auch möglich ist weniger Leitungen umzuschalten. Ich hatte dann aber immer Probleme in irgendeinem Betriebszustand. Außerdem schalte ich noch ein Signal um, daß eine Anzeige steuert, um den angewählten Drucker zu identifizieren. Dafür können einfache Leuchtdioden genommen werden, oder aber man baut sich ein Display für diverse Informationen..... siehe "Eine Anzeige"

UG

Carrera am MTX

(Herbert Herberg, 2000)

Wie Uwe Grass habe auch ich mir mittlerweile die Typenradmaschine Carrera von Olympia zugelegt. Nein, ich werde das Info nicht damit Drucken!!! Die schafft nur 10 Zeichen je Sekunde - das dauert zu lang. Außerdem ist diese recht kleine Maschine nicht für harten Dauerbetrieb gedacht, obwohl damit auch mal 100 Seiten in einem Schwung (d.h. viele Stunden lang) ausgedruckt werden können. Nun kostet diese Schreibmaschine ca. DM 780.-, und ist ein sehr gutfunktionierendes Gerät. Mit Interface ist sie DIABOLO-Kompatibel in der Ansteuerung - und NewWord unterstützt diesen Drucker! Also kann ich schon auf die schönen Dinge des Druckens zugreifen: Schattenschrift, Microspacing, verschiedene ansteuerbare Schrittweiten in 1/48" vertikal und 1/120" horizontal, Oder einfacher: .CW, .LH aus NewWord laufen!

Aber da war doch noch das Interface. Nun dafür kann man schon satte DM 200.- berappen, aber es geht auch billiger. Man nehme einen LötKolben, Kabel, einen Widerstand 470 Ohm), zwei Kondensatoren (je 100 nF keramisch!) und zwei 74LS00. Ach so zwei Stecker natürlich auch noch: einen 34-poligen und einen 20-poligen Quetsch-Pfostenverbinder (also so ein lüftes Ding wie am MTX-seitigen Ende des Druckerkabels und ein noch kleineren derart) und Kabel.

ABER dabei ist eine kleine Schummelei. Eigentlich müßten alle Datensignale D0-D7 noch invertiert werden (also 74LS?? her!), nur bin ich dazu zu bequem gewesen, und wozu auch? Um die Carrera zu betreiben ist sowieso ein Overlay im System angeraten - und Dank meiner Bequemlichkeit erzwungen. Es muß das Zeichen, welches an die Carrera geschickt wird zuerst noch komplementiert werden (Z80-Assembler-Befehl CPL A).

Also z.B. in meinem Overlay auf Seite 14, Info 15 bei PRINT:

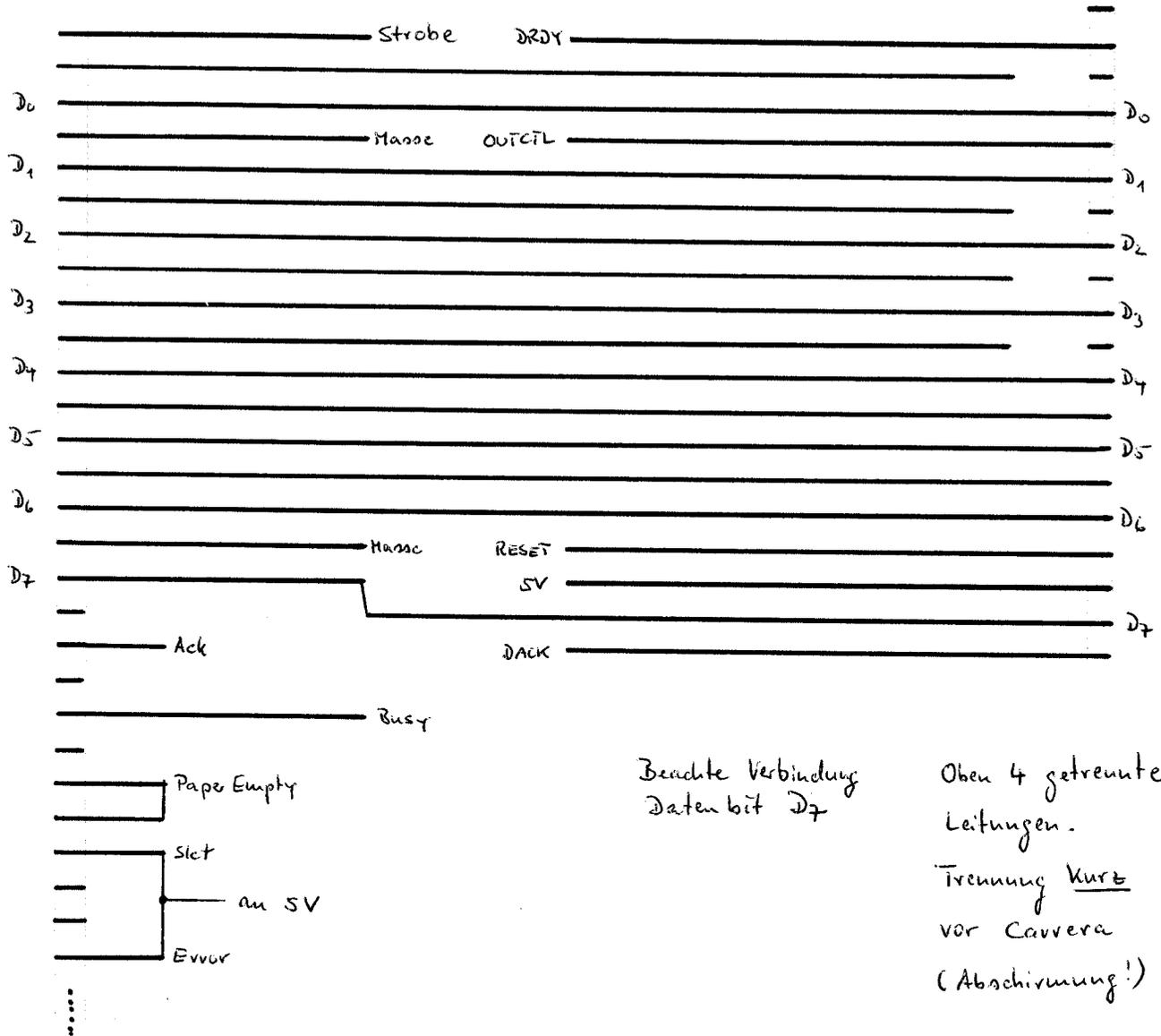
```
PRINT: LD      A,(HL)
        OR      A
        RET     Z
        CPL     A          ;Dieser Befehl muß noch hinein! Er erspart
                          ;die 8 Inverter für die Datenleitungen!
        LD      C,A
        u.s.w.
```

Bei dem Treiber von Andreas Viebke, der auch für die Carrera angepaßt werden kann, müssen im "Treiber" alle Befehlsfolgen LD C,wert durch die drei Befehle LD A,wert / CPL A / LD C,A ersetzt werden.

HARDWARE: Carrera-Interface

MTX

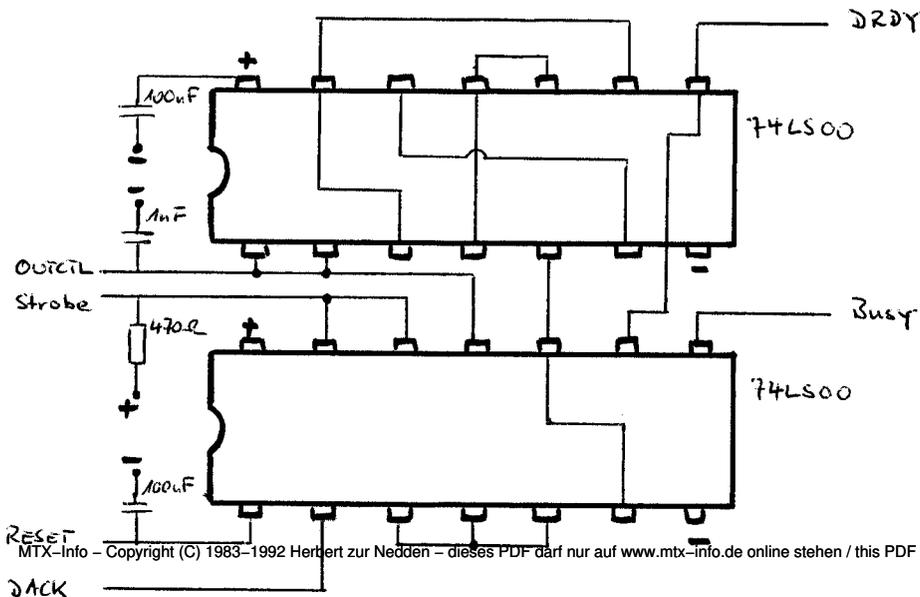
Carrera



Beachte Verbindung Daten bit D7

Oben 4 getrennte Leitungen.
Trennung Kurz vor Carrera
(Abschirmung!)

Schaltung von Oben.



Beachte

Die beiden Pins 14 und freies Ende des Wickelstandes an +5V,
die Pins 7 und die freien Enden der 3 Kondensatoren an MASSE.

HARDWARE: Eine Anzeige / 32k-Karte**Eine Anzeige** (auch Display genannt) (Uwe Grass, 3300)

Ja, hier geht es nicht um Knöllchen oder so. Hier geht es um ein Display. Viele Betriebszustände bleiben lange Zeit im Dunkeln, wenn man nicht irgendwann auf die Idee kommt sie anzuzeigen. Da ich inzwischen ein umschaltbares Laufwerk habe (umschaltbar ist die Umdrehungsgeschwindigkeit und Aufzeichnungstechnik der Floppy, für 77-Spur und 80-Spur), geschah es häufig, daß ich nicht so genau wußte, oder mit einem Blick sehen konnte, welcher Betriebszustand gewählt war. Abhilfe hat ein Display gebracht, wie es in Weckern (Digital natürlich, nicht der mit der großen Rassel obendrauf) eingesetzt wird. Dieses Ding gab es bei Völkner billig zu kaufen. Aber Achtung, einfach ist alles nur, wenn es sich um eine Exemplar handelt, das nicht im Multiplex-Verfahren angesteuert wird (oder einfacher gesagt, jedes Leuchtsegment hat seinen eigenen Anschluß). Nun habe ich dieses Ding dort in die FDX eingebaut, wo in dem Plastik-Lüftungsschlitz der Aufkleber mit dem Namen FDX sitzt. Die Kathode (Minus) hatten alle Segmente gemeinsam, die Anoden wurden über Vorwiderstände an 5V angeschlossen. Fest verdrahtet werden können alle Segmente, die eine 77 bilden. Für den 80-Spur Modus werden die restlichen Segmente nur zusätzlich eingeschaltet. Als Umschalter für das Laufwerk habe ich einen 2-poligen Kippschalter genommen, ein Pol für die Laufwerksumschaltung, der andere für die Anzeige. Durch diese Methode waren keine Eingriffe in das Laufwerk nötig. Wie jedes Uhrendisplay gibt es jetzt auch noch Doppelpunkte, in diesem Fall ein Paar vor der Anzeige, ein Paar in der Mitte. Die vorderen Punkte benutze ich, um anzuzeigen, welcher Drucker angewählt ist (oben = Typenrad, unten = Matrix), die mittleren Punkte zeigen an, welche Taktfrequenz ich gewählt habe.

