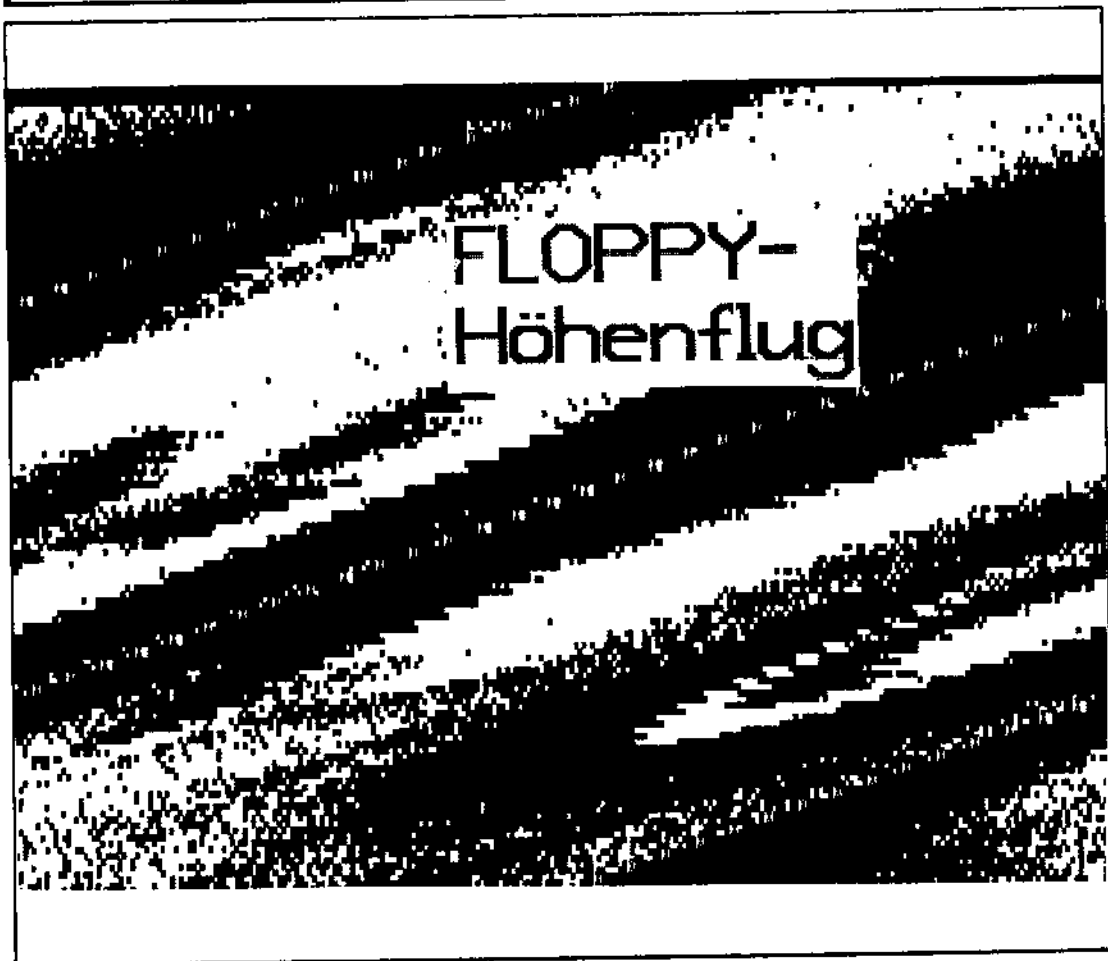


# 80-bus journal

Zeitschrift für NASCOM, GEMINI und andere  
Z80-Anwender

1. JAHRGANG \* SEPTEMBER 1983 \* AUSGABE 9



Der Heftpreis beträgt DM 5,-. Ein Abonnement erhalten  
Sie für DM 60,- im Jahr.

# 80-bus journal

## Intern

Liebe Leser,

Diese Ausgabe des Journals erreicht Sie sehr spät. Da es sich um's Septemberheft handelt, das Sie voraussichtlich erst im November erhalten, ist mir die Sache schon sehr peinlich. Es gibt dafür aber zwei Entschuldigungsgründe, die Sie hoffentlich akzeptieren können:

1. Seit Juli handelt es sich bei der Redaktion um einen Ein-Mann-Betrieb. Dafür ist die Arbeit aber weit zu umfangreich, um sie noch termingerecht zu vollenden.

2. In dieser Ausgabe sollte unbedingt eine ausführliche Beschreibung des Floppy-Controllers erscheinen, um die zahlreiche Leser gebeten haben, die mit der Vorabinformation recht wenig anfangen konnten. Diese Floppy-Geschichte hat uns aber so aufgehalten, daß alle Terminpläne über den Haufen geworfen wurden. Mehr dazu auf der Einstelgerseite.

Bei der augenblicklichen Situation erhebt sich natürlich die Frage, wie es mit dem Journal im nächsten Jahr weitergehen soll.

Ich habe nun schon seit einiger Zeit mit vielen Lesern gesprochen, die einen Fortbestand des Journals unbedingt begrüßen würden. Da Günter Kreidl auf keinen Fall weiter mitarbeiten wird, hat sich meine Frau entschlossen, das Heft möglicherweise zu übernehmen. (Dazu braucht man ja einen Gewerbeschein wegen der Mehrwertsteuer und dem ganzen Kram). Da ich aber keine Zusagen von Mitarbeitern erhalten konnte, sich in der Redaktionsarbeit das ganze Jahr über termingerecht zur Verfügung zu stellen, müßte ich voraussichtlich die Arbeit größtenteils alleine übernehmen. Dies ist bei der gegenwärtigen Form des Heftes und der Erscheinungsweise nicht möglich.

Nach vielfältigen Überlegungen und Gesprächen haben wir nun eine Kompromißlösung gefunden, wie wir die Gemeinschaft der Journalleser zusammenhalten könnten:

Das Heft erscheint nur vierteljährlich als Doppelausgabe. Damit wäre genügend Spielraum, die Artikel entsprechend aufzubereiten und auch weiterhin für Platinienservice, Cassettentausch und in Zukunft auch einen Floppy-Dienst zu sorgen. Viele Leser begrüßen die Form des Heftes, die es auch erlaubt, Programme und Artikel, die im Augenblick nicht interessieren, zu späterem Zeitpunkt nachzuschlagen.

Um aber aktuell zu bleiben, sollte nebenher zu einem festgelegten Zeitpunkt monatlich ein Rundbrief erscheinen (möglicherweise 4-8 DIN A 4 Seiten), der Kleinanzeigen enthält, Ankündigungen von Veranstaltungen, Programmbeschreibungen mit Floppy- oder Cassettenangebot und weitere Dinge, auf die man nicht ein Vierteljahr warten möchte.

In dieser Form würden wir auch von der Firma LAMPSON Unterstützung finden, die uns z.B. das CLD-DOS und weitere interessante Erweiterungen auf unserem Diskettenformat zu akzeptablen Preisen zur Verfügung stellen würde.

Druck- und vor allem Portokosten setzen aber trotz der Einsparung von vier normalen Heften weiterhin den Abopreis von DM 60,- im Jahr voraus. (Verglichen mit anderen Beitragsgebühren von Computerclubs wäre das allerdings nicht übermäßig viel).

Da ich im Augenblick schon genug Schwierigkeiten habe, meine vorgelegten Ausgaben zurückzubekommen, wäre das Vorhaben nur möglich, wenn ich die Zusage einer notwendigen Anzahl von Abonnenten hätte, denn ich will kein weiteres finanzielles Risiko eingehen. Wenn Sie also an einem Fortbestand des Journals in oben beschriebener Form interessiert sind, schicken Sie mir bitte ein Kärtchen mit Ihrer Zusage. Das neue Jahr ist nicht mehr fern, so sollten Sie Ihre Antwort nicht hinausschieben.

Falls Sie aus irgendwelchen Gründen kein Interesse haben, teilen Sie uns die Gründe bitte mit. Vielleicht erhalten wir dadurch neue Anregungen.

Dieses Heft ist jedenfalls in der alten Form erschienen und sagt Ihnen hoffentlich zu; es enthält ja außer der Floppy-Beschreibung einiges, was Sie verwerten können. Das

nächste ist nun schon wieder in Arbeit, und verspricht auch einiges; u.a. einen Druckerbuffer, eine 256x256 Punkt-Grafik für weniger als 80.-, eine ECB-PIO/SIO-Karte, eine Verbesserung der N2-Grafik, ein Buchführungsprogramm und einiges mehr.

Für's Jahresende könnte noch eine Grafikkarte erscheinen (in Holland entwickelt, in Österreich zum Layout verarbeitet), die eine 256x512-Punkte-Grafik erlaubt, mit Lichtgriffel und ausgereifter Software. Es tut sich so einiges in unserer "Gemeinde". Tragen Sie dazu bei, daß sie sich nicht in alle Winde zerstreut.

Ihr Günter Böhm

### Zur Floppy Karte

Die Floppy-Karte ist fertig durchkontaktiert mit den beiden Steuerproms für DM 60,- zu beziehen.

Überweisen Sie einfach den Betrag an Gabi Böhm

PSchA Kirh  
mit dem Vermerk "Floppy"

Der Benötigte Baustein FDC 9229B ist nur bei einer Firma erhältlich. Er kostet als Einzelstück DM 47,74 + MwSt.

Wir hätten den Baustein auch gerne für Sie besorgt, aber dann müßten Sie über 60,- dafür bezahlen. So laufen Sie günstiger mit einer Direktbestellung bei der Firma

Tekelec Airtronic GmbH

Postfach 152027

80000 München 15

### INHALT

2	80-Bus Journal Intern	
3	Inhalt/Impressum	
4	Fragebogenauswertung	Karl Mark
5	METEOR	Peter Brendel
	Tips und Tricks	
6	Seite(n) für Floppy-Einsteiger	Günter Böhm
14	FDC-Aufbau	Dieter Oberle
15	RTTY II	Jörg Wittich
18	Cass.-Tastatur-Interface	Karl Schulmeister
19	Interrupt-Uhr	Jörg Wittich
22	User Keys	Gerhard Klement
	Kettennetz	Gerhard Klement
23	NASCOM Praxis Teil 5	Wolfgang Mayer-Gürr
25	TDL-Basic	Peter Huntemann
26	NASCOMPL	
27	GEMINI	
28	LAMPSON Sonderangebote	

### IMPRESSUM

#### HERAUSGEBER:

Günter Böhm  
75 Karlsruhe

Ludwigshafener Str. 21d  
Tel. [REDACTED]

Gabi Böhm

Redaktion  
ebendort  
Layout u. Versand  
Bertenweg 18

Günter Kreidl  
4172 Straelen

Tel. [REDACTED]  
Redaktion u. Buchhaltung

#### KORRESPONDENTEN:

Karl Georg Englmann  
Mutterstadt

Tel. [REDACTED]  
Reinzeichnungen

Wolfgang Mayer-Gürr  
Recklinghausen

Tel. [REDACTED]

Clemens u. Max Ballarin  
Ueberlingen

Tel. [REDACTED]

Michael Bach  
Stegen

Tel. [REDACTED]

Peter Brendel  
Mannheim

Tel. [REDACTED]

Hans-Jürgen Plath  
Kiel

Hans Schneider  
Eens

Oesterreich:

Gerhard Klement  
A-Wien

Tel. [REDACTED]

Niederlande:

Eric v.d.Vaart  
NL-Waddixveen

England:

Frank M. Butler  
Mansfield Woodhouse/Notts

Luxemburg:

Rene Claus  
L-Bonneweg

Schweiz:

Markus Zimmer  
CH-Basel

Tel. [REDACTED]

Jugoslawien:

Gilvazi Istvan  
YU-Becej

#### VERLAG:

Günter Kreidl 4172 Straelen

#### VERTRIEBSWEISE und BEZUGSPREIS:

Einzelheft DM 5,-  
Doppelheft DM 10,-

Jahresabonnement In- und Ausland DM 60,-

Es erscheinen 10 Hefte pro Jahr, davon zwei Doppelhefte. Es können jeweils nur ganze Jahrgänge abonniert werden. Bei Bestellungen nach dem Erscheinungsdatum des ersten Heftes eines Jahrgangs werden die bereits erschienenen Hefte nachgeliefert. Die Lieferung von Einzelheften durch den Verlag ist nicht möglich. Bitte zahlen Sie direkt bei der Bestellung auf des Postscheckkonto:

Günter Kreidl PSchA Essen

#### HAFTUNG und RECHTE:

Für Fehler in Texten, Bildern, Programmen und Schaltungen und daraus entstehende Schäden kann keine Haftung übernommen werden.

Alle Rechte verbleiben grundsätzlich bei den Autoren der Beiträge. Die Veröffentlichung von Programmen und Schaltungen geschieht nur für den persönlichen Gebrauch der Abonnenten des 80-BUS-Journals; jede kommerzielle Auswertung ist nur mit Genehmigung des Verfassers erlaubt. Beiträge, die nicht mit einem Copyright-Vermerk versehen sind, dürfen für nichtkommerzielle Verwendung vervielfältigt werden, wenn als Quelle das 80-BUS-Journal und der Verfasser angegeben werde.

# FRAGEBOGEN

VON KARL MARK

Es haetten schon mehr sein koennen, aber die Urlaubszeit hat wohl viele davon abgehalten, ihre Fragebogen an mich zu schicken, oder hatten nicht mehr daran gedacht, wie auch immer, es laesst sich doch eine Art Statistik aufgrund der Zuschriften erstellen. Ich versichere Ihnen, lieber Leser, diese Auswertung dient nur dazu festzustellen, wie weit der einzelne Anwender sein System ausgebaut hat und welche Wuensche er im Bezug auf Hard- und Software hat. Zwar war Nascompl auch ein wenig skeptisch, aber wir wissen, dass es oft ohne Fragen nach den Wuenschen des Einzelnen nicht geht. Selbst eine Zeitschrift wie mc muss solche oder aehnliche Fragen stellen, damit die Redaktion weiss, was der Leser und Anwender von Rechnern wuenscht; erst danach kann man eine Zeitschrift gestalten und gleichzeitig die Wuensche der Leser erfuellen. Und nun zur "Analyse".

Es laesst sich feststellen, dass sich vom Schueler bis hin zum Rentner alle Altersgruppen, mit diesem schoenen Hobby befassen. Hervorzuheben waere, dass viele der Anwender technische Berufe haben, somit also entsprechend "vorbelastet" sind. Die Frage "Welchen Rechner haben Sie?" wurde zu 80 % mit N 1 und N 2 beantwortet, aber einige der Fragebogen sagten aus, dass auch Zweitrechner (hauptsaechlich Z 80 Systeme) vorhanden sind.

Ca. 90 % haben ihren Speicher mit mindestens 32 K (jeder weiss, was gemeint ist. Wozu also KByte schreiben, wie die mc-Redaktion immer wieder mit Eifer verlangt? Red.) belegt, 10 % mit mehr als 48 K.

Die "wichtigste Frage" nach einer Floppy wurde mit nein beantwortet, somit steht fest: Hier muss verstaerkt im 80-BUS-Journal in Sachen Soft- und Hardware etwas getan werden, damit unser System auf den neuesten Stand gebracht wird, was ja in erfreulicher Weise im Heft 7/8 des Journals zu lesen ist. (...und in diesem! Red.) Es hat mich doch gefreut, dass die Spielernaturen unter uns nicht so haeufig vorkamen, wie erst angenommen. Dafuer dominiert die technische Anwendung des Rechners, wie Steuern und Regeln, astronomische Berechnungen, Amateurfunk und

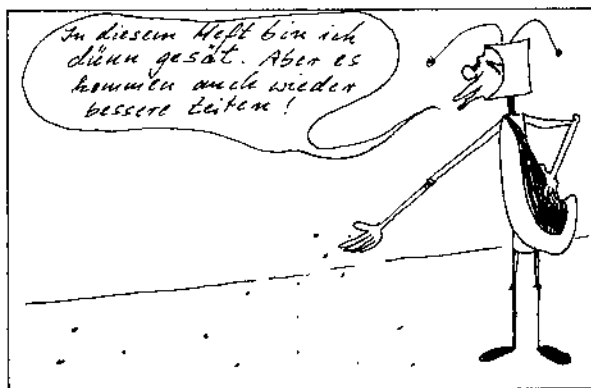
RTTY, Lagerhaltung, Kartelen, Datenverwaltung und Textverarbeitung. Hier geht die Hobbyanwendung in das Gebiet der ernsthaften Anwendung ueber.

Zum Thema Drucker waere zu sagen: Fast jeder besitzt einen Drucker; die meisten von uns haben einen "schnellen" Printer. Das Arbeiten in Maschinensprache und Basic haelt sich die Waage. Aber jetzt: "Welche Kenntnisse haben Sie im Programmieren?" Traurig, traurig, kann man da nur sagen (mich einbezogen). Es sind nur ein paar (oder waren es zwei Paar?), die mit gut geantwortet haben. Hier sei sofort gesagt: Seiten fuer Einsteiger (und Zugestiegene) sollten nicht vernachlaessigt werden. (Wer macht's? Red.) Wichtig also fuer den Erhalt der Substanz. Viele baten auch um mehr Programmierbeispiele, sei es in Basic oder in Maschinensprache. Man weiss ja, was der Rechner kann, aber nur, wenn er die richtige Software "eingefloesst" bekommt.

