

# **nascocom**

## **JOURNAL 3/80**

---

ZEITSCHRIFT FÜR BENUTZER DES NASCOM 1 ODER NASCOM 2

Herausgegeben von: M K - Systemtechnik, Michael Klein, Waldstraße 20  
6728 Germersheim

Heftpreis: 4.- ; Abonnement für 1980 nur 39.-, sonst 48.- pro Jahr  
für 12 Ausgaben

Redaktion: c/o M K - Systemtechnik, Waldstr. 20 6728 Germersheim  
Bitte alle Zuschriften an diese Adresse

Freie Mitarbeiter: Ulrich Krüger, Jörg Donandt, Bernd Ploss, H.-J. Dietmann

---

Liebe Leser,

der schöne Sommer läßt nicht viel Gelegenheit, den Bauch in die Sonne zu hängen, oder mal ausgiebig schwimmen zu gehen. Wir sehen dies an den anhaltend zahlreichen Telephonen, Briefen und Postkarten, die uns täglich erreichen.

Etwas vorab: Das Format des letzten Heftes ist einem Unfall zuzuschreiben. Der Drucker vergaß, die Bögen nach dem Druck zu schneiden. Daher das Oberformat.

Der redaktionelle Teil konnte fast durchweg wieder mit Beiträgen unserer Kunden gefüllt werden. Vielen Dank für die zahlreichen Beiträge. Noch eine Bitte: Wenn's geht wären wir bei den Zeichnungen für Tusche-Reinzeichnungen dankbar. Eine Zeichnung nochmal anfertigen zu lassen, kostet nämlich ein Ver-

mögen

Eine gallige Bemerkung am Rande: Ein deutsches Unternehmen, das NASCOMs hier in Deutschland vertreibt, wollte unsere Zeitschrift zensieren! Was halten Sie davon? Wenn der I M P - Drucker einige ganz offensichtliche Mängel hat, warum zum Teufel darf man es dann nicht sagen? Ein Geschäft zu betreiben, heißt doch nicht, sich und die Kunden zu belügen! Eine etwas traurige Episode - fürwahr. Unsere Zeitschrift läuft weiter wie bisher und wenn ein Testbericht eben kritisch ausfallen muß, dann wird er eben kritisch ausfallen.

Ihr

*Michael Klein*

I n h a l t s v e r z e i c h n i s

- 1 Editorial
- 2 Inhaltverzeichnis - Bezugsbedingungen
- 3-7 Ein Plotter für den NASCOM - von Peter Bentz, Mühlheim
- 8,9 Entfernungsberechnung mittels QTH-Kennern  
von Winfried Widmann, Hattenhofen
- 10 Hell-Dunkel-Umschaltung für NASCOM 1 - Anzeige rd.
- 11 BASIC-ROM auf normaler Speichererweiterung  
von R. Böttchers, Essen
- 12 Verbesserung der Kompatibilität des Disassembler-  
Programmes mit ZEAP beim NSCOM 1  
von Peter Bentz, Mühlheim
- 13 17 & 4 ; von Peter Szymanski, Rehsiepen
- 14 Logeleien; rd
- 15 Verschiedene Angebote
- 16 Kleinanzeigen

---

Bezugsbedingungen

1. Das Jahresabonnement des NASCOM - JOURNAL kostet für 1980 DM 39.-. <sup>Ab</sup> 1.1.1981 kostet es 48.- pro Jahr.
2. Der Betrag ist nach Rechnungsstellung unter Angabe der Rechnungsnummer (nicht vergessen, sonst ist Bearbeitung unmöglich !!) auf unser Sonderkonto 299 26 - 674 beim PSchA Ludwigshafen zu überweisen.
3. Ein Abonnement läuft für 1 Jahr. Wird es bis 6 Wochen vor Ablauf des Kalenderjahres nicht gekündigt, dann erhalten Sie das NASCOM-JOURNAL auch für das nächste Jahr.
4. Es erscheinen 12 Ausgaben pro Jahr. Auslieferung erfolgt jeweils am Monatsende.

Einige Firmen bieten hervorragende Plotter für Mikro- und Minicomputer an, die für den Computer-Amateur sicher von großem Interesse sind, aber die Preise von über 10 000 DM machen solche Geräte unerschwinglich. Für weniger Geld gibt es zwar schon einige Drucker, die die Möglichkeit bieten, einfache Zeichnungen auszugeben, aber deren Qualitäten befriedigen nicht alle Vorstellungen, die man sich von einem Plotter macht. Deshalb wurde versucht, mit einfachsten Mitteln einen Trommelplotter zu bauen, der auch höheren graphischen Ansprüchen genügt. Da einem Computer-Amateur selten bis nie eine feinmechanische Werkstatt zur Verfügung steht, um einen Plotter selbst zu bauen, wurde hier ein alter Kompensationslinienschreiber vor seinem traurigen Ende auf dem Schrottplatz bewahrt und entsprechend umgebaut.

Die Mechanik :

Aus einem Linienschreiber "Speedomax Typ G" der Firma Leeds & Northrup Co , Philadelphia, USA wurde zunächst die defekte Elektronik und die überflüssige Mechanik entfernt. Zur Bewegung der Trommel wurde direkt auf die Trommelachse ein Schrittmotor mit Getriebe fest angekuppelt. Ein zweiter Schrittmotor dreht über ein gleiches Getriebe ein Schnurrad mit dem gleichen Durchmesser wie die Trommel. Dieses Schnurrad mußte als einziges Teil zusätzlich angefertigt werden, aber prinzipiell läßt sich auch ein Rad mit einem anderen Durchmesser verwenden, wenn die entstehende Verzerrung per Software korrigiert wird. Das Schnurrad bewegt mit einem dünnen Stahlseil über vier Umlenkrollen einen Schlitten, der auf zwei Laufschiene parallel über der Trommel gleitet. Er trägt ein Gewinde, in das ein Tuschefüller eingeschraubt werden kann. Ein zusätzlich angebrachter Elektromagnet kann eine der beiden Laufschiene kippen, und damit den Schlitten mit Tuschefüller um einige Millimeter anheben.

Zum Zeichnen wird ein Stück passendes Papier mit der entsprechenden Transportlochung über die Trommel gelegt, und die beiden frei herunterhängenden Enden werden mit je einer Metallklammer beschwert.

Leider haben die Getriebe ein Spiel von ca  $1^{\circ}$  ( entspr. 0,5 mm). Diesen Nachteil kann man beseitigen, indem man die beiden Papierenden mit unterschiedlich schweren Klammern versieht. Dadurch wird die Trommel immer an einem Anschlag des mechanischen Spieles geführt. Für den zweiten Motor, bei dem eine solche Kompensation nicht durchzuführen war, wurde eine elegante Lösung per Software gefunden.

Die verwendeten Schrittmotoren sind eigentlich umfunktionierte reversierbare Synchronmotoren ( für 48 Volt Wechselspannung ) der Firma Valvo, Hamburg. ( Typnr. 9904 111 07413 ). Sie lassen sich aber wie Schrittmotoren ansteuern. Diese Motoren befinden sich zusammen mit einem Getriebe in Fernschalteinheiten, die die Firma Völkner, Braunschweig zum Schrottpreis anbietet.

Einige technische Daten :

benutzbare Papierbreite 25cm ; 1/2 inch Transportlochung

Trommelumfang 6 inches = 48,5 mm Durchmesser

Motoren : Schrittwinkel  $7,5^{\circ}$  entspr. 48 Schritte / Umdrehung

Getriebeuntersetzung 1:125

daraus ergibt sich eine Auflösung von 1/1000 inch entspr.