Die von mir gewählten Vorwiderstände sind:

Für 1 Segment	510 Ohm	Die Widerstände dürfen aber auch größer sein.
Für 2 Segmente	270 Ohm	
Für 3 Segmente	180 Ohm	
Für 4 Segmente	150 Ohm	
Für 5 Segmente	100 Ohm	

Die Taktfrequenz habe ich umschaltbar gemacht, indem ich an einen 3-poligen Umschalter den 4 MHz (aus der MTX- Hauptplatine) und einen 5 MHz Quarz angeschlossen habe (je 2 Pole). Den 3. Kontakt benutze ich zur Steuerung der Anzeige, mittlerer Punkt oben = 5 MHz, unten = 4 MHz. Die Verbindung der Quarze zur Hauptplatine erfolgt über 15 cm abgeschirmtes Kabel. Dieser Aufbau funktioniert bei mir problemlos. In absehbarer Zeit werde ich versuchen, ob ich 6 MHz laufen lassen kann. Darüber lest ihr dann in einem Info mehr.

UG

Die 32k-Karte (Herbert Herberg, 2000)

Hier nun endlich der Schaltplan der 32k-Karte, eigentlich der 64k- bzw. 128k-Karte. Wie Ihr seht, generiert das PROM aus der Page-Adresse und den Adreßbits A14, A15 die oberen Adressen der RAMs und das Signal, mit dem die eine oder die andere Speicherbank angewählt wird.

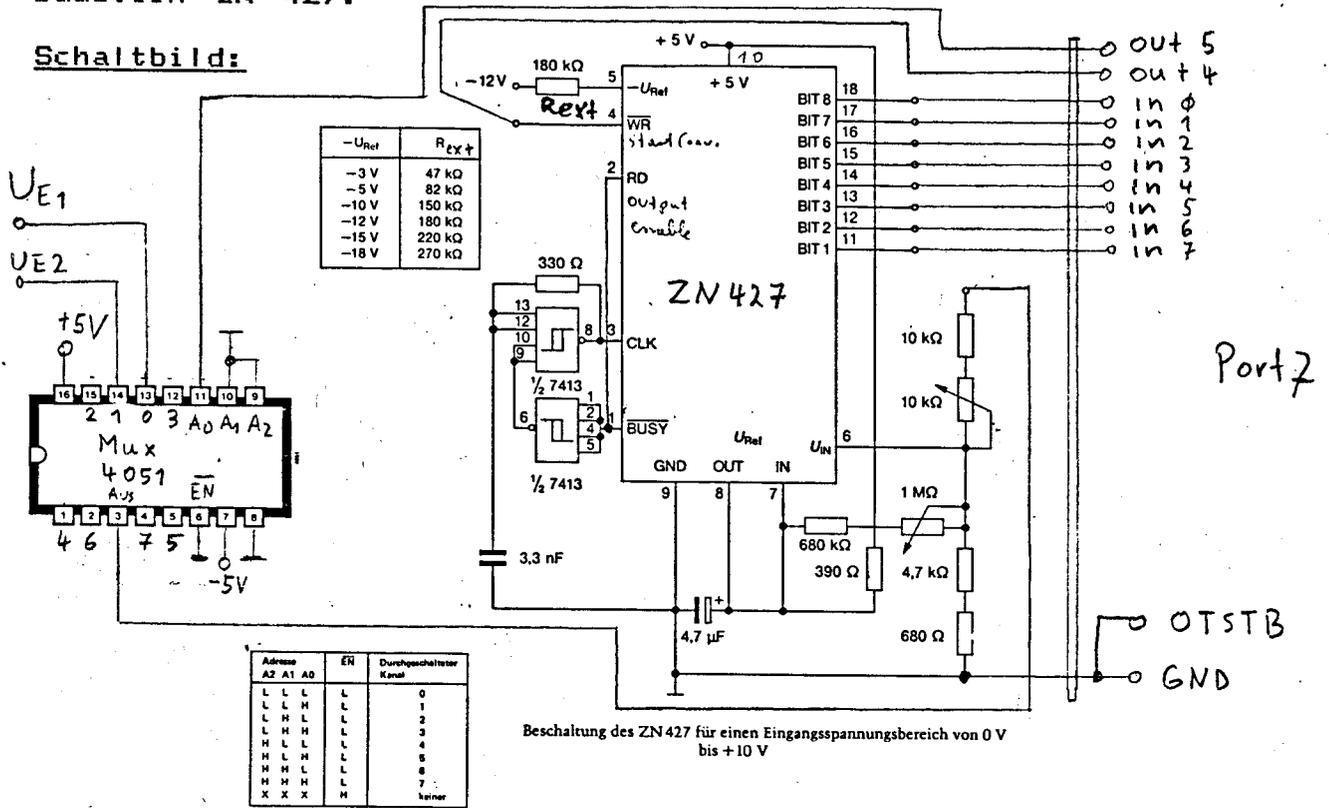
Sorry, aber diese Seite musste ich entfernen, weil ich für eine Veröffentlichung außer in der ehemaligen, gedruckten Form keine Freigabe habe.

HARDWARE: A/D-Wandler mit Multiplexer

A/D-Wandler mit Multiplexer am Port 7 (Klaus Muerling, 8702)

Nach A/D-Wandlung lassen sich Analogsignale über den Port 7 digitalisiert einspeisen und weiterverarbeiten. Zum Bau eines solchen A/D-Wandlers benötigt man nur wenige Bauteile. Ich verwendete den Baustein ZN 427.

Schaltbild:



Beschaltung des ZN 427 für einen Eingangsspannungsbereich von 0 V bis +10 V

Funktion: Auf ein Convert Command Signal (\overline{WR} low) vom Computer wird der ZN 427 reseriert, \overline{BUSY} low und die Clock aktiv. Nach Beenden der A/D-Wandlung wird \overline{BUSY} high, damit auch RD und der digitale Spannungswert liegt auf den Datenbits 1-8. Der Baustein 7413 wird als schaltbarer Oszillator verwendet.

Mehr Informationen über A/D-Wandler findet man in dem Buch "Hardwareerweiterungen für Z80-Rechner" von Martin Aschoff.

Um mehrere Analogkanäle mit nur einem Wandlerbaustein übernehmen zu können, ist ein Analogmultiplexer 4051 vorgeschaltet. Mit diesem Baustein lassen sich 8 Analogkanäle multiplexen. (16 Kanäle kann man mit einem 4067 multiplexen).

Ich verwende von Port 7 die Ausgänge out 5 zur Auswahl der Analogkanäle 0 und 1 und out 4 für das Convert Command. Das Programm zur Übernahme in Basic (2 Kanäle) ist sehr einfach:

```

40 OUT 7,0 ;Auswahl Kan.0,Low für Convert Command
50 OUT 7,16 ;High für Convert Command
60 LET A=INP(7) ;Übernahme Analogwert von Kan. 0 in A
70 OUT 7,32 ;Auswahl Kan.1,Low f. Conv. Command
80 OUT 7,48 ;High für Convert Command
90 LET B=INP(7) ;Übernahme Analogwert von Kan.1 in B
    
```

Bei der angegebenen Beschaltung des 7413-Oszillators beträgt die Frequenz 835 kHz und die Wandlung pro Kanal braucht etwa 11 μ S. Bei BASIC-Programmen ist die Wandlung bis zur nächsten Abfrage **keine** Rücksicht nehmen, oder \overline{BUSY} abfragen.

HARDWARE: ECB-Bus

Der ECB-Bus oder Wie man Fluchen lernt ? (Herbert Herberg, 2000)
 Da haben wir diese schöne Platine von Detlef Harms, und nach langem Suchen und Löten lief der MTX auch, wenn ich die Platine an den Bus stöpselte. Später fand ich dann noch heraus, das ein Eingriff in der RS 232-Karte unerlässlich ist. Nun war ich glücklich und zufrieden, da schließlich mein teures Programmiergerät für PALs und (E)PROMs funktionierte. Aber warum nicht die 1.5MByte-RAMDisc von EDICTA. Blindes Huhn trinkt auch mal Korn, und ich untersuchte mal den Bus mit dem Oszilloskop. Ich sah buchstäblich NICHTS-VOR-DURCHEINANDER!

Trotz Takteiler (zwecks Anpassung des Widerstandes/Impedanz) und herumspielen mit dem Trigger - es blieb WÜST. Bis dato hatte ich noch keine Rechteckschwingungen mit Rechteckigen Oberwellen gesehen - aber es gibt ja immer ein erstes mal. Jedenfalls konzentrierte ich meine Forschung auf das wichtigste Signal des Busses, das IOREQ, das durch ein Low-Pegel anzeigt, daß auf einen Port zugriffen werden soll. Wenn nämlich IOREQ loslegt, sind die anderen Leitungen stabil.

Beknackt wie ich bin, dachte ich, daß vielleicht ein 74S241 Treiber-IC anstelle des 74LS241 gut wäre. Dieses Treiber-IC ist notwendig, damit die Signale den langen Weg über die Leitung überstehen - ist also ein Leitungsverstärker (einfach ausgedrückt). NICHTS! Da ich nach dem Wenn-Schon-Denn-Schon Motto einkaufte steckte ich also einen 74HC241 in die Fassung (die Fassung habe ich eingelötet!). EI-GUCKE-DA, es wird doch DEUTLICHER. Keine Oberwellen, saubere Pegel - aber nicht genug, da eigentlich noch nichts lief.