Mit Forth, CP/M und Pascal arbeiten zur Zeit noch sehr wenige unter uns. In dieser Richtung muesste auch was getan werden. Hardware-Wuensche gehen dahin, dass nachbaeuere Schaltungen veroeffentlicht werden, nicht solche, die im Verlaufe eines Jahres in sechs Heften immer wieder Berichtigt werden. Es soll ja auch mal funktionieren, denn sonst verliert man leicht die Uebersicht und die Laune dazu. Unter Software-Wuenschen versteht man fast einstimmig: Die Zeit des Spielcomputers ist vorbei, wir wollen den Rechner jetzt dazu benutzen, dass er uns im alltaeglichen Bedarf zur Seite steht. Warum soll er nicht auch energiesparend unsere Heizung steuern? Und dem Wunsch nach mehr Programmen fuer den Funkamateurl kann ich mich nur anschliessen. Hier gibt es zur Zeit noch sehr wenig Amateurfunk-spezifische Programme, sei es zum Steuern oder Regeln von Geraeten, oder Berechnungs-Software fuer im Orbit befindliche Satelliten. Der Wunsch, welche Artikel im 80-BUS-Journal erscheinen sollten, kommt schon einer flehenden Bitte gleich. Zunaechst sollte die zur Zeit existierende Anwender - Software besser beschrieben werden (Basic, Zeap, NAS-DIS usw.) in Form von kurz erklarten Beispielen. Man sollte auch daran denken, dass staendig Newcomer zu uns kommen, Ihnen sollte der Einstieg nicht zu schwer gemacht werden.

Die breite Palette weiterer Wünsche erstreckt sich auf "Mehr einfache Hardware Tips, Interface zu anderen BUS-Systemen, wobei die Bitte nach ECB-BUS dominiert, Hardware-Uhr, Programmiertricks, Arbeitsweise einer Floppy, mehr Software fuer Grafik, Erklarung zur Steuerung ueber PIO-Ports, ebenso ueber In- und Out-Routinen in Basic, Info ueber CP/M, Basic, Lehrgang fuer Fortgeschrittene, DMA-und CTC-Anwendungen, sowie mehr Werbung fuer unsere Zeitschrift" und einiges mehr.

Zum Schluss noch eine Bemerkung (sie haette eigentlich in die erste Zeile wegen des Eigenlobes gehoert): Es sind durchweg alle mit dem 80-BUS-Journal zufrieden. Es heisst da im einzelnen: Sehr gut, weiter so, ausgeglichene Beitrage, fuer Jeden etwas.... Es hoert sich also gut an. Diese positive Bewertung sollte eigentlich Auftrag zur Kreativitaet und Initiative der Z80-Spezialisten sein, denn sie koennen dafuer sorgen, dass wir an unserem System noch lange Freude haben werden. Es gibt also noch viel zu tun. Wer packt es an?



## METEOR

von PETER BRENDDEL

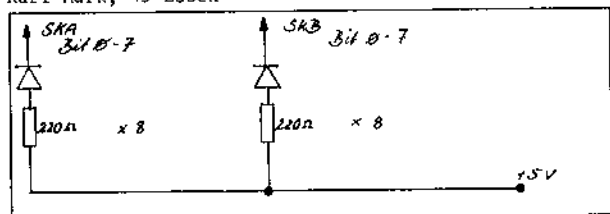
```

30 REM DAS SPIELCHEN BRAUCHT NICHT MAL NAS-GRA
40 REM
50 REM DAS "UFO" WIRD MIT CURSOR LINKS/RECHTS
60 REM GESTEUERT + MIT LEERTASTE GESTOPPT.
70 REM (AB 200 NICHT MEHR)
80 REM DU MUSST DEN METEOREN AUSWEICHEN
90 REM
100 FOR I=3200 TO 3208 STEP 2: READ A: DOKEI, A: NEXT
110 DATA 25311,312,18351,-15441,-3854
120 POKE 3210,233: DOKEI 4100,3200: REM INKEY 1
130 CLS: P=2400: D=0: U=0
140 PRINT TAB(RND(1)*70)"*"
150 U=U+1
160 IF PEEK(P) < 232 THEN 280

```

## TIPS UND TRICKS

Wer hat nicht schon 'mal versucht, mit den PIO-Ports zu experimentieren, ohne gleich Geräte wie Drucker, Motoren oder andere elektrische Verbraucher über die PIO zu steuern? Beim Erstellen von diesbezüglichen Programmen kann man sich auf einfache Weise mit einer LED-Kette helfen, deren optische Anzeige alle Schaltvorgänge der PIO nachvollziehen läßt. Der Aufbau erfolgt am einfachsten mit einer Lochraster-Platine. 16 LEDs, 16 Widerstände 220 Ohm, 2 DIL-Stecker 16 Pin und etwas Kabel: einfach, billig, aber praktisch. Karl Mark, 43 Essen



## KLEINANZEIGEN

SUCHE: FDC + Floppy für CP/M mit NASCOM Videokarte 80x24 für NASCOM ASCII-Selectric Interface für IBM Kugelpapier-Schreibmaschine  
Rene Claus

VERKAUFE FDC-Karte 80-Bus Format für Controller FD 1771-1 (softwarekompatibel mit WD 1793) mit Schaltplan DM 50,-  
Günter Böhm

```

170 POKEP,7
180 S=USR(0)
190 IFS=17 THEN D=-1
200 IFS=18 THEN D=1
210 IFS=32 THEN IF U<200 THEN D=0
220 T$=STR$(U)
230 FOR I=2 TO LEN(T$)
240 Q=VAL(MID$(T$,I,1))+48: POKE 3018+I,Q: NEXT
250 POKEP,32: P=P+D
260 IF PEEK(P)=43 THEN 370
270 GOT0140
280 H=100: POKEP,43: GOSUB 400
290 POKEP,150: GOSUB 400
300 POKEP+66,138: POKEP-66,137: POKEP-62,136
310 POKEP+62,139: GOSUB 400
320 G=46
330 FORK=1 TO 2
340 FOR J=1 TO 5
350 FOR I=P-130 TO P-124 STEP 2: POKEI,G: NEXT
360 P=P+64: NEXT: G=32: P=P-320: NEXT
370 SCREENP-2378,6: PRINT "ENDE": H=2000: GOSUB 400
380 S=USR(0): IFS=0 THEN 380
390 GOT0130
400 FOR I=1 TO H: NEXT: RETURN

```

# Seite(n) für Floppy- Einsteiger

von GÜNTER BÖHM

Dieser Artikel und die entsprechenden Programme sind der Grund für die Verspätung dieser Ausgabe. Um Ihnen den Aufbau und die Benutzung des FD-Controllers zu vereinfachen, sollte eine Karte zunächst einmal an einem Mini-Laufwerk getestet werden. Als Grundlage dienten die Routinen von A. Zippel (Heft 5), die allerdings in CPM eingebunden auf einem Rechner in der Uni Karlsruhe laufen und für unsere Zwecke isoliert und ergänzt werden mußten, wobei natürlich Fehler auftraten.

Vier Wochen hat nun die Fehlersuche gedauert, wobei ich als absoluter Floppy-Neuling gewaltig ins Schwitzen kam. Helmut Emmelmann hat mich mit viel Zeitaufwand tatkräftig unterstützt. Ohne seine Erfahrungen und seine Vorgaben, wie man an so eine Sache überhaupt herangeht, wäre das Projekt noch lange nicht am Laufen. Dieter Oberle wurde auch laufend belastigt und hat durch viele Tips und die Lieferung der notwendigen Informationen und Datenblätter zum Gelingen beigetragen.

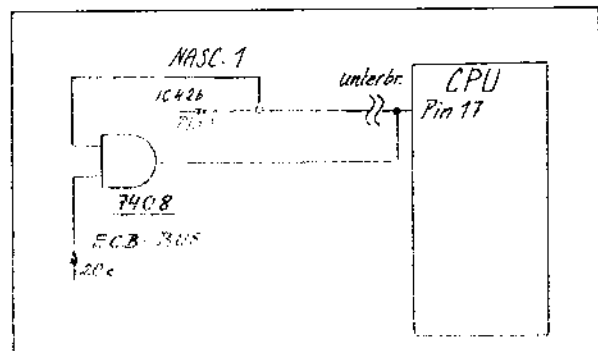
Dieser Artikel kann keine Information bis ins letzte Detail beinhalten, das würde den Rahmen sprengen und auch das Erscheinen der Heftes noch weiter verzögern. Es sollen aber die Informationen geliefert werden, die zum Verständnis der grundlegenden Funktion des Controllers dienen und die auch zum Testen der Software unerlässlich waren. Vielleicht können sie Ihnen helfen, etwaige Fehler bei Ihrem Nachbau zu entdecken. Sie sind allerdings in dem Vorteil zu wissen, daß die Schaltung und die angebotene Software hier schon läuft, während wir beim Testen weder Hard- noch Software als funktionstüchtig voraussetzen durften und so einer Rechnung mit zwei Unbekannten gegenüberstanden.

Die Funktion des Controllers soll anhand der Schreib/Leseroutinen erklärt werden. Zunächst aber zum Aufbau und Anschluß der Controller-Karte.

Die DMA wurde nicht bestückt. Welche Drahtbrücken dabei zu machen sind, wurde bereits im letzten Heft beschrieben. In diesem Heft

finden Sie den Bestückungsplan und eine neue Stückliste. Die beigefügten Änderungen gegenüber dem Schaltplan sind z.T. auf der Platine schon durchgeführt oder aber leicht vorzunehmen.

Wenn der Controller Daten anfordert oder abgeben möchte (beim Lesen oder Schreiben), gibt er ein DRQ- (Data Request) Signal an Pin 38 aus. Dieses Signal soll an der CPU einen NMI (Hardware Interrupt mit absolutem Vorrang) auslösen. Deshalb muß es über die ECB-Busleitung 20c zur CPU geführt werden. Da der NMI im NASCOM über Port 0 zur Single Step Routine verwendet wird, muß man diese Verbindung lösen oder aber die Leitungen von Port 0 und dem Controller über ein Gatter verbinden, sodaß beide wahlweise den Interrupt auslösen können. Ich habe auf der Tastaturkarte von List/Niemann einen Jumper in dieser Leitung, den ich öffnete und damit auf die Single Step Möglichkeiten verzichtete. Das erwähnte Gatter soll aber auf meiner Karte noch nachträglich eingebaut werden.

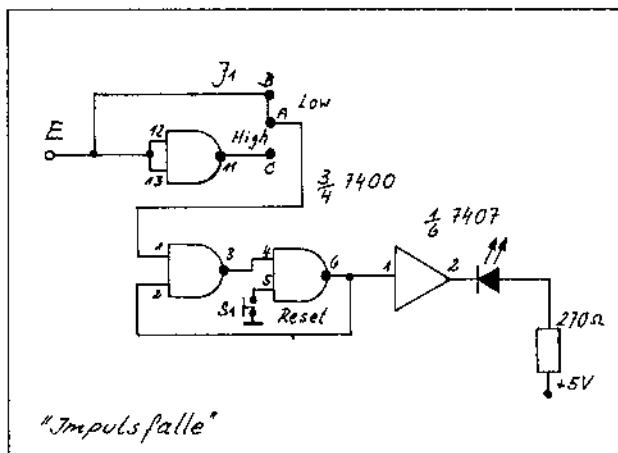


Den Anschluß der Cannon-Buchse über ein 34-poliges Flachbandkabel an den Preßstecker des BASE 6106 Laufwerkes zeigt die Tabelle. Zwischen allen Steuerleitungen liegen Masseanschlüsse, die man je zu zwei oder drei gebündelt an die vorgesehenen Stifte des Cannon-Steckers löten kann.

Shugart Bus	BASE 6106
23	MOTOR
9	HEAD LOAD
6	DIR.
7	STEP
8	WRITE GATE
25	LOW CURRENT
	15
	2
	18
	20
	20
	offen

10	WRITE DATA	22
4	READY	6
5	WRITE PROT	28
3	TRACK 00	26
2	INDEX	8
1	RD DATA	30
16, 17, 18	} GND	} alle Anschl. mit unger.Zahl
19, 20, 21		
22		
12	DR0	10
13	DR1	12
11	DR2	14
24	SIDE SEL	offen
15	DISK CHANGE	34
14	DOUBLE SIDE	offen

Seien Sie beim Anschluß des Laufwerkes an das Netzteil vorsichtig: mein Laufwerk döst nun in den ewigen Jagdgründen, weil ich die Anschlüsse nicht nachgemessen habe. Mir lagen Unterlagen zu einem neueren Laufwerk vor, auf dem die Steckeranschlüsse vertauscht sind. Beim Anlöten des Steckers lagen nun 12V an den TTL-Bauteilen. Die ertrugen das 5 Minuten, dann begann die Sache zu stinken. Messen Sie also vor dem Einschalten nach, ob die TTL-ICs auch an der richtigen Leitung hängen!



Wenn der Controller eine Aufgabe erledigt hat (z.B. Spur suchen oder Schreib/Lesekopf auf Spur 00 positionieren, gibt er über seinen INT-Ausgang ein Signal an die PIOA Bit7, die dadurch einen Software-Interrupt auslösen soll. Da beim Testen des Programms hier zunächst nichts geschah (das Programm

blieb irgendwo hängen), baute ich eine kleine Schaltung, die es gestattet, einen Impuls zu "speichern", denn das Oszilloskop ist bei einem einmalig auftretenden sehr schnellen Impuls keine Hilfe. Der Eingang der Schaltung wurde dann an die Interruptleitung (21c) oder die NMI-Leitung (20c) gelegt. Erfolgte ein Interrupt, so leuchtete die LED und zeigte damit den erfolgten Impuls an.

Auf diese Weise konnte ich einen gewichtigen Fehler im Programm beseitigen. Vielleicht kann Ihnen die Schaltung auch einmal beim Verfolgen ähnlicher Signale dienlich sein. Durch Umstecken eines Jumpers läßt sie sich auch durch positive Signale ansteuern.

Die Hardware ist aufgebaut, das Laufwerk ist angeschlossen. Nun müssen wir es zum Laufen bringen. Am besten geht das, wenn eine formatierte Diskette zur Verfügung steht, dann hat man eine standardisierte Kontrolle. Aber auch mit einer leeren Diskette können wir arbeiten.

Zunächst müssen die PIO, der Controller und der FDC 9229B (Floppy Disk Interface Circuit) initialisiert werden.

Die PIO arbeitet im Controlmode, d.h. es werden bestimmte Leitungen als Ein- bzw. Ausgänge benutzt. Die Bits der I/O-Masken lassen sich anhand des Schaltbildes nachvollziehen; 1 ist dabei jeweils ein Eingang. Weiterhin muß PIOA interruptfähig gemacht werden; die entsprechenden Kontrollwörter machen sie dafür bereit, als Interruptvektor wird für PIO und I-Register in unserem Programmbeispiel 8000H eingegeben; d.h. dort ist die Adresse eingetragen, zu der das Programm bei einem Interrupt durch Bit7 der PIO springt.

Der WD-1793 wird durch einen zeitlich genau definierten Impuls zurückgesetzt.

Als letzter Schritt der Initialisierung wird die Laufwerknummer selektiert. (Die Nummer muß auf dem Laufwerk selbst mittels einer Drahtbrücke eingestellt werden, Fabrikmäßig ist DRIVE0 vorgesehen). Hier muß man aufpassen, weil durch die PIO auch der FDC9229 programmiert wird. Lange Zeit ist uns beim Testen entgangen, daß eine 1 am Pin "MINI" (durch Invertierung eine 0 an der PIO A3) der Controller für 8" Laufwerke angesteuert wird. Damit Ihnen hier kein Denkfehler unterläuft, sollten Sie nachfolgende Tabelle zu Rate ziehen.

INT A7	HEADLD A6	MR A5	DDENSE A4	MINI A3	DRZ A2	DR1 A1	DR0 A0		MINI SD	MINI DD	8" SD	8" DD
0	0	1	1	0	0	0	1	DRIVE0	31	21	39	29
0	0	1	1	0	0	1	0	DRIVE1	32	22	3A	2A
0	0	1	1	0	1	0	0	DRIVE2	34	24	3C	2C
PIO A (ADDR. 10H)								STEUERWORT				

Nun ist der Controller bereit, vom Rechner Befehle entgegenzunehmen. Dafür stellt er folgende Register zur Verfügung:

FDCCMD Kommandoregister  
 FDCTRK speichert die Spurnummer  
 FDCSEC speichert die Sektornummer  
 FDCDAT Speicher für Schreib/Lesedaten  
 FDCSTA gibt Fehlermeldungen aus

Mit dem Out-Befehl (Taste O) und der entsprechenden Portadresse (sind dem Anfang des Assemblerlistings zu entnehmen) können zu Testzwecken direkt Kommandos an den FDC ausgegeben werden.

Als Beispiel:

O 0C 03 läßt den Schreib/Lesekopf auf Spur 00 zurückfahren (dazu sollte man die Schnecke des Laufwerks manuell vorstellen, damit's auch eine Strecke zum fahren gibt)

Die möglichen Befehle sind ebenfalls am Beginn des Listings zusammengestellt.

Nach der Ausführung eines solchen Befehls macht der FDC einen Interrupt, der ihn in unserem Programm in die Routine FLPINT springen läßt. Zu Testzwecken wird dort das Wort "INT" auf den Bildschirm geschrieben gefolgt von der Adresse, die nach dem Programmteil steht, der durch den Interrupt unterbrochen wurde. Dieser Teil der Interruptroutine wird nach erfolgten Testläufen wieder entfernt.

Die Hauptaufgabe des Interruptprogramms ist es festzustellen, ob die Unterbrechung innerhalb einer Schleife geschah. Ist dies der Fall, wird das Programm außerhalb der Schleife fortgesetzt.

Dieser Programmiertrick erlaubt es, mit einer Endlosschleife auf einen Interrupt zu warten. (z.B. in Zeile 1880). Handelt es sich um NMIs, so wird nach Rückkehr aus der NMI-Routine (hier Lesen eines Bytes von Diskette) in die Schleife zurückgekehrt und auf den nächsten NMI gewartet. Erst der Interrupt von der PIO, der anzeigt, daß die gesamte Aktion abgearbeitet ist, entläßt das

Programm auf oben erwähnte Weise aus der Schleife.