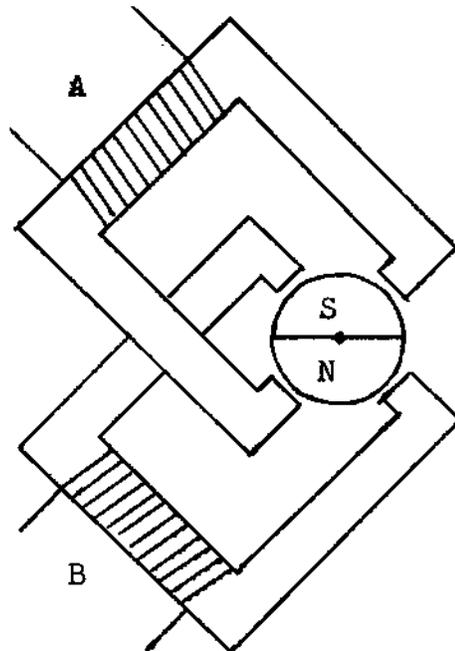
0,0254 mm/ Schritt

maximale Schrittfrequenz ca 280 Hz

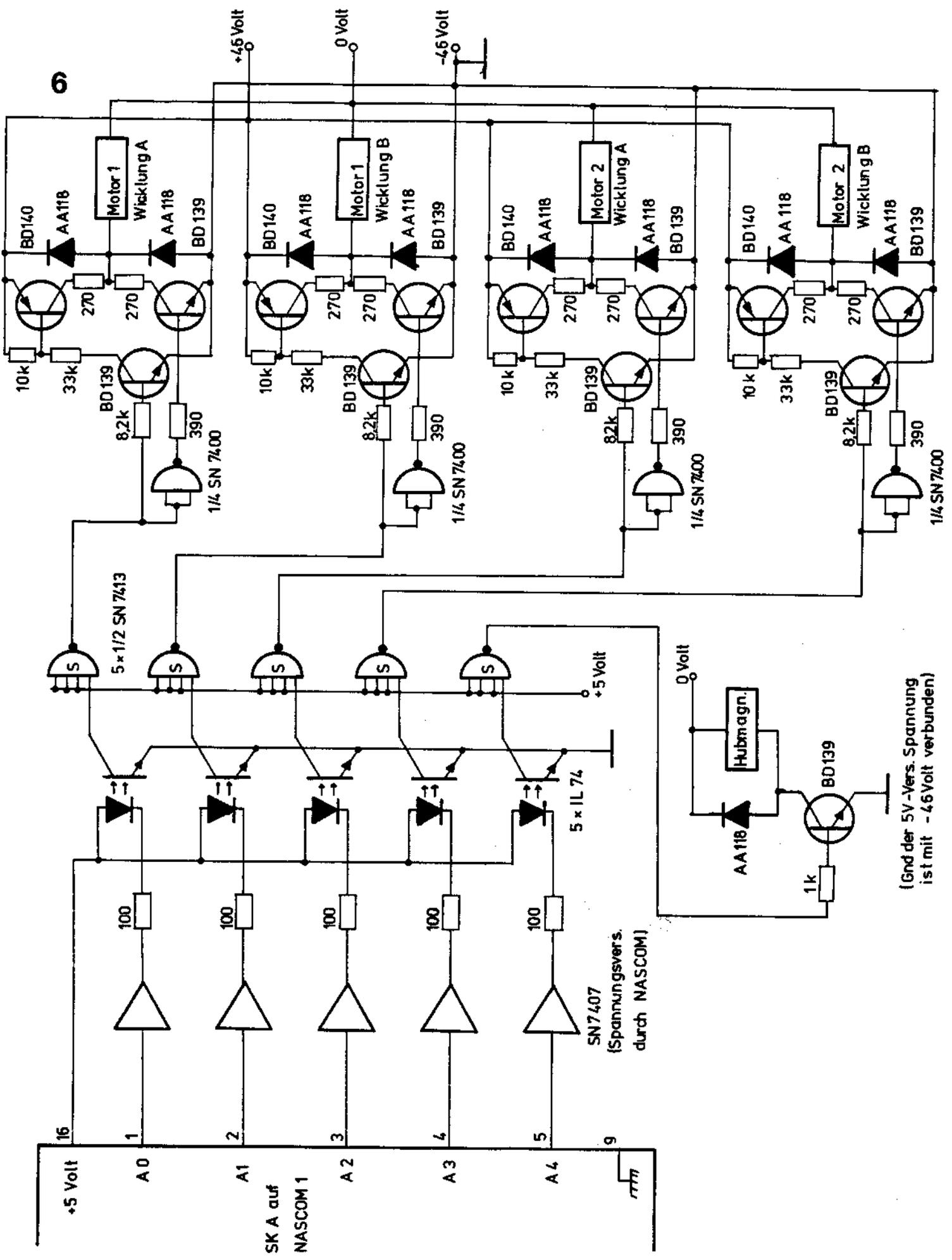
maximale Schreibgeschwindigkeit ca 7,2 mm / sec

Die Elektronik :

Schematische Darstellung der Gegentakt-2-Strang-Ansteuerung  
von Schrittmotoren



Mit den beiden Spulen A und B können zwei Magnetfelder erzeugt werden, die um den Schrittwinkel ( im Schema  $90^\circ$  ) zueinander verdreht sind. Der Rotor, der einen Dauermagneten trägt, stellt sich entsprechend dem gesamten magnetischen Feld ein und erreicht so eine stabile Lage. Durch abwechselnde Umkehrung der Stromrichtung in den beiden Spulen wird das Gesamtmagnetfeld um jeweils einen Schritt nach rechts oder links verdreht. Dadurch befindet sich der Rotor momentan in einer instabilen Lage und dreht sich sofort um einen Schritt weiter und erreicht so wieder eine stabile Stellung. Nach vier solchen Schritten in eine Richtung haben die beiden Magnetfelder alle Kombinationen durchlaufen und der Zyklus kann wieder von vorne beginnen. Die Drehrichtung ist davon abhängig, welches Magnetfeld dem anderen im laufenden Wechsel voraussetzt.

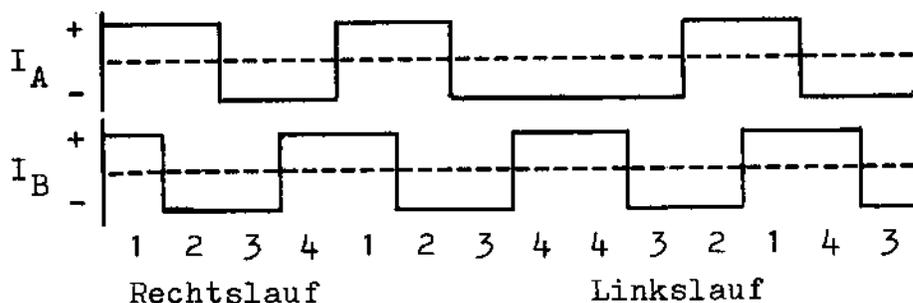


6

SK A auf  
NASCOM 1

SN7407  
(Spannungsvers.  
durch NASCOM)

(Gnd der 5V-Vers. Spannung  
ist mit -4.6Volt verbunden)



Literatur zur Ansteuerung von Schrittmotoren :