In der Zeitschrift MC las ich etwas davon, daß 4MegaHertz ja Hochfrequenz sind, und also ich es mit HF-Leitungen zu tun habe. Das Antennenkabel des Fernsehers ist ein sog. 75Ohm Kabel, weil 75Ohm für HF das beste sind. Jedenfalls fand ich in MC einen Schaltplan für eine aktive Buserminierung - allerdings mit 220Ohm, da die Digital-ICs bei 75Ohm eingehen würden. Lange Rede (für Kenner verständlich), kurzer Unsinn: Ich habe also die unten gezeigte Schaltung zusammengelötet und auf den Bus gesteckt: **HURRA** Die Signale wurden schlagartig vernünftiger!!! Und wie ich das gemerkt habe? Meine c't-EPROM-Disc lief!

Danach habe ich den 74LS241, also das Original-IC in die Platine gesteckt, was den Output meines Oszillographen umgehend demolierte - aber die EPROM-Disc läuft trotzdem. Oh, eh ich es verschweige:

Die 12 Volt klaue ich mir aus dem Grundgerät am Stecker RS232--32k (genauer 512k) -Karten, die 5 Volt bekomme ich auf der ECB-Buskarte durch einen Festspannungsregler 7805 aus den 12 Volt. (Kühlkörper!).

Hier nun noch einmal der Überblick über meine Anordnung:

1. ECB-Busanschluß, Platine von Detlef Harms
2. RS 232 - Karte mit ECB-Option (s. Info 13, Seiten 3 & 4)
3. 12 Volt aus MTX, 5 Volt aus diesen 12 Volt mit IC 7805
4. Buserminierung nach Schaltplan nächste Seite
5. c't-EPROM-Floppy
6. Sonst keine Veränderungen (d.h. kein IC-Tausch auf RS 232)

---> LÄUFT!!!

ABER - ABER - LEIDER - LEIDER

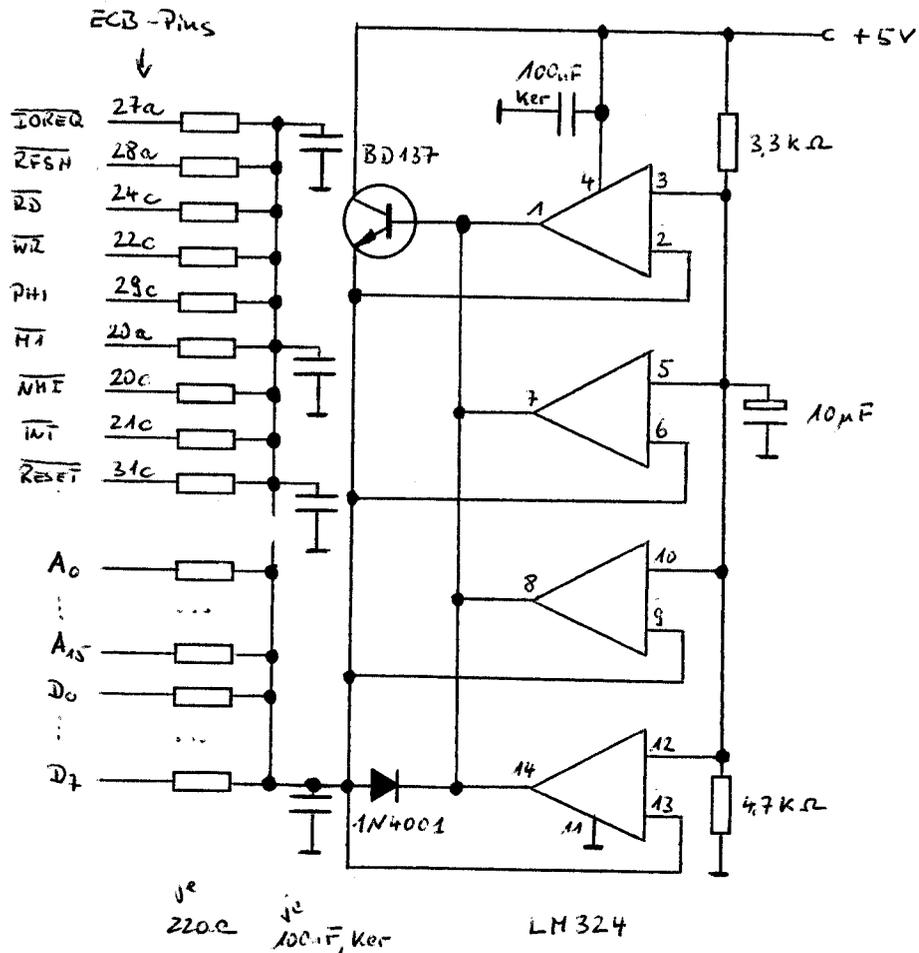
reicht das noch nicht, da meine 1.5M-Karte noch nicht will.

HARDWARE: ECB-Bus

Schaltplan Buserminierung

(Herbert Herberg, 2000)

Wichtig ist die Verwendung von keramischen Kondensatoren - bis auf den Kondensator mit 10 µF.



*D₀-D₇ sind auf Pins 2c, 14c, 4c, 4a, 5a, 2a, 3a, 3c
 A₀-A₁₅ --- 5c, 7c, 6a, 6c, 7a, 8a, 9a, 9c,
 8c, 30a, 18c, 17c, 27c, 29a, 18a, 28c*

Bei mir laufen:

(Herbert Herberg, 2000)

1. Mei Programmier für PALS und (E)PROMs einwandfrei!
2. Die c't-EPROM-Floppy.
3. Die EDICTA-Grafik-Karte (s.u.)

nicht laufen tut:

4. Die EDICTA-RAM-Floppy.

Hoffentlich im Info 17 die Auflösung. Für Tips bin ich jederzeit dankbar!

P.S. Die c't 1.0M-RAM-Floppy läuft (mit Buserminierung e.t.c. o.o.)

HARDWARE: ECB-Karten / RAM-Floppies**Folgende ECB-Ports sind vergeben:**

(Herbert Herberg, 2000)

80 - 83	EDICTA	Grafik-Karte	1024 x 1024 Pixel
98 - 9B	c't	RAM-Floppy	1 MByte
A0 - A3	EDICTA	RAM-Floppy	1.5 MByte
A4 - A7	c't	EPR0M-Floppy	256 kByte
CC - CF	Janich & Klass	Programmer	

Die EDICTA-Grafikkarte

(Herbert Herberg, 2000)

In der Grundaufüstung kann die Platine 1024 x 1024 Pixel monochrom produzieren. Da die meisten Monitore diese Auflösung nicht schaffen (benötigt wird eine Zeilenfrequenz von ca. 30 kHz - üblich sind 15 kHz und eine Grenzfrequenz von ca. 30 MHz) kann ein beliebiger Ausschnitt - je nach Können des Monitors z.B. 600 x 300 - auf den Monitor ausgegeben werden. Die Lage und Größe des Ausschnittes ist leicht und schnell zu verändern. Davon ist die Zeichenfläche von 1024 x 1024 Pixel nicht betroffen. Aufrüstbar ist die Karte z.B. auf 1024x1024 RGB.

Die Karte mit dem Prozessor 7220, 16 MHz zeichnet sich durch mächtige und schnelle Befehle aus: Vektor-Grafik, Kreissegmente, Kursivschrift und -zeichnungen,

Software dafür (Assembler/Turbo-Pascal) werde ich zusammen mit Olaf Krumnow schreiben.

Die Hardware direkt von EDICTA ist recht teuer - Dank der RAMs die die verwenden, aber ich habe etwas Gehirnschmalz verbraucht, und eine skurril aussehende aber deutlich billigere Lösung zusammengebastelt - UND GETESTET! Läuft!

Ulrich Hönisch und ich, wir wollen versuchen eine Bausatz und ein Fertigerät anzubieten. Der Bausatz kostet ca. DM 400.-. Interessenten möchten sich bitte bald an mich wenden, damit ich abschätzen kann, wer interessiert ist - und evtl. den Preis der Platine mit 3 PROMs drücken (DM 200.-)

RAM-Floppies

(Herbert Herberg, 2000)

Wo ist der Unterschied zwischen der RAM-Erweiterung im Grundegerät und den RAM-Floppies von EDICTA, C'T und Memotech ?

Die im Grundgerät untergrbrachten Erweiterungen liegen im Adreßraum der Z80-CPU, über Port 0 gepaged. Die CPU kann nur 64 kB adressieren, also wird größerer Speicher auf mehrere RAM-Seiten verteilt - wobei allerdings die oberen 16 kB immer die selben bleiben, damit sich ein Programm bei Seitenwechsel nicht selbst wegpagen muß. Die CPU kann darum auch Programme, die in diesem Speicher stehen direkt ausführen - solange diese entweder nur auf einer Seite stehen, oder selbst die Seiten umschalten. Das ist genau eines der Geheimnisse von RAM 3 und den Folgeversionen.

Die sog. RAM-Floppies hingegen werden über I/O-Ports angesprochen und sind deshalb dazu prädestiniert als RAM-Floppies zu dienen (was die internen Erweiterungen auch können), da sie RAM-Speicher liefern, ohne den Adreßraum der CPU - unsere popeligen 64kB - zu dezimieren. Ein dort stehendes Programm mußn erst in den Arbeitsspeicher geladen werden, bevor es ausgeführt werden kann.

BASIC: Korrektur der VAL-Routine

KORREKTUR der VAL-Routine aus Info 15

Andreas Viebke

Knapp daneben ist auch vorbei: Fast hätt' ich's noch geschafft, diesen Beitrag ins Info 15 zu kriegen - doch Herbert sonorte "Info 15 ist schon im Druck". Dann eben hier und jetzt.

```

LD HL,END
LD BC,VAL
PUSH BC
AND A
SBC HL,BC
EX (SP),HL
POP BC
LD DE,#5F9E
LDIR
RET
VAL: RST 28
DB #B9
LD (HL),#FF
INC HL
LD (#FA81),HL
CALL #10A6
PUSH DE
EX DE,HL
PUSH HL
CALL #58BB
POP HL
PUSH HL
CALL #599E
POP HL
PUSH HL
RST 28
DB #B6
POP DE
CALL Z,#5D84
CALL Z,#128F
POP HL
LD (#FA81),HL
RET
DW 0
END: RET

```

Die VAL-Routine vom letzten Mal ist zwar nicht so richtig falsch, aber ganz korrekt funktionieren tut sie auch nicht. Ich hatte vergessen, das Ergebnis vom Stack in den Accu zu bewegen, was dazu geführt hat, daß Ausdrücke aus einer einzigen Variablen (z.B. VAL("A")) nicht berechnet werden konnten. Hier die (diesmal hoffentlich) richtige Routine. (MTXler brauchen sich darum keine Gedanken zu machen, bei denen funktioniert alles richtig.)