Nach diesen Schleifen, die in allen Programmteilen in gleicher Art auftauchen, wird jeweils der Inhalt des Statusregisters (FDCSTA) in den Akku gelesen. In der gegenwärtigen Form des Programms wird diese Information noch nicht weiterverarbeitet. Setzt man aber einen BREAKPOINT dahinter (EZH) kann man nach Ausführung des Kommandos im Akku die Fehlermeldungen ablesen, die zeigen ob der Befehl vom Controller richtig durchgeführt wurde.

Mit dem QUERY-Befehl (Taste Q) kann dieses Register auch direkt auf den Bildschirm geholt werden. (Q 0C).

Die wichtigsten Fehlermeldungen, die uns häufig großes Kopfzerbrechen bereitet haben, finden Sie in der folgenden Tabelle:

80 Not Ready ; Laufwerk nicht bereit  
 10 Record Not Found ; Sektor nicht gefunden  
 04 Lost Data ; Daten verloren  
 08 CRC Error ; Prüfsummenfehler  
 00 ; Hurra, gewonnen

Da die Fehlermeldungen bei den verschiedenen Befehlen unterschiedlich sind, folgt eine Tabelle mit der Bedeutung der Bits:

Bild 5

Bit	Type I Commands	Read Address	Read	Read Track
S7	Not Ready	Not Ready	Not Ready	Not Ready
S6	Write Pr.	0	Rec.Type	0
S5	Head Eng.	0	Rec.Type	0
S4	Seek Err.	ID Not Fd	Rec,Not F.	0
S5	CRC Err.	CRC Err.	CRC Err.	0
S2	Track 0	Lost Data	Lost Data	Lost Data
S1	Index	DRQ	DRQ	DRQ
S0	Busy	Busy	Busy	Busy



	Write	Write Track
	Not Ready	Not Ready
	Wr.Protect	Write Protect
	Write Fau.	Write Fault
	Rec.Not F.	Ø
	CRC Error	Ø
	Lost Data	Lost Data
	DRQ	DRQ
	Busy	Busy

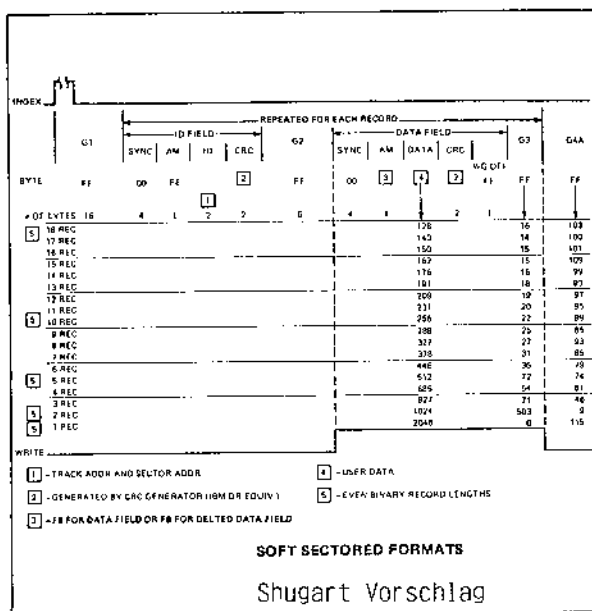
Von den TypeI-Kommandos habe ich nur die benutzt, die im Listing aufgeführt sind (Z100-203).

Nun sollten wir an die praktische Erprobung des Controllers gehen.

Setzen Sie einen Breakpoint auf 8115H und starten Sie mit E810A. Die Initialisierung wird durchgeführt (vorher auf Ihr Laufwerk einstellen!), die Schreibdichte auf Single Density eingestellt (für DD müßte im Akku 01 stehen), und der Kopf wird auf Spur 00 gefahren. Im Akku müßte 00 stehen (kein Error in FDCSTA).

Nun entfernen Sie den Breakpoint und setzen ihn auf 811DH. Mit gleicher Startadresse wie oben und einem zusätzlichen Argument zwischen 0 und 27H können Sie den Kopf nun eine der 40 Spuren aufsuchen lassen.

Der Test der Leseroutine muß aufgeschoben werden, bis eine formatierte Diskette vorliegt. Deshalb zunächst zum Formatieren.



Bis Zeile 4340 ist der Programmablauf bekannt. Mit dem Aufruf von "SAV66" wird die NMI-Routine von NASSYS in Sicherheit gebracht und mit "RE66HL" durch eine eigene Routine ersetzt. Bei einem NMI, der ja einen Restart bei der Adresse 66H erzeugt, wird nun ein OUTI (Inhalt von Register HL nach Port (C) und INC HL) abgearbeitet, der die Daten auf die Diskette bringt. Am Ende der Routine wird der alte Zustand bei Adresse 66H wiederhergestellt ("RE66").

Das Formatieren geht so vor sich, daß im Speicher (hier ab 1000H) eine Spur vollständig aufgebaut wird. Diese wird durch das WRITE-TRACK-Kommando mithilfe der NMIs in einem Stück auf die Diskette geschrieben. Dann wird die nächste Spur aufgebaut etc., bis alle 40 Spuren beschrieben sind.

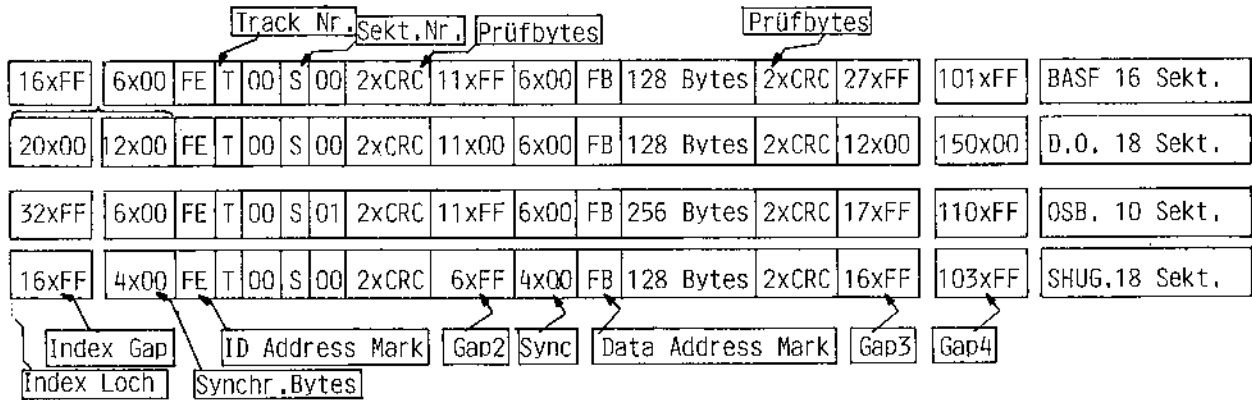
Da man die Spuren aber nicht jeweils vollständig wieder einlesen möchte (was durch das Kommando E4H möglich ist), sondern auch kleinere Datenmengen, sind die Spuren in einzelne Sektoren eingeteilt. Und hier beginnt das Problem: Auf welches Format möchte man sich einigen? Für uns scheint klar, daß wohl aus der Ecke der mc-CP/M-Benutzer die meisten mit ähnlich gelagerten Interessen kommen, und deshalb wollen wir uns in Zukunft an deren Format anschließen. Mehr darüber im nächsten Heft, denn der Aufbau dieses Formates ist etwas komplizierter. Hier sollten wir ein Format wählen, mit dem wir den Controller testen können, ohne Rücksicht auf Kompatibilität.

Mir lagen Disketten vor von folgenden Formaten:

1. Dieter Oberle Spezialformat (18 Sektoren 128Bytes)
2. Osborne 10 Sektoren/256 Bytes;Single Density wie oben
3. Osborne Double Density (Format nicht weiter nachgeprüft)

Von diesen Disketten konnte ich die beiden letzteren lesen, Dieter Oberles Diskette war nicht zu entziffern, auch eine selbst beschriebene in diesem Format brachte keinen Erfolg. Obwohl Dieter schon seit Jahren damit arbeitet, scheint sich dieses Format mit unserem Controller nicht zu vertragen.

Ich habe mich nun zum weiteren Testen für das Osborne-Format Single Density entschieden. Den Vergleich mit einem BASF-Vorschlag, der Oberle-Diskette und einem Shugart-Vorschlag entnehmen Sie folgender Tabelle:



Die Tabelle bedarf wohl einiger Erläuterung.

Nach dem Indexloch auf der Diskette, durch das sich das Laufwerk mit einer Lichtschranke orientiert, folgt einmalig das Index-Gap (gap=Spalte, Freiraum). Danach einige Nullen zum Synchronisieren und das Byte FE, das als ID Address Mark den Beginn des Sektors angibt. Darauf folgt die Spurnummer (0-39) und ein Zero zum Trennen. Die Sektornummer (1-max,18 bei SD) wird von einem Byte für die Sektorlänge gefolgt.

Ist Bit3 in den Schreib/Leseroutinen 1, dann gilt für die Sektorlänge:

- 00 128 Bytes
- 01 256 Bytes
- 02 512 Bytes
- 03 1024 Bytes

Dies ist bei uns der Fall, und mit den Längen bei rückgesetztem Bit3 sollten wir uns nicht belasten.

Nach der Sektorlänge wird F7 ausgegeben. Mit diesem Befehl erzeugt der Controller eine 2Byte-Prüfsumme.

Die folgenden FFs (Gap 2) trennen den Sektorkopf vom Datenblock ab, der wiederum durch Nullen zur Synchronisation eingeleitet wird.

FB signalisiert den eigentlichen Beginn der Daten (Data Address Mark), und an die Daten schließt sich wieder die Prüfsumme an (beim Schreiben durch F7 erzeugt). Gap 3 schließt den Sektor ab.

Alle Sektoren einer Spur sind so gleich aufgebaut. Ans Ende des letzten Sektors wird nun noch das Pre-Index-Gap angehängt, das bis zum Indexloch reicht.

Dieses Format ist in der Format-Tabelle (TRKTAB) angelegt. Die "Expand"-Routine nimmt zwei Bytes und interpretiert das erste

als Anzahl, das zweite als Zeichen. So wird in Zeile 4820 32 mal das Byte FF in den Speicher geschrieben. Trifft die Routine ein 00, dann kehrt sie zurück.

Die Spurnummern werden fortlaufend erhöht, die Sektornummern werden der Sektortabelle in Zeile 5000 entnommen. Im Falle der Osborne-Formatierung sind sie ebenfalls fortlaufend, in nachstehendem Programmausschnitt sind sie so gelegt, daß sie schneller eingelesen werden können (Oberle). Das Format des Beispiels entspricht dem Shugart-Vorschlag.

ZBAP 280 Assembler - Source Listing

```

323E FE13      4630      CP   18+1 ;18 SEKTOREN FERTIG?

4800 ;
4810 ;
323D 10FF      4820 TRKTAB DEF 16,#FF
323F 00        4822 DEF 0
3240 0400      4824 TRKSTB DEF 4,0
3242 01FE      4830 DEF 1,0FEH
3244 01        4840 DEF 1
3245 00        4850 FTRK DEF 0
3246 0100      4860 DEF 1,0
3248 01        4862 DEF 1
3249 00        4864 SECNR DEF 0
324A 0100      4866 DEF 1,0 ;SEKTORLAENGE 128
324C 01F7      4870 DEF 1,0F7H
324E 06FF      4880 DEF 06,#FF
3250 0400      4885 DEF 4,0
3252 01FB      4890 DEF 1,0FBH
3254 80E5      4900 DEF 128,0E5H; DATA
3256 01F7      4920 DEF 1,0F7H
3258 10FF      4930 DEF 16,#FF
325A 00        4940 DEF 0
325B 67FF      4950 DEF 103,#FF
325D 00        4960 DEF 0
4970 ;
4980 ;
4990 ;
325E 010C0510 5000 SEKTAF DEF 1,12,5,16,9,2,13,6,17,10
09020D06
110A
3268 030E0712 5010 DEF 3,14,7,18,11,4,15,8
0B040F08

```

Experimentieren Sie mit der Formatierung. Testen Sie das Ergebnis zunächst mit READ TRACK, indem Sie in Adresse 807BH anstatt 8CH → E4H eintragen. Dann wird Ihnen nach E810A E (als Beispiel) die ganze Spur 14 in den Speicher ab 1000H eingelesen, und Sie können das Format kontrollieren (es sei denn, Sie erhalten eine Fehlermeldung). Haben Sie Erfolg mit READTRACK, ändern Sie das Programm wieder und Starten Sie mit E810A E 3

Nun werden nur die Daten von Sektor 3 der Spur 14 in den Speicher gelesen. Vor diesen Experimenten sollten Sie allerdings alle Speicheradressen durch einen Copy-Befehl mit einem bestimmten Wert füllen, damit Sie auch sofort sehen, wenn der Speicher nicht richtig eingelesen wird. (Helmut Emmelmann empfiehlt 55).

Teilen Sie uns bald Ihre (hoffentlich positiven) Erfahrungen mit, damit wir sie in der folgenden Ausgabe noch berücksichtigen können; denn die muß diesmal sehr bald raus. Melden Sie sich auch, wenn Sie meinen, daß ich in meinen Ausführungen etwas Wichtiges vergessen haben sollte. Dies wäre bei der Fülle der Dinge, die man beachten muß, sehr leicht möglich.

ZEAP Z80 Assembler - Source Listing

```

0001 ;FDC CONTR. VER 1.6 23.10.83
0002 ;G.Bohm/H.Emmelmann
0003 ;nach Routinen von Oberle/Zippel
34DF 000E 0005 ARG2 EQU 000EH
34DF 000C 0010 FDCCMD EQU 00CH
34DF 000D 0020 FDCCTR EQU 00DH
34DF 000E 0030 FDCSEK EQU 00EH
34DF 000F 0040 FDCDAT EQU 00FH
34DF 000C 0045 FDCSTA EQU 00CH
0050 ;
34DF 0010 0060 PIOAD EQU 10H
34DF 0011 0070 PIOAC EQU 11H
34DF 0012 0080 PIOBD EQU 12H
34DF 0013 0090 PIOBC EQU 13H
0095 ;
34DF 0003 0100 THOME EQU 03 ;SPEED STEPPER MOTOR
34DF 001B 0120 FSEEK EQU 1BR;SPUR SUCHEN
34DF 001F 0140 FSEEKV EQU 1FR;SUCHEN MIT VERIFY
34DF 003C 0160 FREAD EQU 8CH ;SEKTOR LESEN
34DF 00AC 0180 FWRITE EQU 0ACH;SEKTOR SCHREIBEN
34DF 00D0 0200 FINT0 EQU 0D0H ;FDC RÜCKSETZEN
34DF 00F4 0202 FWRTRK EQU 0F4H;GANZE SPUR SCHREIBEN
34DF 00E4 0203 FRPTRK EQU 0E4H;GANZE SPUR LESEN
0220 ;
0240 ;
8000 0260 ORG 8000H
8000 D980 0270 DEFW FLPINT;INTERRUPT TABELLE
8002 F3 0280 INIT DI ;PIOs u. FDC initiieren
8003 3ECF 0300 LD A,0CFH
8005 D311 0302 OUT (PIOAC),A ;CONTROLMODE
8007 3EC0 0304 LD A,0C0H
8009 D311 0306 OUT (PIOAC),A ;I/O Maske