Das Schrittmotoren Handbuch, Sigma Instruments, Bad Tölz 1973

E. Schmid:Elcomp, Juni 1979, 18 - 21

#### Schaltungsbeschreibung

Die Steuerimpulse für die Schrittmotoren werden über die PIO des Nascom ausgegeben und mit Hilfe der Opto-Koppler in den vom Nascom galvanisch getrennten Teil der Schaltung übertragen. Diese Opto-Koppler sind nur aus Sicherheitsgründen installiert; sie sollen den Nascom bei unbeabsichtigten Kurzschlüssen oder bei evtl. defekten Bauteilen vor unzulässigen Spannungen schützen. In den anschließenden vier identischen Schaltungen werden die einzelnen Wicklungen der Motoren angesteuert. Die Invertergatter bewirken dabei, daß jede Wicklung nur an die positive oder negative Versorgungsspannung gelegt wird und verhindern dadurch einen evtl. Kurzschluß über je einen pnp- und npn-Transistor. Die Dioden sollen die Transistoren vor induktiven Spannungsspitzen schützen. Der fünfte Kanal steuert über einen Transistor als Schaltverstärker den Hubmagneten zur Zeichenstiftanhebung. Wegen der galvanischen Trennung vom Nascom benötigt man noch eine zusätzliche Versorgungsspannung für die TTL-ICs. Das Gnd-Potential dieser Spannung muß mit der negativen Versorgungsspannung der Motoren verbunden werden. Für die 5V-Spannung empfiehlt sich eine Spannungsstabilisierung, für die +/- 48V-Spannungen reicht je ein Elektrolytkondensator nach der Gleichrichtung um die Spannungen ausreichend zu glätten.

Peter Bentz

## Entfernungsberechnung mittels QTH-Kennern

8

Nach dem Starten des Programmes, teilt der Computer dem Anwender mit, welche Eingaben und in welcher Form vom Bediener erwartet werden.

Wichtig ist hierbei die richtige Interpretation der Zeile 20 im Programmlisting.

Es muß lauten:

```
20 PRINT"Der QTH-Kenner wird in der Form 'EI28d' eingegeben."
```

Dabei ist der Kleinbuchstabe am Ende des QTH-Kenners von Wichtigkeit.

Wenn hier ein Großbuchstabe eingegeben wird, führt dies in den Zeilen 172 und 173 zu einer Fehler-Bedingung und der Computer verzweigt erneut zu der Eingabe QTH-Kenner bzw. eigener QTH-Kenner.

Bei allen Fragen vom Computer nach dem QTH-Kenner ist nach diesem Schema zu verfahren.

Winfried Widmann, DC2SU

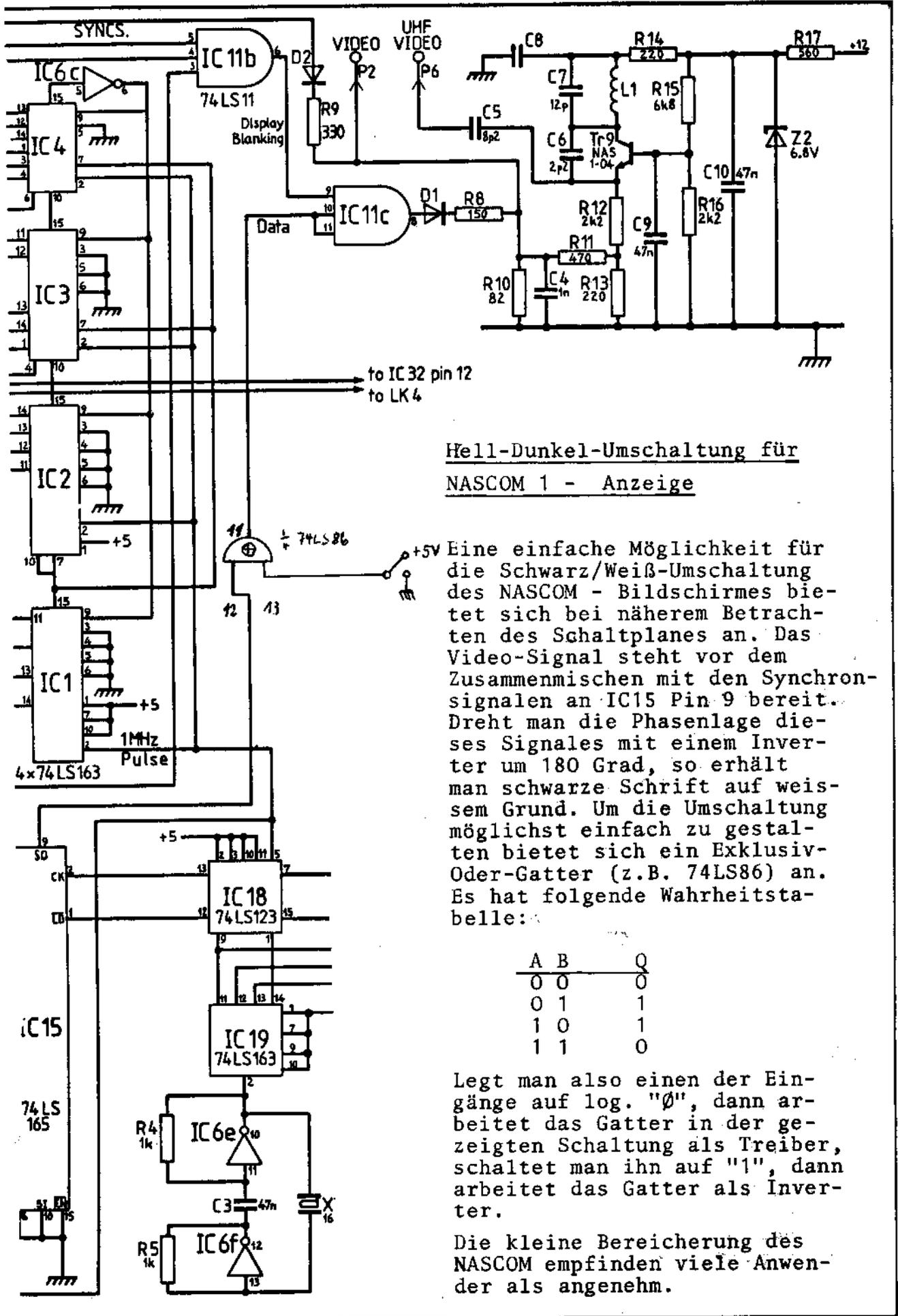
```
1 cls: width (64)
2 rem programm entfernungsberechnung mittels qth-kennern
3 rem nascom 8k microsoft-basic von winfried widmann (dc2su)
4 rem zeilen 7 bis 18 sind nascom spezifisch und dienen nur
5 rem zur ueberschrift erzeugung in der kopfzeile bei anderen
6 rem system kann dies entfallen
7 screen 1,15
8 print' entfernungsberechnung mittels qth-kennern ':
9 for c=2954 to 3000
10 doke c+64, deek(c)
11 next c
12 print:print:print:print:print:print:print:print
13 print:print:print:print:print:print:print:print
14 print'der qth-kenner wird in form 'ei28d' eingegeben.'
15 print'ein fehler in der eingabe wird mit einer'
16 print'erneuten frage nach dem qth-kenner beantwortet.'
17 print'jede eingabe wird mit der taste 'new line' ab-'
18 print'geschlossen.':print
19 clear
20 input'eigener qth-kenner ':fd
21 ad=fd:goto 110
22 r9=r7:r8=r6
23 goto 100
24 input'qth-kenner ':ad
25 if ad=xd then 100
26 if len(ad>()5 then 100
27 w=asc(leftd(ad,1))
28 b=asc(midd(ad,2,1))
29 r2=val(midd(ad,3,1))
30 r3=val(midd(ad,4,1))
31 e=asc(rightd(ad,1))
32 if r3=0 then r3=10
33 if r3=10 then r2=r2-1
34 if e)=107 goto 100
35 if e(=96 goto 100
36 if r2)8 goto 100
```

```

175 if a(85 or a)64 then r0=a-64
177 if b(87 or b)64 then r1=b-64
179 if a)84 or a(89 and r0=0 then r0=a-90
180 if b)86 or b(91 and r1=0 then r1=b-90
195 goto 1000
200 rem berechnung geogr. laenge
210 r6=((3*r3-r4+.5)/15+2*r0-2)
220 rem berechnung geogr. breite
230 r7=((3*r2+r5-.5)*-1)/24+r1+40)
235 if ad=fd then print'entfernung= 0 km.':goto 60
240 rem berechnung der entfernung
245 p=3.14159/180
250 z=(cos((r8-r6)*p)*cos(r7*p)*cos(r9*p)+sin(r7*p)*sin(r9*p))
260 def fnt(x)=1.5708-atn(x/sqr(-x*x+1))
270 t=int(fnt(z)*6368+.5)
280 sum=sum+t
285 print'lwenge ':r6:'grad - breite ':r7:'grad':print:print
290 print'die entfernung betraegt ':t:'km.'
300 printtab(24)'=====
310 print:print'die summe der entfernungen betraegt'
320 print'=:sum:'km.':print'.....'
330 print:print:print
340 xd=ad
350 goto 100
1000 rem tabelle
1530 if e=97 then gosub 1640
1540 if e=98 then gosub 1650
1550 if e=99 then gosub 1660
1560 if e=100 then gosub 1670
1570 if e=101 then gosub 1680
1580 if e=102 then gosub 1690
1590 if e=103 then gosub 1700
1600 if e=104 then gosub 1710
1610 if e=106 then gosub 1660
1615 if a=(64 then 100
1620 if a=)91 then 100
1625 if b=(64 then 100
1630 if b=)91 then 100
1635 goto 200
1640 r4=2:r3=1
1645 return
1650 r4=1:r5=1
1655 return
1660 r4=2:r5=2
1665 return
1670 r4=1:r5=3
1675 return
1680 r4=2:r5=3
1685 return
1690 r4=3:r5=3
1695 return
1700 r4=3:r5=2
1705 return
1710 r4=3:r5=1
1715 return