Außerdem kann VAL jetzt keine Fehlermeldung mehr erzeugen. Das ist gut so, denn INPUT soll ja bei Syntax-Fehlern (z.B. nur <RET>) noch ein '?' ausgeben und nicht abbrechen. Ach ja: Habe ich etwa nicht gesagt, daß INPUT und READ von der VAL-Routine Gebrauch machen? Dadurch ergeben sich doch völlig neue Perspektiven!

Das wäre dann wohl BASIC 1.2. Wer will, kann diese Version und das neue NEWCOM kostenlos bei mir bekommen. (Die übliche Pauschale in Höhe von 8,- DM fällt für die Nicht-Einsender von D+P+V trotzdem an.)

BASIC: Fractals

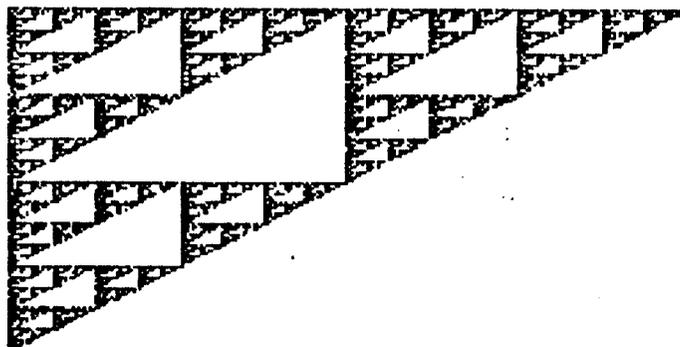
EINFACHE FRACTALS

von David Fisher und Christian Woehlbier

MUCS (Memotech User Club Schweiz)

```

10 VS 4: CLS : RAND -3535234
15 LET X=0: LET Y=0
20 LET R=RND
30 IF R>.995 THEN GOTO 200
40 IF R>.2 THEN GOTO 300
50 IF R>.12 THEN GOTO 500
60 IF R>.025 THEN GOTO 600
200 LET U=0: LET V=.16*Y
210 GOTO 1000
300 LET U=.849*X+.037*Y: LET V=-.037*X+.849*Y+1.6
310 GOTO 1000
500 LET U=.197*X-.257*Y: LET V=.226*X+.223*Y+1.6
510 GOTO 1000
600 LET U=-.15*X+.283*Y: LET V=.26*X+.238*Y+.44
1000 VS 4: PLOT V*25,U*25+80: LET X=U: LET Y=V: GOTO 20
    
```



```

4 VS 4: CLS : RAND -4535355
10 LET X=0: LET Y=0
20 LET R=INT(33RND)+1
30 IF R=1 THEN GOTO 200
40 IF R=2 THEN GOTO 300
50 IF R=3 THEN GOTO 500
60 GOTO 500
200 LET U=I/2: LET V=Y/2
210 GOTO 1000
300 LET U=I/2+.25: LET V=Y/2
310 GOTO 1000
500 REN
510 LET U=I/2+.25: LET V=Y/2+.5
520 GOTO 1000
1000 VS 4: PLOT V*25,U*25+40: LET X=U: LET Y=V: GOTO 20
    
```

BASIC: Fractals

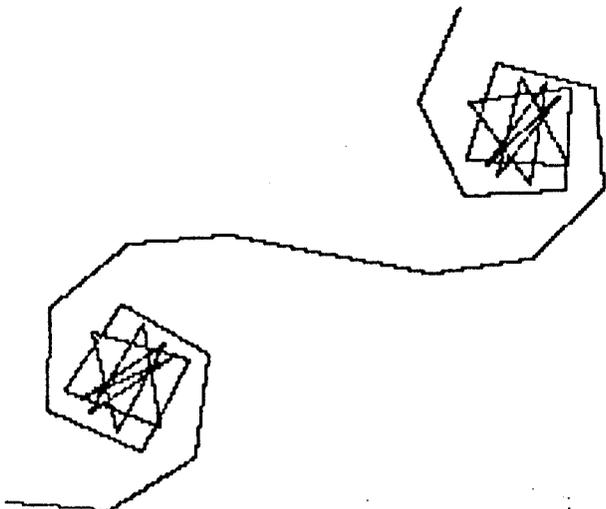
Fractal?

von Michael Kuenzli

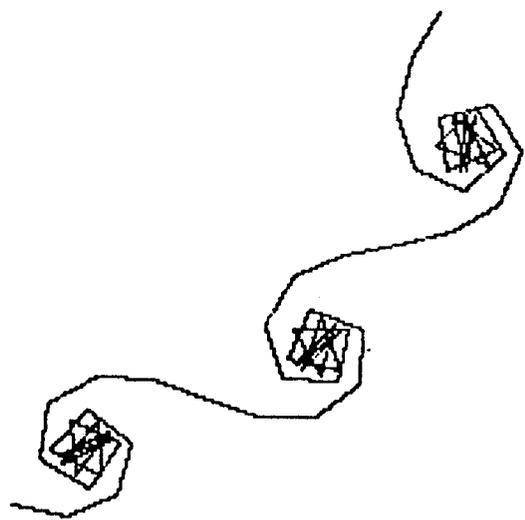
```

500 REM n.k. 3.8.86
600 INPUT "l, a, nx, ny ";L,A,NX,NY
710 VS 4: CLS
720 LET B=0: LET KV=0
740 PLOT NX,NY
760 LET B=B+.3: LET A=A+B: IF A>65000 THEN LET A=A
-10000*PI
780 ANGLE A
790 LET NX=NX+COS(A)*L: LET NY=NY+SIN(A)*L
795 IF NX>255 OR NX<0 OR NY<0 OR NY>191 THEN LET K
V=1: GOTO 820 ELSE IF KV=1 THEN PLOT NX,NY: LET KV
=0: GOTO 820
800 DRAW L: PLOT NX,NY: LET AX=NX: LET AY=NY: REM
*** Ver{nderbar
820 GOTO 760
999 REM
1000 REM HARDCOPY
1001 VS 4
1005 LPRINT CHR$(27);"A";CHR$(8)
1006 LPRINT CHR$(27);"P";CHR$(0)
1010 FOR Y=24 TO 1 STEP -1
1020 LPRINT CHR$(27);"K";CHR$(0);CHR$(1);
1030 FOR X=0 TO 255
1040 LET A#=GR$(X,8*Y-1,8)
1045 LPRINT A#;
1050 NEXT X
1055 LPRINT
1060 NEXT Y
    
```

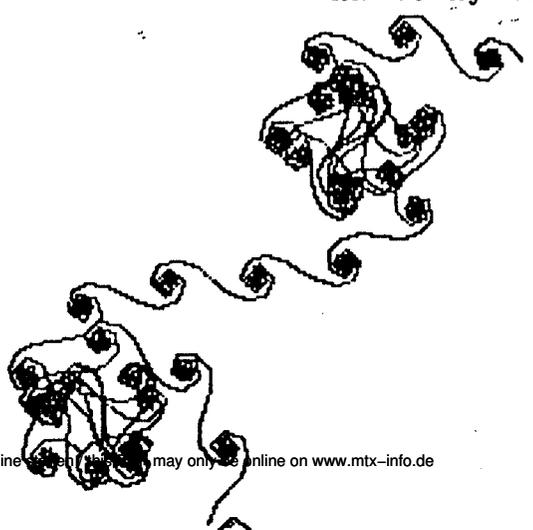
MUCS [Memotech User Club Schweiz]



Hardcopy 1: l=37 a=-0.37
nx=10 ny=10



Hardcopy 2: l=20 a=-0.5
nx=10 ny=10



Hardcopy 3: l=7 a=-1.2
nx=20 ny=20

BASIC: Das Spiel Light Riders

LIGHT RIDERS ist ein Spielprogramm für MTX 512/500 und ist im Konzept "TRON" aus Spielhallen sehr ähnlich. Du mußt verhindern die Lichtspur Deines Gegners zu treffen und nicht in die Hindernisse auf dem Spielfeld hineinlaufen.

Das Programm beinhaltet eine vollständige Bedienungsanleitung, und Du wirst Dich freuen festzustellen, daß es ein Zweipersonenspiel ist.

Gib das Programm ohne Zeile 40 ein, bis Du sicher bist, daß es läuft. Dann kannst Du Zeile 40 eingeben (für FDXB natürlich mit dem DISC SAVE -Befehl), Zeile 1000 neu assemblieren und RUN eingetippen. Dann hast Du ein Autostart-Programm.

MTX 500-Besitzer und FDXB-Benutzer geben das Programm genauso ein - lediglich das Maschinenprogramm wird bei 83D8 statt 43D8 beginnen, und alle Hex-Adressen der Zeile 1000 um 4000 (Hex) höher.

(Englischer MEMOPAD-Originaltext von Herbert Herberg übersetzt und ergänzt für FDX.)