```

```

800B 3ECF 0310 LD A,0CFH
800D D313 0312 OUT (PIOBC),A ;CONTROLMODE
800F 3EF0 0314 LD A,0F0H
8011 D313 0316 OUT (PIOBC),A ;I/O Maske
0318 ;
8013 3EB7 0320 LD A,0B7H;INTERRUPT CONTROL MODE
8015 D311 0322 OUT (PIOAC),A
8017 3E7F 0324 LD A,7FH
8019 D311 0326 OUT (PIOAC),A ;MASKE:BIT7 macht Interr.
0328 ;
801B 3E08 0360 LD A,8 ;FDC RÜCKSETZEN
801D D310 0380 OUT (PIOAD),A
801F 0E01 0400 LD C,1
8021 CDC580 0420 CALL DELAY
8024 3E28 0440 LD A,28H
8026 D310 0460 OUT (PIOAD),A
8028 3ED0 0480 LD A,FINT0
802A D30C 0500 OUT (FDCCMD),A
802C E3 0520 EX (SF),HL
802D E3 0540 EX (SP),HL
802E DB0C 0560 IN A,(FDCSTA)
8030 3E80 0580 LD A,80H ;INTERR.VECTOR MSB
8032 ED47 0600 LD I,A
8034 3E00 0620 LD A,00 ;LSB
8036 D311 0640 OUT (PIOAC),A
8038 3E34 0660 LD A,34H ;LAUFWERKNUMMER HIER 3
803A D310 0680 OUT (PIOAD),A
803C FB 0700 EI
803D ED5E 0710 IM 2;INTERRUPT MODE
803F C9 0720 RET
0730 ;
0740 ;HOME (KOPF AUF TRACK00)
0760 HOME LD A,PHOME
0780 OUT (FDCCMD),A
0790 EI
0805 18FE 0820 HOMEW JR HOMEW
0847 DB0C 0840 IN A,(FDCSTA)
0849 C9 0860 RET
0880 ;
0900 ;
0920 ;
0940 ; SEEK PARAMETER IN A
804A D30F 0960 SEEK OUT (FDCDAT),A ;TRACK NUMBER
804C 3E1B 0980 LD A,FSEEK
804E FB 1000 EI
804F D30C 1020 OUT (FDCCMD),A
8051 18FE 1040 SEEKW JR SEEKW
8053 DB0C 1060 IN A,(FDCSTA)
8055 C9 1080 RET
1100 ;
1120 ;
1140 ;
1160 ;
8056 D30E 1180 SETSEK OUT (FDCSEK),A ;SEKTORNUMMER
8058 C9 1200 RET
1280 ; SETDEN (SINGLE oder DOUBLE)
1300 ;
8059 C5 1320 SETDEN PUSH BC
805A CB47 1340 BIT 0,A
805C 0E10 1360 LD C,10H
805E 2802 1380 JR Z,SETDEL
8060 0E00 1400 LD C,0
8062 DB10 1420 SETDEL IN A,(PIOAD)
8064 E6EF 1440 AND 0EFH
8066 B1 1460 OR C
8067 D310 1480 OUT (PIOAD),A
8069 C1 1500 POP BC
806A C9 1520 RET
1540 ;
1560 ;
1580 ;
806B E5 1600 READ PUSH HL
806C CDA180 1620 CALL SAV66
806F 00 1680 NOP
8070 00 1681 NOP ;PROG. CORR.
8071 21BD80 1700 LD HL,NMIR
8074 CDB080 1760 CALL RE66HL
8077 E1 1780 POP HL
8078 0E0F 1800 LD C,FDCDAT
807A 3E8C 1820 LD A,FREAD

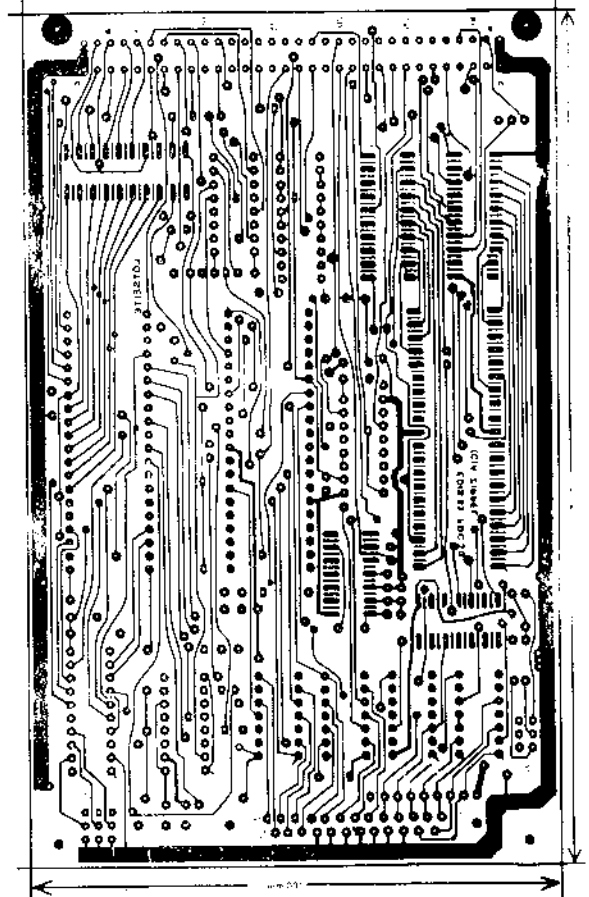
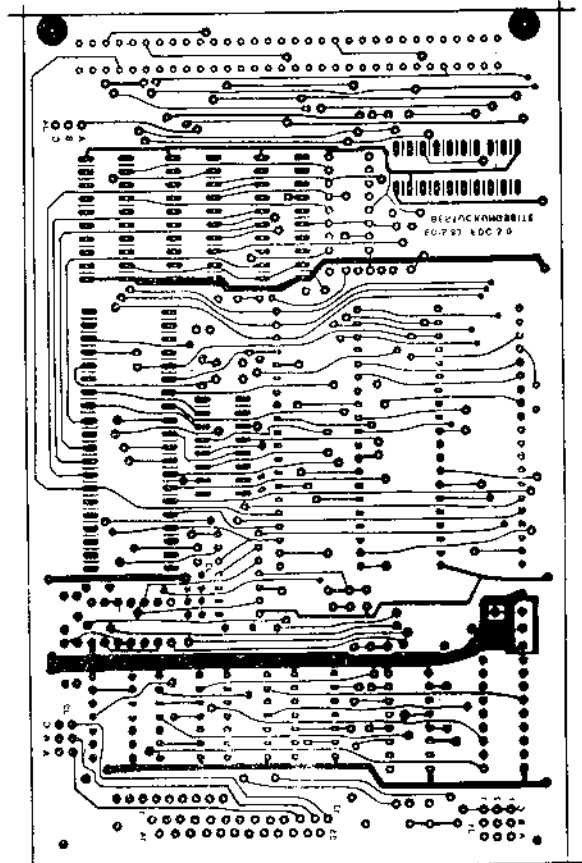
```

807C	FB	1840	EI			80F7	2004	2920	JR	NZ,NOLOOP
807D	D30C	1860	OUT	(FDCCMD),A		80F9	03	2930	INC	BC
807F	18FE	1880	JR	READW		80FA	70	2940	LD	(HL),B
8081	DB0C	1900	IN	A,(FDCSTA)		80FB	2B	2950	DEC	HL
8083	CDAD80	1980	CALL	RE66		80FC	71	2960	LD	(HL),C
8086	C9	2000	RET			80FD	78	2961	NOLOOP	LD A,B
		2020				80FE	DF68	2962	DEFW	#68DF
		2040				8100	79	2963	LD	A,C
		2060				8101	DF68	2964	DEFW	#68DF
8087	E5	2080	WRITE	PUSH HL		8103	EL	2970	POP	HL
8088	CDA180	2100	CALL	SAV66		8104	CL	2980	POP	BC
808B	21C180	2120	LD	HL,NMIW		8105	F1	2990	POP	AF
808E	CDB080	2140	CALL	RE66HL		8106	FB	3000	EI	
8091	E1	2150	POP	HL		8107	ED4D	3010	RETI	
8092	0E0F	2151	LD	C,FDCDAT				3030		
8094	3EAC	2152	LD	A,FWRITE				3040		
8096	FB	2153	EI					3050		
8097	D30C	2154	OUT	(FDCCMD),A				3060		
8099	18FE	2155	WRITEW	JR WRITEW		8109	00	3070		
809B	DB0C	2156	IN	A,(FDCSTA)				3080	FLPSTA	DEFB 0
809D	CDAD80	2160	CALL	RE66				3090		
80A0	C9	2180	RET					4000		TEST DER ROUTINEN
		2200						4010		
80A1	216600	2220	SAV66	LD HL,66H		810A	CD0280	4020		I N I T
80A4	11B980	2240	LD	DE,SAVE		810D	3E00	4030	CALL	INIT
80A7	010400	2260	LD	BC,4		810F	CD5980	4040	LD	A,0 ;SINGLE DENSITY
80AA	EDB0	2280	LDIR			8112	CD4080	4050	CALL	SETDEN
80AC	C9	2300	RET			8115	0000	4060	CALL	HOME
		2320						4070	DEFW	0 ;CORR.
80AD	21B980	2340	RE66	LD HL,SAVE				4080		
80B0	116600	2360	RE66HL	LD DE,66H		8117	3A0E0C	4090		S E E K
80B3	010400	2380	LD	BC,4		811A	CD4A80	4100	LD	A,(ARG2)
80B6	EDB0	2400	LDIR			811D	00	4110	CALL	SEEK
80B8	C9	2420	RET					4120	DEFB	00 ;CORR.
		2440						4130		
80B9	0004	2460	SAVE	DEFS 4				4140		S E T S E K
80BD	EDA2	2500	NMIR	INI		811E	3A100C	4150	LD	A,(#CL0) ;ARG3
80BF	ED45	2510	RETN			8121	CD5680	4160	CALL	SETSEK
80C1	EDA3	2520	NMIW	OUTI		8124	00	4170	DEFB	0 ;CORR.
80C3	ED45	2530	RETN					4180		
		2540						4190		R E A D
		2550				8125	210010	4200	LD	HL,1000H
80C5	E5	2560	DELAY	PUSH HL		8128	CD6B80	4210	CALL	READ
80C6	D5	2570		PUSH DE		812B	E7	4220	DEFB	0E7H
80C7	C5	2580		PUSH BC				4230		
80C8	0664	2590	DELAY1	LD B,100				4240		W R I T E
80CA	17	2600	DELAY2	RLA		812C	210010	4250	LD	HL,1000H
80CB	29	2610		ADD HL,HL		812F	CD8780	4260	CALL	WRITE
80CC	29	2620		ADD HL,HL		8132	E7	4270	DEFB	0E7H
80CD	05	2630		DEC B				4280		
80CE	C2CA80	2640		JP NZ,DELAY2				4290		
80D1	0D	2650		DEC C				4300		F O R M A T (hier OSBORNE SD/10Sekt)
80D2	C2C880	2660		JP NZ,DELAY1		8133	EF	4310	RST	28H
80D5	C1	2670		POP BC		8134	464F524D	4320	DEFM	'FORMATIEREN'
80D6	D1	2680		POP DE				41544945		
80D7	E1	2690		POP HL				52454E		
80D8	C9	2700		RET		813F	0D00	4330	DEFB	0DH,0
		2710				8141	CD0280	4335	CALL	INIT
		2720				8144	3E00	4336	LD	A,0 ;SD
		2730				8146	CD5980	4337	CALL	SETDEN
		2740				8149	CD4080	4340	CALL	HOME ; AUF SPUR 0
80D9	F5	2750	FLPINT	PUSH AF		814C	CDA180	4341	CALL	SAV66
80DA	C5	2760		PUSH BC		814F	21C180	4342	LD	HL,NMIW
80DB	E5	2770		PUSH HL		8152	CDB080	4343	CALL	RE66HL
80DC	DB0C	2780		IN A,(FDCSTA)		8155	AF	4350	F050	XOR A ; BEGINNE MIT SPUR 0
80DE	320981	2790		LD (FLPSTA),A		8156	32A781	4360	F100	LD (FTRK),A ; ZWISCHENSPP.
80E1	EF	2791		DEFB #EF		8159	CD4A80	4370	CALL	SEEK
80E2	0D	2792		DEFB #D		815C	CD6F81	4390	CALL	EXPTRK ; SPUR IM SP. AUFBAUEN
80E3	494E54	2793		DEFM "INT		815F	CDCC81	4400	CALL	WRITRK ; SPUR SCHREIBEN
80E6	00	2794		DEFB 0		8162	3AA781	4410	LD	A,(PTRK) ; NAECHSTE SPUR
80E7	210600	2800		LD HL,6		8165	3C	4420	INC	A
80EA	39	2810		ADD HL,SP		8166	FE28	4430	CP	40
80EB	4E	2830		LD C,(HL)		8168	20EC	4440	JR	NZ,FL00
80EC	23	2840		INC HL		816A	CDAD80	4441	CALL	RE66
80ED	46	2850		LD B,(HL)		816D	DF5B	4450	DEFB	0DFH,5BH
80EE	0A	2860		LD A,(BC)				4470		
80EF	FE18	2870		CP 18H				4480		
80F1	200A	2880		JR NZ,NOLOOP				4490		EXPAND THE FORMAT-TABLE IN MEMORY
80F3	03	2890		INC BC				4500		
80F4	0A	2900		LD A,(BC)		816F	110010	4510	EXPTRK	LD DE,BUF
80F5	FEFE	2910		CP 0FEH		8172	219F81	4520	LD	HL,TRKTAB
						8175	CD8F81	4530	CALL	EXPAND ; INDEX PART

```

8178 010100 4540 LD BC,1 ; SEKT. 1
817B 21C181 4550 TRRSEC LD HL,SEKTAF-1
817E 09 4560 ADD HL,BC
817F 7E 4570 LD A,(HL)
8180 32AB81 4580 LD (SECNR),A
8183 21A281 4590 LD HL,TRKSTB
8186 CD8F81 4600 CALL EXPAND ; EXPAND SEKTOR PART
8189 0C 4610 INC C
818A 79 4620 LD A,C
818B FE0B 4630 CP 10+1 ; OSBORNE 10 SEKTOREN
818D 38EC 4640 JR C,TRKSEC
4650 ;
818F 7E 4660 EXPAND LD A,(HL)
8190 23 4670 INC HL
8191 B7 4680 OR A
8192 C8 4690 RET Z
8193 C5 4700 PUSH BC
8194 47 4710 LD B,A
8195 7E 4720 LD A,(HL)
8196 23 4730 JNC HL
8197 12 4740 EXPA LD (DE),A
8198 13 4750 INC DE
8199 05 4760 DEC B
819A 20FB 4770 JR NZ,EXPA
819C C1 4780 POP BC
819D 18F0 4790 JR EXPAND
4800 ;
4810 ;
819F 20FF 4820 TRKTAB DEFB 32,#FF ;INDEX GAP
81A1 00 4822 DEFB 0
81A2 0600 4824 TRKSTB DEFB 6,0 ;SYNC
81A4 01FE 4830 DEFB 1,0FEH ;ADR.MARK
81A6 01 4840 DEFB 1
81A7 00 4850 FTRK DEFB 0 ;TRACK NUMBER
81A8 0100 4860 DEFB 1,0 ;ZERO
81AA 01 4862 DEFB 1
81AB 00 4864 SECNR DEFB 0 ;SEKTOR NUMBER
81AC 0101 4866 DEFB 1,1 ;SEKTORLAENGE OSBORNE 256
81AE 01F7 4870 DEFB 1,0F7H ;CRC
81B0 0BFF 4880 DEFB 11,#FF ;GAP2
81B2 0600 4885 DEFB 6,0 ;SYNC
81B4 01FB 4890 DEFB 1,0FBH ;DATA ADDR.MARK
81B6 FFE5 4900 DEFB 255,0ESH ;DATA
81B8 01E5 4910 DEFB 1,0E5H ;255+1
81BA 01F7 4920 DEFB 1,0F7H ;CRC
81BC 11FF 4930 DEFB 17,#FF ;GAP3
81BE 00 4940 DEFB 0
81BF 6EFF 4950 DEFB 110,#FF ;GAP4
81C1 00 4960 DEFB 0
4970 ;
4980 ;
4990 ;TABELLE DER SEKTOR REIHENFOLGE
81C2 01020304 5000 SEKTAF DEFB 1,2,3,4,5,6,7,8,9,10
05060708
090A
5020 ;
5030 ;
5040 ;
5050 ; GANZE SPUR ZUM FORMATIEREN SCHREIBEN
5060 ;
81CC 0E0F 5100 WRTRK LD C,FDCDAT
81CE 3EF4 5110 LD A,FWRTRK ;WRITE A TRACK COMMAND
81D0 FB 5115 EI
81D1 210010 5117 LD HL,BUF
81D4 D30C 5120 OUT (FDCCMD),A
81D6 18FE 5130 WRTRW JR WRTRW
81D8 C9 5160 RET
81D9 1000 5200 BUF EQU 1000H ;BUFFER ZUM FORMATIEREN

```



Die abgebildete Platine können Sie fertig gebohrt und durchkontaktiert bei mir bestellen. Inklusive der beiden programmierten Steuerproms für DM 60,- (MwSt, Verpackung und Porto eingeschlossen).

Die Überweisung des Betrags auf das Konto Gabi Böhm Konto 148029-751 PSchA Kirh mit dem Hinweis "Floppy" genügt als Bestellung.

# FDC-Aufbau

von DIETER OBERLE

Änderungen am FDC-Schaltplan 7/8-83

Anscheinend ist es nie möglich, eine Hardware-schaltung mit Niveau ohne Nachlese zu veröffentlichen. Im Falle des Floppy-Controllers ist das nicht so schlimm, da die Platine erst jetzt erhältlich ist und die Änderungen noch vor der frustrierenden Fehlersuche vorgenommen werden können.

Nur wenige Änderungen betreffen die Platine selbst; hauptsächlich handelt es sich um leicht änderbare Korrekturen im Schaltplan oder Bestückungen, die nicht vorgenommen werden müssen. Die wirklichen Hardwareänderungen sind (z. B. auf der Rückseite der Platine) leicht vorzunehmen.

1. Pull-up Widerstand 1K an 7438 (6) fehlt im Lay-out (einlöten)
2. Pull-up Widerstand R24 (1K) WD 1793 WE (2) gegen +5V. Im Layout ist dieser Widerstand an RE angeschlossen (4), muß aber an (2) angelegt werden wie angegeben (ändern).
3. Im Schaltplan ist R17 (1K) Pull-up an 74LS241 (14,13) gegen +5V noch nicht eingezeichnet (nachtragen). Im Layout ist dieser Widerstand schon vorgesehen.
4. Im Schaltplan fehlt eine Verbindungsleitung von P2 (2) nach DMA (38) IEI. (einzeichnen).
5. In der Stückliste müssen noch R20 (1k) und C8 (100 pF) nachgetragen werden.
6. Die beiden Tiefpässe an P1 (11) und P1 (12) 100 Ω /100 pF entfallen und sind im Layout nicht vorhanden.
7. 470K Widerstand und Quarz am 9229 darf nicht bestückt werden.

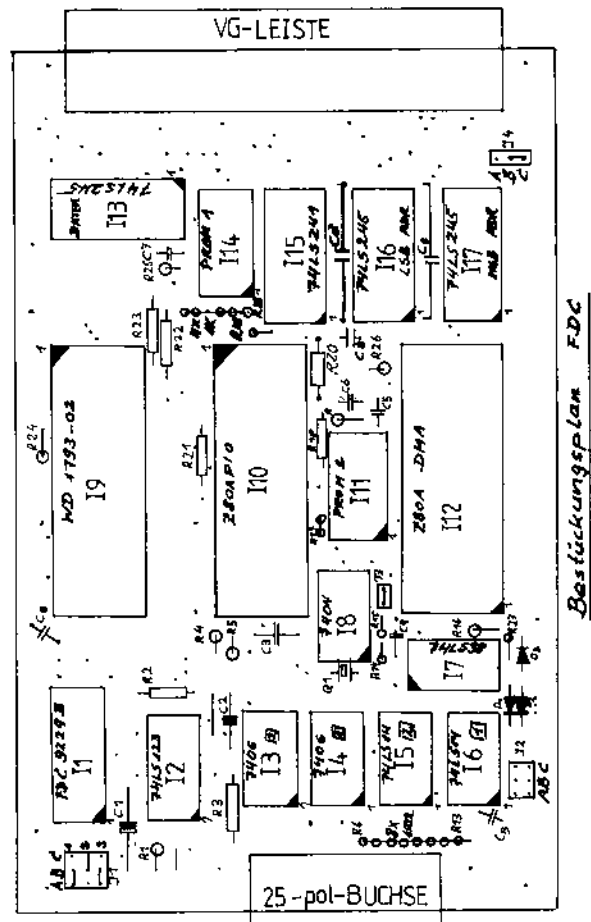
Die Zuordnung der Widerstände und Kondensatoren kann mit Hilfe des Bestückungsplanes und des Layouts leicht in den Schaltplan übertragen werden, falls dies notwendig ist. Das Layout war zum Erscheinungszeitpunkt des letzten Heftes leider noch nicht fertig, sonst hätten die Änderungen noch vor Abdruck in den Schaltplan eingetragen werden können.