```

□ △ \*



Hell-Dunkel-Umschaltung für NASCOM 1 - Anzeige

Eine einfache Möglichkeit für die Schwarz/Weiß-Umschaltung des NASCOM - Bildschirmes bietet sich bei näherem Betrachten des Schaltplanes an. Das Video-Signal steht vor dem Zusammenmischen mit den Synchronsignalen an IC15 Pin 9 bereit. Dreht man die Phasenlage dieses Signales mit einem Inverter um 180 Grad, so erhält man schwarze Schrift auf weißem Grund. Um die Umschaltung möglichst einfach zu gestalten bietet sich ein Exklusiv-Oder-Gatter (z.B. 74LS86) an. Es hat folgende Wahrheitstabelle:

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

Legt man also einen der Eingänge auf log. "0", dann arbeitet das Gatter in der gezeigten Schaltung als Treiber, schaltet man ihn auf "1", dann arbeitet das Gatter als Inverter.

Die kleine Bereicherung des NASCOM empfinden viele Anwender als angenehm.

## BASIC-ROM auf normaler Speichererweiterung

11

Diese Methode hat den Vorteil, daß sie ohne Schalter auskommt: Gebraucht werden je ein 7400, 7432 und ein 24p DIL-Sockel; von beiden ICs werden alle Anschlüsse ausser 7 und 14 waagrecht gebogen und gekürzt. Sodann kann man sie auf eine IC löten, die schon auf einer Platine liegt, am besten ein 7402, 7406 o. ä. mit gleicher Spannungsversorgung auf 7/14. Die DBDR-Leitung von Punkt 13 des Karten-Sockels wird unterbrochen, ebenso wird der Anschluss vom Kartensockel mit der 7400, pin 6 verbunden, der nun offene Anschluss an pin 2 ( mit "A" bezeichnet). Pin 3,4,5 werden zusammengeschaltet. Vom 24p-Sockel werden die Anschlüsse 18,19,20,21 hochgebogen und so verdrahtet:

pin 19.....	IC 24 p2
pin 18.....	IC 24 p3
pin 21.....	IC 25 p7
pin 20.....	7432 p3

Der Rest des Sockels wird mit dem BASIC-ROM in einen freien EPROM-Sockel gesteckt. Pin 1 der 7432 führt zu den Decode-Pads 12 & 11, Pin 2/7432 an IC 33 pin 9. Als letztes wird p3/7432 noch mit p1/7400 verbunden.

Funktion: ab Adresse  $E000$  bekommt IC 7432 eine  $\emptyset$  von den Decodern und eine  $\emptyset$  vom IC 33 ( mrqe und rd). Dies gibt ein CE für das ROM. Die NANDs erzeugen noch ein Signal für die Umschaltung der Datenbus-treiber auf dem Buffer-Board.

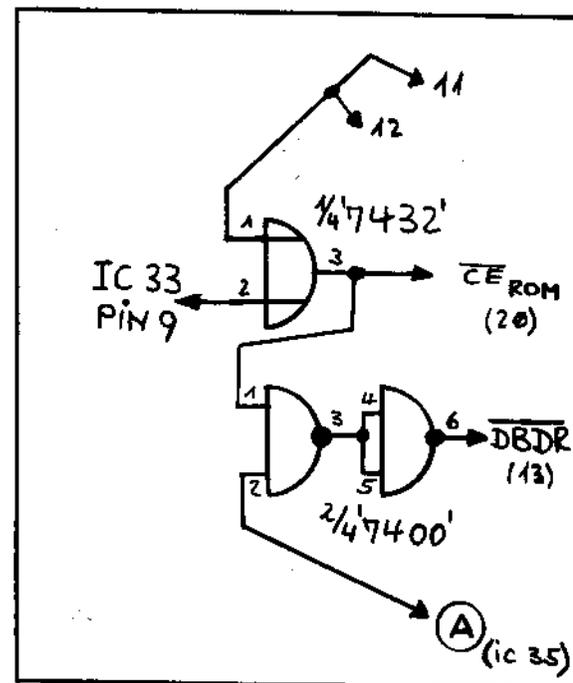
Gibt es Arger, so hilft folgende Notlösung:

Die Datenanschlüsse für den Sockel des ROMs werden hochgebogen wie 18,19,....

und die freiliegenden Anschlüsse des Sockels direkt auf den Datenbus gelegt (am besten über Anlöten auf IC 19, 3/5/7/11/13/15/17).

Meine persönliche Erfahrung ist, daß die Treiber 8197 sehr viel Arger machen, wenn das ROM zu puffern ist, was sich eventuell erst zeigt, wenn Befehle wie "? SQR(2)" gegeben werden.

Die angegebene Schaltungsänderung läuft bei mir nun einige Tage zur Probe, ohne daß irgendwelche Störungen auftraten. Der EPROM-Assembler ist nun weiterhin in Betrieb, ohne daß Schalter notwendig sind, das heisst, auch Unterbringung in einem geschlossenen Gehäuse ist möglich.