MUCS [Memotech User Club Schweiz]

```

10 REM *****
20 REM ***** LIGHT RIDERS *****
30 REM *****
40 SAVE "Light-Riders"
100 VS 4: PAPER 4: COLOUR 4,4: INK 15: CLS
110 CSR 5,0: PRINT "L I G H T R I D E R S"
130 CSR 10,5: INK 7: PRINT "INSTRUCTIONS"
140 CSR 1,7: PRINT " Race your Light-Bikes around the Games Arena, avoiding your opponent, light trails and walls and obstacles."
150 PRINT : PRINT " LEFT PLAYER      RIGHT PLAYER"
160 PRINT "      use                use"
170 PRINT " Left joystick   Right joystick"
180 PRINT "      or                or"
190 PRINT "      Z,C,B,M        CURSOR KEYS"
200 PAPER 9: INK 4: CSR 11,18: PRINT "GOOD LUCK"
210 CSR 4,20: PRINT "Author Barry Young. '85"
220 GENPAT 3,1,0,16,40,255,40,16,0,0
230 GENPAT 3,2,0,8,20,255,20,8,0,0
240 GENPAT 3,3,16,16,56,84,56,16,16,16
250 GENPAT 3,4,16,16,16,56,84,56,16,16
260 GENPAT 1,129,129,66,153,60,165,90,36,60
270 GENPAT 1,130,127,191,223,239,247,251,253,254
280 GENPAT 1,131,254,253,251,247,239,223,191,127
290 GENPAT 1,132,255,255,255,255,255,255,255
300 INK 15: CSR 4,22: PRINT "PRESS ANY KEY TO START"
310 IF INKEY="" THEN GOTO 310
320 PAPER 4: INK 15: COLOUR 4,1: CLS
1000 CODE

4308 INIT:  RST 10
4309      DB 24C
430A      LD HL,MESS1
430D      LD DE,20106
43E0      CALL MESS
43E3      LD HL,MESS2
43E6      LD DE,20403
43E9      CALL MESS
43EC      LD HL,MESS3
43EF      LD DE,20704
43F2      CALL MESS
43F5      LD HL,MESS4
43F8      LD DE,20B04
43FB      CALL MESS
43FE      LD HL,MESS5
4401      LD DE,20904
4404      CALL MESS
4407      LD HL,MESS6
440A      LD DE,20A04
440D      CALL MESS
4410      LD HL,MESS7
4413      LD DE,20D04
4416      CALL MESS
4419      LD HL,MESS8
441C      LD DE,20F04
441F      CALL MESS
4422 SPRCL: LD A,2C0
4424      OUT (2),A
4426      LD A,2B1
4428      OUT (2),A
442A      LD DE,23F00
442D      LD A,E
442E      OUT (2),A
4430      LD A,D
4431      OR 64
4433      AND 127
4435      OUT (2),A
4437      LD B,128
4439 CLATT: XOR A
443A      CALL OUT
443B      DEC B
443E      LD A,B

```

Light Riders Barry Young

BASIC: Das Spiel Light Riders

443F	JR NZ,CLATT	44FO B5:	LD HL,BLOCK1	4599	LD (RX),A
4441	XOR A	44F3	LD DE,£0A0F	4593	JP POINTR
4442 SETSC:	LD (LSCORE),A	44F6	CALL MESS	4596 LR:	CP 2
4445	LD (RSCORE),A	44F9 B6:	LD HL,BLOCK2	4598	JR NZ,UR
444B INP:	LD A,£FE	44FC	LD DE,£0B0F	459A	LD A,(RI)
444A	OUT (S),A	44FF	CALL MESS	459D	DEC A
444C	IN A,(S)	4502 B7:	LD HL,BLOCK1	459E	LD (RX),A
444E INP1:	CP 254	4505	LD DE,£0F08	45A1	JP POINTR
4450	JR NZ,INP3	4508	CALL MESS	45A4 UR:	CP 3
4452	LD BC,2000	450B B8:	LD HL,BLOCK2	45A6	JR NZ,DR
4455	LD (SPEED),BC	450E	LD DE,£1008	45A8	LD A,(RY)
4459	JR GAME	4511	CALL MESS	45AB	INC A
445B INP3:	CP 253	4514 B9:	LD HL,BLOCK1	45AC	LD (RY),A
445D	JR NZ,INP5	4517	LD DE,£0F16	45AF	JP POINTR
445F	LD BC,500	451A	CALL MESS	45B2 DR:	CP 4
4462	LD (SPEED),BC	451D B10:	LD HL,BLOCK2	45B4	JR NZ,POINTR
4466	JR GAME	4520	LD DE,£1016	45B6	LD A,(RY)
446B INP5:	CP 251	4523	CALL MESS	45B9	DEC A
446A	JR NZ,INP2	4526 INITPOS:	LD A,32	45BA	LD (RY),A
446C	LD BC,2	4528	LD (LX),A	45BD POINTR:	LD A,(RX)
446F	LD (SPEED),BC	452B	LD A,R	45C0	LD (POINTX),A
4473	JR GAME	452D	ADD A,50	45C3	LD A,(RY)
4475 INP2:	LD A,253	452F	LD (LY),A	45C6	LD (POINTY),A
4477	OUT (S),A	4532	LD A,1	45C9	CALL POINT
4479	IN A,(S)	4534	LD (LDIR),A	45CC	CP 01
447B	CP 253	4537	LD A,223	45CE	JP Z,CRAHR
447D	JR NZ,INP4	4539	LD (RX),A	45D1	LD A,(RX)
447F	LD BC,1000	453C	HALT	45D4	LD (PLOTX),A
4482	LD (SPEED),BC	453D	LD A,R	45D7	LD A,(RY)
4486	JR GAME	453F	ADD A,50	45DA	LD (PLOTY),A
449B INP4:	CP 251	4541	LD (RY),A	45DD	CALL PLOT
448A	JR NZ,INP	4544	LD A,2	45E0	CALL RSPROUT
448C	LD BC,250	4546	LD (RDIR),A	45E3 GETLTJ:	LD A,127
448F	LD (SPEED),BC	4549	CALL LSPROUT	45E5	OUT (S),A
4493 GAME:	RST 10	454C	CALL RSPROUT	45E7	IN A,(S)
4494	DB £4C	454F	CALL CHEER	45E9 LEFTL:	CP 254
4495 BORDER:	RST 10	4552	CALL SOUNDON	45EB	JR NZ,RIGHTL
4496	DB £A5,2,0,16,255,16	4555 GETRTJ:	LD A,£F7	45ED	LD A,2
449C	DB £A5,2,255,16,255,183	4557 LEFTR:	CALL STROBE	45EF	LD (LDIR),A
44A2	DB £A5,2,255,183,0,183	455A	JR NZ,RIGHTR	45F2 RIGHTL:	CP 253
44A8	DB £85,2,0,183,0,16	455C	LD A,2	45F4	JR NZ,UPL
44AE CROWDT:	LD HL,CROWDD8	455E	LD (RDIR),A	45F6	LD A,1
44B1	LD DE,£0000	4561 RIGHTR:	LD A,£EF	45FB	LD (LDIR),A
44B4	CALL MESS	4563	CALL STROBE	45FB UPL:	CP 251
44B7 CROWD6:	LD HL,CROWDD8	4566	JR NZ,UPR	45FD	JR NZ,DOWNL
44BA	LD DE,£1600	4568	LD A,1	45FF	LD A,3
44BD	CALL MESS	456A	LD (RDIR),A	4601	LD (LDIR),A
44C0	LD HL,PLAYERS	456D UPR:	LD A,£FB	4604 DOWNL:	CP 247
44C3	LD DE,£1700	456F	CALL STROBE	4606	JR NZ,TESTL
44C6	CALL MESS	4572	JR NZ,DOWNR	4608	LD A,4
44C9	CALL PRSCR	4574	LD A,3	460A	LD (LDIR),A
44CC BLOCKS:	LD HL,BLOCK1	4576	LD (RDIR),A	460B TESTL:	LD A,(LDIR)
44CF	LD DE,£0508	4579 DOWNR:	LD A,£BF	4610	CP 1
44D2	CALL MESS	457B	CALL STROBE	4612	JR NZ,LL
44D5 B2:	LD HL,BLOCK2	457E	JR NZ,TESTR	4614	LD A,(LI)
44D8	LD DE,£0608	4580	LD A,4	4617	INC A
44DB	CALL MESS	4582	LD (RDIR),A	4618	LD (LI),A
44DE B3:	LD HL,BLOCK1	4585 TESTR:	LD A,(RDIR)	461B	JP POINTL
44E1	LD DE,£0516	4588	CP 1	461E LL:	CP 2
44E4	CALL MESS	458A	JR NZ,LR	4620	JR NZ,UL
44E7 B4:	LD HL,BLOCK2	458C	LD A,(RI)	4622	LD A,(LI)
44EA	LD DE,£0616	458F	INC A	4625	DEC A
44ED	CALL MESS			4626	LD (LI),A