Hier nun die genaue Liste der Bauteile, die im letzten Heft dem Schaltplan noch nicht eindeutig zugewiesen wurden:

R6 - R13 8fach SIL 150 Ω  
 R16, R23 10K  
 R3, R4, R5, R17, R22, R24, R25, R26, R28, R29 u. 4\*1K (SIL) 1K  
 R18, R19, R21 100 Ω  
 R14, R15 330 Ω  
 R1, R2 220K

CB 4\* 100µF  
 C4 10 µF  
 C3 1 µF  
 C5, C6, C7 100pF  
 C2 15 µF  
 C1 47 µF  
 Q1 16MHz Quarz

Zur Treibersoftware erfahren Sie etwas in dem Artikel "Seite für Floppy-Einsteiger" an anderer Stelle dieses Heftes. Als Programmgrundlage dazu dient das gut kommentierte Listing in Heft 5/83 Seite 10. (Beachten: die Zeile "inr hl ;niederer Byte von Adresse" in der Interruptroutine muß gestrichen werden!) Red.



# RTTY II

von JÖRG WITTICH

Der Grundgedanke bei der Originalversion des im letzten Heft abgedruckten RTTY-Programms war, dass es ohne Hardwareänderungen auf der Grundplatte des Nascom 1 laufen koennen sollte. Durch diese Beschaerung liessen sich jedoch solche Annehmlichkeiten wie Vorschreiben bei Empfang, Abruf von Festtexten, ASCII/Baudot-Umschaltung usw. nicht realisieren. Das hier vorgestellte Programm bietet diesen Komfort, erfordert aber eine Hardwaremodifikation. Fuer die seriell/parallel- und parall./ser.-Wandlung befindet sich auf dem Nascom bereits ein spezieller Baustein, der UART. Dieser UART koennte prinzipiell direkt ueber den Bus fuer verschiedene Formate, z.B. 5 Bit-Daten und 1,5 Stopbits, wie es bei Baudot ueblich ist, programmiert werden. Leider ist diese Moeglichkeit von den Entwicklern des Nascom nicht vorgesehen worden und der UART fest fuer 8 Bit-Daten verdrahtet. Ich habe daher SBS (Pin 36) direkt und CLS 1, CLS 2 (Pin 37,38) ueber einen Inverter (z.B. 74LS04 oder Transistorinverter) mit Bit 2 von Port 0 verbunden. Die alten Verbindungen muessen natuerlich auf irgendeine Weise aufgetrennt werden. Nach dem Einschalten oder Reset stellt sich automatisch das vom Cassetten-Interface benoetigte Format ein. Zum Einstellen der Baudrate verwende ich Kanal 1 eines Z80A-CTCs (ZC/T01 an die entsprechenden Clock-eingaenge des UARTs). Wer keinen CTC besitzt, kann die notwendigen Clockfrequenzen auch z.B. mit einem NE 555 erzeugen und die Baudraten von Hand umschalten. Das Programm kennt folgende Befehle, die bis auf die ersten beiden sowohl bei Empfangs- als auch bei Sendebetrieb Gueltigkeit haben:

Ctrl A: ASCII  
Ctrl B: Baudot  
Ctrl C bei Baudot: Aendern der Bu/Zi-Ebene  
bei ASCII: K1-Option von  
NAS-SYS ein/ausschalten  
Ctrl L: Baudrate erniedrigen (45,45,50,75,  
100,110,150,300,600 Baud)  
Ctrl I: Baudrate erhoehen

Ctrl P: Text 1  
Ctrl Q: Text 2  
Ctrl R: Text 3  
Ctrl S: Text 4  
Ctrl T: Text 5  
Ctrl Z: Sender ein/(aus), Bit 4 von Port 0  
kann zum Schalten des Senders verwendet werden.

Der Vorschreibspeicher ist etwa 2 kByte lang und beginnt bei 17D8H, der Stack liegt bei 1000H. Festtexte koennen beliebig lang sein und muessen mit einer 0 abgeschlossen werden. Bei 0DH (CR) wird immer auch ein 0AH (LF) ausgesendet. Die Anfangsadressen der 5 Texte muessen zwischen 0F00H und 0F09H eingetragen werden. Den Abruf von nicht definierten Texten sollte man vermeiden. Fuer die Texte 2-5 koennen auch die entsprechenden Cursor-Tasten verwendet werden. Bei Erreichen eines 'a' im Vorschreibspeicher schaltet das Programm nach einer kurzen Verzoegerungszeit auf Empfang um. Falls nicht von Hand gegeben, fuegt das Programm am Ende einer RTTY-Zeile automatisch CR/LF ein. Am Beginn einer neuen Zeile wird bei Baudot immer die Ebenenken- nung gegeben, um die Synchronisation mit der Gegenstation sicherzustellen. Im Assemblerlisting befindet sich eine Zeile mit SCAL 7DH, welches NAS-SYS 1 Benutzer durch SCAL 61H ersetzen muessen.

## Z80 Assembler - Source Listing

```
0010 ;RTTY-PROGRAMM
0020 ;(c) Joerg Wittich
0030 ;9-83
0040 ;
0050 ORG 0C80H
0060 ENT
0070 LD A,0CH
0080 RST 30H
0090 LD HL,1800H-40
0100 LD (HL),20H
0110 LD DE,1800H-39
0120 LD BC,40
0130 LDIR
0140 LD (VZEI),HL
0150 LD DE,4400H
0160 XOR A
0170 RCAL RT0
0180 LD A,2
0190 RCAL ASCII
0200 EX AF,AF'
0210 LOOP LD HL,0C00H
0220 BIT 4,(HL)
0230 JP NZ,SEND
0240 SCAL 70H
0250 JR NC,JTAST
0260 LD HL,0BF4H
0270 BIT 1,(HL)
0280 JR Z,ASK
0290 AND 1FH
0300 CP 1FH
```

QCB5 2004	Q310	JR	NZ, NOBU	QD69 2C003A28	1120	DEFB	2CH, 0, 3AH, 28H
QCB7 3642	Q320	LD	(HL), "B	QD6D 352B2932	1130	DEFB	35H, 2BH, 29H, 32H
QCB9 1827	Q330	JR	JTAST	QD71 08363031	1140	DEFB	8, 36H, 30H, 31H
QCB8 FE1B	Q340	NOBU	CP	1BH	1150	DEFB	39H, 3FH, 0, 0
QCB8 2004	Q350	JR	NZ, NOZI	QD79 2E2F3D00	1160	DEFB	2EH, 2FH, 3DH, 0
QCBF 3662	Q360	LD	(HL), "b	QD7D 07080A0D	1170	CRLCHR	DEFB 7, 8, 0AH, 0DH
QCC1 181F	Q370	JR	JTAST	QD81 10111213	1180	DEFB	10H, 11H, 12H, 13H
QCC3 0600	Q380	NOZI	LD	B, 0	1190	DEFB	14H
QCC5 4F	Q390	LD	C, A	QD86 203435FB	1200	RTTAB	DEFB 20H, 34H, 35H, 0FBH
QCC6 CB6E	Q400	BIT	5, (HL)	QD8A 203530FA	1210	DEFB	20H, 35H, 30H, 0FAH
QCC8 213D0D	Q410	LD	HL, CODTAB	QD8E 203735B6	1220	DEFB	20H, 37H, 35H, 0B6H
QCCB 2802	Q420	JR	Z, MOZ	QD92 31303082	1230	DEFB	31H, 30H, 30H, 82H
QCCD CBE9	Q430	SET	5, C	QD96 31313074	1240	DEFB	31H, 31H, 30H, 74H
QCCF 09	Q440	MOZ	ADD	HL, BC	1250	DEFB	31H, 35H, 30H, 4EH
QCD0 7E	Q450	LD	A, (HL)	QD9E 3330301A	1260	DEFB	33H, 30H, 30H, 1AH
QCD1 217D0D	Q460	ASK	LD	HL, CRLCHR	1270	DEFB	36H, 30H, 30H, 0
QCD4 E67F	Q470	AND	7FH	QDA6 1C	1280	BRATE	DEFB 28
QCD6 FE20	Q480	CP	20H	QDA7 0002	1290	VZEI	DEFS 2
QCD8 3007	Q490	JR	NC, OUT	QDA9 0002	1300	TEXZEI	DEFS 2
QCD8 010400	Q500	LD	BC, 4	QDAB D5	1310	TAST	PUSH DE
QCD8 EDB1	Q510	CPIR		QDAC DF7D	1320	SCAL	7DH
QCDF 2001	Q520	JR	NZ, JTAST	QDAE DL	1330	POP	DE
QCE1 F7	Q530	OUT	RST	30H	1340	JP	NC, LOOP
QCE2 C3AB0D	Q540	JTAST	JP	TAST	1350	CP	9
QCE5 3AF40B	Q550	CHANGE	LD	A, (0BF4H)	1360	CALL	Z, RATE
QCE8 EE20	Q560	XOR	20H	QDB7 FE04	1370	CP	4
QCEA 32F40B	Q570	LD	(0BF4H), A	QDB9 C0C00D	1380	CALL	Z, RATE
QCED CB47	Q580	BIT	0, A	QDBC FE01	1390	CP	1
QCEF 2808	Q590	JR	Z, CMA	QDBE CCFB0C	1400	CALL	Z, ASCII
QCF1 3A270C	Q600	LD	A, (0C27H)	QDC1 FF02	1410	CP	2
QCF4 EE01	Q610	XOR	1	QDC3 CCFB0C	1420	CALL	Z, ASCII
QCF6 32270C	Q620	LD	(0C27H), A	QDC6 FE03	1430	CP	3
QCF9 AF	Q630	CMA	XOR	A	1440	CALL	Z, CHANGE
QCF9 C9	Q640	RET		QDC8 CCE50C	1450	CP	1AH
QCFB C640	Q650	ASCII	ADD	A, 40H	1460	JR	NZ, NOS
QCFD 32F40B	Q660	LD	(0BF4H), A	QDCF DF5F	1470	SCAL	5FH
QD00 17	Q670	RLA		QDD1 AF	1480	XOR	A
QD01 E604	Q680	AND	4	QDD2 FE20	1490	NOS	CP
QD03 32000C	Q690	LD	(#C00), A	QDD4 300A	1500	CP	20H
QD06 00	Q700	NOP		QDD6 217D0D	1510	JR	NC, OK
QD07 AF	Q710	XOR	A	QDD9 010900	1520	LD	HL, CRLCHR
QD08 32270C	Q720	LD	(0C27H), A	QDDC EDB1	1530	LD	BC, 9
QD0B C9	Q730	RET		QDDE 2025	1540	CPIR	
QD0C 1F	Q740	RATE	RRA	QDE0 2AA70D	1550	OK	JR
QD0D 3AA60D	Q750	LD	A, (BRATE)	QDE3 FE08	1560	LD	HL, (VZEI)
QD10 3802	Q760	JR	C, ERH	QDE5 200A	1570	CP	8
QD12 D608	Q770	SUB	8	QDE7 2B	1580	JR	NZ, NOBS
QD14 C604	Q780	ERH	ADD	A, 4	1590	DEC	HL
QD16 E61F	Q790	AND	1FH	QDE8 CB5C	1590	BIT	3, H
QD18 21860D	Q800	RT0	LD	HL, RTTAB	1600	INC	HL
QD1B D5	Q810	PUSH	DE	QDEB 2803	1610	JR	Z, NOBS1
QD1C 32A60D	Q820	LD	(BRATE), A	QDED 2B	1620	DEC	HL
QD1F 1600	Q830	LD	D, 0	QDEE 1807	1630	JR	ROLL
QD21 5F	Q840	LD	E, A	QDF0 14	1640	NOSB1	INC
QD22 19	Q850	ADD	HL, DE	QDF1 CB6C	1650	NOSB	BIT
QD23 11F60B	Q860	LD	DE, 0BF6H	QDF3 2010	1660	JR	NZ, JLOOP
QD26 010300	Q870	LD	BC, 3	QDF5 77	1670	LD	(HL), A
QD29 EDB0	Q880	LDIR		QDF6 23	1680	INC	HL
QD2B D1	Q890	POP	DE	QDF7 22A70D	1690	ROLL	LD
QD2C 7E	Q900	LD	A, (HL)	QDFA 2B	1700	LD	HL, (VZEI), HL
QD2D C61A	Q910	ADD	A, 1AH	QDFB D5	1710	PUSH	DE
QD2F F5	Q920	PUSH	AF	QDFC 11F10B	1720	LD	DE, 0BC9H+40
QD30 3E25	Q930	LD	A, 25H	QDFE 012800	1730	LD	BC, 40
QD32 3802	Q940	JR	C, RAB	QE02 EDB8	1740	LDDR	
QD34 3E05	Q950	LD	A, 5	QE04 DL	1750	POP	DE
QD36 D309	Q960	RAB	OUT	(9), A	1760	JLOOP	JP
QD38 F1	Q970	POP	AF	QE08 DB02	1770	SEND	IN
QD39 D309	Q980	OUT	(9), A	QE0A CB77	1780	BIT	6, A
QD3B AF	Q990	XOR	A	QE0C 289D	1790	JR	Z, TAST
QD3C C9	1000	RET		QE0E CB43	1800	BIT	0, E
QD3D 00450D41	1010	CODTAB	DEFB	0, 45H, 0DH, 41H	1810	JR	Z, MA1
QD41 20534955	1020	DEFB	20H, 53H, 49H, 55H	QE12 3E0D	1820	LD	A, 0DH
QD45 1744524A	1030	DEFB	17H, 44H, 52H, 4AH	QE14 184B	1830	JR	TX
QD49 4E46434B	1040	DEFB	4EH, 46H, 43H, 4BH	QE16 CB4B	1840	MAL	BIT
QD4D 545A4C57	1050	DEFB	5AH, 5AH, 4CH, 57H	QE18 2806	1850	JR	Z, MA2
QD51 48595051	1060	DEFB	48H, 59H, 50H, 51H	QE1A 3E0A	1860	LD	A, 0AH
QD55 4F424700	1070	DEFB	4FH, 42H, 47H, 0	QE1C CB8B	1870	RES	1, E
QD59 4D585600	1080	DEFB	4DH, 58H, 56H, 0	QE1E 1841	1880	JR	TX
QD5D 00330D2D	1090	DEFB	0, 33H, 0DH, 2DH	QE20 AF	1890	MA2	XOR
QD6J 20273837	1100	DEFB	20H, 27H, 38H, 37H	QE21 08	1900	EX	AF, AF'
QD65 17003407	1110	DEFB	17H, 0, 34H, 7	QE22 B7	1910	OR	A
				QE23 2047	1920	JR	NZ, NOMAL



```

QE25 CB53 1930 BIT 2,E
QE27 2810 JR 2,MA3
QE29 2AA90D 1950 LD HL,(TEXZEI)
QE2C 7E 1960 LD A,(HL)
QE2D 23 1970 INC HL
QE2F 22A90D 1980 LD (TEXZEI),HL
QE31 B7 1990 OR A
QE32 202D 2000 JR NZ, TX
QE34 CB93 2010 RES 2,E
QE36 C3AB0D 2020 JP TAST
QE39 2AA70D 2030 MA3 LD HL,(VZEI)
QE3C 2B 2040 DEC HL
QE3D CB5C 2050 BIT 3,H
QE3F CAAB0D 2060 JP 2,TAST
QE42 3A0018 2070 LD A,(1800H)
QE45 22A70D 2080 LD (VZEI),HL
QE48 23 2090 INC HL
QE49 D5 2100 PUSH DE
QE4A 11D817 2110 LD DE,1800H-40
QE4D B7 2120 OR A
QE4F ED52 2130 SBC HL,DE
QE50 44 2140 LD B,H
QE51 4D 2150 LD C,L
QE52 21D917 2160 LD HL,1800H-39
QE55 EDB0 2170 LDIR
QE57 D1 2180 POP DE
QE58 FE10 2190 CP 10H
QE5A 3805 2200 JR C, TX
QE5C FE15 2210 CP 15H
QE5E DAE50E 2220 JP C,TEXT
QE61 FE40 2230 TX CP 40H
QE63 2006 2240 JR NZ,NOMA
QE65 DF5D 2250 SCAL 5DH
QE67 DF5F 2260 SCAL 5FH
QE69 1877 2270 JR JAST
QE6B F7 2280 NOMA RST 30H
QE6C FE0D 2290 NOMAL CP 0DH
QE6E 2006 2300 JR NZ,MA4
QE70 1644 2310 LD D,68
QE72 CBCB 2320 SET 1,E
QE74 CB83 2330 RES 0,E
QE76 FE20 2340 MA4 CP 20H
QE78 200B 2350 JR NZ,MA5
QE7A 7A 2360 LD A,D
QE7B FE08 2370 CP 8
QE7D 3E20 2380 LD A,20H
QE7F 3004 2390 JR NC,MA5
QE81 CBC3 2400 SET 0,E
QE83 185D 2410 JR JAST
QE85 F5 2420 MA5 PUSH AF
QE86 3E44 2430 LD A,68
QE88 BA 2440 CP D
QE89 3002 2450 JR NC,MA6
QE8B CBC3 2460 SET 0,E
QE8D F1 2470 MA6 POP AF
QE8E 21F40B 2480 LD HL,0BF4H
QE91 CB46 2490 BIT 0,(HL)
QE93 204A 2500 JR NZ,ASS
QE95 FE0D 2510 CP 0DH
QE97 2006 2520 JR NZ,MA7
QE99 3E08 2530 LD A,8
QE9B CBDB 2540 SET 3,E
QE9D 1840 2550 JR ASS
QE9F FE0A 2560 MA7 CP 0AH
QE A1 2004 2570 JR NZ,MA8
QE A3 3E02 2580 LD A,2
QE A5 1838 2590 JR ASS
QE A7 FE20 2600 MA8 CP 20H
QE A9 2004 2610 JR NZ,MA9
QE AB 3E04 2620 LD A,4
QE AD 1830 2630 JR ASS
QE AF 217C0D 2640 MA9 LD HL,CODTAB+3FH
QE B2 014000 2650 LD BC,40H
QE B5 EDB9 2660 CPDR
QE B7 2029 2670 JR NZ,JAST
QE B9 47 2680 LD B,A
QE BA 3AF40B 2690 LD A,(0BF4H)
QE BD A9 2700 XOR C
QE BF CB6F 2710 BIT 5,A
QE C0 79 2720 LD A,C
QE C1 CBAF 2730 RES 5,A