## Verbesserung der Kompatibilität des Disassembler-Programmes mit ZEAP beim NASCOM 1

Übersetzt man mit dem Disassembler ein größeres ( $\sim 1K$ ) Maschinencodeprogramm in ein Quellprogramm zurück, und liest es mit ZEAP wieder in den NASCOM 1 ein, so stellt man fest, daß nach einer gewissen Anfangsphase, in der alles korrekt verläuft, die einzelnen Zeilen nicht mehr vollständig erfasst und auf den Bildschirm geschrieben werden.

Dies rührt daher, daß ZEAP eine bestimmte Zeit benötigt, um eine Zeile, die eingelesen wurde, in den Speicher zu laden. Diese Zeit wächst mit der Länge des Quellprogrammes. Deshalb empfiehlt es sich ja auch, vor Abspeichern eines Quellprogrammes auf Kassettenrecorder mit dem Q-Befehl eine Pause nach jeder Zeile zu programmieren.

Das Disassembler-Programm erzeugt ebenfalls eine solche Pause, die aber ziemlich kurz ist. Bei kleinen Programmen spielt dies keine Rolle, da hier ZEAP noch mitkommt. Für größere Programme ist es aber wichtig, den Zeitfaktor bzw. den Schleifenzähler, der bei  $41C6_h$  und  $41C7_h$  in das Register A geladen wird, zu erhöhen. Man ersetzt also den Wert  $20_h$ , der im Speicherplatz  $41C7_h$  des Disassembler-Programmes steht, durch einen höheren Wert. Für ein zu disassemblierendes Programm von  $1K$  reicht ein Wert von  $80_h$  aus. Man kann ihn jedoch bis zu  $00$  erhöhen, was einer Pause von rund zwei Sekunden nach jeder Zeile entspricht.

Peter Bentz

Das recht bekannte Kartenspiel 17 und 4 folgt sehr einfachen Regeln und kann deshalb auch ohne Schwierigkeiten in BASIC formuliert werden. Die Spielregeln für das hier vorgestellte Programm sind: Ein Spieler tritt gegen den Computer an, der die Bank hält. Zunächst fragt der Computer nach dem Einsatz, den der Spieler höchstens zu verspielen gedenkt und nach dessen Namen. Dann mischt der Computer die Karten und zeigt die erste Karte an, die der Spieler gezogen hat. Der Spieler kann nun entscheiden, ob er eine weitere Karte ziehen möchte, oder ob er paßt. Wird die Frage "Noch eine Karte?" mit "Nein" beantwortet, so zieht der Computer seine Karten. Gewonnen hat derjenige, der 21 Punkte oder mehr als der Gegner, aber weniger als 21 Punkte hat. Wer mehr als 21 hat, verliert. Die Karten haben folgende Wertigkeit: Bube zählt zwei, Dame drei, König vier, das As zählt 11, wenn der Spieler jedoch mit dem zuletzt gezogenen As mehr als 21 hätte, zählt es nur eins. Alle anderen Karten (zwei bis zehn) haben den entsprechenden Wert. Sind die beiden zuerst gezogenen Karten Assen, so zählt dies 21.

Das Programm ist weitgehend ohne weitere Erläuterungen verständlich, es seien lediglich zwei Besonderheiten erklärt:

Nach der Initialisierung des Basic- Interpreters wird der Zufallszahlengenerator immer auf den gleichen Anfangswert gesetzt. Dies hätte zur Folge, daß nach dem Start des Spiels immer die gleichen Karten gezogen würden. Deshalb wird mit einem kleinen Unterprogramm in Maschinensprache eine Zufallszahl erzeugt, indem der Akkumulator mit dem Inhalt des Refresh- Registers geladen wird. Dieser Wert wird genutzt, um eine zufällige Sequenz von Zufallszahlen auszuwählen (Zeilen 200 bis 310).

Das Unterprogramm ab Zeile 1000 wird zum Mischen des Kartenstapels benutzt. Das eindimensionale Feld B\$ enthält die 52 Karten. In einer IF ... THEN ... - Schleife wird 51 mal eine zufällig ausgewählte Karte mit einer anderen vertauscht. Damit ist gewährleistet, daß die Karten recht gleichmäßig gemischt sind.

```

5 REM +++++ 17 und 4 +++++
10 REM (c) 1980 Peter Szymanski
15 REM Eröffnung
20 CLS: CLEAR500
30 PRINT "Willkommen zu +++17 und 4+++!"
40 PRINT "Ich halte die Bank."
50 PRINT "Der Einsatz pro Spiel betraegt DM 10."
60 PRINT "Wieviel willst Du insgesamt einsetzen?";
70 INPUT E: F=E
80 IF E < 90 THEN PRINT "Sei nicht so geizig!": GOTO 60
90 INPUT "Und wie ist Dein Name"; A$
100 DIM B$(52)
200 REM Start einer zufälligen Sequenz von
210 REM Zufallszahlen
220 FOR Z=0 TO 7: READY(Z): NEXT
230 DATA 237,95,6,0,205,242,240,201
240 Z=0
250 FOR J=3328 TO 3335
260 POKE J, Y(Z)
270 Z=Z+1
280 NEXT J
290 DOKE 4100,3328
300 A=USR(0)
310 Z=RND(-1/A+1)
320 PRINT "Ich mische die Karten..."
330 GOSUB 1000
340 REM Ziehen der Karten
350 K=1
360 PRINT A$; ", Du ziehst "B$(K)
370 G=VAL(RIGHT$(B$(K),1))
380 IFC=OTHERN GOSUB 1300
385 IF G=11 AND S+G>21 THEN G=1
390 S=S+G: K=K+1
400 IF G=11 OR G=1 THEN R=R+1
410 IF K=53 THEN K=1: PRINT "Ich mische neu.": GOSUB 1000
500 IFS=22 AND R=2 THEN S=21
510 IFS=21 THEN 700
520 IFS>21 THEN 800
530 INPUT "Neue Karte"; D$
540 IF LEFT$(D$,1) = "J" THEN 360
550 PRINT "Ich ziehe "B$(K)
560 G=VAL(RIGHT$(B$(K),1))
570 IF G=0 THEN GOSUB 1300
575 IF G=11 AND C+G>21 THEN G=1
580 C=C+G: K=K+1
590 IF G=11 OR G=1 THEN Q=Q+1
600 IF K=53 THEN K=1: PRINT "Ich mische neu.": GOSUB 1000
610 IFC=22 AND Q=2 THEN C=21
620 IFC=21 THEN 800
630 IFC>21 THEN 700
640 IFC>S THEN 800
650 GOTO 550
700 REM Spieler gewinnt
710 PRINT A$; ", Du gewinnst."
720 E=E+10: F=F-10
730 PRINT "Du hast "E" DM und ich "F" DM."
740 S=0: C=0: R=0: Q=0
750 IF E <= OTHERN PRINT "Ich bin geschlagen!": END
760 IF E <= OTHERN INPUT "Neuer Einsatz"; E
770 INPUT "Noch ein Spiel"; E$: PRINT: PRINT
780 IF LEFT$(E$,1) = "J" THEN 360
790 END
800 REM Computer gewinnt
805 IFC=OTHERN PRINT "Du bist kaputt!": GOTO 820
810 PRINT "Ich gewinne mit "C" Punkten!"
820 E=E-10: F=F+10: GOTO 730
1000 REM Unterprogramm Mischen
1010 A=1: RESTORE 1100: FOR Z=1 TO 4: FOR J=1 TO 13
1050 READ B$(A): A=A+1: NEXT J: RESTORE 1100: NEXT Z
1100 DATA 2,3,4,5,6,7,8,9,10,Bube,Dame,Koenig,As
1110 C$="Karo": FOR A=1 TO 52
1140 IFA=14 THEN C$="Pik "
1150 IFA=27 THEN C$="Herz "
1160 IFA=40 THEN C$="Kreuz "
1165 B$(A)=C$+B$(A): NEXT A
1180 FOR A=1 TO 51
1190 Z=A+INT(RND(1)*(52-A+1))
1200 C$=B$(A): B$(A)=B$(Z): B$(Z)=C$
1210 NEXT A: RETURN
1300 IF VAL(RIGHT$(B$(K),2))=10 THEN G=10
1310 IF RIGHT$(B$(K),2) = "As" THEN G=11
1320 IF RIGHT$(B$(K),4) = "Bube" THEN G=2
1330 IF RIGHT$(B$(K),4) = "Dame" THEN G=3
1340 IF RIGHT$(B$(K),6) = "Koenig" THEN G=4
1350 RETURN
1360 END