BASIC: Das Spiel Light Riders

4629	JP POINTL	4680	LD (LDISP),A	4742	CALL NORMAL	4782	CALL ZBF6
462C UL:	CP 3	4683	LD DE,£170C	4745	CALL SOUNDOFF	478E	RET
462E	JR NZ,DL	4686	LD HL,LDISP	4748	RET	47BF	SOUNDOFF:LD B,2
4630	LD A,(LY)	4689	CALL MESS	4749 ROW:	LD A,L	47C1	QUIET: LD A,B
4633	INC A	468C	LD A,(RSCORE)	474A	OUT (2),A	47C2	LD (£FE14),A
4634	LD (LY),A	468F	CP 10	474C	LD A,H	47C5	LD HL,0000
4637	JP POINTL	46C1	JP Z,RIGHTWIN	474D	OR 64	47CB	LD (£FE16),HL
463A DL:	CP 4	46C4	ADD A,48	474F	AND 127	47CB	LD (£FE18),HL
463C	JR NZ,POINTL	46C6	LD (RDISP),A	4751	OUT (2),A	47CE	PUSH BC
463E	LD A,(LY)	46C9	LD DE,£1713	4753	LD B,255	47CF	CALL ZBF6
4641	DEC A	46CC	LD HL,RDISP	4755 VOL:	LD A,R	47D2	POP BC
4642	LD (LY),A	46CF	CALL MESS	4757	SLA A	47D3	DEC B
4645 POINTL:	LD A,(LX)	46D2	PET	4759	SLA A	47D4	LD A,B
464E	LD (POINTL),A	46D3 LSPROUT:	LD DE,£3F00	475D	SLA A	47D5	CP 255
464B	LD A,(LY)	46D6	LD A,E	475D	SLA A	47D7	JR NZ,QUIET
464E	LD (POINTY),A	46D7	OUT (2),A	475F	ADD A,4	47D9	RET
4651	CALL POINT	46D9	LD A,B	4761	CALL OUT	47DA	POINT: RST 10
4654	CP 1	46DA	OR 64	4764	DEC B	47DB	DB £85,27,67
4656	JP Z,CRASHL	46DC	AND 127	4765	LD A,B	47DE	POINTX: DB 0
4659	LD A,(LX)	46DE	OUT (2),A	47E6	JR NZ,VOL	47DF	POINTY: DB 0
465C	LD (PLOTX),A	46E0	LD A,(LY)	47E9	RET	47E0	DB 1
465F	LD A,(LY)	46E3	CPL	47E9	NORMAL: LD A,L	47E1	LD A,(£FE1A)
4662	LD (PLOTY),A	46E4	SUB 68	47EA	OUT (2),A	47E4	RET
4665	CALL PLOT	46E6	CALL OUT	47EC	LD A,H	47E5	PLOT: RST 10
4668	CALL LSPROUT	46E9	LD A,(LX)	47ED	OR 64	47E6	DB £83,1
466B PAUSIN:	CALL £0079	46EC	SUB 3	47EF	AND 127	47E8	PLOTX: DB 0
466E	CP 13	46EE	CALL OUT	4771	OUT (2),A	47E9	PLOTY: DB 0
4670	JR NZ,DELAY	46F1	LD A,(LDIR)	4773	LD B,B	47EA	RET
4672 PAUSE:	HALT	46F4	CALL OUT	4775	NORM1: LD C,32	47EB	CRASHR: CALL SOUNDOFF
4673	HALT	46F7	LD A,B	4777	LD HL,FANCOL	47EE	CALL BANG
4674	HALT	46F9	CALL OUT	477A	NORM2: LD A,(HL)	47F1	CALL SOUNDOFF
4675	HALT	46FC	RET	477B	SLA A	47F4	CALL CHEER
4676	CALL £0079	46FD RSPROUT:	LD DE,£1F04	477D	SLA A	47F7	LD A,(LSCORE)
4679	CP 13	4700	LD A,E	477F	SLA A	47FA	INC A
467B	JR NZ,PAUSE	4701	OUT (2),A	4781	SLA A	47FB	CP 10
467D DELAY:	LD BC,(SPEED)	4703	LD A,D	4783	ADD A,4	47FD	JP Z,LEFTWIN
4681 DELOOP:	DEC BC	4704	OR 64	4785	CALL OUT	4800	LD (LSCORE),A
4682	XOR (IX+0)	4706	AND 127	47E8	INC HL	4803	JP GAME
4685	LD A,B	4708	OUT (2),A	4789	DEC C	4806	CRASHL: CALL SOUNDOFF
4686	OR C	470A	LD A,(RY)	478A	LD A,C	4809	CALL BANG
4687	JR NZ,DELOOP	470D	CPL	478B	JR NZ,NORM2	480C	CALL SOUNDOFF
4689	BACK: JP GETRTJ	470E	SUB 68	478D	DEC B	480F	CALL CHEER
468C	MESS: LD (COORD),DE	4710	CALL OUT	478E	LD A,B	4812	LD A,(RSCORE)
4690	RST 10	4713	LD A,(RX)	478F	JR NZ,NORM1	4815	INC A
4691	DB £83,3	4716	SUB 3	4791	RET	4816	CP 10
4693	COORD: DW 0000	4718	CALL OUT	4792	STROBE: OUT (5),A	4818	JP Z,RIGHTWIN
4695	NLOOP: LD A,(HL)	4718	LD A,(RDIR)	4794	IN A,(5)	481B	LD (RSCORE),A
4695	CP 0	471E	CALL OUT	4796	CP 127	481E	JP GAME
4698	RET Z	4721	LD A,11	4798	RET	4821	LEFTWIN: RST 10
4699	LD (CHAR),A	4723	CALL OUT	4799	SOUNDON: XOR A	4822	DB £4C
469C	RST 10	4726	RET	479A	LD (£FE14),A	4823	LD HL,LWINNESS
469D	DB £81	4727	CHEER: LD D,150	479D	LD HL,500	4826	LD DE,£0A09
469E	CHAR: DB 0	4729	UNFIN: LD HL,£2000	47A0	LD (£FE16),HL	4829	CALL MESS
469F	INC HL	472C	CALL POW	47A3	LD A,200	482C	LD HL,STARS
46A0	JR NLOOP	472F	LD HL,£3600	47A5	LD (£FE18),A	482F	LD DE,£0809
46A2	PET	4732	CALL ROW	47A8	CALL ZBF6	4832	CALL MESS
46A3	OUT (1),A	4735	DEC B	47AB	LD A,1	4835	LD HL,CROWOOD
46A5	RET	4736	LD A,B	47AD	LD (£FE14),A	4838	LD DE,0000
46A6	PRSCR: LD A,(LSCORE)	4737	JR NZ,UNFIN	47B0	LD HL,750	483D	CALL MESS
46A9	CP 10	4739	PLAIN: LD HL,£2000	47B3	LD (£FE16),HL	483E	LD HL,CROWOOD
46AB	JP Z,LEFTWIN	473C	CALL NORMAL	47B6	LD A,200	4841	LD DE,£1600
46AE	ADD A,48	473F	LD HL,£3600	47B8	LD (£FE18),A	4844	CALL MESS

BASIC: Das Spiel Light Riders

```

4847 CALL CHEER
484A CALL SOUNDOFF
484D JP INIT
4850 RET
4851 RIGHTWIN:RST 10
4852 DB 44C
4853 LD HL,RWINMESS
4856 LD DE,20A09
4859 CALL MESS
485C LD HL,STARS
485F LD DE,20B09
4862 CALL MESS
4865 LD HL,CROWDOB
4868 LD DE,0000
486B CALL MESS
486E LD HL,CROWDOB
4871 LD DE,21600
4874 CALL MESS
4877 CALL CHEER
487A CALL SOUNDOFF
487D JP INIT
4880 BANG: LD BC,1500
4883 REPT: LD A,0
4885 LD (EFE14),A
4888 LD A,R
488A LD H,A
488D LD L,A
488C LD (EFE16),HL
488F LD A,R
4891 LD (EFE18),A
4894 PUSH BC
4895 CALL 28F6
4898 POP BC
4899 DEC BC
489A LD A,B
489B OR C
489C JR NZ,REPT
489E RET
489F NOP
48A0 NOP
48A1 NOP
48A2 NOP
48A3 NOP
48A4 NOP
48A5 MESS1: DB "L I G H T R I D E R S",0
48B0 MESS2: DB "Please choose the Race Speed.",0
48DA MESS3: DB "1. Easy.",0
48E3 MESS4: DB "2. More Difficult.",0
48F6 MESS5: DB "3. Getting interesting.",0
490E MESS6: DB "4. Masochists only.",0
4922 MESS7: DB "5. Forget it!!!",0
4932 MESS8: DB "Use RETURN to pause game.",0
494C LSCORE: DB 0
494D RSCORE: DB 0
494E SPEED: DW 0000
4950 CROWDOB: DB 129,129,129,129,129,129,129,129
4958 DB 129,129,129,129,129,129,129,129
4960 DB 129,129,129,129,129,129,129,129
4968 DB 129,129,129,129,129,129,129,129
4970 DB 0
4971 PLAYERS:DB "$$ PLAYER 1) $$$$ 0<PLAYER 2 $$",0
4992 BLOCK1: DB 130,131,0
4995 BLOCK2: DB 131,130,0
    
```

```

4998 LX: DB 0
4999 RX: DB 0
499A LY: DB 0
499B RY: DB 0
499C LDIR: DB 0
499D RDIR: DB 0
499E LDISP: DW 0000
49A0 RDISP: DW 0000
49A2 FANCOL: DB 9,9,9,9,9,2,2,2
49AA DB 9,9,9,9,9,8,8,8
49B2 DB 9,9,9,9,9,7,7,7
49BA DB 9,9,9,9,9,13,13,13
49C2 LWINNESS:DB "PLAYER 1 WINS!",0
49D1 RWINNESS:DB "PLAYER 2 WINS!",0
49E0 STARS: DB "*****",0
49EF RET
49F0 RET
    
```

Symbols:

```

INIT 43D8 MESS1 48A5
MESS 46BC MESS2 48BC
MESS3 46DA MESS4 48E2
MESS5 48F6 MESS6 490E
MESS7 4922 MESS8 4932
SPRCL 4422 CLATT 4439
OUT 46A3 LSCOR1 494C
SETSC 4442 RSCOR1 494D
INP 444B INP1 444E
INP3 445B SPEED 494E
INP5 446B INP2 4475
INP4 449B GAME 4493
BORDB 4495 CROWDOB 4950
CROWDT 44AE CROWDB 44B7
PLAYERS 4971 PRSCR 46A6
BLOCK1 4992 BLOCKS 44CC
BLOCK2 4995 B2 44D5
B3 44DE B4 44E7
B5 44F0 B6 44F9
B7 4502 B8 450B
B9 4514 B10 451D
INITPOS 4526 LX 4998
LDIR 499C RX 4999
RDIR 499D LSPROUT 46D3
LSPROUT 46F0 CHEER 4727
SOUNDOB 4799 GETRTJ 4555
STROBE 4792 LEFTR 4557
PIGHTR 4561 UPR 456D
DOWNR 4579 TESTR 45B5
LR 4596 POINTR 45B0
UR 45A4 DR 45B2
RY 499B POINTX 47DE
POINTY 47DF POINT 47DA
CRASHR 47EB FLOTI 47EB
FLOTY 47E9 PLOT 47E5
GETLTJ 45E3 LEFTL 45E9
RIGHTL 45F2 UPL 45FB
DOWNL 4604 TESTL 460D
LL 461E POINTL 4645
UL 462C DL 463A
LY 499A CRASHL 4806
FAUSH 466B DELAY 467D
FAUSE 4672 DELOOP 46B1
PACZ 4699 COCFD 4693
    
```

```

MLOOP 4695 CHAR 469E
LEFTWIN 4821 LDISP 499E
RIGHTWIN 4851 RDISP 49A0
UNFIN 4729 ROW 4749
PLAIN 4739 NORMAL 4769
SOUNDOFF 47BF VOL 4755
NORM1 4775 FANCOL 49A2
NORM2 477A QUIET 47C1
BANG 4880 LWINNESS 49C2
STARS 49E0 RWINNESS 49D1
REPT 48B3
    
```

SuperCalc: Kurs Teil 2 von Wolfgang Gieger

Kurs-02.cal Fortsetzung Wolfsburg den, 28.07.86

Damit es nicht allzu langweilig wird wollen wir an dieser Stelle eine kleine Aufgabe als Beispiel loesen. Dieses Beispiel speichern wir dann auch auf Diskette ab, damit wir es spaeter noch einmal heranziehen koennen. Nun aber zur Aufgabe, wir geben folgendes ein:

Die Eingaben genauso abtippen, dazu benoetigen wir einen leeren Formatbogen. (Der / ist der Schraegstrich, und nicht RETURN!)