```

```

QEC3 2006 2740 JR NZ,OAE
QEC5 CB5B 2750 BIT 3,E
QEC7 2816 2760 JR 2,ASS
QEC9 1803 2770 JR SHIFT
QECB CDE50C 2780 OAE CALL CHANGE
QECE 21F40B 2790 SHIFT LD HL,0BF4H
QED1 78 2800 LD A,B
QED2 08 2810 EX AF,AF'
QED3 CB9B 2820 RES 3,E
QED5 3E1F 2830 LD A,1FH
QED7 CB6E 2840 BIT 5,(HL)
QED9 2805 2850 JR 2,MX
QEDB 3E1B 2860 LD A,1BH
QEDD 1801 2870 JR MX
QEDF 15 2880 ASS DEC D
QEE0 D301 2890 MX OUT (1),A
QEE2 C3AB0D 2900 JAST JP TAST
QEE5 CBA7 2910 TEXT RES 4,A
QEE7 87 2920 ADD A,A
QEE8 260F 2930 LD H,#F
QEEA 6F 2940 LD L,A
QEEB 4E 2950 LD C,(HL)
QEEC 23 2960 INC HL
QEE D 46 2970 LD B,(HL)
QEE E ED43A90D 2980 LD (TEXZEI),BC
QEF2 CBD3 2990 SET 2,E
QEF4 18EC 3000 JR JAST

```

#### Nachtrag zum RTTY-Programm 7/8-83

Durch das Umschreiben des Programms für NASSYS ist es etwas länger geworden, sodaß es bei zwischenzeitlicher Benutzung anderer Programme durch den Stack überschrieben wird.

Dies ist vor Ausdruck des Listings geschehen, und so müssen wir die überschriebenen Bytes nachliefern.

FE0 0F 0F 0F C3 2F 0D

Vom Autor kamen noch einige kosmetische Korrekturen, die allerdings mit dem Fuktionieren des Programms selbst nichts zu tun haben. Hier die nachgelieferte Fassung mit Anpassung auf 1760 Hz (genau 1748,25 Hz).

D90 20 20 20 20

D9C 31

E03 26

E07 23

E0B 17

E0F 11

E62 BF 99 16 02

EBC 26

EC1 00

DAE 34 35

DBD 55 54 43

Nachdem einige Leser, die das ursprüngliche T4-Programm nicht besitzen, auch an der Anwendung des RTTY-Programms interessiert sind, wollen wir nochmals kurz auf den Anschluß der Hardware eingehen (wobei man

beachten sollte, daß das RTTY-Programm aus diesem Heft weit luxuriöser in der Anwendung ist).

Start: EC80 HHMM -Uhrzeit

Befehle bei Empfang:

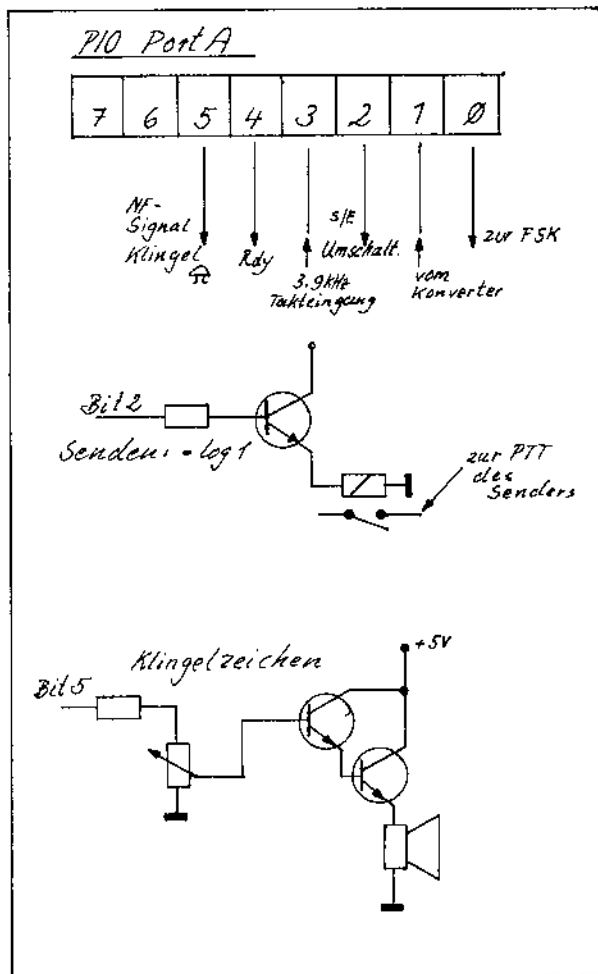
- a Umschalten auf Senden
- C Ändern der Ziffern-/Buchstabenebene
- I Inverse Schrift
- N Normale Schrift
- Q QRM- Betrieb ein/aus (Bu/Zi vor jedem Zeichen)
- + Baudrate erhöhen (45,45, 50, 75, 100)
- Baudrate erniedrigen

Befehle bei Senden:

- a Umschalten auf Empfang
- % Wiederholung einer Zeile mit WR/ZV
- & Klingel

Nach spätestens 68 Zeichen automatisch WR/ZV, wenn nicht von Hand gegeben (NL).

Und hier noch der Anschluß der benötigten Hardware:



# I/O-Interface

von KARL SCHULMEISTER

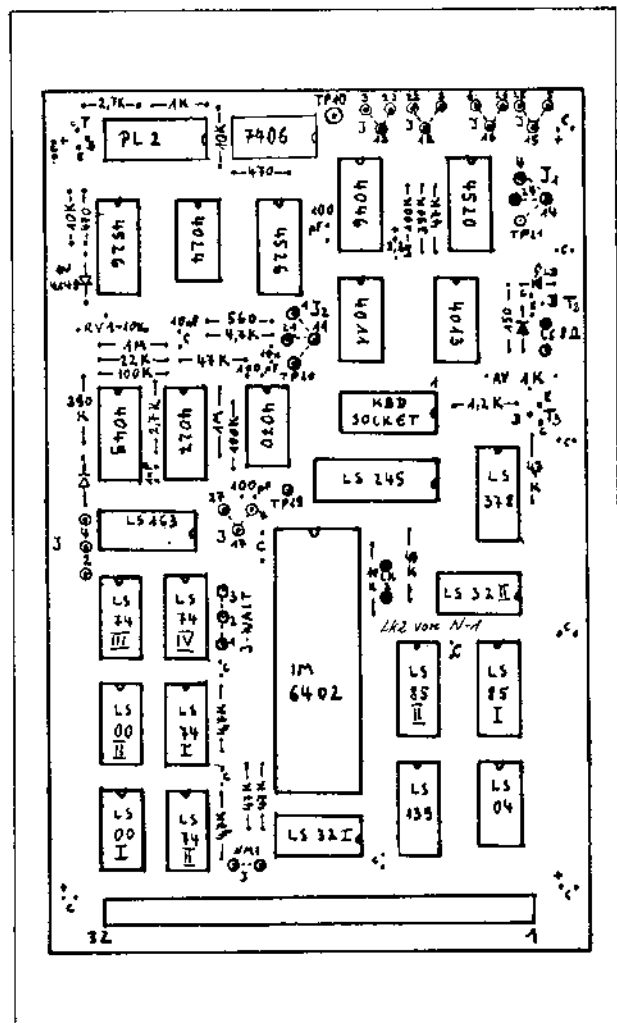
Tastaturanschluß und NASCOM-2 Kassetteninterface-ECB Karte:

1. Diese Karte wurde bezüglich des Kassetteninterface genau nach dem Schaltplan des NASCOM-2 gebaut. Die Baud-Rate ist jeweils über eine Steckbrücke für INPUT bzw. OUTPUT auf 300, 1200 oder 2400 Baud einzustellen.

Da die Karte den Takt für den UART aus dem CPU Takt gewinnt und mit 500 kHz weiterarbeitet, ist es wichtig, den CPU Takt bei 2 MHz mit Pin 12 des LS 163, bei 4 MHz mit Pin 11 zu verbinden (gegenüberliegende Lötunkte unter LS 163).

2. Der Anschluß des NASCOM-Keyboards erfolgt über den 16 poligen KBD-Socket, die Belegung entspricht dem NASCOM-1 Schaltplan. Außerdem wurde das Bit 2 an einen Transistor geführt und kann so einen kleinen Lautsprecher ansteuern.

3. Auf der Karte werden WAIT-Signale erzeugt (Schaltung gem. techn. Manual von ZILOG). Es ist das WAIT-Signal für M1 oder MRQ jeweils über eine Steckbrücke ansprechbar.



Informationen zum Bestückungsplan:

T = Schalttransistor PNP

T2 = BC239

T3 = BC517

C = Abblockkondensator 10n - 4,7uF

Wichtig: Jumper 4 bzw. 2 je nach Systemtakt setzen!

3000 Bd: J1 14-4

J2 11-1

1200 Bd: J1 14-24

J2 11-21

2400 Bd: J1 u. J2 mit TP 20,21

J-NMI setzen für SINGLE-STEP NASCOM1

J-WAIT: 1-2 MRQ-Zyklus

2-3 M1-Zyklus

Als Jumper-Material können IC-Einzelfassungen und Abfalldraht verwendet werden.

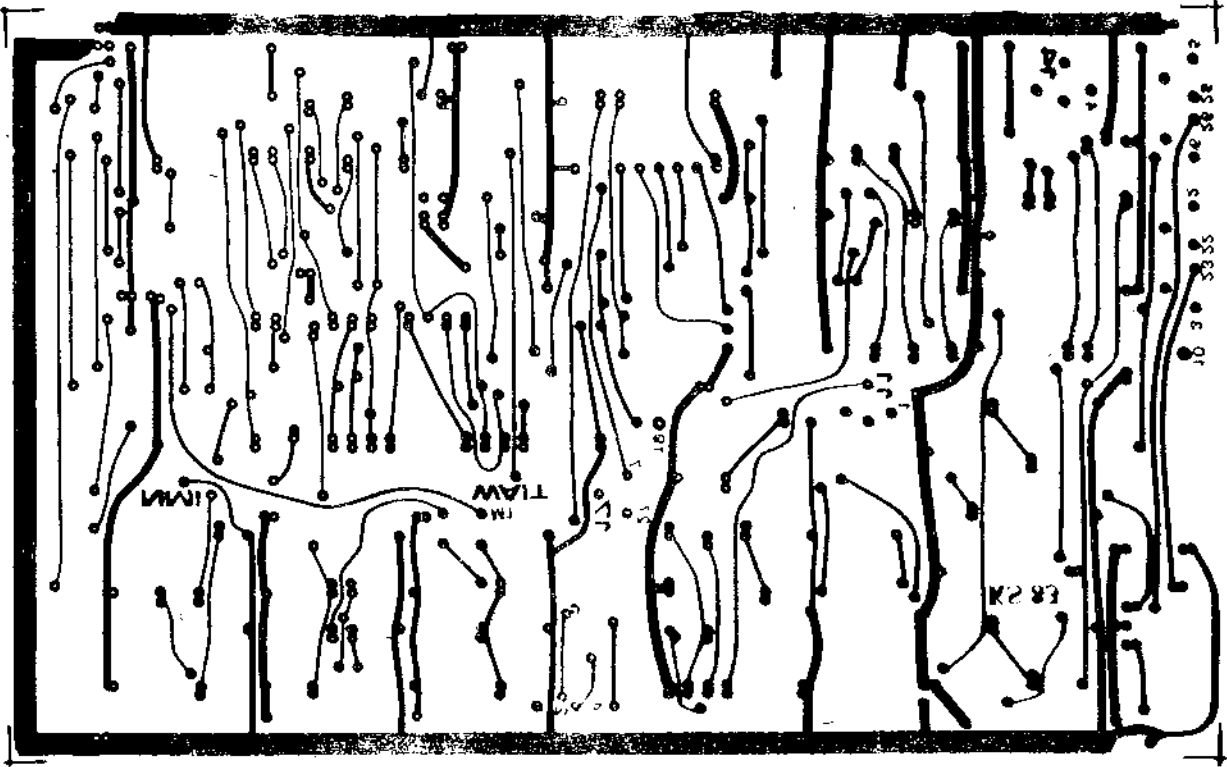


# Interr.-Uhr

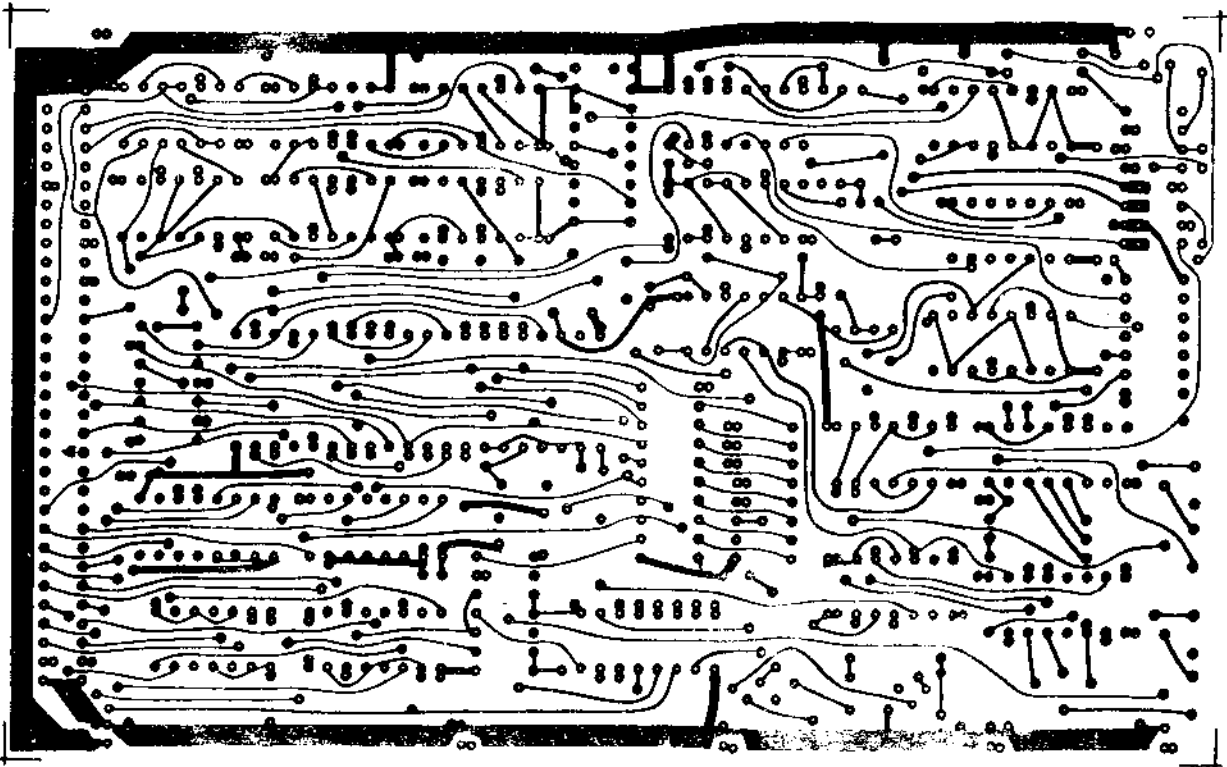
von JÖRG WITTICH

Dieses Programm zeigt ständig in der Kopfzeile die aktuelle Uhrzeit an. NAS-SYS (i oder 3), ZEAP, Basic usw. laufen davon unbeeinflusst weiter. Lediglich beim Basic ist zu beachten, dass ein Kaltstart bei der Adresse E004H durchgeführt werden muss, da sonst alle Interrupts gesperrt werden und die Uhr dadurch stehenbleibt. Das Maschinenprogramm ist im RAM voll verschieblich, sofern sich die Anfangsadresse durch 8 teilen lässt. Die Uhrzeit wird beim Start des Programms als zweiter Parameter (Stunde und Minute) angegeben. Gesteuert wird die Uhr von einem 1 Sekunden-Takt, der von zwei hintereinandergeschalteten CTC-Kanälen erzeugt wird (4 MHz Systemtakt).