```

# 14 logeleien

## 8 Damen Problem aus Heft 0 / 80

Inzwischen haben uns zum 8 Damen Problem aus Heft 0 / 80 einige Zuschriften erreicht. Allen Lesern möchten wir an dieser Stelle für die zahlreichen Hinweise danken. Einige von Ihnen haben sich tatsächlich sehr viel Mühe gemacht. Herzlichen Dank !

Zunächst ist zu sagen, daß in dem vorge - stellten BASIC-Programm eine Zeile verges - sen wurde. Dadurch fehlt eine Diagonale und zwar die nach rechts abfallende. Die fehlende Zeile muß lauten:

```
613 IF A(C)+C = A(B)+B GOTO 400
```

Dann bringt das Programm die 92 möglichen Lösungen, wobei 23 Lösungen "echte" Lösungen sind, die anderen entstehen durch 90° Dre - hungen des Spielbrettes.

Lothar Bayer aus München hat die lange Ausführungszeit des TINY BASIC-Programmes nicht gefallen und er hat für das JOURNAL den Algorithmus des vorgestellten TINY BASIC Programmes in Maschinensprache formuliert. Der Erfolg ist durchschlagend: Das Maschinen - programm läuft im Vergleich zum BASIC-Pro - gramm um einen Faktor 400 schneller ! Es braucht für alle möglichen 92 Lösungen nur ca. 9 Sekunden Rechenzeit. Es mußte sogar eine Verzögerungsroutine in die Ausgabe eingebaut werden, damit man sich die Lösungen überhaupt anschauen kann.

Das Maschinenprogramm wird bei Adresse 0C50 gestartet. Die Anfangsadresse ist 0C50 , die Endadresse 0DB8. Das Programm läuft mit NASBUG T2 oder NASBUG T4.

```
0C50 31 00 10 00 00 00 21 00
0C58 0C 22 4B 0C 03 00 0D 00
```

```
0C60 CD 3B 01 FE 1F 20 05
0C68 06 90 FF 10 FD C9 00
```

oder

```
C60 CD 3B 01 FE 1F 20 05 CD
C68 09 00 30 FB C9 00 00 00
```

Bei dieser Alternative wird bei jedem Betätigen der SPACE - Taste (Zwischenraumtaste) die nächste Lösung angezeigt.

```
0CD0 // // // // 21 00 0F 0E
0CD8 10 36 00 23 10 FB EF 1E
0CE0 1F 00 EF 20 38 2D 44 41
0CE8 4D 45 4E 20 2D 20 50 52
0CF0 4F 42 4C 45 4D 1F 00 C9
```

```
0D00 00 00 CD 04 0C 21 00 0F
0D08 36 00 34 E5 7E 6F 36 00
0D10 E1 E5 7E 6F 7E E1 FE 08
0D18 38 07 35 7E B7 28 59 18
0D20 F0 E5 7E 6F 34 E1 7E FE
0D28 02 38 33 46 0E 00 10 01
0D30 04 E5 7E 6F 5E E1 E5 0C
0D38 69 7E E1 BB 28 D3 E5 7E
0D40 EF 7E 95 5F E1 E5 69 7E
0D48 E1 91 BB 28 C4 E5 7E 6F
0D50 7E 85 5F E1 E5 69 7E E1
0D58 81 38 28 B5 10 D3 7E FE
0D60 08 38 A7 CD A0 0D 0F 08
0D68 E5 23 CD 90 0D 7E CE 00
0D70 CD 60 0C 10 F4 21 18 A2
0D78 EF 1F 1F 20 45 4E 44 45
0D80 20 33 CD E2 0C 27 30 02
0D88 00 02 00 00 00 00 00 00
0D90 C5 3F 21 3E 20 CD 60 0C
0D98 10 F9 C1 C9 00 00 00 00
0DA0 E5 EF 1F 00 00 2E 09 7E
0DA8 0F 01 27 77 E1 CD 44 02
0DB0 EF 2E 00 00 00 00 00 00
0DB8 C9 00 00 00 00 00 00 00
```

01.	1	5	8	6	3	7	2	4	47.	5	1	4	6	8	2	7	3
02.	1	6	8	3	7	4	2	5	48.	5	1	8	6	4	2	7	3
03.	1	7	4	6	8	2	5	3	49.	5	5	2	4	6	3	7	1
04.	1	7	5	8	2	4	6	3	50.	5	5	2	4	6	3	7	1
05.	2	4	6	8	3	1	7	5	51.	5	5	2	2	4	7	1	1
06.	2	5	7	1	3	8	6	4	52.	5	5	2	2	4	7	1	1
07.	2	5	7	4	1	8	6	3	53.	5	5	2	2	4	7	1	1
08.	2	6	1	7	4	8	3	5	54.	5	5	3	1	7	2	4	7
09.	2	6	8	3	1	4	7	5	55.	5	5	3	1	7	2	4	7
10.	2	7	3	6	8	5	1	4	56.	5	5	3	1	7	2	4	7
11.	2	7	5	8	1	4	6	3	57.	5	5	7	1	3	8	6	4
12.	2	8	6	1	3	5	7	4	58.	5	5	7	1	4	2	8	6
13.	3	1	7	5	8	2	4	6	59.	5	5	7	2	4	8	1	3
14.	3	5	2	8	1	7	4	6	60.	5	5	7	2	4	8	1	3
15.	3	5	2	8	6	4	7	1	61.	5	5	7	2	4	8	1	3
16.	3	5	7	1	4	2	8	6	62.	5	5	7	4	1	3	8	6
17.	3	5	8	4	1	7	2	6	63.	5	5	8	4	1	3	6	2
18.	3	6	2	5	8	1	7	4	64.	5	5	8	4	1	7	2	6
19.	3	6	2	7	1	4	8	5	65.	5	5	1	5	2	8	3	7
20.	3	6	2	7	5	1	8	4	66.	6	2	7	1	3	5	8	4
21.	3	6	4	1	8	5	7	2	67.	6	2	7	1	3	5	8	4
22.	3	6	4	2	8	5	7	1	68.	6	2	7	1	7	5	8	3
23.	3	6	8	1	4	7	5	2	69.	6	3	1	7	5	8	2	7
24.	3	6	8	1	5	7	2	4	70.	6	3	1	8	5	2	4	7
25.	3	6	8	2	4	1	7	5	71.	6	3	1	8	5	2	4	7
26.	3	7	2	8	5	1	4	6	72.	6	3	5	7	1	4	2	8
27.	3	7	2	8	6	4	1	5	73.	6	3	5	8	1	4	2	7
28.	3	8	4	7	1	6	2	5	74.	6	3	7	2	4	8	1	5
29.	4	1	5	8	2	7	3	6	75.	6	3	7	2	4	8	1	5
30.	4	1	5	8	6	3	7	2	76.	6	4	1	5	8	2	7	3
31.	4	2	5	8	6	1	3	7	77.	6	4	1	5	8	2	7	3
32.	4	2	7	3	6	8	1	5	78.	6	4	2	8	5	1	3	6
33.	4	2	7	3	6	8	5	1	79.	6	4	2	8	5	1	3	6
34.	4	2	7	5	1	8	6	3	80.	6	4	7	1	8	2	5	3
35.	4	2	8	5	7	1	3	6	81.	6	4	7	1	7	5	3	6
36.	4	2	8	6	1	3	5	7	82.	7	1	3	8	6	4	2	5
37.	4	6	1	5	2	8	3	7	83.	7	2	4	1	8	5	3	6
38.	4	6	8	2	7	1	3	5	84.	7	2	6	3	1	4	8	5
39.	4	6	8	3	1	7	5	2	85.	7	3	1	6	8	2	4	4
40.	4	7	1	8	5	2	6	3	86.	7	3	8	2	5	1	6	4
41.	4	7	3	8	2	5	1	6	87.	7	4	2	5	8	1	3	6
42.	4	7	5	2	6	1	3	8	88.	7	4	2	8	6	1	3	5
43.	4	7	5	2	1	6	8	2	89.	7	5	3	1	6	8	2	4
44.	4	8	1	3	6	2	7	5	90.	8	2	4	1	7	5	3	6
45.	4	8	1	5	7	2	6	3	91.	8	2	5	3	1	7	4	6
46.	4	8	5	2	1	7	2	6	92.	8	4	1	3	6	2	7	5