Eingabe: /F G \$ RET dann: /F C A , 12 RET dann: /F G 10 RET

Eingabe: /F R 1 , TR RET dann: /F R 2 , TR RET

Den Cursor nach Zelle A3 bringen und die Eingabe: '- RET

Den Cursor nach Zelle A12 bringen und die Eingabe: '= RET

Nun folgen Formeln, die wir in die Zellen eingeben.

Eingabe in Zelle E4: G4+H4 RET

Eingabe in Zelle F4: SUM(B4:E4) RET

Eingabe: /R E4:F4 , E5:E11 RET

Eingabe in Zelle B13:SUM(B3:B12) RET

Eingabe: /R B13 , C13:F13 RET

Alle Eingaben hinter dem Doppelpunkt ; werden in der Statuszeile hinter der 1>.....also hier eingegeben!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 Wo nur Eingabe steht, also keine Zellenangabe dahinter, kann der Cursor in der letzten Zellenposition stehen bleiben!!!!!!!!!!!!!!!!!!!!!!
 Sonst wird der Cursor zur neuen Zellenposition mit den Pfeiltasten auf der Tastatur gebracht, oder mit CTRL + E,S,D,und X !!!!!!!!!!!!!!!!!!!!!!!

Wir schreiben nun den Text und die Zahlenwerte aus folgender Tabelle ab,d.h. wir uebernehmen sie in den Arbeitsbogen.

!	B !!	C !!	D !!	E !!	F !!	G !!	H !
1!	Neue	Mehr als	Mehr als	Mehr als	Gesamt-	Arb.Ber.	Alt 90
2!	Forderg.	30 Tage	60 Tage	90 Tage	Forderg.	alt 60	Tage
3!							
4!	100.00	200.55	300.33	400.40	1001.28		400.40
5!	100.30	200.00	300.05	.00	600.35		.00
6!	300.00	400.45	100.00	.00	800.45		.00
7!	285.40	600.00	30.35	.00	915.75		.00
8!				.00	.00		.00
9!				.00	.00		.00
10!				.00	.00		.00
11!				.00	.00		.00
12!	=====						
13!	785.70	1401.00	730.73	400.40	3317.83		

In den Zellen **A4 bis A11** schreiben wir den Text :

"Kunde Nr.1 bis Kunde Nr.8

Aus Platzgruenden hier auf dem Papier werden die Spaltenreihe A nicht aufgefuehrt.

Jetzt speichern wir den Arbeits-Bogen auf die Diskette ab mit folgender

Eingabe: /S TEST RET

und verlassen SC. mit der Eingabe: /Q Y

SuperCalc: Kurs Teil 2 von Wolfgang Gieger

Um eine solche Tabelle zu bearbeiten, um auf dem laufenden zu sein bedienen wir uns einer automatischen Funktion. Diese Funktion kann aber auch manuell, also per Hand von der Tastatur eingegeben werden, was aber manchmal sehr Aufwendig sein kann.

Dazu Schreiben wir in Newword eine Datei mit den Namen (MONAT-1.XQT) im NON-Documenten Modus.

Die sieht so aus:

```

/LTEST, A
/B G4:G11
/C D4:E11, G4, V
/B D4:D11
/C C4:C11, D4
/B C4:C11
/C B4:B11, C4
/B B4:B11
    
```

Wir kopieren uns die Datei MONAT-1.XQT auf unsere Arbeits-diskette, wo sich unser Supercalc befindet.

So und geben wir folgendes ein: SC MONAT-1 und RET

Den folgenden Ablauf nun nicht unterbrechen, sondern sehen wir uns den Ablauf in Ruhe an und warten auf den Stillstand, d.h bis das Laufwerk nicht mehr anzeigt, also steht.

Wir sehen dann ein neues folgendes Bild auf dem Monitor:

!	B !!	C !!	D !!	E !!	F !!	G !!	H !
	Neue Forderung.	Mehr als 30 Tage	Mehr als 60 Tage	Mehr als 90 Tage	Gesamt-Forderung.	Arb.Ber. alt 60	Alt 90 Tage
		100.00	200.55	700.73	1001.28	300.33	400.40
		100.30	200.00	300.05	600.35	300.05	.00
		300.00	400.45	100.00	800.45	100.00	.00
		285.40	600.00	30.35	915.75	30.35	.00
				.00	.00		.00
				.00	.00		.00
				.00	.00		.00
				.00	.00		.00
=====							
	.00	785.70	1401.00	1131.13	3317.83		

Wir haben nun die Bearbeitung einer Tabelle (automatisch) gesehen und die Tabelle auf den neusten Stand gebracht.

Wenn auf ihrem Bildschirm, die Zahlen genauso berechnet wurden mit neuen Ergebnissen, dann waren die Eingaben alle richtig und es koennen nun in den Zellen B4 bis B11 neue Zahlenwerte eingegeben werden.

Das nur als kleines Beispiel, um sehen zu koennen wie STARK supercalc ist und wie umfangreich. Erklarungen kommen spaeter.

SuperCalc: Kurs Teil 2 von Wolfgang Gieger

Wir kommen jetzt zu den Rechenoperationen und Rechenfunktionen

1. Rechenoperation:

- + = Addition
- = Subtraktion
- * = Multiplikation
- / = Division
- ^ = Potenzieren

Hier gelten die ueblichen Rechenregeln:

- von links nach rechts
- Multiplikation und Division vor Addition und Subtraktion
- Klammerausdruecke sind moeglich

2. Rechfunktionen (Auswahl)

- ABS (Wert) = ermittelt den absoluten Wert
- Average (Liste) = ermittelt das arithmetische Mittel
- EXP (Wert) = gibt die 'e'-Potenz an
- INT (Wert) = gibt den Wert ganzzahlig ohne Runden an
- LN (Wert) = gibt den natuerlichen Logarithmus an
- LOG10 (Wert) = gibt den 10er Logarithmus an
- MAX (Liste) = gibt den maximalen Wert an
- MIN (Liste) = gibt den minimalen Wert an
- PI = verwendet PI mit 16 Stellen Genauigkeit
- SIN (Wert) = berechnet trigonometrischen Wert
- TAN (Wert) = " " "
- ASIN (Wert) = " " "
- ATAN (Wert) = " " "
- COS (Wert) = " " "
- ACOS (Wert) = " " "
- SQRT (Wert) = berechnet Quadratwurzel
- SUM (Liste) = addiert Bereich

Mit der jeweiligen Eingabe:

- + = bewirken wir eine Addition von zwei Werten, die auch aus Zelleninhalten bestehen koennen.Z.b:(7+9 oder A1+C3)
- = veranlassen wir die Subtraktion der Werte, die auch aus Zelleninhalten bestehen koennen.Z.b:(23-10 oder B13-F3)
- * = loesen wir die Multiplikation von Werten, Ausdruecken oder Zelleninhalten aus.Z.b:(12*34 oder D2*f44)
- / = loesen wir die Division von Werten, Ausdruecken oder Zelleninhalten aus.Z.b:(34/12 oder E4:G44)
- ^ = veranlassen wir das Potenzieren eines Wertes, der auch aus einer anderen Zelle entnommen werden kann.Z.b:(^5 oder ^E4)

SuperCalc: Kurs Teil 2 / LeserbriefeEINGABE: /B und RET

Diese Eingabe ruft den Loeschvorgang. Wir muessen dann den Bereich der geloeschet werden soll eingeben.

Wir haben hier die Moeglichkeit einzelne Zellen, Zeilen, Spalten, oder ganze Bloecke zu loeschen:

Zelle : G12

Zeile : B13:G13 -der Bereich kann wahllos sein (A-BK), die Zeile muß 13 sein. (also von A13 bis BK13, hier: B13 bis G13)

Spalte: G2:G17 -hier ist genau umgekehrt (Die Buchstaben Gleich)

Block : D2:G9 -hier wird alles von D2 bis G9 geloeschet (Block)

Der Loeschvorgang bezieht sich jedoch nur auf die Zelleninhalte, wie Formeln, Zahlen, oder Ergebnisse, D.h: das vorher erstellte Format bleibt auf jeden Fall erhalten.

Bei geschuezten Zellen ist der Blank-Befehl nicht wirksam !!!, man kann also bestimmte Bereiche auch verblocken, also dort wo wichtige Formeln stehen, damit sie nicht ueberschrieben werden und somit verloren gehen. Mehr ist zu dem Blank-Befehl nicht zu sagen.

Weiter gehts im Kurs-03: viel Spass!

W.Gieger

Hartmut Traber, 5270 Gummersbach schrieb mir:

1. Nachträcken von Farbbändern:

Nach dem Verbrauch des ersten Farbbandes für meinen DMX 80 und Inbetriebnahme des zweiten habe ich versuchsweise das verbrauchte, also zum zweiten Mal verbrauchte (daß das Farbband nachgrträckt werden kann, sollte jeder wissen) Farbband mit gewöhnlicher Stempelkissenfarbe sukzessive unter Weiterdrehen grträckt. Dann eine Weile liegen gelassen, jetzt schreibt es wieder.

(o. Garantie!)

Anm.d.HH. Toll, leider hat sich bei mir jetzt ein Farbband durchgeschrieben.

2. Alma Gelbe Mater (DBP)

Sie verkauft Schrott ! Was ist Schrott ?

Alles was noch betriebsfähig ist, nicht angeschlossen wertden darf, aber für MTX'ler höchst brauchbar ist!

Z.B. kann man beim FZZA Elmshorn alte Modems zum Stückpreis von DM 1,- erhalten. (FZZA = Fernmeldezentralzeugamt)

Ganz außerordentlich geeignet sind D 1200S, aber besonders DAG 2400M. Ein Gehäuse, das zur Aufnahme aller Europakarten und des ECB-Busses tauglich ist. Ein Netzteil mit +/-12 V, +5V. Keine mechanischen Probleme mehr! Nur alles rauswerfen, was drin war. Die bauteile kann man natürlich verwenden, sicher alle einwandfrei.

Na, was ist das ? Andere Sachen gibt's da auch, nur mal anfragen!

Dür DM 1,- bis, was weiß ich, DM 3,- ein konkurrenzloses Angebot!