0C80	0010	ORG	0C80H
0C80 DF	0020 STU	RST	18H
0C81 60	0030	DEFB	60H
0C82 7D	0040	LD	A,L
0C83 E607	0050	AND	7
0C85 2804	0060	JR	Z,OK
0C87 DF	0070	RST	18H
0C88 6B	0080	DEFB	6BH
0C89 DF	0090	RST	18H
0C8A 5B	0100	DEFB	5BH
0C8B 013800	0110 OK	LD	BC,ISR-STU-1
0C8E 09	0120	ADD	HL,BC
0C8F 7D	0130	LD	A,L
0C90 D308	0140	OUT	(8),A
0C92 23	0150	INC	HL
0C93 72	0160	LD	(HL),D
0C94 23	0170	INC	HL
0C95 73	0180	LD	(HL),E
0C96 23	0190	INC	HL
0C97 3600	0200	LD	(HL),0
0C99 23	0210	INC	HL
0C9A 54	0220	LD	D,H
0C9B 5D	0230	LD	E,L
0C9C 23	0240	INC	HL
0C9D 23	0250	INC	HL
0C9E 7C	0260	LD	A,H
0C9F ED47	0270	LD	I,A
0CA1 ED5E	0280	IM	2
0CA3 73	0290	LD	(HL),E
0CA4 23	0300	INC	HL
0CA5 72	0310	LD	(HL),D
0CA6 3E37	0320	LD	A,37H
0CA8 D30A	0330	OUT	(0AH),A
0CAA 3E7D	0340	LD	A,125
0CAC D30A	0350	OUT	(0AH),A
0CAE 3EF7	0360	LD	A,0F7H
0CB0 D30B	0370	OUT	(0BH),A
0CB2 3E7D	0380	LD	A,125
0CB4 D30B	0390	OUT	(0BH),A
0CB6 FB	0400	EI	
0CB7 DF	0410	RST	18H
0CB8 5B	0420	DEFB	5BH
0CB9 0003	0430 ISR	DEFS	3
0CBC E5	0440	PUSH	HL
0CBD 210000	0450	LD	HL,0
0CC0 D5	0460	PUSH	DE
0CC1 C5	0470	PUSH	BC
0CC2 F5	0480	PUSH	AF
0CC3 016003	0490	LD	BC,360H
0CC6 AF	0500	XOR	A
0CC7 2B	0510 LOOP	DEC	HL
0CC8 380D	0520	JR	C,FUN
0CCA 3C	0530	INC	A
0CCB B8	0540	CP	B
0CCC 2002	0550	JR	NZ,DG
0CCE 0E24	0560	LD	C,24H
0CD0 86	0570 DG	ADD	A,(HL)
0CD1 27	0580	DAA	
0CD2 B9	0590	CP	C
0CD3 2001	0600	JR	NZ,FAN
0CD5 AF	0610	XOR	A
0CD6 77	0620 FAN	LD	(HL),A
0CD7 10EE	0630 FUN	DJNZ	LOOP
0CD9 11ED0B	0640	LD	DE,0BEDH
0CDC 013A03	0650	LD	BC,33AH
0CDF 79	0660	LD	A,C
0CE0 13	0670 PA	INC	DE
0CE1 ED6F	0680	RLD	
0CE3 B9	0690	CP	C
0CE4 2803	0700	JR	Z,PQ
0CE6 12	0710 PP	LD	(DE),A
0CE7 18F7	0720	JR	PA
0CE9 23	0730 PQ	INC	HL
0CEA 10FA	0740	DJNZ	PP
0CEC F1	0750	POP	AF
0CED C1	0760	POP	BC
0CEE C1	0770	POP	DE
0CEF E1	0780	POP	HL
0CF0 FB	0790	EI	
0CF1 ED4D	0800	RETI	



IN/OUT-KARTE NASCOM ECB  
 (Tastaturanschluß, Cassetteninterface, V24-  
 Schnittstelle, WAIT-Erzeugung)  
 (c) Karl Schulmeister





# User Keys

von GERHARD KLEMENT

Die Tektronix 4051, an der ich mehrere Jahre arbeitete, hatte 10 sogenannte User Keys, die sich bei der Benutzung von BASIC-Programmen als sehr nützlich erwiesen. Die Philosophie war, dass durch Druecken einer Taste bestimmte Zeilen im Programm aufgerufen wurden, die der Benutzer mit entsprechenden BASIC Befehlen belegen konnte. Die Taste 1 rief die Zeile 4, die Taste 2 die Zeile 8 u.a.w. auf. Ich hatte die Tasten so programmiert, dass Taste 1 mit RUN 100 belegt war, wobei die Zeile 100 der Beginn des eigentlichen Programmes war. Taste 2 war LIST, Taste 3 bis 5 listeten mir Programmteile, die ich gerade editierte, Taste 6 startete die eigentliche Rechenroutinen, und Taste 10 rief die Ausgabe der Rechenergebnisse auf. In die entsprechenden Programmzeilen schrieb ich nur die notwendigen GOTO oder GOSUBS.

Die Initialisierung veraendert den Input Vektor, der nunmehr auf die User Routine INROUT weist. In dieser wird geprueft, ob eine Zifferntaste zugleich mit der Graphic Taste gedruickt ist. Ist dies nicht der Fall, erfolgt eine sofortige Rueckkehr zu BASIC. Andernfalls wird der Keyboard Buffer geloescht, das RUN Token eingeschrieben und -je nach gedruickter Taste- die entsprechende BASIC Line number in der EXTAB errechnet, die ebenfalls in den KBUF geladen wird. Nach Adjustierung des Stackpointers wird die Token-Execution Routine aufgerufen, und BASIC laeuft normal weiter.

```

0010 ; === USER KEYS ==
0020 ; Teddysoft
0000          0030          ORG #0
0000 0C75      0040 $IN      EQU #0C75
0000 0C7B      0050 $UIN     EQU #0C7B
0000 1060      0060 KEYBUF   EQU #1060
0000 E816      0070 TOKEX    EQU #E816
0000 FFFD      0080 BWARM    EQU #FFFD
00C0          0090          ORG #CC0
0100 ;
00C0 21CFOC    0110 INI      LD   HL,INRU
00C3 227BOC    0120          LD   ($UIN),HL
00C6 21140D    0130          LD   HL,INTAB
00C9 22750C    0140          LD   ($IN),HL
00CC C3FDFD    0150          JP   BWARM
0160 ;
00CF DF7D      0170 INRU     SCAL #7D
00D1 D0        0180          RET  NC
00D2 FEB1      0190          CP   #B1
00D4 D8        0200          RET  C
00D5 FEBA      0210          CP   #BA
00D7 3032      0220          JR   NC,EXIT
00D9 210D0D    0230          LD   HL,STRI
00DC 116010    0240          LD   DE,KEYBUF
00DF 010700    0250          LD   BC,7
00E2 EDB0      0260          LDIR
00E4 D6B1      0270          SUB  #B1
00E6 110000    0280          LD   DE,0
00E9 47        0290          LD   B,A
00EA 21160D    0300          LD   HL,EXTAB
00ED 7E        0310 REPT    LD   A,(HL)
00EE 5F        0320          LD   E,A
00EF 78        0330          LD   A,B
00F0 A7        0340          AND  A
00F1 2805      0350          JR   Z,FOUND
00F3 19        0360          ADD  HL,DE
00F4 23        0370          INC  HL
00F5 05        0380          DEC  B
00F6 18F5      0390          JR   REPT
0400 ;
00F8 D5        0410 FOUND    PUSH DE
00F9 C1        0420          POP  BC
00FA 23        0430          INC  HL
00FB 116210    0440          LD   DE,KEYBUF+2
    
```

```

00FE EDB0      0450          LDIR
0D00 060D      0460          LD   B 13
0D02 E1        0470 SPO      POP  HL
0D03 10FD      0480          DJNZ SPO
0D05 216010    0490          LD   HL,KEYBUF
0D08 C316E8    0500          JP   TOKEX
0510 ;
0D0B 37        0520 EXIT     SCF
0D0C C9        0530          RET
0540 ;
0D0D 208800    0550 STRI     DEFB " ,#88,0
0D10 00000000  0560          DEFB 0,0,0,0
0570 ;
0D14 7600      0580 INTAB    DEFB #76,0
0590 ;
0D16 0134      0600 EXTAB    DEFB 1,"4
0D18 0138      0610          DEFB 1,"8
0D1A 023132    0620          DEFB 2,"1,"2
0D1D 023136    0630          DEFB 2,"1,"6
0D20 023230    0640          DEFB 2,"2,"0
0D23 023234    0650          DEFB 2,"2,"4
0D26 023238    0660          DEFB 2,"2,"8
0D29 023332    0670          DEFB 2,"3,"2
0D2C 023336    0680          DEFB 2,"3,"6
    
```

# Kettennetz

von GERHARD KLEMENT

KETTENNETZWERK ist eine praktische Nutzenanwendung, ein Programm, das Impedanzen von Netzwerken errechnet. Als erstes Element muss ein Parallelement eingegeben werden. Die Schaltung wird dann via Eingabe weiter aufgebaut, wobei die Impedanzen fuer jede Aufbauphase angezeigt werden. Zum Testen vielleicht zuerst eine einfache Schaltung aus Parallel- und Serienwiderstaenden versuchen.

```

1 REM --- KETTENNETZWERK FILE F ---
2 CLS:PRINTTAB(10)"USER KEYS ACTIVE !!
3 GOTO899 : REM *** HIDDEN MCODE ***
4 GOSUB100
5 END
8 OUT0,32:END
12 GOSUB230
13 END
16 GOSUB500
17 END
20 GOSUB370
21 END
24 GOSUB300
25 END
28 GOSUB430
29 END
32 GOSUB560
33 END
36 CLS:LIST
37 END
100 CLS:PRINTTAB(12)"KETTENNETZWERK"
110 P1=3.1415926
120 PRINT" (3) R ser (4) L ser
130 PRINT" (5) C ser (6) R par
140 PRINT" (7) L par (8) C par
150 PRINT:PRINT" FREQUENZ";
160 INPUT
170 O=PI*F
180 PRINT"Beginn mit Parallelelement"
190 I1=0
200 R1=0
210 I1=0
220 RETURN
230 REM - SERIENWIDERSTAND --
240 SCREEN1,8:PRINTCHR$(27)
250 SCREEN1,8:PRINT "R ser";
    
```

```

260 INPUT X
270 X=1/X
280 F1=1
290 GOTO 340
300 REM - PARALLELWIDERSTAND --
310 SCREEN1,8:PRINTCHR$(27)
320 SCREEN1,8:PRINT "R par";
330 INPUT X
340 X2=1/X
350 Y2=0
360 GOTO 620
370 REM - SERIENKONDENSATOR --
380 SCREEN1,8:PRINTCHR$(27)
390 SCREEN1,8:PRINT "C ser";
400 INPUT X
410 F1=1
420 GOTO 400
430 REM PARALLELE SPULE --
440 SCREEN1,8:PRINTCHR$(27)
450 SCREEN1,8:PRINT "L par";
460 INPUT X
470 X2=0
480 Y2=-1/(0*X)
490 GOTO 620
500 REM SERIEN SPULE
510 SCREEN1,8:PRINTCHR$(27)
520 SCREEN1,8:PRINT "L ser";
530 INPUT X
540 F1=1
550 GOTO 600
560 REM PARALLELE KONDENSATOR
570 SCREEN1,8:PRINTCHR$(27)
580 SCREEN1,8:PRINT "C par";
590 INPUT X
600 Y2=0*X
610 X2=0
620 REM BERECHNUNG
630 Y3=11
640 X3=11
650 REM ** SBR CALL COM **
660 GOSUB 830
670 Y3=Y3+Y2
680 X3=X3+X2
690 REM ** SBR CALL COM **
700 GOSUB 830
710 I1=Y3
720 R1=X3
730 F1=1
740 REM ** SBR CALL COM **
750 GOSUB 830
760 F1=0
770 R=SQR(X3*X3+Y3*Y3)
780 F=COS(PI*HENTX)/R-30
790 Z=ATH(Y3/X3)
800 SCREEN1,11:PRINTCHR$(27):SCREEN1,11
810 PRINT "Z ein "R"Winkel = "I/P1*180
820 RETURN
830 REM -- SBR COM --
840 (FF) (PI*HENTX)
850 D=1/(X3*Y3+Y3*Y3)
860 X3=X3*D
870 Y3=Y3*D
880 RETURN
890 (DOKF4)0,4352
910 U=USR(0):FHD

```

KETTENFÜR HIDDEN CODE

```

105E 70 11 20 40 00 00 00 00 AF
1086 3A 20 2E 20 2A 2A 20 1C
106E 4B 4B 46 44 45 4F 20 4D 37
1076 43 4F 44 45 20 2A 2A 3F
107F 05 00 00 20 2A 2A 20 2C
1085 21 21 00 30 47 4F 5A 4F 4B
108F 35 30 30 00 00 30 00 3F
109F 03 00 00 03 00 00 30 00 46
1091 00 05 00 00 00 00 00 4E
1096 30 00 00 00 00 11 00 01 08
10AE 00 FF AF B3 10 03 FF EA 18

```

```

1086 10 00 00 00 00 00 00 06
10BE 00 03 FF EA 10 FF AF 00 38
10C6 0F FF FF FF FF 00 00 00 E1
10CE 6F 11 00 0F A0 00 00 00 CD
1006 4E 17 4E 17 4E 17 6F 11 95
100E 00 00 FF FF FF FF 93 10 AD
10E6 00 30 EC 20 38 39 30 00 D3
10EE 30 30 30 20 FF FF FF AA
10F6 F5 0A 94 00 70 11 00 00 1A
10FE 00 00 21 0F 11 22 70 0C F8
1106 21 54 11 22 75 0C C3 FD 00
110E FF 0F 70 00 FE B1 08 FE CF
1116 8A 30 32 21 40 11 11 60 33
111E 10 01 07 00 ED 00 06 B1 68
1126 11 00 00 47 21 56 11 7E 95
112E 5F 78 A7 28 05 19 23 05 28
1136 18 F5 05 C1 23 11 62 10 90
113E ED 00 06 00 E1 10 FD 21 0E
1146 60 10 C3 16 E8 37 C9 20 A8
114E 80 00 00 00 00 00 76 00 5D
1156 01 34 01 38 02 31 32 02 3C
115E 31 36 02 32 30 02 32 34 A2
1166 02 32 38 02 33 32 02 33 7F
116E 36 00 96 11 01 00 8E 20 08

```

Nähere Hinweise zur Benutzung des "Hidden M-Code" finden Sie im Programm "BASIC-OFFSET" von Gerhard Klement, 80-Bus Journal 3/83 Seite 16. Red.

## NASCOM-Praxis

TEIL 5 von  
WOLFGANG MAYER-GÜRR

Vorbemerkung der Redaktion:

Wolfgang Mayer-Gürr, Lehrer an einem Gymnasium, findet die Artikel über die NASCOM-Praxis persönlich nicht von allgemeinem Interesse, da der Vollzug zur Praxis nicht direkt nachvollziehbar sei. Deshalb hat er uns auch nichts über seine Arbeit an der Schule berichtet (z.B. Verwaltung der reformierten Oberstufe), die er mit dem NASCOM durchführte, sondern nur über die Rechnerkopplung mit dem APPLE, den er in der Schule zur Verfügung hat.

Dieser Bericht dürfte tatsächlich eine breitere Leserschicht interessieren.

Rechnerkopplung NASCOM - APPLE

Da ich beruflich mit Apple Computern arbeite, mußte, bestand der Wunsch, Artikel über herkömmliche auch zu Hause erledigte Arbeit zu haben. Mit dem Erscheinen sogen-

nannter Apple kompatibler Platinen bot sich eine preisgünstige Möglichkeit an, wenn man schon vorhandene Nascomteile mit einschließen kann. Die Tastatur des Apple (oder besser die Dekodierung auf der dazu gehörenden Platine) ist nicht sehr attraktiv, man kann auf sie verzichten. Die Verbindung zur Hauptplatine entspricht nämlich einer 7-Bit ASCII Schnittstelle. Hier kann also bequem der Nascom über eine PIO angekoppelt werden. Der Apple erkennt ein ansteigendes Zeichen an einer Flanke (steigend oder fallend). Die PIO wird auf Ausgabe gesetzt, die Ausgaberroutine kann so aussehen:

```

AUS      SET      7,A      Zeichen in A
          OUT      (PORT),A
          RES      7,A
          OUT      (PORT),A
          RET

```

Bit 7 wird als Strobe benutzt. Vor der Ausgabe muß lediglich der Code zur Cursorsteuerung geändert werden. Im Originalzustand hat der Apple keine Kleinschrift obwohl sie im Zeichengenerator enthalten ist. Die Monitorroutine wandelt automatisch in Großbuchstaben. Pascal verwendet einen Eingabeteil, der nicht im ROM steht und deshalb leicht geändert werden kann.

Attraktiv ist das Pascalpaket (neben der Unmenge an leicht zu erhaltender Software). Als Laufwerke lassen sich die bereits vorhandenen preiswerten BASF 6106 verwenden. Allerdings braucht man dazu einen anderen Controller (HDCO). Probleme können gegenüber den teuren Originallaufwerken bei Pascal auftreten. Der Zugriff auf einen Sektor nach dem Einschalten des Motors ist kürzer als beim DOS 3.3. Es gibt dazu eine Modifikation auf der Floppyplatine, die aber nicht nötig ist, wenn man den Schreib/Lesekopf ständig aufgesetzt läßt. Dadurch erhöht sich zwar der Verschleiß durch Abrieb der Disketten, dafür verringert sich aber auch das Betriebsgeräusch. Bei dieser Betriebsweise habe ich nach 3 Jahren einen Ausfall von etwa 2 %, das dürfte tolerabel sein.

Pascal bietet die Möglichkeit, ein externes Terminal zu betreiben, sieht dafür aber eine serielle Schnittstelle vor. Ich verwende aber ein modifiziertes Parallellinterface. Priorität hat die Ausgabe vom Apple. Dies läßt sich durch die Interruptfähigkeit der PIO leicht bewerkstelligen. Das Interface verwendet 8 Bit (es werden aber nur 7 gebraucht). Strobe, ACKN bei der Ausgabe und BUSY bei der Eingabe.