Minifloppy für NASCOM 1 und NASCOM 2  
Ist jetzt erhältlich. Die seit langem angekündigte Floppy-Disk aus England ist leider noch nicht lieferbar, sodaß wir uns vor längerer Zeit zu einer Eigenentwicklung entschlossen haben.

Die Grundausstattung besteht aus einem Floppy-Disk Laufwerk (BASF Typ 6106) sowie einer Controller-Karte mit Bootstrap-Loader. Sämtliche Steuersoftware ist auf Diskette gespeichert und umfaßt ein Floppy-Betriebssystem (DOS) sowie ein sehr leistungsfähiges BASIC. Im Preis eingeschlossen ist eine einjährige Software-Pflege. D.H.: Sie bekommen ein Jahr lang, vom Kaufdatum an gerechnet, immer die neuste Software nachgeliefert. Geplant sind u.a. ein leistungsfähiger MACRO-Assembler, eine kleines Textverarbeitungssystem sowie ein sog. "Supermonitor".

Der Hardware-Aufwand ist minimal, denn die Controller-Karte, die das übliche Format der NASCOM-Erweiterungskarten hat, ist fertig aufgebaut und getestet. Mit minimalen Hardware-Änderungen am System ist die Floppy nach ca. 30 min. betriebsbereit. Wer keinen Platz mehr im vorhandenen Gehäuse hat, kann auch die aufgebaute und getestete Version mit eigenem Gehäuse und Netzteil wählen. Das ist natürlich etwas teurer.

Bei Bestellung geben Sie bitte an, ob Sie die Floppy-Disk an einem NASCOM 1 oder an einem NASCOM 2 betreiben wollen. Versteht sich, daß eine ca. 200-seitige deutsche Beschreibung im Preis inbegriffen ist.

Floppy 1. Laufwerk mit Controller und Steuersoftware 1548.- + MWSt

Floppy 1. Laufwerk mit Controller und Steuersoftware sowie einem Floppy-Gehäuse, in dem Floppy-Laufwerk und Netzteil montiert sind 1898.- + MWSt.

2. Laufwerk 880.- + MWSt.

Eine Controller-Karte kann 2 Laufwerke bedienen. Bei den angebotenen Laufwerken handelt es sich um 5" Minifloppies mit einer Speicherkapazität von ca. 100 kByte.

#### PIO-Board

Auf der I/O-Karte sind Plätze für 3 PIO-Bausteine, einen UART und einen CTC-Zähler-Zeitgeber. Die Karte wird für 189.- + MWSt ohne alle Optionen geliefert, sodaß man sich für die jeweiligen Anwendungen ein System individuell zusammenstellen kann. Zu jeder Option werden auch die zugehörigen Stecker und Kabel gleich mitgeliefert.

#### EPRM-Programmiergerät für NASCOM 1 und NASCOM 2

Das EPRM-Programmiergerät ist nun schon bei einigen hundert Anwendern in Betrieb. Mittlerweile gibt es schon eine Version, die mit dem NASCOM 2 läuft.

Das Gerät wird über die PIO angesteuert und kann 2708 oder 2716 - EPROMs (1k x 8 bzw. 2k x 8) programmieren. Es können allerdings bislang nur Typen mit 3 Betriebsspannungen programmiert werden.

Die Steuerung des Programmierers erfolgt über ein EPROM, das voll relokaterbar ist. Es ist also gleichgültig, auf welche Adresse Sie die Steuersoftware legen, sie funktioniert immer.

Auch die Programmierspannung stellt kein Problem dar. Die - 26Volt werden aus der 12Volt-Versorgung des NASCOM mit einem einfachen Spannungswandler gewonnen.

Bausatz: 168.- + MWSt; Typ des Rechnersystems bzw. Betriebssystem angeben.

Wandler auf der Karte !

Umschaltkarte zum Umschalten zwischen den Betriebssystemen NAS-SYS 1 und NASBUG T4. Mancher Benutzer möchte die Vorteile nutzen, die NAS-SYS 1 für den Betrieb mit BASIC bietet (z.B. frei beweglicher Cursor), aber andererseits nicht auf den gewohnten NASBUG T4 verzichten, für den vielleicht schon zahlreiche Programme vorliegen. So bietet sich als Kompromiß die Umschaltkarte an. Auf diese Karte werden beide Betriebssysteme aufgesteckt. Von der Karte führen zwei Flachsteckverbinder in die Sockel, in denen bislang der NASBUG T4 steckte.

Platine, Bausatz mit allen Teilen einschließlich Flachsteckern: 68.- + MWSt

#### 3k ASSEMBLER für NASCOM 1 und NASCOM 2

Der EPROM-Assembler ist mit ZEAP kompatibel. Das heißt, Sie können die schon vorhandenen Bänder mit symbolischen Code weiterverwenden. Zusätzlich hat der EPROM-Assembler noch einen erweiterten Editor, der z.B. mit einem verbesserten F-Befehl arbeitet. Noch einige andere Dinge sind verbessert worden. So kann es bei ZEAP passieren, daß man mit dem Assembler den Assembler selbst überschreibt und zerstört. Das kann z.B. geschehen, wenn man ein Programm assembliert, dessen Maschinencode bei 0C50 beginnt, aber weiter als bis 0F00 reicht. Einerseits kann man vom symbolischen Code her nicht ohne weiteres auf einen Blick sagen, wie weit der Maschinencode in den Speicher hinein laufen wird, andererseits "merkt" ZEAP das ohnehin nicht und zerstört sich selbst.

Der EPROM-Assembler verhindert dies. Er schützt zusätzlich noch die Variablenfelder, die er im Speicher ablegt.