Es wird nur nach Gewicht abgerechnet, Versandkosten kommen hinzu.

(D 1200S ist ein geschlossenes Alu-Gehäuse, z.B. für Floppy einbau ideal; DAG 2400M-Gehäuse ist auch aus Metall, aber rundherum perforiert - damit für Mega-Cards u.ä. geeignet. Beide enthalten Schienen für Europakarten, SPITZE!)

Leserbriefe

An Herbert Herberg
Sonnensau 2
2000 Hamburg 76

Hallo Herbert und alle anderen MTX-User !

Mit Tips und Tricks bzw. einer kleinen Maschinenroutine kann ich diesmal infolge eines inzwischen behobenen Defekts an meinem Massenspeicher leider nicht aufwarten, jedoch hat sich mittlerweile genügend "Material" angesammelt, das ich jetzt endlich einmal loswerden möchte.

Zunächst einmal ein dickes Lob an Herbert Oppmann, der mit den kommentierten SAVE/LOAD - Routinen endlich einmal damit angefangen hat, den ROM zu analysieren (zu fast jedem Computer gibt's inzwischen ein Data-Becker-Buch "sowieso - Intern", nur nicht für unsere Kiste). Ich hoffe sehr, daß dies der Anfang einer Serie über den MTX-ROM war.

Kritik: Bekanntlich besteht bei der Bestellung von Programmen immer noch die Möglichkeit, eine Leercassette zu schicken. Das sieht dann u.U. so aus: Als Warensendung deklariert kostet das Porto für die Leercassette DM 1,10 . So kommt es, daß der Preis einer Datencassette manchmal unter den Portokosten liegt. Ich z.B. nutze günstige Gelegenheiten und kaufe des öfteren Cassetten für DM 1,- ein! Herr Schwarz-Schilling (unser werter Postminister) freut sich!

Mein Vorschlag: Schafft den Unsinn ab und paßt den Preis für Porto, Datencassette/Discette, Verpackung der Realität etwas an. M.E. müßten DM 6,- eigentlich ausreichen.

Gag: Nachdem mir in den letzten Wochen bereits jemand freundlicherweise eine Discette zugesand hat, warte ich nur noch auf zugehörige Floppy, aber vielleicht schickt mir demnächst ja irgendwer im Tausch gegen Programme eine Discettenbox? Haben sich genügend Discettenutensilien bei mir angesammelt, dann wird der Kauf einer Floppy schon zum Sachzwang!

Anregung: Ich wünsche mir mehr Programmbesprechungen (-kritiken), insbesondere was die kommerziell angebotene Software betrifft. Was taugt z.B. das Schachspiel, das K.H. Harter anbietet (in Info 2 jedenfalls, wie es im Inhaltsverzeichnis steht, gibt es keine Besprechung von Schachprogrammen. "Schachprogramme...im Meinungsstreit" vermute ich in Info 3)? Es muß doch Leute geben, die das Programm gekauft haben!

Ich habe bis jetzt davon zurückgeschreckt, dort (bei K.H.H.) einzukaufen, da ich in dem Angebot einige Programme vorfand, die mir bekannt sind und die ich nicht allzu gut finde, die dennoch keineswegs billig sind. Ich denke da z.B. an "Star-Command", ein recht stupidies Kriegsspiel, bei dem man zudem noch Schwierigkeiten hat, sich dort hineinzufinden. "Kilopede" ist zwar weitaus packender (manch Kind wird davon süchtig), aber ebenso gewaltverherrlichend.

Narrung: Bei jeder Bestellung von Programmen, die in letzter Zeit bei mir eingingen, war Bargeld dabei! Die Auftraggeber, deren Geld beim Posttransport nicht abhanden kam, sollten sich beim lieben Gott bedanken! Daß der Geldtransfer per Post ein risikoreicher - und dank den Vorzügen des modernen bargeldlosen Zahlungsverkehrs darüber hinaus ein völlig überflüssiger - Weg ist, zeigte sich wieder einmal bei der Sendung von St. Ch. aus M. (er soll sich bitte bei mir melden). Hier kam noch als risikoerhöhendes Moment hinzu, daß die Sendung als Warensendung deklariert und offen (!) war. Was Wunder, daß ich vom Geld nichts mehr gesehen habe. Tja, man kann eben nicht erwarten, daß bei der Post lauter Engel arbeiten. Vielleicht könnte manch einer von uns bei einer solchen Gelegenheit nicht widerstehen?!

Tschüß

Thomas

Thomas Mäurer
Duisburger Str. 296
4200 Oberhausen 1

Leserbriefe

Ich will auch mal "Leserbriefe" schreiben

Zu VIEW.COM

Ich hoffe, jeder kennt inzwischen VIEW2.COM. Besonders die Vielschreiber und Textveränderer werden dieses PD-Utility bald sehr zu schätzen wissen! Mit sehr ausführlichem DOC.

Zu den "merkwürdigen Typenradmaschinen"

Ich habe das dumpfe Gefühl, daß mit Herberts "Filter" was nicht stimmt. NORM, CODE1 und CODE2 sind doch absolute Werte, die sich auf den Programmanfang 100H beziehen - bei LD HL,NORM und anschließendem LD (HL),A landet der Inhalt von A mitten im laufenden Programm. Wenn ich nicht irre, dann wäre es sinnvoller gewesen, das zwar umständlichere - aber funktionierende - Original von Uwe Grass abzudrucken. Siehe meine Version für die Olympia electronic compact 2 (ec2).

Zu Olaf Krumnows Zeitrechnung

Ich bin ganz und gar nicht seiner Meinung, und werde die bei mir bestellte Software weiterhin am Tage nach Erhalt der Bestellung und des Schecks abschicken. Ich hab' das Vertrauen in die Clubmitglieder, daß der auf dem Scheck ausgewiesene Betrag auch tatsächlich auf dem Konto ist. Und ich bin noch nicht enttäuscht worden.

In Bezug auf die manchmal vergessene Pauschale für Disk, Porto und Verpackung gebe ich ihm Recht. Doch da sollte man auf der Disk einen Brief mit Hinweis hinterlassen, anstatt wie eine Behörde zu reagieren. Niemand wird den Hinweis ignorieren, wir sind doch ein Club. Gibt's denn nichts mehr auf der Welt, woran man glauben kann?

Zu XBASIC für SDX

Da HedeB (Herbert, der Beschäftigte) meine Leserbriefe auf Diskette ignoriert - neiiiin, bitte keine Anm.d.HH. hierher... -, muß ich ein paar Sachen an dieser Stelle loswerden: Hauke Ahrensfield fragte, ob es XBASIC auch für die SDX gäbe (ich schreibe "die SDX", weil ich dabei immer noch an die Diskettenstation denken muß, anstatt an das Diskettensystem):

Dazu kann ich zwei, nein drei Dinge sagen: 1. Ich hab' kein 80-Spur-Laufwerk und weiß nicht, ob man mit der SDX auch 40-Spur-Disks bearbeiten kann. (Ich lese die Sachen über die SDX nicht.) 2. Wenn SDXB (oder wie das BASIC da heißt) genauso verpfuscht ist wie das FDXB, wenn nur gepatcht und verlängert wurde und nicht verschoben, wenn also die Einsprungsadressen die gleichen sind wie beim FDXB, dann besteht eine recht gute Chance, daß XBASIC läuft. 3. Wenn sich ein mutiges Clubmitglied findet, das das mal testet, dann bekommt es (das Mitglied) im Falle des Fehlschlagens das Geld für die Software zurück. Nicht zurück bekommt es die D+P+V-Pauschale. Das ist mir zu umständlich, zu "portolastig", müßte ich dann die Disk zurückverlangen.

Leserbriefe / Tips*Zum "Supercalc-Kurs" von Wolfgang Gieger*

Ich frage mich, wozu der gut sein soll. Nur wenige Clubmitglieder benutzen Supercalc, und von denen finden sich sicherlich viele im Englischen zurecht. Eine kleine Gruppe von Clubmitgliedern, die nämlich, die Supercalc brauchen und nicht Englisch können, werden vielleicht von diesem Kurs profitieren. Aber selbst das wage ich zu bezweifeln, denn in so einen grauslichen Stil, in so eine schreckliche Sprache, die einen wegen ihrer Defekte vom Inhalt ablenkt, kann man sich einfach nicht einlesen.

Nun will ich nicht richten, soll doch jeder so und das schreiben, wie und was er will: aber bitte nicht auf Kosten anderer. Nein, ich will die Clubleitung nicht zur Zensur aufrufen. Vielmehr appelliere ich an die Vernunft jedes einzelnen, der meint, ein Projekt größeren Umfangs allen zugänglich machen zu müssen: Überlegt Euch, wie groß das allgemeine Interesse sein könnte, und wenn Ihr zu keinem Ergebnis kommt, dann bietet eure "Machwerke", eventuell mit Auszügen, doch einfach im Club an. Schränkt ein nur bescheidenes allgemeines Interesse Euren Tatendrang nicht ein, dann könnt Ihr immer noch versuchen, Disketten mit Euren Erzeugnissen an den Mann oder die Frau zu bringen. So geschah es mit Christian Löhrmanns Kompendium, mit dem deutschen Newword-Handbuch und anderem, und es war gut so. Möge Wolfgang mir meine bissige Ehrlichkeit verzeihen.

Genug des Lobes

Dann wollte ich noch sagen - nachdem die Laudatio-Welle zurückgegangen ist - daß ich die Clubleitung sehr schätze. Wegen ihrer stets offenen Augen und Ohren. Und das ist mein Ernst und mein letztes Wort - für heute.

Andreas Viebke

Tips

TURBO-Pascal-Quelprogramme vom Schneider Joyce laufen (nach Completion) problemlos auf dem MTX.

Nützliche BASIC-Einzeiler: (hab ich wo gelesen)

Formatiere positive Zahlen:

```
X=INT(X*100+.5)/100:X#=RIGHT$(" "+STR$(INT(X+.005)),5)+MID$(STR$(X+1E5+.005),8,3
```

Wandle X in vierstellige Hex-Zahl H#

```
H#="":FOR I=3 TO 0 STEP -1:R=INT(X/16^I):X=X-R*16^I:H#=H#+CHR$(R+48+INT(R/10)*7):NEXT
```

Wandle Hex-Zahl H# in Zahl X