CLD-DOS arbeitet zur Tastaturabfrage mit Interrupts im Modus 2. Die Controllerplatine hat keinen IEI-Eingang, deshalb muß zur Bildung einer "Daisy Chain" (gibt's da einen guten deutschen Ausdruck?) die PIO

Platine hinter dem Controller liegen. Der Start der Vektortabelle zur Interruptverarbeitung liegt bei der Adresse 0C80H, im Grundzustand besteht sie nur aus einer Eintragung. Als Acknowledge könnte theoretisch der RDY Ausgang der PIO dienen, dieser Puls ist aber für das verwendete Interface etwas zu kurz. Ich verwende deshalb eine Datenleitung mit einer Softwarelösung.

Das Koppelprogramm hat folgende Struktur:

```

ADATA    EQU     14H      Daten PIO 1A
ACTRL    EQU     16H      Status PIO 1A
BDATA    EQU     15H      Daten PIO 1B
BCTRL    EQU     17H      Status PIO 1B
CDATA    EQU     18H      Daten PIO 2A
*        Bit 6, Acknowl. Eingabe
*        Bit 5, Busy Ausgabe
VEC      EQU     0C82H    2.Eintrag Vektor

ORG      START

INIT     EQU     *
LD       HL,INT          Interruptrou.
LD       (VEC),HL
LD       A,11111111B     Modus 3 (control)
OUT      CCTRL
LD       A,00100000B     Bit 5 Abfrage Busy
OUT      CCTRL
LD       A,00111111B     Ausgabe
OUT      BCTRL
LD       A,01111111B     Input
OUT      ACTRL
LD       A,82H           Vek.Tab. Low
OUT      ACTRL
LD       A,11110011B     Interr. ermöglichen
OUT      ACTRL

IF       Rest des Systems nicht im
        Interruptmodus 2

LD       A,0CH           High Byte VEC
LD       I,A
IM       2

ENDIF

JMP     HAUPT

INT      PUSH     AF
        IN       A,(ADATA)
        CALL    VDU
        CALL    ACKN
        POP     AF
        EI
        RETI

ACKN     RES      6,A
        OUT     CDATA
        SET     6,A
        OUT     CDATA
        RET

VDU     sonderzeichen?
        cursorsteuerung
        editieren
        ausgabe an schirm
        etc.
        RET

HAUPT   tastaturabfrage
        umlaute?
        parameter?
        codewandlung
        PIO B busy?
        ausgabe PIO B
        jr haupt

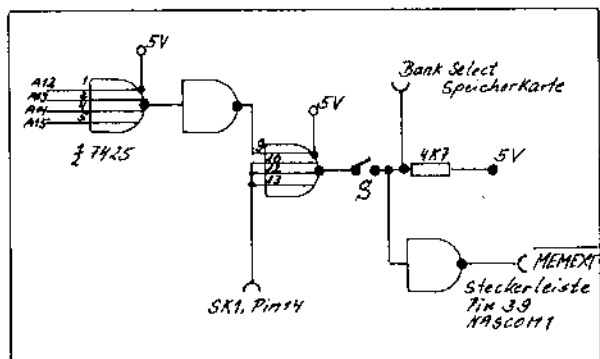
```



# TDL-Basic

von PETER HUNTEMANN

Ich habe die 64KByte-Speicherkarte aus "mc" nachgebaut und deren BANK-SELECT-Eingang mit MEMEXT usw. folgendermaßen verschaltet:



S deselektiert die Speicherkarte und ermöglicht RESET ohne Geräteabschalten (nach Amoklauf).

Mit IC 41 (74LS378, Pin 15) bzw. SK1, Pin 14 wird per Programm zwischen CPU-Platine und Speicherkarte hin- und hergeschaltet.

Mit diesem System habe ich das TDL-BASIC zum Laufen gebracht.

Die Anpassung an NASSYS 3 steht oberhalb FD00 (freifliegend, ohne Jede Optimierung).

(D.h. zwischen den notwendigen Programmteilen befinden sich belanglose RAM-Bereiche, die auch im abgedruckten Listing ausgelassen wurden. Red.)

Das mit W-Befehl auf Cassette geschriebene Programm wird nach CD00 - FFFF geladen und mit Start bei FDED nach 100H verschoben; der Interpreter meldet sich automatisch.

Nach Sprung von BASIC nach NASSYS mit  $\text{C}$  erfolgt der Rücksprung (Warmstart) mit "J". BASIC-Programme werden mit SAVE ohne Kontrolle auf Cassette geschrieben (es hat bei mir nie Probleme gegeben), und der Drucker wird über dieselbe Schnittstelle betrieben ("off line" bei Cassettenaufnahme). Bei Betrieb des Druckers wird dessen BUSY-Signal über SK1, Pin 8 abgefragt und gegebenenfalls gewartet.

Soll der Interpreter oder die Anpassung verändert werden, so ist "Highest Memory" <CD00 einzugeben. Nur "Newline" setzt "Highest Memory" =FBFF; im ersten Fall kann mit NASSYS bei FDED (wie nach Einlesen von

Cassette) gestartet werden.

Zur Tastatur-Eingabe: "↑" =  $\text{C}$

Rubout= BS

Beim Abdruck des Programms sind nur die Teile wiedergegeben, die zur Anpassung an NASSYS dienen. Das vollständige TDL-BASIC inklusive Anpassung können Sie durch Einsen-dung einer Cassette mit Rückporto an den Autor erhalten (allerdings leider nur im N1-Format).

```

C D00 C3 C9 2F C3 73 04 C3 44
C D08 0A C3 43 FD C3 40 FC C3
C D10 00 FD C3 30 FC C3 50 FC
C D18 C3 83 FD AF C9 00 AF 79
C D20 C9 C3 A0 FC C3 DD FC C3
C D28 2D 0A C3 1E 12 A3 22 AD
C D30 23 CE 22 06 03 F6 11 D4
C D38 16 2F 12 0E 2A 0E 2B EB
    
```

↑  
BASIC wie in R.D.Klein  
"BASIC- Interpreter"

```

FC30 79 F5 D3 01 DB 02 CB 77
FC38 28 FA F1 C9 00 42 00 42
FC40 DB 02 CB 7F 28 FA DB 01
FC48 C9 42 00 42 00 42 00 42
FC50 F5 79 F5 D3 01 DB 02 CB
FC58 77 28 FA F1 DB 00 E6 8C
FC60 28 FA F1 C9
    
```

```

FCA0 06 FB 3E FF C9 FE 19 CA
FCA8 B5 FC FE 08 CA BD FC 32
FCB0 12 FC C3 64 FD 3E 5E 32
FCB8 12 FC C3 64 FD 3E 7F 32
FCC0 12 FC C3 64 FD FD E1 DD
    
```

```

FCC8 E1 E1 D1 C1 F1 D9 E1 D1
FCD0 C1 D9 3E 20 D3 00 ED 7B
FCD8 20 FC C3 03 03 ED 73 20
FCE0 FC 3E 00 D3 00 31 00 10
FCE8 D9 C5 D5 E5 D9 F5 C5 D5
FCF0 E5 DD E5 FD E5 E7 00 42
FCF8 00 42 00 42 00 42 00 42
FD00 ED 73 20 FC 31 E0 FF 00
FD08 79 32 10 FC F5 C5 D5 E5
FD10 DD E5 FD E5 D9 C5 D5 E5
FD18 D9 08 F5 08 3E 00 D3 00
FD20 3A 10 FC F7 3E 20 D3 00
FD28 08 F1 08 D9 E1 D1 C1 D9
FD30 FD E1 DD E1 E1 D1 C1 F1
FD38 3A 10 FC ED 7B 20 FC C9
FD40 E5 C9 31 ED 73 20 FC 31
    
```

```

FD48 E0 FF 00 F5 C5 D5 E5 DD
FD50 E5 FD E5 D9 C5 D5 E5 D9
FD58 08 F5 08 3E 00 D3 00 DF
FD60 7B C3 A5 FC 3E 20 D3 00
FD68 08 F1 08 D9 E1 D1 C1 D9
FD70 FD E1 DD E1 E1 D1 C1 F1
FD78 3A 12 FC ED 7B 20 FC C9
FD80 F5 C5 D5 ED 73 20 FC 31
FD88 E0 FF 00 F5 C5 D5 E5 DD
FD90 E5 FD E5 D9 C5 D5 E5 D9
FD98 08 F5 08 3E 00 D3 00 DF
FDA0 62 3E 20 D3 00 DA C5 FD
FDA8 3E 00 32 14 FC 08 F1 08
FDB0 D9 E1 D1 C1 D9 FD E1 DD
FDB8 E1 E1 D1 C1 F1 3A 14 FC
FDC0 ED 7B 20 FC C9 3E FF 32
FDC8 14 FC 08 F1 08 D9 E1 D1
FDD0 C1 D9 FD E1 DD E1 E1 D1
  DD8 C1 F1 AF 37 3F 3A 14 FC
FDE0 ED 7B 20 FC C9 3E 00 D3
FDE8 00 E7 42 00 42 3E 20 D3
  DFO 00 01 00 40 11 00 00 21
FDF8 00 CA ED B0 C3 00 03 00

```

```

FE0 00 3E 20 D3 00 C3 03 03
FE8 FE 6E 6E 6E 6E 6E 6E 6F
FF0 00 64 64 64 24 64 64 64
FF8 6E 6E C3 C5 FC 6E 6E A7

```

Hier die Einsprungadressen für die einzelnen Anpassungsroutinen:

```

FC30 Cassette schreiben
FC40 Cassette lesen
FC50 Drucker
FCA0 "Highest Memory"
FD00 Konsolenausgabe
FD43 Tastatureingabe
FDED Kaltstart
FFE1 Warmstart mit J aus
  NASSYS 3

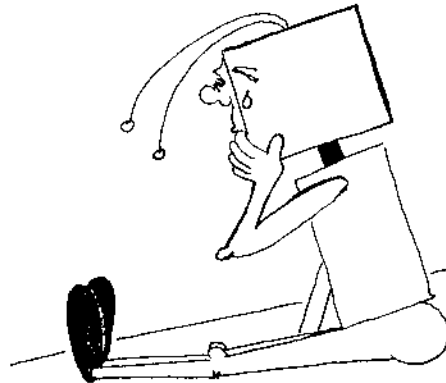
```

```

VERKAUFE
original NASCOM 32K- RAM- A- Karte
Erhard Thumm
██████████
██████████
Tel. ██████████

```

# nascompl



Hallo liebe Leser,  
traurige Zeiten sind angebrochen: seit Monaten in der Redaktionsarbeit völlig alleingelassen, finanzielle Außenstände, die in die Tausende gehen, Ausblick nur auf weitere Ausgaben, totaler Zeitdruck... und da soll man noch etwas Lustiges schreiben?

Sie sind mir hoffentlich nicht böse, wenn ich Sie unter diesen Umständen diesmal mit ein paar Kleinanzeigen abspelse. Wenn da ein Geschäft zustande kommt, haben wenigstens die Beteiligten etwas zu lachen.

In diesem Sinne Ihr NASCOMPL

VERKAUFE Siemens 100 S Fernschreiber mit Lochstreifensender- und Empfänger.

Generalüberholt; Fernschaltgerät auf 20mA Current Loop Schnittstelle umgerüstet.

Assmann tags. ██████████  
abends ██████████

VERKAUFE

NASCOM 1 mit 16K RAM, NASBUG-T4, NASSYS-1, 20K-ECB-Bus-EPROM-Karte mit BASIC, Disassembler, NASPEN, DEBUGGER. Grafikmöglichkeiten des NASCOM 2, Cassettenrecorder inkl. Cassetten und Programme.

Komplettes NASCOM- Journal, Fernschreiber (LO-15)

VB 700,- DM für Selbstabholer

Jochen Münster Tel. ██████████

# Gemini Microcomputer

Vertriebs - GmbH

## SONDERANGEBOTE

solange der Vorrat reicht

RAM 'C' - Platine mit 64 KBytes, Bausatz	DM 450,--
EPROM 'B' - Platine, Bausatz	DM 330,--
SUPERMUM Erweiterung für NASCOM 1, Bausatz ohne Netzt.	DM 299,--
Paketpreis für NASCOM 1 Erweiterung, bestehend aus RAM 'C', EPROM 'B' und SUPERMUM	DM 998,--
EPROM - Programmiergerät für NASCOM oder GEMINI, programmiert 2708 und 2716 (5V), Bausatz einschl. Software (Betriebssystem angeben !)	DM 149,--
RTC Real Time Clock Bausatz, stellt über die PIO Urzeit und Datum zur Verfügung, Quarzgesteuert mit Akku einschl. Software	DM 149,--
BASIC ROM V 4.7 für NASCOM	DM 99,--
GRAFIK ROM für NASCOM	DM 47,--
Ersatzteile für NASCOM und GEMINI MULTIBOARD	
Z80 CPU	DM 9,--
Z80 A CPU	DM 9,50
Z80 PIO	DM 6,50
Z80 A PIO	DM 9,--
Z80 A CTC	DM 12,--
UART 6402	DM 25,--
DIL Plattform 16-pol.	DM 2,--
Tasten für Nascom 1/2 - Tastatur 10 Stück	DM 60,--
Centronics - Stecker (Weibchen)	DM 19,--

Achtung! Jetzt besonders günstig!

Original Gemini Floppydiskstation mit 2 Laufwerken (jeweils 350 KBytes form.), anschlussfertig im Gehäuse, mit Netzteil und Kabel einschl. Original Gemini FDC, fertig aufgebaut und getestet

DM 2800,--

Vero- Frame Einschubrahmen für 80-Bus Platinen

DM 189,--

Alle Preise einschl. ges. Mehrwertsteuer, zuzüglich Porto/Verpackung, Lieferung nur gegen Nachnahme.

Bitte fordern Sie unser neuestes 80 - Bus Info an !

Schluderstr.10 • 8000 München 19

Tel. 089 / 168595



# NASCOM

NASCOM - Sonderangebote OKTOBER '83

## NASCOM-C, der neue Maßstab für CP/M -und 80-BUS-Systeme !

- NASSYS-kompatibel und derzeit stärkstes CP/M-System !
- \* NASCOM-C mit Z80A CPU, 64KB RAM, MMU, 2xV24 und eine CENTRONICS-Schnittstelle, Videoteil, NUCLEOSYS.....DM 1.298,-
  - \* NASCOM-C wie oben als Bausatz.....DM 998,-
  - \* NASCOM-C Leerplatine mit Firmware & Dokumentation.DM 298,-
  - \* NASCOM-C DMA-Floppy-Controller Option 5" oder 8"..DM 298,-
  - \* Floppycontroller-Option für 5"+8" gemischt.....DM 348,-
  - \* NASCOM-C 64KB-Erweiterung und Paritylogik.....DM 198,-
  - \* Deutsche Tastatur für NASCOM-C (Cherry).....DM 198,-
  - \* NASCOM-AVC Farbgrafik mit BASIC, -ASSEMBLER-und CP/M Softwareschnittstellen.....DM 798,-
  - \* Hi-Res Farbmonitor für AVC (Zenith).....DM 1.998,-
  - \* NASCOM-2a, NASCOM-2 mit 8KB CMOS-RAM ,ZEAP-Editor/Assembler und Microsoft-Basic in ROM als Bausatz.....DM 1.098,-
  - \* 80-BUS, 4 Steckplätze, Busrahmen, Führungsleisten.DM 148,-
  - \* CLD-Hardcontroller für hardsekt. Minidisketten..DM 498,-
  - \* CLD-Softcontroller mit DMA, Echtzeituhr und Interface für Festplattenlaufwerke.....DM 998,-
  - \* Softcontroller ohne DMA und CTC, Bausatz.....DM 698,-
  - \* Softcontroller als Leerplatine + Firmware.....DM 198,-
  - \* Minidiskettenlaufwerk BASF-6106, 200KB.....DM 498,-
  - \* CLD-BANKED-Epromkarte für 16 Stück 2708 /16 /32, 2532 sowie 8KB ROMs in vier Banks, Bausatz.....DM 248,-
  - \* Leerplatine Epromkarte mit Dok.....DM 148,-
  - \* CLD-256KB-Ramkarte, Bausatz ohne DMA und Paritylogik mit 64KB RAM.....DM 698,-
  - \* CLD-256KB-Ramkarte als Leerplatine mit Dok.....DM 148,-
  - \* BLS-Pascal auf EPROM.....DM 298,-
  - \* Page-Mode-Kit für LUCAS RAM-B Karte.....DM 79,-
  - \* Grafik-Erweiterung für NASCOM-1 (ohne Grafgen.)...DM 98,-
  - \* Grafik-Erweiterung für NASCOM-1 (mit Grafgen.)....DM 119,-
  - \* CP/M 2.2 Betriebssystem mit ADM-31 Terminalemulator fuer AVC-Board, Screen-Editing auf CP/M Kommandoebene, Interface für Centronics-Drucker..DM 498,-
  - \* BIOS-Anpassung bei Zusendung eines liz. CP/M.....DM 99,-

Alle Platinen mit Lötstopplack, vergoldeten Kontakten und Bestückungsdruck gefertigt und für alle NASCOM und GEMINI-Systeme verwendbar, alle Bausätze und neuen CLD-Platinen mit (gedrehten) Präzisionssockeln. FORTH -und BASIC EPROMS für NASCOM-C in Vorbereitung, desgleichen GSX für AVC & CP/M+

Preise inklusiv MwSt., exklusiv Versandkosten

LAMPSON Digitaltechnik  
 Odenwaldstrasse 21-23 Tel.: 06152/56730  
 6087 Büttelborn

Wir sind Ausst.

auf der  
 Hobbyelektronik83  
 Stand 1448/Halle  
 14