Der NASCOM 2 - Assembler hat den großen Vorzug, daß man die vollen Editiermöglichkeiten nutzen kann, die NAS-SYS 1 bietet. Man kann mit dem Cursor überall auf dem Bildschirm herumlaufen, den Text verändern und als neue Eingabe verwenden.

Beide ASSEMBLER kommen mit deutscher Beschreibung. Bei Bestellung bitte Typ des Betriebssystems angeben !  
Jeder EPROM-Assembler 240.- + MWSt

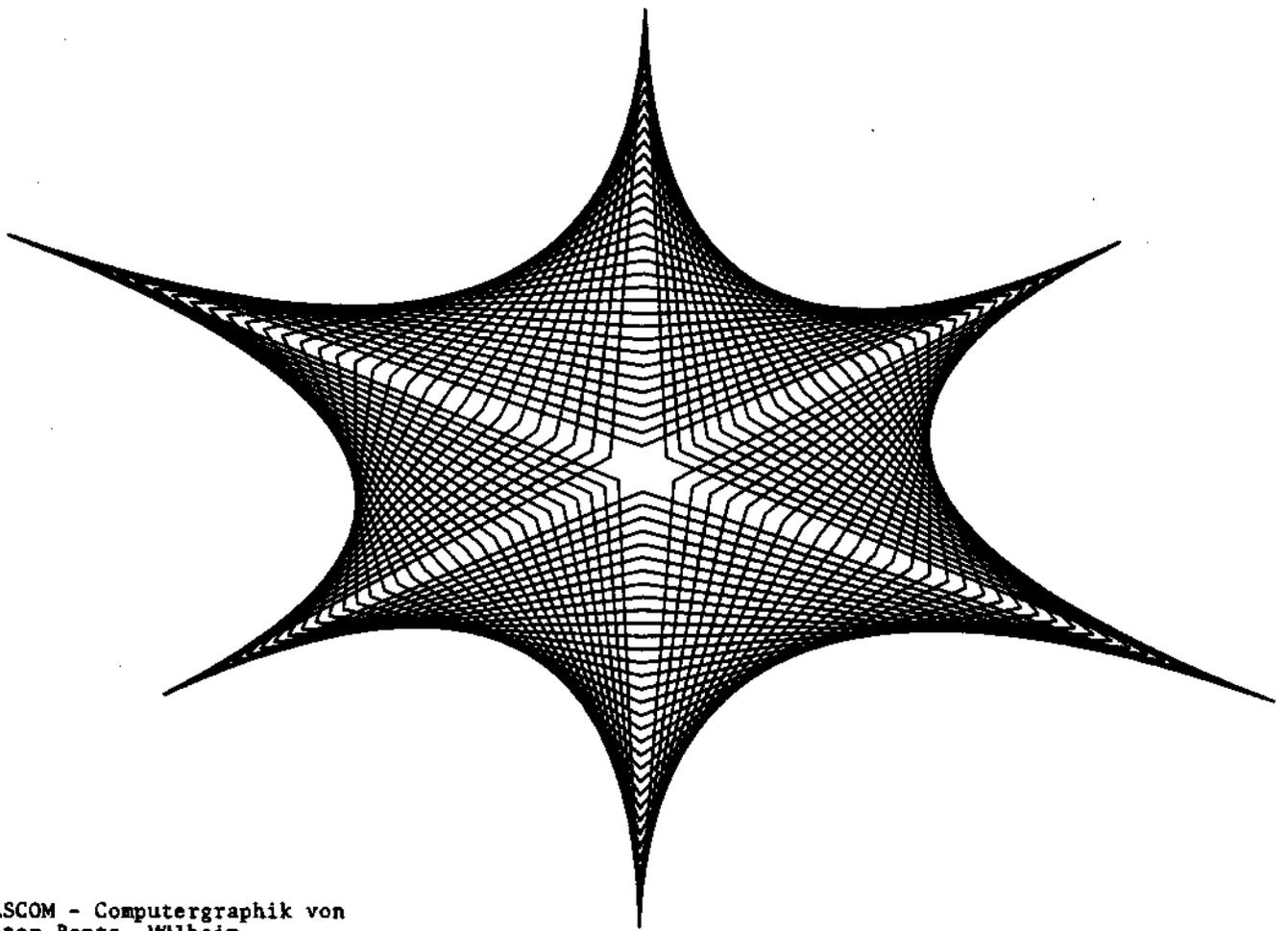
#### 2k Disassembler für NASCOM 1 und NASCOM 2

Der Disassembler dient dazu, Maschinenprogramme in Assemblerprogramm zurück zu übersetzen. Unser Disassembler kann sogar Bänder erzeugen, die man mit ZEAP oder unseren EPROM-Assemblern gleich weiterverarbeiten kann.

Den Assembler gibt es für NASCOM 1 auf Band mit den Startadressen 1000H und 4000H.

Für NASCOM 2 ist der Assembler auf EPROMs erhältlich.

NASCOM 1 - Disassembler 79.80 + MWSt  
NASCOM 2 - Disassembler 120.00 + MWSt



NASCOM - Computergraphik von  
Peter Bentz, Mülheim

---

## kleinanzeigen

RAFI-Tastatur (HALL-Effekt-Tasten) z.T. schon ASCII-dekodiert, neu über 600.- jetzt für 150.- zu haben.

Schneller Impulsverstärker aus der Kernforschung (Hämmer-Verstärker) und Differenzierer. Röhren-Gerät 50.-, Transistorgerät 98.-

Zwei regelbare Hochspannungsmetzgeräte 0.5kV - 5kV ca. 1mA; Stück 35.-

Lochstreifenleser, superschnell (ca. 1500 Zeichen/s) fabrikneu, 5,6,7,8 - Kanal mit Service-Karten und allen Unterlagen 1250.-

M. Klein, Waldstraße 20, 6728 Germersheim/Rhein

---

NASCOM 1 mit NASBUG T4, aufgebaut und getestet 680.- Buffer Bus Board, aufgebaut, getestet 120.- Super TINY BASIC 115.- ZEAP 99.-

Uwe Wurditsch, [redacted], [redacted]

---

19-Zoll Gehäuse, blau, 6 Höheneinheiten, 60.- mit Tragegriffen, brauchbarer Zustand

Hans-Jörg Dietmann, [redacted]

Nimm-Spiel für NASCOM 1 erläutert sich selbst (0.7kB) HARD-Copy des Bs. über PIO (Interrupt) möglich. Incl. Dokumentation wie HEX-Dump u. Prog.-Listing aller U-Prog. wie blinkender Cursor gegen Assembler oder 35.-

Lothar Bayer, [redacted]

---

ZEAP - Assembler/Editor Package für NASCOM 1, jetzt zum Schleuderpreis von 50.-

M K - Systemtechnik, Postfach 730, Waldstraße 20, 6728 Germersheim/Rhein

### LEERKASSETTEN



Speziell geeignet für Datenaufzeichnung. Hochwertiges BASF-Band. Cassette 5-fach verschraubt. Cassette C10, d.h. 10 Minuten spieldauer, daher besonders geeignet für Mikrorechnerprogramme.

10 Stk	19.80	Jede Kassette mit selbst-
20 Stk	36.00	klebendem Aufkleber zum
50 Stk	87.50	Beschriften.
100 Stk	160.00	

Bei: M K - Systemtechnik  
Waldstraße 20  
6728 Germersheim/Rhein  
Tel.: 07274/2756